Quantifying Timing and Rate of Deformation and Exhumation in the Central Andes: Insights from Thermokinematic and Landscape Models of Balanced Cross-Sections

by

Victoria M. Buford Parks

Geophysics and Applied Geophysics B.S., University of California- Los Angeles, 2015

Submitted to the Graduate Faculty of the

Dietrich School of Arts and Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2020

UNIVERSITY OF PITTSBURGH

DIETRICH SCHOOL OF ARTS AND SCIENCES

This dissertation was presented

by

Victoria M. Buford Parks

It was defended on

November 20, 2020

and approved by

Dr. J. Ryan Thigpen, Academic Rank, Assistant Professor, Earth & Environmental Sciences, University of Kentucky

Dr. Charles E. Jones, Senior Lecturer, Geology and Environmental Science

Dr. Brian Stewart, Associate Professor, Geology and Environmental Science

Dr. Eitan Shelef, Assistant Professor, Geology and Environmental Science

Thesis Advisor/Dissertation Director: Dr. Nadine McQuarrie, Professor, Geology and Environmental Science

Quantifying Timing and Rate of Deformation and Exhumation in the Central Andes: Insights from Thermokinematic and Landscape Models of Balanced Cross-Sections

Victoria M. Buford Parks, PhD

University of Pittsburgh, 2020

In the central Andes, we couple detailed, sequentially deformed, forward modelled, balanced cross-sections, including flexural isostasy and erosion, with advection-diffusion thermal models (Pecube) and a physics-based surfaces processes model (CASCADE) in order to provide insight on the structural and morphological history evolution. Preserved basin histories in the Altiplano, along with geologic mapping and thermochronology sampling, assist in constraining the location and timing of exhumation, topographic development, and the geometry of active faulting. Thin-skinned surface exposures of Paleozoic folded and faulted rocks are balanced at depth by long basement thrust sheets. These basement thrust sheets, and their associated ramps, are the first-order control on cooling age patterns in the central Andes, as shown through our Pecube modelling. Incision level also has an important impact on the suite of thermal ages measured, and our new methodology allows us to synchronously track and thermally model interfluve and canyon elevations. We additionally use our sequentially deformed cross-section as the kinematic input to landscape evolution models, which apply realistic hillslope and fluvial erosion based on a suite of erosional and climatic patterns to produce a landscape based on the cross-section geometry and kinematics. Extracting geomorphic indices of uplift (eg river channel steepness, Ksn) from the modeled topography and comparing the modelled to modern-day, remotely acquired, Ksn, we are able to link these surface uplift indicators with subsurface geometry

that can reproduce those metrics. We can trace the regional Ksn pattern to gain insight into the regional subsurface geometry and thus better locate active, or recently active, faults in the region along strike. The combination of traditional structural techniques, such as kinematically and flexurally sequentially modelled balanced cross-sections, and newer techniques of thermal and landscape modelling allow us to investigate the geomorphic response of viable thermo-kinematic models and derive potential regional variations in and along strike subsurface geometry and kinematics.

Table of Contents

List of Tables

List of Figures

1.0 Introduction

The Central Andes provide a focal point to characterize the dynamics of the Earth's surface and upper lithosphere. Here, a suite of incised river canyons, exposed Paleozoic rocks in a series of folds and faults, and a unique captured basin history allow us to quantify the age and rates of structural deformation and exhumation histories in southern Peru and central Bolivia. In addition to connecting structural deformation and exhumation, we also evaluate how topography develops through time, both through a kinematic and flexural structural lens as well as through a surface-processes geomorphic lens. Finally, we address the relative importance of active faulting, and the rates at which it is occurring.

In Central Bolivia, our research proposes to identify the lateral variations in subsurface fault geometry, as well as classify the morphological responses to deformation. The subsurface geometry is refined using mapped geological relationships, sequentially restored balanced cross-sections, thermochronology, and geomorphology. The mapped relationships are used to create a balanced cross-section, and the restored section is then imported into Midland Valley's 2DMove. This restored section is sequentially deformed, flexurally loaded, and eroded (McQuarrie and Ehlers, 2015). Because the exhumation, or uplift and erosion, of rocks is focused at ramps (Whipp et al., 2007; Lock and Willett, 2008; McQuarrie and Ehlers, 2015), we can use the cooling ages of rocks to both locate and date the timing of this uplift driven exhumation. In addition, we can use disequilibrium topography to locate current or recent uplift. Thermochronology uses certain responsive minerals, such as apatite and zircon and the decay of radioactive isotopes in these minerals to assign an age for when rocks cool through specific temperatures. As these minerals pass through their closure temperature, diffusion of the daughter products of radioactive decay ceases, and thus from this time onwards the radioactive decay products are preserved. (Reiners and Ehlers, 2005; McQuarrie and Ehlers, 2017). These mineral systems, when combined with balanced cross-sections, allow us to constrain the timing, rate, and geometry of fault motion. Once the 2D restoration of a geologic cross section matches the available surface data, a 0.5x0.5km grid is overlain and deformed in ~10 km steps. This allows us to track individual rock paths from depth to the surface, and constrain the subsurface geometry through time (McQuarrie and Ehlers, 2015). The deformed grids are then imported in to Pecube, a 2D advection-diffusion thermal model, to determine the distribution of temperatures in the modelled subsurface and provide predicted thermochronology ages based on the modelled section (Braun, 2003). The predicted ages are compared to sample data, and the geometry and timing are systematically varied to match measured and predicted cooling ages for a best-fit thermal model. We map the lateral variation in subsurface structures and characterize the deformation history in central Bolivia through a combined structural and thermal model in Chapter 2.0, which is published in *Tectonophysics* as Buford Parks and McQuarrie (2019).

In southern Peru, we follow a similar, but modified, methodology as described above to evaluate the timing of canyon incision into the eastern margin of the actively deforming Andes. This region in southern Peru has been the site of controversy regarding the timing and drivers of incision (Garzione et al., 2017; Jeffery et al., 2013; Lease and Ehlers, 2013; Perez et al., 2016; Schildgen et al., 2007). Using a suite of newly collected thermochronology data, we compare modelled ages for multiple along-strike elevations: interfluve, mean, and canyon elevations. These three topographic elevation profiles are tracked within our Move model, and separately modelled in Pecube so that we can evaluate the timing and rate of canyon incision on modelled thermochronology ages in Chapter 3.0.

In addition to the above structural modelling efforts, which allow a detailed 2D view of the deformation, geomorphology data and modelling can be used to give a 3D picture of the lateral variations in structures across the region, as well as the impacts of structural evolution on the formation of modern day topography. Geomorphic indicators of uplift, such as stream channel steepness and knickpoints, provide data points for active uplift. Typical geomorphic models are limited solely to vertical motion that define the area of uplift based on geomorphic indicators (see Whipple and Gasparini, 2014). We will use geomorphic indicators to inform where regions of uplift are, but the geometry of the uplift will be constrained by viable balanced cross-section structures and thermochronology. The velocity vectors determined from the balanced section and used as input into *Pecube* will also be input into a modified version of the landscape model CASCADE (Braun and Sambridge, 1997) that now allows lateral translation (Eizenhöfer et al., 2019). This approach will allow us to test if the subsurface geometry will actually produce the geomorphic data through physics based surface processes forward modeling. The reconstructed history of cross-sectional deformation must be consistent with all of the available data sources: geologic mapping, thermochronology, and geomorphology. This approach to geomorphic modelling will allow us to evaluate the sensitivity between horizontal motion and advection with the resulting geomorphic features and indices (e.g. Miller et al., 2007). The subsurface geometry that results in the surface geomorphology is inferred independently from these geomorphic analyses, and allows evaluation as to whether upstream increases in stream channel steepness values previously interpreted as varying vertical uplift are related to the horizontal velocity of deformation. We provide additional insight into the structural and morphological history through our investigation coupling structural deformation and the physics-based surfaces process model CASCADE in Chapter 4.0.

2.0 Chapter 1: Kinematic, Flexural, and Thermal Modelling in the Central Andes: Unravelling Age and Signal of Deformation, Exhumation, and Uplift

2.1 Introduction

Quantifying age, rate, and lateral variation of deformation and exhumation in convergent systems relies on integration of geologic map patterns, age and locations of reset thermochronometer systems, and synorogenic sediment distribution. The central Bolivian Andes provide an ideal location to examine the influence of variations in shortening and stratigraphic architecture on the structural evolution of the mountain range due to differential age and rates of shortening, and distinct sedimentary basin geometries along strike. To quantify age and rate of shortening, we link thermokinematic modelling of sequentially deformed, forward -modelled, balanced cross-sections to synorogenic and thermochronologic histories. The preserved basin history in the Altiplano and Eastern Cordillera argues for an early fold-and-thrust belt located in the now-Western Cordillera, with subsequent propagation of shortening eastward around 40-50 Ma. Flexural modelling incorporating isostasy and erosion requires multiple basement thrust sheets with 35-97 km of displacement. A temporally evolving effective elastic thickness, as well as imposed subsidence in the foreland and uplift in the hinterland, are required to reproduce the surface geology, increase Subandean foreland basin depth, limit Altiplano sedimentation, and facilitate Altiplano uplift to modern elevation. Thermokinematic modelling is compatible with initiation of deformation at 50-40 Ma in a marked increase in Subandean velocities from ~5.5 to 8-10 mm/yr from ~12-10 Ma to present. Out-of-sequence thrusting at the westernmost limit of the Subandes is required to match measured young and partially reset zircon helium ages. Out-of-sequence faulting is supported by high Ksn values, indicative of active uplift, and was likely promoted by the abrupt eastern edge of the Paleozoic basin rocks, which limited forward propagation of structures, and/or increased erosion due to focused precipitation. Our results highlight the importance of incorporating detailed structural modelling in differentiating the geometry, kinematics, and timing of deformation to reproduce thermochronologic ages and basin histories.

The Andes mountains, formed in response to compressional strain driven by the subducting Nazca plate, is the modern archetype of a retro-arc fold-thrust-belt-foreland basin (FTB-FB) system. The Central Andes, located in northern Chile, Argentina, Bolivia, and southern Peru is the widest portion of the mountain belt. Both proposed and observed variations in timing and magnitude of deformation, exhumation, and uplift across the Central Andes have generated questions about the processes of lithospheric deformation, and delamination and crustal growth, which are key to understanding the impact of Andean deformation on society from both natural hazards and hydrocarbon extraction. Although results and models from one geographic location are often depicted as applying to the Central Andes as an entity, the map-view expression and style of structures, stratigraphy exposed at the surface, age and magnitude of exhumation, and uplift show a lateral variability along and across strike Figure 1.1)(Anderson, Long, Horton, Calle, & Ramirez, 2017; Barnes, Ehlers, McQuarrie, O'Sullivan, & Tawackoli, 2008; Garzione et al., 2014; Lease, Ehlers, & Enkelmann, 2016; McQuarrie, 2002; Saylor & Horton, 2014; Sundell, Saylor, Lapen, & Horton, 2019). Map patterns of rocks and visible structures are commonly used to understand the extent of subsurface faulting, as well as infer locations of lateral structures (Boyer & Elliot, 1982; Kley, 1996). Additionally, integrative work incorporating well-mapped geology, structural interpretations, rock samples, and thermal and climate modelling can be used to help understand lithospheric processes and evolution (Garzione et al., 2017; Horton, 2018a). While many previous studies have attempted to delineate structural relationships, age of initiation of shortening, and magnitude and timing of deformation, differing interpretations of the existing data and associated uncertainties has led to conflicting interpretations, such as predominantly west-verging surface and basement structures at the western edge of the eastern Cordillera (Armijo, Lacassin, Coudurier-Curveur, & Carrizo, 2015; Müller, Kley, & Jacobshagen, 2002) or east-verging (Anderson et al., 2017; McQuarrie, 2002), differing age and rate of SA shortening (Anderson et al., 2018; Gubbels, Isacks, & Farrar, 1993; Lease et al., 2016; McQuarrie, Horton, Zandt, Beck, & DeCelles, 2005; Oncken, Boutelier, Dresen, & Schemmann, 2012; Oncken et al., 2006; Rak, McQuarrie, & Ehlers, 2017), and the correlation (or lack thereof) between shortening and uplift (Barnes & Ehlers, 2009; Garzione et al., 2008, 2017; S. Lamb, 2011). Our goal with this contribution is to quantitatively link the structural evolution of the FTB at 18°S, with the resulting cooling ages and foreland basin history to evaluate permissible geometries, kinematics, and rates. The process of linking kinematic models of deformation derived from balanced cross -sections to advection–diffusion thermal models in order to calculate the evolving subsurface temperatures and predict cooling ages has been explored recently by several research groups (Almendral et al., 2015; Castelluccio et al., 2015; Chapman, 2017; Erdös, Van Der Beek, & Huismans, 2014; Gilmore, Mcquarrie, Eizenhöfer, & Ehlers, 2018; McQuarrie & Ehlers, 2015; Mora et al., 2015; Rak et al., 2017). We incorporate fault motion, isostasy, and erosion predicted by a geologic cross-section with the predicted thermal evolution in order to characterize the relationship between fault geometry, timing and magnitude of shortening, exhumation, and sedimentation (Gilmore et al., 2018; McQuarrie & Ehlers, 2015; Rak et al., 2017)

Although published cross-sections along the Bolivian Andes show many similarities, there are also pronounced differences in the geometry and location of basement structures and the proposed kinematics that link basement and surface deformation, exhumation and sedimentation (Anderson et al., 2018; Armijo et al., 2015; Baby, Rochat, Mascle, & Herail, 1997; McQuarrie, 2002; McQuarrie, Barnes, & Ehlers, 2008; McQuarrie et al., 2005; Müller et al., 2002; Rak et al., 2017). In addition, across Bolivia, age and magnitude of exhumation gleaned from apatite fission track (AFT) and apatite and zircon (U-Th)/He (AHe, ZHe) thermochronology show significant variations, with initiation of exhumation in the EC at 45-50Ma near both 15-17°S and 18°S, but not until ~36-42 Ma at 19.5-21°S (Anderson et al., 2018; Barnes, Ehlers, Insel, McQuarrie, & Poulsen, 2012; Barnes et al., 2008). Interandean zone (IAZ) exhumation in the north began >25 Ma with a second pulse of exhumation ~15 Ma to present (Barnes, Ehlers, McQuarrie, O'Sullivan, & Pelletier, 2006). In central Bolivia, rapid IAZ exhumation was likely between 18-6 Ma, but exhumation could have initiated as early as 40-50 Ma (Barnes et al., 2012; Eichelberger et al., 2013). A similar exhumation history exists for the IAZ of southern Bolivia: a potential ~25-17 Ma start with rapid exhumation between 20-5 Ma (Anderson et al., 2018; Barnes et al., 2008). In the northern SA, exhumation and deformation started ~19-20 Ma (Barnes et al., 2008; Rak et al., 2017). In the central Bolivian Subandes (SA), exhumation initiated at 14 ± 4 Ma (Barnes et al., 2012; Eichelberger et al., 2013), and in the south, SA exhumation can start potentially as early as 18 Ma (Barnes et al., 2008; Calle et al., 2018), but reset AFT and AHe ages range from 11-6 Ma and 6-2 Ma respectively (Anderson et al., 2018; Lease et al., 2016). The majority of AP sedimentation in the Corque syncline finished around ~10 Ma, with low rates of accumulation until ~5 Ma. Altiplano deformation initiated between ~15 and 10 Ma and continued to ~ 5 Ma. (Garzione, Molnar, Libarkin, & MacFadden, 2006; S. Lamb, 2011; Simon Lamb & Hoke, 1997;

McQuarrie & DeCelles, 2001). Additionally, the San Juan del Oro paleosurface post-dates all EC shortening and is dated at ~ 10 Ma (Gubbels et al., 1993).

Deformation rates depend on the timing of onset of deformation as well as the magnitude of shortening. Shortening estimates in the central Andes range from ~200 to 400 km (Anderson et al., 2017; Baby et al., 1995, 1997; Eichelberger et al., 2015; Gotberg, McQuarrie, & Caillaux, 2010; Kley, 1996; McQuarrie, 2002; McQuarrie, Barnes, et al., 2008; Müller et al., 2002; Perez, Horton, McQuarrie, Stubner, & Ehlers, 2016). With an onset of deformation at ~50 Ma, these shortening magnitudes equate to 4-8 mm/yr at constant rates. While the magnitude of shortening and onset of deformation can provide an initial long-term shortening rate, shortening rates in the Central Andes are likely variable through time (Anderson et al., 2018; Echavarria, Hernandez, Allmendinger, & Reynolds, 2003; Elger, Oncken, & Glodny, 2005; McQuarrie, Barnes, et al., 2008; McQuarrie et al., 2005; Rak et al., 2017; Cornelius E. Uba, Kley, Strecker, & Schmitt, 2009). Active shortening in the Andes, recorded by GPS, is currently accommodated in the SA at ~9-13 mm/yr (Brooks et al., 2011), notably higher than the long term average of 4-8 mm/yr, though Quaternary estimates of convergence rates range from 7-11 mm/yr (Echavarria et al., 2003; Cornelius E. Uba et al., 2009).

The growing suite of structural, sedimentological, geomorphological, and thermochronological data through the central Andes provides an opportunity to evaluate if and how along strike changes in exhumation and sedimentation are related to proposed along strike changes in the geometries of structures and the rates at which they move. We assess the importance of along strike change by evaluating a sequentially-deformed, isostatically balanced, thermo-kinematic model of the central Bolivian Andes near 18°S (Figure 2.1)and compare our results to a similar modelling approach from 15-17°S and published geometries and rates from 19.5-21°S. The combined modelling techniques in this study allows us to evaluate the geometry and kinematic sequence of faulting, permissible timing and rates of deformation, and evaluate the validity of a published cross-section.

2.2 Geologic Background

2.2.1 Central Andes

There are several distinct tectonogeomorphic zones in the Andes; from west to east, these are: Western Cordillera (WC), Altiplano (AP), Eastern Cordillera (EC), Interandean Zone (IAZ), and Subandes (SA). The AP is a low-relief, high-elevation (~3.7 km), internally drained basin consisting of up to 12 km thick Cretaceous and Tertiary synorogenic sedimentary rocks derived initially from sources west of the AP, with upper portions being derived from the EC (DeCelles & Horton, 2003; Horton, 2005; Horton, Hampton, Lareau, & Baldellon, 2002; Horton, Hampton, & Waanders, 2001). The EC and IAZ host a thick (~15 km) continuous succession of Paleozoic marine siliciclastic rocks, and a discontinuous section (2-4 km) of nonmarine Carboniferous through Cretaceous rocks (T. Sempere, 1995) deformed in narrow anticlines and synclines. These zones encompass a bivergent thrust belt system that reaches 6.4 km in elevation in the EC and decreases in elevation towards the east with significant decreases in both topographic and structural elevation in the IAZ and SA. Both the EC and IAZ are argued to be uplifted as the result of basement thrusts faults (Kley, 1996; Kley, Monaldi, & Salfity, 1999; McQuarrie, 2002). The SA is the actively deforming portion of the Andean fold-and-thrust belt (FTB), whose thrust faults carry Cambrian through Cretaceous rocks and fold 4-7 km of Tertiary foreland basin sedimentary rocks (Baby et al., 1995; McQuarrie, 2002).

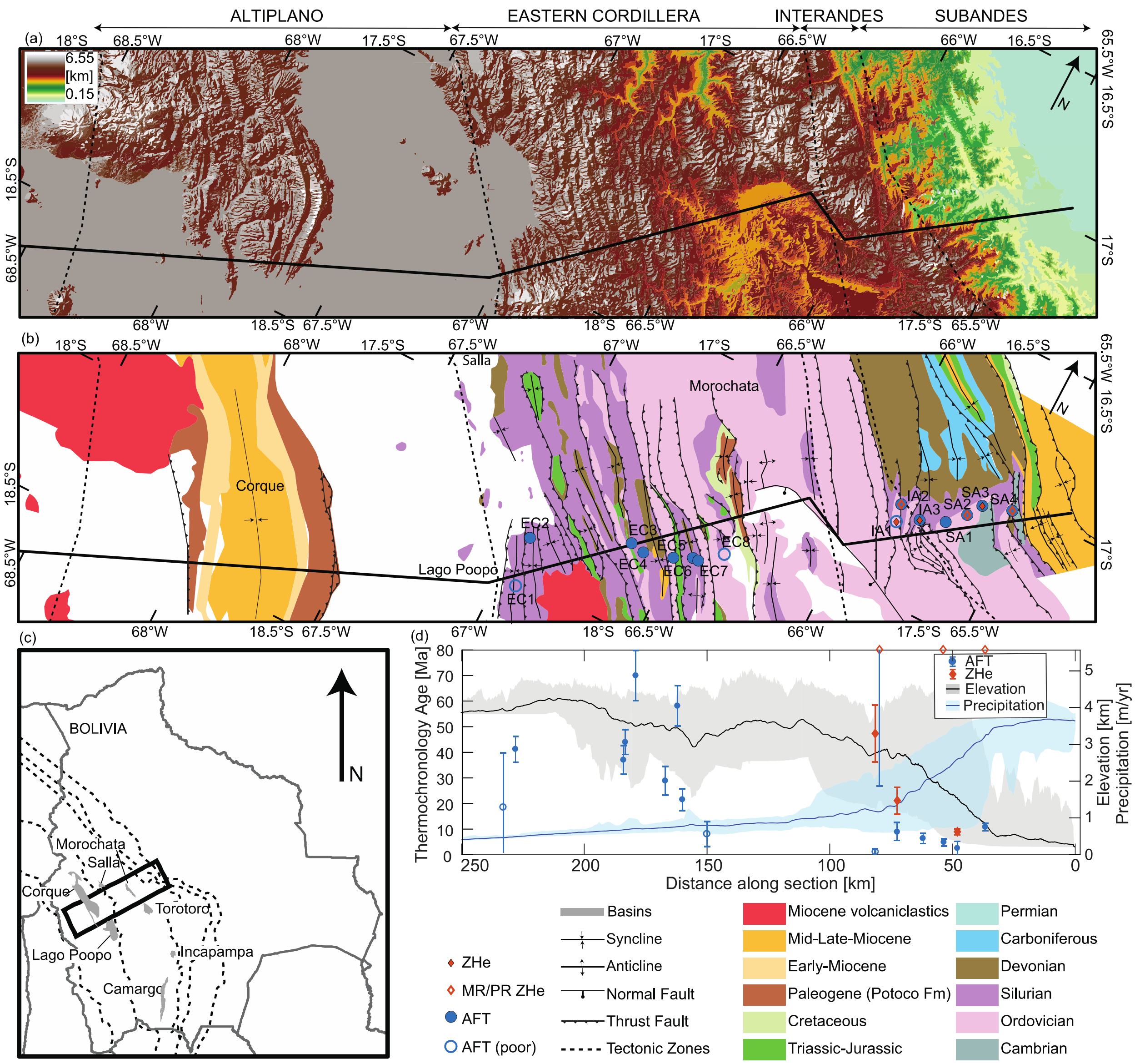


Figure 2.1. Area map

with (a) Elevation, (b) geology and thermochronology sample locations, (c) Regional map highlighting basins and localities referenced in text, and (d) precipitation, elevation and thermochronometer ages across the cross-section; shaded area is the min to max range, with the solid line being the average. Open blue circles indicate samples with poor data quality (Table 2.1); open orange diamonds indicate mixed/partial reset ZHe samples. Dashed lines in (a-c) are tectonogeomorphic zones.

2.2.2 Shortening Estimates and Crustal Thickness

Early estimates of shortening in the Central Andes documented shortening amounts ranging from 191 to 231 km (Baby et al., 1995, 1997; Kley, 1996) that emphasized well-defined structures and detailed shortening estimates in the eastern portions of the system. A wealth of detailed mapping in the western portion of the eastern Cordillera, including improved stratigraphic relationships and fault geometries, have generated larger estimates of total shortening ranging from 265 to 326 km (Anderson et al., 2017; Eichelberger et al., 2013; McQuarrie, 2002; McQuarrie et al., 2008a; McQuarrie and Decelles, 2001; Müller et al., 2002; Rak et al., 2017; and references therein). Estimates of crustal shortening are directly related to the potential for accumulated crustal thicknesses. In Bolivia, the crust ranges from 35 km thick in the foreland to approximately 60 km under the EC and 65+ km thick under the AP (Ryan et al., 2015). The Andean plateau, at 3+ km elevation, is largely in Airy isostatic equilibrium, though shortening estimates and volumetric analyses shown that the uniformly thick crust may be a function of crustal flow from regions of high shortening and over-thickening to areas of lower crustal shortening (Eichelberger et al., 2015).

The balanced cross-section used in this study (original from McQuarrie, 2002) argues that the Paleozoic shortening is balanced at depth by long, ~10 km thick, east-verging basement thrust sheets, where slip along a mid-crustal detachment near the brittle-ductile transition is transferred to upper décollement horizons in the Paleozoic section. The emplacement of these thrust sheets up and over their associated footwall ramps impose a first-order control on topographic uplift, focused exhumation, and thus the pattern of thermochronologic ages (McQuarrie & Ehlers, 2015, 2017b; Rak et al., 2017).

2.2.3 Regional Thermochronology

Thermochronometer cooling ages are a function of the timing, magnitude, and rate of exhumation, and thus directly related to the paths the rocks take to the surface due to the influences of both vertical and lateral transport along faults on the rate and magnitude of exhumation (Ehlers & Farley, 2003; Gilmore et al., 2018; McQuarrie & Ehlers, 2017a; Rak et al., 2017). Thus, subsurface structures impart the first order pattern to cooling ages and can be used to verify or invalidate a cross-section (Gilmore et al., 2018; McQuarrie & Ehlers, 2015). By requiring sequentially deformed balanced cross-sections to produce predicted cooling ages that match measured thermochronologic data, the modelled evolution of a mountain belt can be adjusted to more accurately depict the structural evolution of the area.

Previously published low-temperature thermochronometer samples in the FTB in Bolivia (Barnes et al., 2008, 2006; Eichelberger et al., 2013; Gillis et al., 2006; Lease et al., 2016) have been interpreted as the result of deformation-induced erosional exhumation (Barnes et al., 2012; Eichelberger et al., 2013; McQuarrie, Ehlers, Barnes, & Meade, 2008; McQuarrie et al., 2005). We limit the thermochronometer data used in this study to those located within 75 km of the cross-section to include all potentially relevant cooing ages and exclude significant lateral variability in structure and timing of exhumation captured by the data (Figure 2.1,Table 2.1) (Barnes, 2012; Eichelberger et al., 2013; Lease et al., 2016), and are projected along structure to the cross-section line. Cooling ages and sample locations are shown in Figure 1. Apatite fission track (AFT) pooled ages and zircon (U-Th)/He (ZHe) cooling ages are shown with 2σ error (Figure 2.1c, Table 2.1).

AFT, which has a typical closing temperature of ~110±10°C and a partial annealing range of 60-110°C (Donelick, O'Sullivan, & Ketcham, 2005), are typically reported as pooled ages when concordant ($P(\chi^2)>5\%$), and as mean ages when discordant ($P(\chi^2)<5\%$), as the pooled age may not reflect a distinct geologic event when samples are over-dispersed. However, the AFT data from Barnes et al., 2012, is only available as pooled ages, regardless of whether they are concordant or discordant, as a result of the non-Poissonian counting process used in the LA-ICPMS (Barnes et al., 2012, 2006, 2008). Ages classified as "poor" had n<10 grains measured. Despite these limitations on the quality of data, we use all published AFT data equally. The AFT ages with n>10 grains range from 1.8-8.1 Ma in the SA and IAZ. In the EC, the youngest measured cooling ages have low grain counts (7.3 Ma, n=5; 17.7 Ma, n=2), and the youngest age with n>10 grains is 20.7 Ma (n=37). From east to west, measured AFT ages broadly increase to 68.9 Ma, then decrease to 17.7 Ma at the western most edge of the EC (Barnes et al., 2012; Eichelberger et al., 2013). ZHe cooling ages, with a typical closure temperature of ~180±10°C and partial retention zone of ~130-200°C (Guenthner, Reiners, Ketcham, Nasdala, & Giester, 2013; Peter W. Reiners, 2005; Peter W. Reiners, Farley, & Hickes, 2002; Peter W. Reiners, Spell, Nicolescu, & Zanetti, 2004; Wolfe & Stockli, 2010), increase in age westward from 8.9 to 47.1 Ma over the SA and IAZ (Lease et al., 2016)((Figure 2.1c).

At 18°S, measured AFT cooling ages and associated modelling (HeFTy) argues for onset of EC exhumation around 50-45 Ma (Samples EC3, EC2) (Barnes et al., 2012), as the earliest possible deformation in the EC, set by the age of the earliest fully reset thermochronometric age. Sample EC3 and EC2 have AFT ages 43.0 ± 4.8 Ma and 40.3 ± 5 (respectively) and modelling indicates onset of rapid cooling at 54-45 Ma and 41-25 Ma. Samples with older pooled ages (EC4 and EC6) are best modelled with onset of rapid cooling at 45-35 Ma (EC6) and 32-18 Ma (EC4) (Barnes et al., 2012). EC deformation lasts until ~25 Ma (EC7) (Barnes et al., 2012).

Previous authors suggest that IAZ exhumation initiated as early as ~30 Ma with an acceptable fit, or 14-21 Ma with good fit, and continued exhumation through 2 Ma is supported by HeFTy modelling with AFT and ZHe data (sample IA2) (Barnes et al., 2012; Lease et al., 2016). The pooled AFT ages of samples IA1 (0.5 ± 1 Ma) and IA3 (8.1 ± 3.6 Ma) are a function of the continuing, younger exhumation. The mixed and partially reset ZHe ages are likely a function of sample depth (IA2, SA2, SA1) (Lease et al., 2016) and/or presence around the closure temperature for an extended period of time (IA2) (Barnes et al., 2012). The best estimate for onset of exhumation is 6 ± 2 Ma from integrated age-depth profiles (Lease et al., 2016), with rapid SA exhumation from 7-3 Ma supported by AFT and ZHe HeFTy modelling (Barnes et al., 2012; Lease et al., 2016).

Sample ID	Sample #	Latitude	Longitude	Elevation [m]	Fm age	AFT Age [Ma] 2σ error	ZHe Age [Ma]
AL1	B815-5	-18.49	-68.67	3950	Camb	83.5 ± 20.8[1]	
AL2	B815-1	-18.48	-68.67	4144	Camb	43.2 ± 13.6[1]	
EC1	05JBBL070	-18.00	-66.95	3874	Sil	17.7 ± 21.2[1]	
EC2	05JBBL071	-17.88	-67.02	3684	Sil	40.3 ± 5[1]	
EC3	05JBBL067	-17.71	-66.66	3998	Sil	43.0 ± 4.8[1]	
EC3	Bol10-201	-18.26	-66.17	4150	O	36.1 ± 5.6 [2]	
EC4	B611-5	-17.72	-66.61	3850	Dv	68.9 ± 9.8[1]	
EC5	66-7	-17.69	-66.51	3600	Ord	28.0 ± 5.6[1]	
EC6	05JBBL065	-17.66	-66.45	3559	Dv	57.1 ± 7.8[1]	
EC7	05JBBL064	-17.66	-66.43	3209	Sil/Ord	20.7 ± 4.1[1]	
EC8	05JBBL062	-17.60	-66.36	2761	Ord	7.3 ± 4.8[1]	
IA1	05JBBL060	-17.23	-65.89	3154	Ord	0.5 ± 1[1]	47.1 ± 11[3]
IA2	05JBBL059	-17.17	-65.90	2772	Ord	126 ± 100[1]	MR[3]
IA3	05JBBL058	-17.19	-65.82	1787	Sil/Ord	8.1 ± 3.6[1]	21.0 ± 5.3[3]
SA1	05JBBL056	-17.16	-65.74	1762	Ord	5.6 ± 2[1]	PR[3]

SA2	05JBBL055	-17.10	-65.68	882	Ord	4.1 ± 1.4[1]	MR[3]
SA3	05JBBL054	-17.06	-65.65	611	Camb	1.8 ± 2.6[1]	8.9 ± 1.2[3]
SA4	05JBBL052	-17.02	-65.55	410	Sil/Ord	10.0 ± 1.4[1]	

Table 2.1. Thermochronologic data used in this study.

Sample IDs correspond to Figure 1. All AFT ages are pooled ages. P(χ^2)>5% and concordant. P(χ^2)<5% discordant and may not represent one geologic event. MR= Mixed reset; PR= Partial Reset. ***Sources: [1]Barnes et al., 2012; [2]Eichelberger et al., 2013; [3]Lease et al., 2016.***

2.2.4 Regional Sedimentology

Altiplano. The sedimentary section preserved in the AP records the transition from pre-Andean sedimentation to an early foreland basin (Horton et al., 2001). This includes the ~200-600 m thick El Molino Formation (Fm.), a regionally extensive marginal marine sequence dated at ~72 Ma that marks the end of marine conditions in the AP (Horton et al., 2001; Thierry Sempere et al., 1997). This is topped by a mid-Paleocene, eastward-sourced 50-300 m thick Santa Lucia Fm. overlain by 20-100 m of Potoco paleosols indicative of 15-20 Myr of reduced (<10 mm/yr) sediment accumulations during mid-Paleocene to middle Eocene (Horton et al., 2001). The paleosols are overlain by an upper Eocene through Oligocene phase of rapid fluvial aggradation (sedimentation rates up to 500 m/Myr) of the ~3000-6500 m thick Potoco Fm (Horton et al., 2001). This transition is interpreted as foredeep migration over initial forebulge deposits. The Potoco Fm. is broken into the westward-sourced ~3500-4000 m thick, upper Eocene, Lower Potoco Fm., and the poorly-dated, eastern- and western -derived ~2500 m thick Upper Potoco Fm., dated at ~23-24 Ma via K-Ar/Ar on biotite (Horton et al., 2001; Kennan, Lamb, & Rundle, 1995), and thus requires active deformation and exhumation in the EC prior to this time (Horton et al., 2001; McQuarrie et al., 2005). However, 40-50 Ma cooling ages from the EC from 15-18°S argue that deformation migrated into the EC by 40 Ma at the latest (Barnes et al., 2012; Eichelberger et al., 2013; Gillis et al., 2006; McQuarrie, Barnes, et al., 2008; Rak et al., 2017) which should have provided a new sediment source for the Altiplano from the east (McQuarrie et al., 2005) that are not seen in the lower Potoco Fm (Horton et al., 2001). Preservation of that eastern derived material rests on the distance between the EC uplift and the modern Altiplano basin, as well as in the magnitude of sedimentary material that was recycled as the EC continues to shorten (Rak et al., 2017). The uppermost units of the section are the eastward-derived, early Miocene Coniri/ lower Totora Fms. (~1000 m thick) and up to 5000 m of volcanic-rich deposits of the upper Totora/ Crucero Fms., late Miocene through Quaternary in age(Horton et al., 2001; S. Lamb, 2011).

Eastern Cordillera. Sedimentology of the EC intermontane basins preserved in the Camargo, Incapampa, Torotoro, and Morochata synclines (located throughout the EC from 21.5°S to 17.5°S, Figure 2.1c) is also interpreted to record the transition from backbulge to forebulge to foredeep (Horton, 2005). These rocks include the Paleocene to early Eocene, eastward-sourced Santa Lucia Fm. topped by ~80 m thick paleosols of the Early Eocene Impora Fm., the overlying eastward-sourced Cayara Fm., and the westward-sourced, >2 km thick Camargo Fm. with a clear EC source (DeCelles & Horton, 2003; Horton, 2005). The interpreted depositional environments require that a proto-FTB, initially active west of the AP, produced the backbulge and forebulge depozones identified in the AP and EC. Deformation jumped eastward in the Eocene, encapsulating the AP as a piggyback basin and provided the sediment source for the Camargo Fm. (DeCelles & Horton, 2003; Horton, 2005; McQuarrie et al., 2005). Deformation in the far-west EC is also recorded in the onlapping sedimentary basins in the Lago Poopo (18°S) and Salla (17.5°S) regions. These depocenters contain synorogenic sediments sitting directly on Silurian-Devonian age rocks. The synorogenic sedimentary rocks show growth strata and are dated at 28

Ma (Salla; (Gillis et al., 2006; A. L. Leier, McQuarrie, Horton, & Gehrels, 2010)) and ~25 Ma (Lago Poopo; (Simon Lamb & Hoke, 1997)) requiring erosion of Devonian and younger rocks prior to 25 Ma (A. Leier, McQuarrie, Garzione, & Eiler, 2013; McQuarrie, 2002) and modest amounts of shortening post 29-25 Ma. The flexural kinematic modelling shown in this paper predicts both magnitude of erosion and flexural basin formation for each increment of modelled shortening that we can directly compare to measured cooling ages and basin stratigraphy.

2.3 Approach/Methods

In order to quantitatively link the geometry and kinematics of deformation with the associated sedimentation and thermal histories to derive the age and signal of exhumation in this region, we created a sequentially deformed, flexurally loaded, forward modelled cross-section, as well as thermo-kinematic models for four different kinematic variations.

2.3.1 Kinematic Modelling

McQuarrie (2002) published a balanced cross-section through central Bolivia at 18°S. Using the 2D Kinematic Module in the modelling software Move (Midland Valley), the restored section was deformed sequentially using the Fault Parallel Flow Algorithm and using the passive wedge option for emplacement of the basement thrust sheets underneath the EC backthrust belt. Exact amounts of shortening on each fault are modified to best match the geometry of the structures using the imposed algorithms and differ slightly from the balanced cross-section (in McQuarrie,

2002) due to the limited ability of the software to precisely replicate structures and deformation processes documented in the region (e.g. McQuarrie and Davis, 2002).

2.3.2 Sequential Deformation and Isostasy

2.3.2.1 Model Setup

To replicate proposed foreland basin geometry and deposition in the AP and EC, we used *Move2015.2* (Midland Valley) to model an initial scenario wherein 200 km of shortening in the WC produced a migrating flexural basin. Modelled accommodation space created by flexure permitted 1.1-3.2 km of back bulge, forebulge, and initial foredeep deposition in the AP, (representing the El Molino, Santa Lucia, and some of the basal Potoco), and ~1.3 km of backbulge and forebulge deposition in the eastern EC, (the Santa Lucia and Impora formations) (Figure 2.2), similar to the method in (Rak et al., 2017). Because we evaluate a suite of different ages (40, 45, 50 Ma) for initiation of EC deformation, the initial deposition of the westward-derived ~ 40 Ma Potoco Formation (Horton et al., 2001) could predate (models that start <40 Ma), coincide with (model start of ~40 Ma), or postdate (models that start >40 Ma) early deformation in the EC. This beginning scenario is not thermally modelled and is used simply to create initial model conditions that are consistent with the proposed locations and magnitude of forebulge and backbulge depozones that are interpreted to precede deformation in the EC (DeCelles & Horton, 2003; McQuarrie et al., 2005).

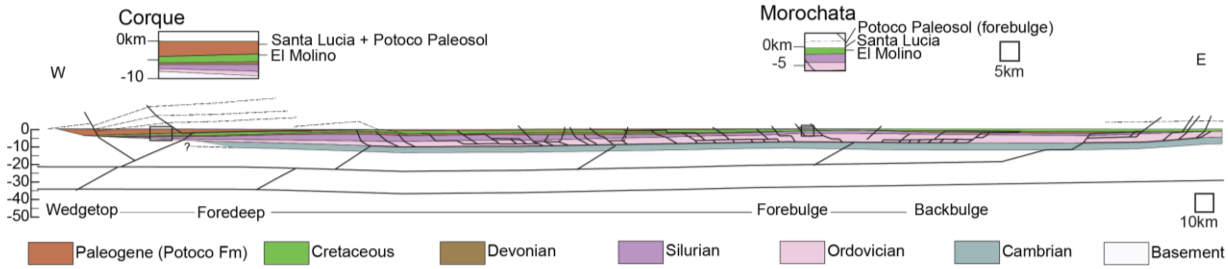


Figure 2.2. Initial *Move* model setup

Showing initial flexural state of lithosphere and sediment thicknesses at initiation of model. No vertical exaggeration. Stratigraphy insets at 2 x Cross-Section Scale.

Our methods for sequential deformation and isostasy build on previous work (Gilmore et al., 2018; McQuarrie & Ehlers, 2015; Rak et al., 2017) and was initially accomplished in ~20 km deformational increments (Figure 2.3). Following each deformation step, the flexural-isostatic load is calculated from the difference between the deformed topography and the previously undeformed topographic surface using the *Move2015.2* 2D Decompaction module that employs a bulk density, and a spatially uniform effective elastic thickness (EET) (Figure 2.3c, Table 2.2). For thrust loading, the bulk density is assigned to the space that defines the load (difference between the deformed topography and the previously undeformed topographic surface) and is hereafter referred to as the load density (ρ_{load}).The equations used in *Move* to model the flexural response follow Turcotte and Schubert (1982) to compute the deflection of the lithosphere caused by the load. Flexural-kinematic modelling is an iterative process, and EET, load and sediment density, and erosion angle (typically between 1 and 3°) are varied systematically for a suite of models to optimize the fit of the final model to the observed surface geology, foreland basin depth, sedimentary history, and surface thermochronology (Gilmore et al., 2018; McQuarrie & Ehlers, 2015, 2017b; Rak et al., 2017). A flexural basin in both the hinterland and foreland is created in response to this loading. The previous elevation horizon, representing the erosional or depositional surface across the model, subsides in these basins in response to the isostatic load. These horizons create a modelled stratigraphy though time. A new eroded surface was modelled following critical taper theory, such that the topography was eroded at a specified angle, α, from the deformation front. The new topographic profile followed the existing topography below the westward-increasing angle while any rocks exposed above the new topographic profile were eroded (Figure 2.3d). Regions that subside below 0 km are filled with sediments up to 0 km. Sedimentation does not occur above 0 km with the exception of the AP, where sedimentation is allowed in topographic lows above sea level, of up to 0.5 km of sediment accumulation per step. The preservation and accumulation of sedimentation is traced throughout the model in these flexurally created basins and can be compared to chronostratigraphic constraints. Elevations are restricted to 6.5 km in the eastern EC, 5 km in the western EC, and 4 km in the AP based on the upper limits of the modern landscape. Westward facing slopes were allowed to increase to 45° when structural deformation locally rotated the topographic slope. Isostatic unloading of eroded material was calculated following the same algorithm, density, and elastic thickness as the loading step, which typically results in ~0.1-0.5 km of additional erosion (Figure 2.3e) due to isostatic rebound. In order to preserve strata in the western EC (backthrust belt), the calculated topographic surface was preserved and was not eroded in the post-isostatic unloading step. This is consistent with a dry, less erosive climate. Sediment loading was also included to account for the accumulation of sediments in foreland and hinterland flexural basins. Using a sediment-appropriate density, calculations are performed with the same elastic thickness to calculate the load associated with the new basin fill (Figure 2.3f; Table 2.2).

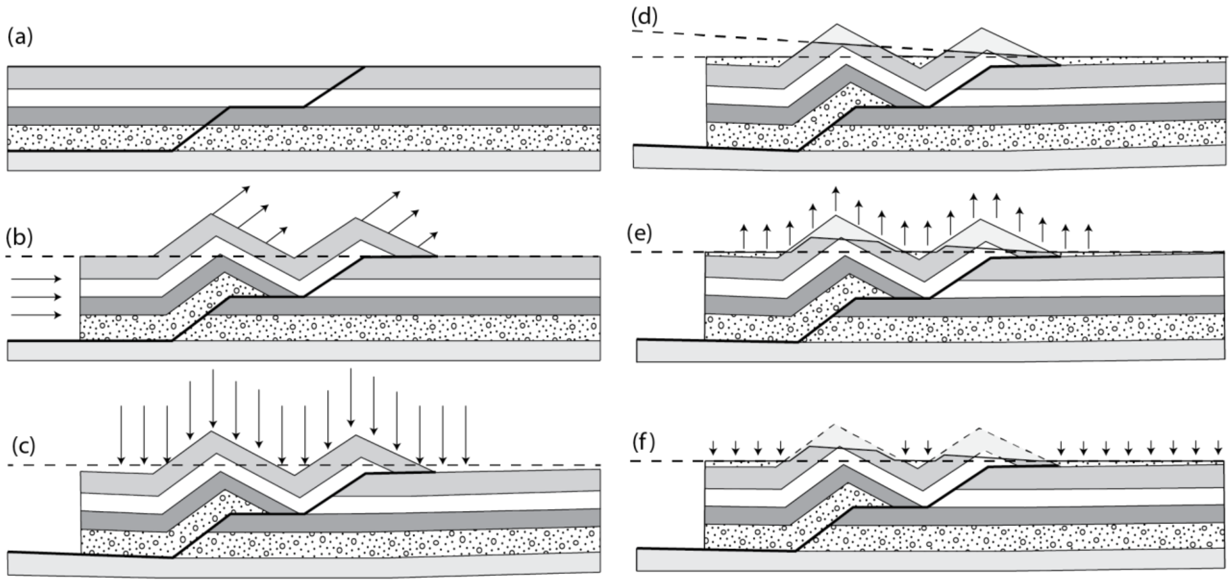


Figure 2.3. Schematic of kinematic-flexural modelling steps

(a) undeformed state, (b) deformation along fault, (c) loading due to isostasy, (d) erosion at critical angle, (e) erosional unloading, (f) loading due to sedimenation, and (g) general shape of imposed uplift and subsidence.

The synorogenic sedimentation modelled by the process described above created a basin that is 10 km thicker than that preserved in the Altiplano (i.e. overfilled) with a topographic surface that remained at 0 km elevation throughout the model. Additionally, deformation -induced loading throughout the model process described above does not produce the necessary subsidence of the foreland basin, resulting in basin that was ~1 km too shallow. To correct AP overfill and SA underfill, imposed uplift (over the AP) and subsidence (over the SA) was incorporated by applying a long-wavelength sinusoidal curve and unfolding the section to this new shape, with maximum subsidence occurring under the SA and maximum uplift occurring under the eastern edge of the AP in order to match modern EC elevation and foreland basin depths (Figure 2.11f) (Rak et al., 2017). This adjustment is required because the *Move* model isostatically accounts for only thrust and sediment loading and erosional unloading. It does not account for the accumulation of mid- to lower-crustal thickness inherently associated with the modelled upper-crustal shortening. A range of both documented and inferred geodynamic processes in the Andes affect accumulation of crustal rock (and thus thicknesses), accumulation of lithosphere, and surface elevations through time and these cannot be flexurally modelled. Dynamic subsidence related to viscous coupling of the mantle wedge can increase the foreland basin load, thus increasing the accommodation space available, particularly in cordilleran orogens (Catuneanu, 2004; DeCelles, 2012; Gurnis, 1993; Mitrovica, Beaumont, & Jarvis, 1989; Rak et al., 2017). Uplift related to mantle delamination has been invoked to account for rapid elevation change of the Andean Plateau(Garzione et al., 2006), and arguments against flexural support of the plateau, such as Airy isostasy due to a thick crustal column (Beck et al., 1996) and thickening of the EC and AP due to lower crustal flow from east to west (Eichelberger et al., 2015; Isacks, 1988; S. Lamb, 2011), have been proposed as necessary to maintain AP and EC elevations in the absence of active structurally induced uplift (Rak et al., 2017).

Parameter		Value
ρ_{load}		2500 kg/m^3 *
ρ_{sed}		2100 kg/m^3
ρ_{mantle}		3300 kg/m^3
Young's Modulus		70 GPa
Shortening/step		~6 km
EET during deformation in:	E. EC/ IAZ	40 km
	W. EC	15-45 km
	AP	100 km
	AP + SA	100 km
	SA	100 km
Imposed uplift and subsidence	under AP	+7.2 km
	under IAZ	-2.7 km
	under SA	-1.3 km
* on steps with only AP loading, ρ = 2300 kg/m^3 was used		

Table 2.2. Move model parameters for all kinematic variations tested.

2.3.3 Thermal Modelling

A 0.5 km x 0.5 km grid of unique points was placed over the extent of the undeformed flexural-kinematic model (following McQuarrie and Ehlers, 2015). The grid extends to the bottom of the basement thrust décollement and above 0 km in locations where basins form. The grid extends a total of 780 km, including 65 km beyond the cross-section extent at the eastern and western edges to ensure that the thermal model boundary conditions do not influence the thermal gradient near sample locations. The grid was deformed following the same steps as the final successful flexural model, but with ~10 km sequential deformation steps (to more accurately capture the behavior and evolution of loading, unloading, and the kinematics of the fault geometry). The grid and surface topography were exported at each step. These deformed grids produce vectors of displacement at each grid point that are converted to velocity fields by differencing the locations and assigning an age at each step. These velocities and topographies, in combination with thermal parameters, were input into a modified version of the thermal advection-diffusion software *Pecube* (Braun, 2002, 2003; McQuarrie & Ehlers, 2015; D. M. Whipp, Ehlers, Braun, & Spath, 2009)(Table 2.3). This modified version of *Pecube* solves the three-dimensional heat transport equation to simulate the evolving crustal thermal field based on the input thermal parameters and velocity fields to derive the time-temperature (t-T) history of exhumed rocks based on their transport paths (McQuarrie & Ehlers, 2015; Rak et al., 2017). Model-predicted ages at the surface for individual thermochronometer systems uses thermochronometer kinetics described in Ehlers et al. (2005) and Braun (2003). Measured AFT and ZHe ages, and the associated HeFTy models, were not used as an input to *Pecube* thermal models. The predicted cooling ages from Pecube are compared directly to measured ages. Matches between measured and modelled ages were identified if the modelled age, with a ±1 Ma and ±2 km error, fell within any portion of the measured age and its error. The match of measured to modelled cooling ages are used to constrain which thermal and velocity parameters provide the best fit.

Parameter	Input Value
Crustal Volumetric Heat Production (A_o)	0.5-4.0 $\mu W/m^3$
e-folding depth (Ef)	0, 12, 15 km
Thermal Conductivity	2.5 W/m/K
Specific Heat	800 J/kg/K
Model Base	110 km
Temperature at base	1300°C
Temperature at surface	23°C
Atmospheric Lapse Rate	5.3°C/km
Kinematic grid spacing	0.5 km x 0.5 km
Displacement Increment	~8-10 km
Model Domain	780 km x 110 km x 5 km
Horizontal node spacing	0.5 km
Vertical node spacing	1.0 km
Model start time (thermal initiation)	100 Ma

Table 2.3. Pecube thermokinematic modelling properties.

The thermal model extends to a base depth of 110 km with a temperature of 1300°C, and up to the surface, where the temperature at sea level is 23°C (Santa Cruz yearly average), and decreases at 5.3°C/km, the mean lapse rate measured in Bolivia (Gonfiantini, Roche, Olivry, Fontes, & Maria, 2001). The model holds the temperature at the surface and base constant. The thermal model is permitted 50 Myr to equilibrate crustal temperatures prior to initiation of Andean shortening. We tested two different methods for modelling the crustal thermal profile; the first applies constant radiogenic heat production (A_o = 0.6-1.0 $\mu W/m^3$) to the entire crustal section, while the second applies a surface radiogenic heat production (A_o = 3.0-4.0 $\mu W/m^3$) that decreases exponentially with depth (e-folding depths of 12 or 15 km).

2.4 Results

2.4.1 Flexural-Kinematic Model

Using the kinematic sequence portrayed in McQuarrie 2002, we produced over 40 different flexural-kinematic models in which EET, erosion angle, density, kinematics, and geometry were varied. Initial models tested with space- and time-invariant EET (four models, a through d, with temporally uniform EETs = 25, 30, 40, and 50 km, respectively) and density (ρ_{load} = 2900 kg/m^3), without calculating the load of sediments filling the basin. Results from these flexural-kinematic models that produced a dramatic mismatch between observed data and model results are described below but not depicted in figures. Models a and b resulting in overfilling the AP (by ~10 km) and under-filling of the FB (only ~2 km thick). Models c and d had over-erosion of the Paleozoic sedimentary cover in the EC that occurred during the initial 10-30 km of basement thrust sheet motion, as well as under-filling of the SA basin (~3 km thick) and overfilling of the AP (by ~5-7 km). Thus, this kinematic sequence (from McQuarrie, 2002) was refined to split the second basement thrust sheet into two thrust sheets, to prevent over-erosion of the Paleozoic sedimentary cover. The location of the new basement footwall ramp was chosen such that the material overlying it had the deepest erosion level in the Paleozoic section. However, this new split basement only resolved over-erosion of the Paleozoic cover, and subsequent models with time-invariant EET still resulted in overfilling of the AP and underfilling of the SA. Thus, we modelled an evolving EET that started at low values (15-40 km, Table 2.2). at the initiation of the model, which gradually increased to 50 km throughout deformation in the backthrust belt. Once deformation in the AP initiates, an EET of 100 km is maintained throughout the model (the maximum in *Move2015.2*). The EET evolution presented here was necessary to accurately reproduce the cross-section and may be representative of changing lithospheric strength such as the transition from an initial weak, faulted lithosphere to one supported by the Brazilian craton.

We iteratively tested another series of flexural-kinematic models which evaluated a suite of temporally varying EET values as well as different load and sediment densities with the goal of matching the surface geology and basin depths (AP and SA), with the most critical constraint to not over erode strata that is currently exposed across the EC and replicating the modern AP and EC elevations. Flexural models with a split basement thrust sheet and time-varying EET still could not replicate the modern AP elevations, nor the SA basin depths. The modern AP elevations were unable to be replicated because Once active deformation in the AP ceases, there is no further mechanism of uplift, and the AP rapidly sinks to at or below 0 km elevation. Imposed uplift (500 m increments) was implemented across the AP and western EC to provide non-flexural support after AP deformation ceases. Imposed subsidence (100 m increments) provided SA accommodation space. The final successful models increase EET from 15 to 100 km, have a load density of 2500 kg/m^3, sediment density of 2100 kg/m^{3,} and a total of up to 7.2 km of imposed uplift under the AP and western EC and up to 1.3 km of imposed subsidence under the SA (Figure 2.11f, Table 2.2).

We then evaluated a series of kinematic scenarios with the goal of replicating the measured thermochronology data in addition to matching the mapped surface geology and measured basin depths (Figure 2.4); all kinematic models have the same EET, density, and imposed uplift and subsidence (Table 2.2, Figure 2.11f). All models have ~2 km of OOS motion in far-west EC (to match Lago Poopo erosion and sedimentation history). The models varied in their SA kinematics and include: (1) in-sequence deformation, (2) ~3 km of OOS on IAZ/SA boundary fault, (3 & 4)

~15 km of OOS motion on IAZ/SA boundary, that is accommodated by two different sequences of faulting.

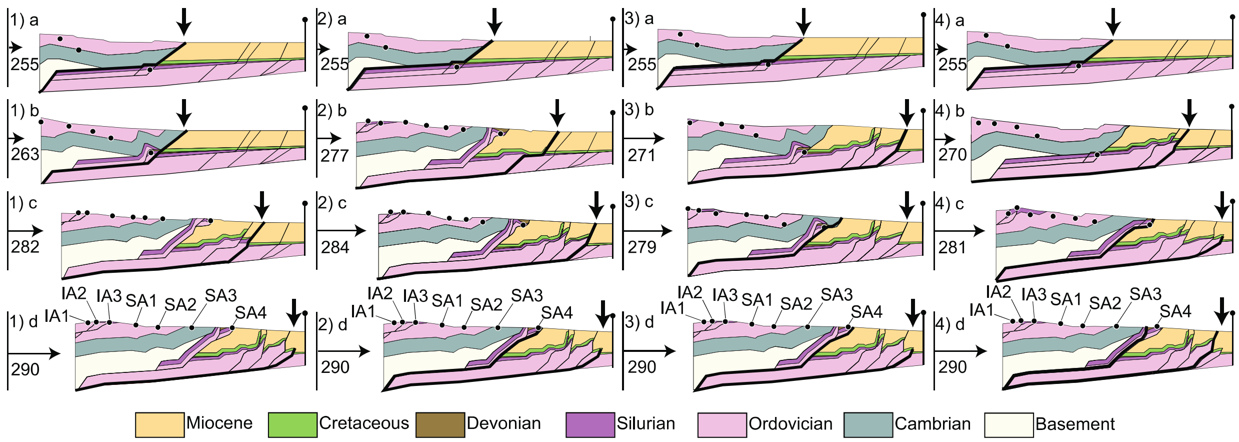


Figure 2.4. Kinematic variations tested

(i) a-d Model (1) – in sequence, (ii) a-d Model (2)- some OOS motion on IAZ/SA boundary, (iii) a-d Model (3) – on IAZ/SA boundary fault, (iv) a-d Model (4) – Same amount of OOS as Model (3) but on different IAZ/SA boundary fault. (a-d) Vertical arrow tracks deformation front through time. Bolded fault is most recently active. Black circles are thermochronology sample locations through time. Straight hatched units are Cambrian/Precambrian, dotted units are Tertiary synorogenic foreland basin sediments, white units are Silurian through Cretaceous. Large Numbers next to arrows on left indicate total amount of shortening [km]. (d) Measured ages [Ma] of AFT (top), and ZHe (bottom, italics).

2.4.2 Modelled Cooling Ages

2.4.2.1 Sequential Deformation and Development of Modelled Thermochronology Ages

In order to understand the effect of geometry on modelled thermochronology ages, we model the evolution of the FTB and predicted cooling ages through time. For samples with sufficient burial temperatures, the earliest possible cooling age is a function of when structures initiate, elevate topography, and facilitate exhumation. If burial was not enough to fully reset the sample, then the partially reset age will be between the age of the detrital sample and the age of exhumation that accompanies motion on a given structure. The initial model uses a constant rate of shortening, approximately 5.7 mm/yr, to accommodate 287 km of shortening since 50 Ma (Figure 2.5). This allows for the initiation of deformation to predate the oldest fully reset cooling ages in the EC. The model runs from 100 Ma to present and allows an initial 50 Myr for the thermal model to equilibrate crustal temperatures; at the time deformation initiates (t = 50 Ma, Figure 2.5a), all thermochronometers are 50 Ma in age. Samples that have already cooled through the closure temperature age with the passage of model time, such that the unreset AFT ages, present at the surface at t = 42 Ma, are ~58 Ma (Figure 2.5b). At t = 42 Ma, shortening is accommodated from west to east through a series of décollement levels. First, exhumation in the WC is driven by uplift over a basement ramp 0 near ~650 km (Figure 2.5b); this basement ramp in the WC drives loading in the AP and provides a source of uplift and exhumation to produce western-derived sediments found in the AP. Moving eastward, slip is transferred on a décollement approximately ~20 km in depth to the next basement ramp, where the basement thrust sheet 1 drives exhumation of the overlying cover and exposes reset predicted AFT ages over the basement ramp (470-490 km, Figure 2.5b). The modelled reset ages form a characteristic U-shaped cooling pattern, with the youngest ages pinned at the ramp (~0.5 Ma at ~490 km) and gently increasing in age in the direction of transport to the tip of the hangingwall ramp (~5 Ma at ~470 km), where there is a break in age and the modelled ages at the surface are no longer reset (Figure 2.5b) (McQuarrie & Ehlers, 2015, 2017b; Rak et al., 2017). Subsidence due to isostatic loading forms a flexural basin in front of and behind basement thrust sheet 1 (~440 km, ~510 km), that fills with accumulated sediments and buries the far-western EC (Figure 2.5b). Slip is transferred further eastward on a décollement at the base of the Ordovician section, approximately 10 km in depth. Duplexing of the Silurian age rocks and shortening in the upper Paleozoic section and the associated exhumation that accompanies it exposes partially reset AFT from ~110-160 km (Figure 2.5b). As each individual fault only has a small amount of motion on it (~2 km), there is not enough deformation to induce exhumation of fully reset AFT ages; the youngest partial reset age exposed (~38 Ma at 160 km) is on the hinterland (west) side of the duplex, as this has the deepest exhumation, and tapers to an unreset age of ~60 Ma at 110 km (Figure 2.5b).

At t = 27 Ma (Figure 2.5c), uplift and exhumation in the WC is ongoing due to active basement ramp 00, which continues to provide sediments to the AP. Slip along the basement décollement has propagated further east to the next basement ramp where emplacement of basement thrust sheet 2 and associated erosion exhumes fully reset AFT ages in a broad-wavelength, U-shaped cooling pattern. The youngest AFT ages, ~0.5 Ma, are pinned at the ramp at ~360 km, and gently increase in age eastward to the tip of the hangingwall ramp to ~20 Ma, near ~275 km (Figure 2.5c). The locus of deformation, and thus isostasy, jumped forward with the emplacement of basement thrust 2. The rocks initially overlying this ramp experienced the highest amount of exhumation in the Paleozoic cover, eroding Cretaceous through upper Silurian rocks. The deepest exhumation is seen in the reset ZHe cooling signal, which is ~22 Ma near the tip of the hangingwall ramp (~275 km), due to erosional exhumation over the basement ramp, and at ~360 km, driven by both the basement ramp uplift and OOS motion in the far-west EC Paleozoic section (Figure 2.5c). As motion continues over this basement ramp, flexural subsidence increases, deepening the décollement and creating flexural basins in front of and behind the ramp, promoting AP and foreland basin sedimentation all the way into the IAZ and burying the previously deformed and exhumed strata near ~110-160 km . This flexural subsidence dampens the amount of subsequent erosional exhumation over the ramp, limiting the extent of reset ZHe ages to only above the hangingwall and footwall ramps of basement thrust sheet 2 (250-350 km) (Figure 2.5c). In front of basement sheet 2, westward-propagating shortening in the Paleozoic section provides a mix of partial reset and unreset ages driven by small amounts of deformation on individual faults.

At t = 24 Ma (Figure 2.5d), uplift in the WC due to basement ramp 00 is ongoing, continuing to provide a western sediment source for the AP. Basement thrust sheets 1 and 2 are no longer actively deforming, as deformation propagated eastward again with the emplacement of basement thrust sheet 3, producing reset AFT ages between 200-220 km (Figure 2.5d). Because rocks in this region had already undergone deformation-induced exhumation which exposed partially reset AFT ages (Figure 2.5c, ~210 km), the AFT signal due to emplacement of basement thrust sheet 3 has a broad-wavelength pattern driven by the basement ramp, with narrower wavelength patterns set by previous shortening and exhumation focused in Paleozoic age rocks. This creates a mix of fully (~10 Ma) and partially reset AFT (~40 Ma) and ZHe (~50-70 Ma) ages at t = 24 Ma (Figure 2.5d, 200-220 km). Exhumation induced by westward-propagating deformation in Paleozoic rocks above the hanging wall of basement thrust 2 predicts a reset AFT signal and partially reset ZHe signal between 240-250 km (Figure 2.5d). The predicted ZHe and AFT signals between 250-310 km experience no additional exhumation and thus age with model time.

At t = 19Ma (Figure 2.5e), the predicted AFT signal due to exhumation induced by the emplacement of basement thrust sheet 1 is dampened by onlapping sedimentation in the AP (320-330 km, Figure 2.5e). Shortening in the westernmost EC drives erosional exhumation of ~3 km of synorogenic sediments and the model predicts partially reset (~40 Ma) AFT ages at 315-320 km (Figure 2.5e). Westward-propagating shortening in the Paleozoic cover overprints and narrows the signal from the emplacement of basement thrust sheet 2, resulting in a mix of partially reset (~50-

60 Ma) and fully reset (~10-20 Ma) AFT ages between 310 and 260 km (Figure 2.5e). This westward-propagation displaces the thin-skinned cover to the west, shifting the signal of ages reset due to emplacement of basement thrust sheet 3 to the west. The final emplacement of basement thrust sheet 3 drives exhumation in the previously deformed overlying Paleozoic cover between 200-250 km, predicting younger populations of partially (~35-40 Ma) and mostly unreset (~70 Ma) AFT ages (Figure 2.5e). The ZHe signal across the entire section remains unchanged, as deformation in this step is insufficient to drive exhumation deep enough to expose rocks with reset ZHe ages.

At t = 10 Ma (Figure 2.5), shortening in AP, first on the east and then west side of the Corque syncline, drives erosion and exhumes reset AFT and ZHe ages (385-375 and 330-320 km, respectively). Deformation in the EC has ceased, and thus the modelled thermochronology signal present in the EC in Figure 2.5e is increasing in age. At t = 10 Ma, emplacement of basement thrust 4 has been ongoing for ~ 4 Myr, and uplift over the SA ramp at ~70 km exhumes reset AFT ages between 60 and 80 km, with the youngest ages (~0.5 Ma) at the ramp at ~80 km, increasing gently in age to ~4 Ma at 60 km (Figure 2.5f). The predicted ZHe signal is partially reset to 40 Ma in the center of the U-shaped AFT pattern where the erosion level is the deepest at ~70 km (Figure 2.5f).

At t = 3Ma (Figure 2.5g), continued emplacement of basement thrust sheet 4 over ramps in basement (~100 km), Paleozoic (~80 km), and Tertiary (~40 km) strata exhumes the reset AFT signal between 90 and 38 km. The AFT signal is reset between 90 and ~45 km due to the ramp through Cambrian and Ordovician rocks at 90 km in a westward-younging U-shaped pattern, with the youngest ages (~0.5 Ma) increasing gently to ~3 Ma at ~45 km (Figure 2.5g). The easternmost SA ramp, located at ~40 km, reset AFT to ~0.5 Ma at 30-40 km. The ZHe signal is reset between 90 and ~45 km due to deformation over the SA ramp at ~90 km inducing the deepest exhumation levels in the Paleozoic strata (Figure 2.5g). The break between predicted fully reset (~0.5 Ma) and unreset (>80 Ma) ages is pinned at 30 km by the surface breaking fault.

At t = 0 Ma (Figure 2.5h), the final 17 km of shortening in the SA brings thrust sheet 3 up and over the active basement ramp at ~120 km, which drives additional exhumation in the Paleozoic cover, resetting the AFT signal to ~20 Ma at 100-110 km (Figure 2.5h). Shortening in the SA and continued emplacement of thrust sheet 4 exhumes the fully reset ZHe signal between 50 and 75 km to ~3-6 Ma (Figure 2.5h). The final SA shortening, within the Tertiary foreland basin, does not produce enough exhumation to expose reset ZHe ages, and only exhumes partially reset AFT ages at ~14 km to ~42 Ma because of limited displacement on individual faults (Figure 2.5h).

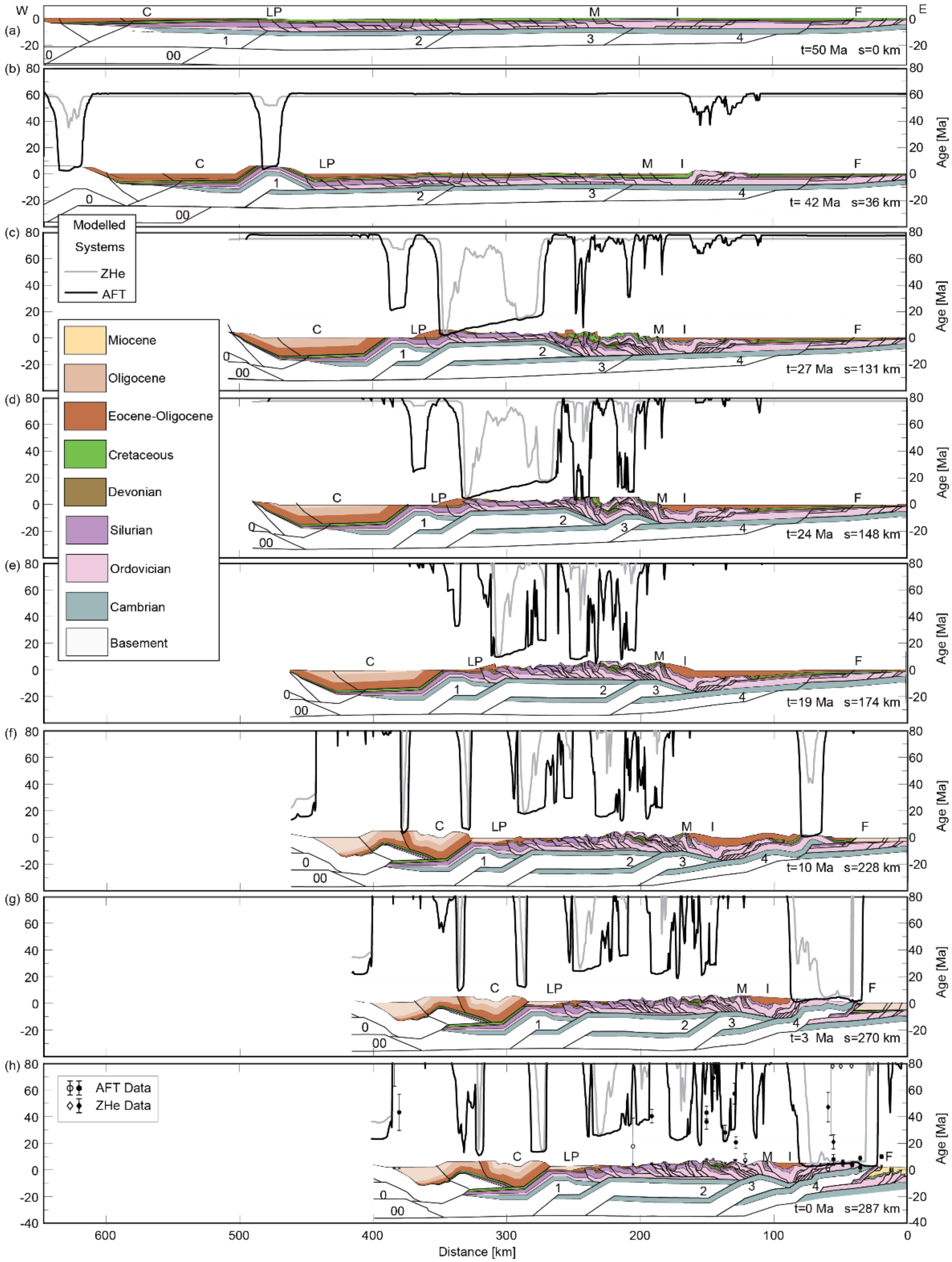


Figure 2.5. Structural and thermochronologic evolution in the central Bolivian FTB

(Figure 2.5 (continued)) using model (1) kinematics. The (a) restored cross-section is (b)-(h) sequentially deformed with the top panel displaying the modelled thermochronology (AFT & ZHe) and the bottom panel showing the geometry of the deformed model. Sedimentary basins: Corque (C), Lago Poopo (LP), Morochata (M), Incapampa (I), Foreland (F).

2.4.2.2 Kinematic Variation and Modelled Thermochronology Ages

We evaluated the effect of in-sequence and out-of-sequence (OOS) faulting on the predicted ages in the SA and IAZ by testing four different kinematic models, while holding s focused over the fourth SA fault (at 12 km) with predicted ages as young as 45 Ma, with less exhumation and older ages (80-90 Ma) over the second and third SA faults (2.5 and 9.5 km, respectively). In-sequence deformation (Model 1) predicts the youngest ages in this region, with similar ages predicted by Model 2 which has limited (3 km) of OOS motion. The larger amount of OOS motion in Models 3 and 4 post-date all but 1.9 km of motion on the frontal faults. The shift from active exhumation to subsidence in front of the OOS fault depresses the predicted AFT ages. At 19 km from the deformation front in Model 4 and 22 km in Models 1-3, the AFT signal shifts from unreset (~100 Ma) to fully reset (~0.5 Ma). These young predicted AFT ages extend to ~75-80 km from the deformation front (Figure 2.6). The westward shift in the location of partially to fully reset ages, at 19 km for model 4 and 22 km for models 1-3, is controlled by both the amount of OOS motion and the specific SA/IAZ boundary fault that is last active. Model 4 has 15 km of OOS motion, with the final motion of OOS on the fault that breaks the surface 20 km from the deformation front. This amount of deformation and accompanying erosion exposes fully reset AFT cooling age behind the fault, such that sample SA4 is in front of the fault on the transition to unreset AFT ages. Models 1-3 have this transition from unreset to fully reset near ~22 km because of no additional OOS motion on the fault at 20 km.

The modelled ZHe signal is completely unreset in the frontal ~20 km of the cross section but exhibits partially reset ages ~55-80 Ma (Figure 2.6a) over the proposed OOS fault. Model 4, with the largest and youngest component of OOS motion produces the youngest modelled cooling ages in this region. All four models predict young reset cooling ages of 8 Ma at ~30 km from the deformation front. The predicted ZHe ages are fully reset (~8-10 Ma) until ~45 km (Model 4), ~63 km (Model 3), ~66 km (Model 2), and 70 km (Model 1), where the predicted cooling ages increase to significantly older ages, with the exception of narrow excursions of partially reset ~55 Ma ages at 55 and 53 km (for Models 2 and 3). Model 4 produces a more complicated predicted pattern of cooling ages with younger ~20 Ma ages at ~55 km before predicted ages continue to irregularly increase to ~55 Ma at 70 km from the deformation front.

Though all four kinematic models result in the same surface geology and erosion level in the end, the paths and erosion level through time are different (Figure 2.4). This is particularly important for capturing the ZHe signal for samples IA1 and IA3. All four models are uplifted over ramps located at ~80, ~63, and ~30 km from the deformation front during SAZ deformation but the kinematic order controls which ramp is active when. Model 1 (in-sequence) predicts a smooth, fully reset ZHe signal set by the most recent shortening occurring only on the ramp at ~80 km, while Models 2 and 3 have a more complicated ZHe pattern imparted by recent exhumation over the ramp at ~62 km during OOS faulting (Figure 2.6). The kinematics of Model 4, specifically no motion on the fault that repeats the upper Ordovician and Silurian strata, limits erosion early on in the sequence (Figure 2.4(iv) c), with final uplift towards the end of the model in a combination that allows the modelled ZHe ages corresponding to samples IA1 and IA3 to rest in the partial retention zone resulting is a suite of mixed and partial reset ages. While each kinematic model produces a different pattern of predicted ZHe ages from 35-75 km, the strong uplift signal over the ramp at from 30-40 km reproduces the young 8 Ma ZHe age of sample SA3 at ~33 km in all four kinematic models. The predicted ZHe ages to the west (35-75 km) are unique, allowing us use to samples IA1 and IA3, located 58 and 52 km from the deformation front, respectively, to differentiate between the viability of the kinematic models (Figure 2.6a). From 70-80 km from the deformation front (Figure 2.6) the modelled AFT signal highlights the effect of varying the most recent active ramp locations (bolded lines, Figure 2.4 (i-iv) d). Model 1, which has in-sequence SA motion only, has the predicted young AFT signal pinned to near 80 km because this signal is driven by motion over the most recent ramp, which for Model 1 is located at ~80 km. However, Models 2-4 predict the young AFT signal near 75 km (Figure 2.6); this is because these models have OOS motion that continues up the SA ramp near 62 km (Figure 2.6), which overprints the signal from the ramp at ~80 km. Model 4 best captures the ZHe signals in samples IA1-SA3 (Figure 2.6a), and is the only kinematic model to accurately represent AFT sample SA4.

Eastern Cordillera. For all four of the proposed kinematic variations, the deformation in the EC is exactly the same, yet the predicted cooling ages in the EC have subtle variations due to different kinematics of IAZ and SAZ deformation that postdate deformation and exhumation in the EC. The EC has a low total amount of exhumation, making it particularly sensitive to small variations in erosion. This is highlighted by modelled ZHe ages that are never fully reset and only show slight resetting at 160-170 and ~190 km along the cross-section line. The youngest predicted ZHe ages are at 230 km, directly above the footwall ramp of basement thrust 3 (Figure 2.5h; Figure 2.6a). Individual faults that displace Paleozoic strata in the EC have small magnitudes of displacement and thus the primary exhumation is imparted by motion of basement structures and the associated erosion as illustrated in Figure 2.5.

Modelled reset EC AFT ages are concentrated in 3 bands from ~110-135, ~150-175, and ~190-230 km (Figure 2.6). The final modelled cooling ages in the eastern portion of the EC are a function of two periods of exhumation: initially as partially reset ages due to deformation-induced exhumation from shortening of the Paleozoic section (Figure 2.6c, 240-250 km), and then exhumed further with the emplacement of basement thrust sheet 3 (Figure 2.5d, e, 200-215 km). The central band of predicted reset ages was exhumed as west-vergent thrusting in Paleozoic rocks allowed for thrust propagation over the hanging wall ramp of basement thrust sheet 2 driving erosional exhumation (Figure 2.5d, e). The western most band was exhumed and cooled due to the motion of basement thrust sheet 2 over its footwall ramp concentrating the youngest reset AFT ages in this region. Because of this shared history, there are only minor differences in the predicted cooling ages between the 4 different kinematic models in the two western most populations of reset AFT ages in the EC. Larger variations in the predicted ages from the four different kinematic models are present in the eastern population of reset ages between 110 and 135 km from the deformation front. This region is located at the edge of the critical wedge of increasing topography in response to uplift of basement thrust sheet 4 and SA shortening, and the timing of erosion is controlled by the timing of propagation of the deformation front. The deformation front of model 1 (in-sequence) propagates forward slower than that of Model 4, resulting in older ages (Model 1: ~80 Ma vs Model 4: ~20 Ma) due to exhumation earlier in Model 1 between 110 and 135 km (Figure 2.4). The amount of OOS motion, and on which fault, controls the pattern of modelled AFT ages in the EC between 110 and 135 km. Specifically, SAZ OOS motion affects the age of the partially reset AFT system, and where that resetting occurs, due to the region’s sensitivity to variation in deformation front propagation and differences in loading associated with different kinematics. Modelled ZHe is unreset, or nearly unreset at >90 Ma, in all kinematic models across the entire EC (80-210 km).

Two regions have particularly limited exhumation: 80-100 km, where foreland basin sedimentary rocks are preserved, and 135-150 km, where there is preservation of nearly the full thickness of a Paleozoic thrust sheet. The sedimentary layers accumulated in a foredeep/wedgetop position of the growing FTB (160-180 km, Figure 2.5b-d), that experienced maximum subsidence during motion and loading from basement thrust 2 as well as initial loading from basement thrust 3. The partially reset AFT (40-80 Ma) ages at ~80-110 km (Figure 2.6) are due to both the preservation of sedimentation as well as lack of exhumation as the basin is never fully exhumed over basement thrust sheet 3 (160-180 km, Figure 2.5e), nor is the uplift over basement thrust sheet 4 enough to exhume the full thickness of synorogenic sediments (80-110km, Figure 2.5g). The basin preserved at 80-100 km in our model is not present at 18°S, however, similar basins preserving the Cretaceous through Tertiary sedimentary section exist further south (e.g., the Incapampa basin). Similarly, between 135-150 km, the clusters of partially reset ages (~60-80 Ma) were located in a region of low exhumation between basement thrust 2 and basement thrust 3 (Figure 2.5d) and was never displaced over the footwall ramp of basement thrust 3 (Figure 2.5d, 210-225 km) limiting exhumation. Additionally, these structures deformed early on in the EC backthrust sequence, and thus never had synorogenic sedimentation from Andean deformation deposited on top. This combination of lack of burial and lack of exhumation preserves nearly two full thicknesses of Paleozoic strata and leads to the cluster of partially reset ages (~60-80 Ma) between 135-150 km (Figure 2.6).

The model-predicted EC AFT signal has multiple broad wavelength patterns primarily controlled by the emplacement and eastward propagation of basement thrust sheets, as illustrated in Figure 2.5h and Figure 2.6. This signal is augmented by smaller wavelength patterns that are a function of individual thrusts in the Paleozoic section. However, due to the lower overall amount of exhumation, this signal is overprinted by time since exhumation. As Model 4 is best able to capture the ZHe measured ages in the SAZ and IAZ and is equally good at capturing the AFT ages throughout the model space, we use kinematic Model 4 in subsequent sections.

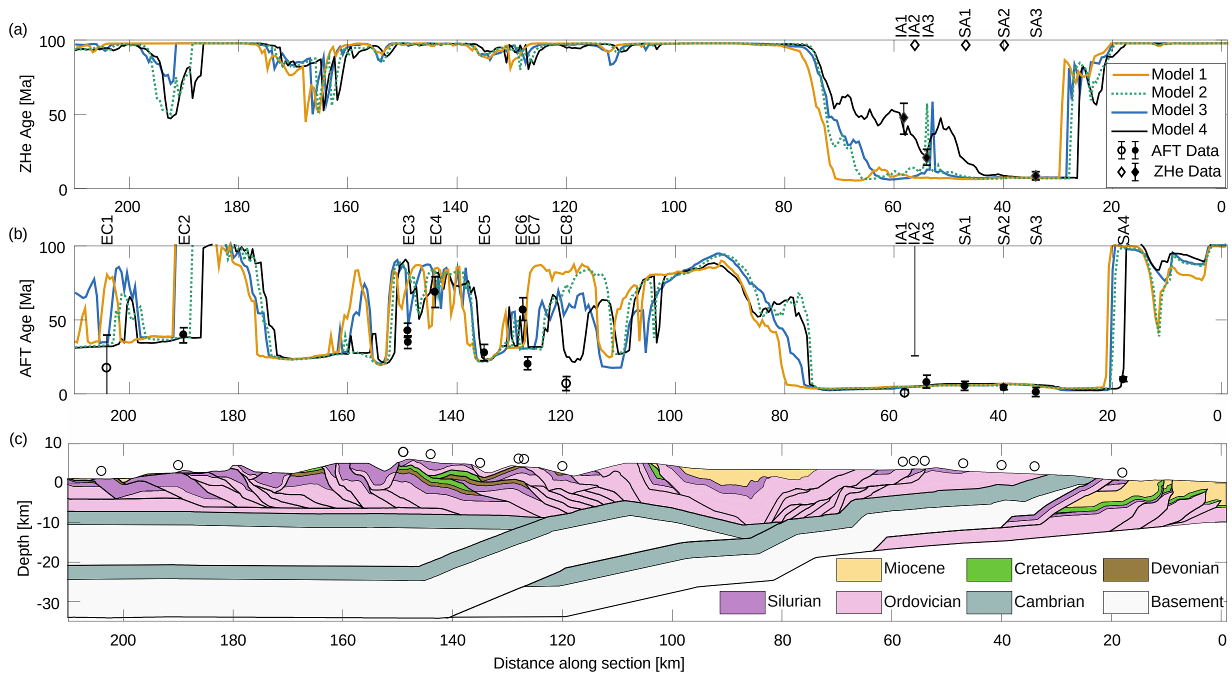


Figure 2.6. Effect of kinematics on thermal modelled ages

thermal structure and velocity are the same between models. (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages, (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section with no vertical exaggeration. Open circles indicate sample locations.

2.4.2.3 Effect of Radiogenic Heat Production on Modelled Thermochronology

The effect of radiogenic heat production on the predicted cooling ages was evaluated using kinematic Model 4, and by modelling two different estimates of how heat production changes with depth; the first has a constant radiogenic heat production ($A_o = 0.6\text{-}1.0\ \mu W/m^3$) through the entire crustal section, while the second applies a radiogenic heat production (A_o = 3.0-4.0 $\mu W/m^3$) that decreases exponentially with depth (e-folding depths of 12 or 15 km). Measured A_o values in the lower crust are difficult to obtain, and variation with in the lower crust is not negligible (Ashwal, Morgan, Kelley, & Percival, 1987; Jaupart & Mareschal, 2011; Ketcham, 1996). In the Andes, surface heat flow measurements change significantly from the foreland (~35-60 mW/m^2) to the AP (70-140 mW/m^2), with an average surface heat flow across the FTB between 60-80 mW/m^2 (Ehlers, 2005; Henry & Pollack, 1988; Jaupart & Mareschal, 2011; Mareschal & Jaupart, 2013). Previous studies have found that surface heat flow modelled by a constant A_o applied to the entire crust (A_o = 0.9 $\mu W/m^3$) or an exponentially decaying A_o (A_o = 3.0 $\mu W/m^3$, ef = 10 km) produce nearly identical surface heat flow of ~56 mW/m^2 (Springer, 1999). Similarly, thermal models that use an e-folding depth require a higher heat production value to produce the same modelled ages as a thermal model that applies a constant heat production value to the entire crust (Figure 2.7). Increasing A_o or e-folding depth (ef) in the crust raises geothermal gradients and results in younger modelled thermochronometer ages.

The modelled AFT pattern is most sensitive to changes in heat production in the EC (70-210 km, Figure 2.7b), while ZHe is most sensitive in the IAZ and SA (18-70 km). With the highest heat production (A_o = 1.0 $\mu W/m^3$) the AFT signal becomes fully reset (17-30 Ma) across the entire EC, except for 90-100 km due to basin preservation and 135-150 km due to the kinematics discussed in 4.2.2. (Figure 2.7b). Decreasing the heat production in the crust reduces the width of the cooling signal imparted by the basement thrust sheets, and instead allows visualization of the effect smaller scale structures have on cooling ages. Changes in heat production have the largest effect on the predicted cooling ages where the magnitude of exhumation is the least, such as between 70 and 160 km. Model-predicted ages become older with lower heat production as well as more variable (e.g. between 150 and 100 km, Figure 2.7). The cooling ages are predominantly partially reset (~40-80 Ma), with small wavelength fully reset ages (20-30 Ma) over individual Paleozoic structures (Figure 2.7b). This same effect on predicted cooling ages, where model-predicted ages become older with cooler heat production, is also present in models with an e-folding depth. The warmest of these models, $A_o = 3.0\ \mu W/m^3$, ef = 15 km, produces a more variable pattern, containing more partially reset ages, than that of the hottest model ($A_o = 1.0\ \mu W/m^3$). This warmest e-folding depth model similarly highlights exhumation on smaller scale structures in the Paleozoic rocks (like $A_o = 0.6\ \mu W/m^3$), though with younger partially reset ages. Reducing the heat production by decreasing the e-folding depth by 3 km, from 15 to 12 km, with $A_o = 3.0\ \mu W/m^3$, produces the same pattern of small-structure dependent AFT ages, but with the overall magnitude of ages shifted older by up to ~16 Ma (Figure 2.7b). A comparable result is seen by decreasing A_o, from 3.5 – 3.0 $\mu W/m^3$, with the same ef = 12 km, where ages are less reset via cooler temperatures in the crust (Figure 2.7). The coolest model without an e-folding depth, $A_o = 0.6\ \mu W/m^3$, produces similar modelled AFT ages, in both pattern and magnitude, as the coolest model with an e-folding depth, $A_o = 3.0\ \mu W/m^3$, ef = 12 km (Figure 2.7b).

For the ZHe system, the warmest modelled heat production, $A_o = 1.0\ \mu W/m^3$, fully resets the entire SA and IAZ predicted ages to ~7 Ma (Figure 2.7a), analogous to the predicted age pattern of Model 1 with in-sequence kinematics (Figure 2.7a). Decreasing the available heat in the crust produces increasingly older populations of partial reset ages in the IAZ and on the western edge of the SA (from 40-70 km). Decreasing the e-folding depth by 3 km, from 15 to 12 km, for the same $A_o = 3.0\ \mu W/m^3$, produces similar patterns of ZHe ages, with the overall magnitude of ages shifted older by up to ~34 Ma (Figure 2.7a). Similarly, decreasing A_o from 3.5 to 3.0 $\mu W/m^3$ with ef = 12 km increases the predicted ages to ~37 Ma, but retains the cooling age pattern prescribed by exhumation on individual structures.

As discussed in section 4.2.1, modelled ages are primarily controlled by the emplacement of basement thrust sheets, while changing heat production changes the magnitude of the ages and to which cooling signal the predicted ages are the most sensitive. High heat production values (e.g. $A_o = 1.0\ \mu W/m^3$) highlights the control that basement thrust sheets impart on the exhumation. As heat production values are reduced, second order features become amplified as only individual structures (1-5 km in width), that focus exhumation, predict reset ages, separated by narrow regions of partially reset ages.

Models with constant radiogenic heat production simplify the thermal regime by applying this input radiogenic heat value across the whole crustal column. However, as radiogenic minerals tend to be at higher concentrations in the upper crust than in the lower crust (Heier & Adams, 1965; Jaupart, Mareschal, & Iarotsky, 2016), it is reasonable to wonder if this simplification may affect cooling ages. Therefore, to test the effects of this variance in crustal thermal structure, we compare models of a constant heat production versus those that have an exponential decay from the surface radiogenic heat value. For modelled AFT ages at the surface, the difference between $A_o = 0.6\ \mu W/m^3$ with no e-folding depth and is negligible for an individual AFT chronometer (compare dashed to the two lighter grays, Figure 2.7b). Using an e-folding depth produces cooler A_o values with depth and thus the difference in the predicted ages using surface heat production values and an e-folding depth or a constant heat production value is seen in the ages of the deeper chronometers like ZHe. In our model, this is reflected as a greater span in ages between the predicted AFT and the predicted ZHe ages using an e-folding depth than at constant heat production values.

Cooler crustal temperatures can be produced by decreasing the e-folding depth or lowering A_o. This lower heat predicts older AFT and ZHe ages, which better capture AFT sample EC6 and ZHe samples IA1 and IA3. Additionally, at depth, the temperatures achieved with A_o=0.6 μW/m^3 and no e-folding depth are unrealistic for crustal temperatures. For example, the crustal temperatures for A_o = 0.6 μW/m^3 is ~720°C at 40 km depth under AP, while with an e-folding depth of 12 km and A_o =3.5 μW/m^3, the crust is only ~533°C (nearly 200°C cooler).

Some portions of the model are, to a first-order, insensitive to the small changes we evaluated in how heat in the crust is distributed; the modelled AFT ages do not change in the SA (between 20-70 km), or in the western half of the EC (150-220 km). AFT samples EC1, EC2, EC5, and all IAZ and SA samples do not differentiate between heat production values. Both EC3 AFT samples are only predicted by the warmest heat production values (A_o = 0.6 μW/m^3, and A_o = 3.5 μW/m^3, ef = 12 km within error). Only the coolest AFT models match EC6 (A_o = 3.0 μW/m^3, ef = 12 km; A_o = 0.6 μW/m^3), while none of the models are warm enough to predict ages that match EC8, though this may be due to an anomalously young measured age. The AFT measured ages, while not a strong distinction, are best fit by models with cooler heat production.

Modelled ZHe ages are unaffected in the frontal portion of the SA between 30-40 km (Figure 2.7). Because much of the EC has not exhumed enough to reset ZHe ages, these predicted ages are also insensitive to changes in heat production. However, from 40-70 km, the ZHe modelled ages are responsive enough to changes in radiogenic heat production that with increased heat production the sensitivity to the kinematics (Figure 2.6) is removed. ZHe sample SA3 does not distinguish between heat production, but samples IA1 and IA3 are only predicted in the coolest heat production values (A_o = 3.0 μW/m^3, ef = 12 km and A_o = 3.5 μW/m^3, ef = 12 km, respectively). Predicted AFT and ZHe modelled ages provide the best fit to measured thermochronology samples when using a thermal heat production slightly warmer than the lowest shown in Figure 2.7 therefore, $A_o = 3.2$ and an e-folding depth of 12 km is used in subsequent sections.

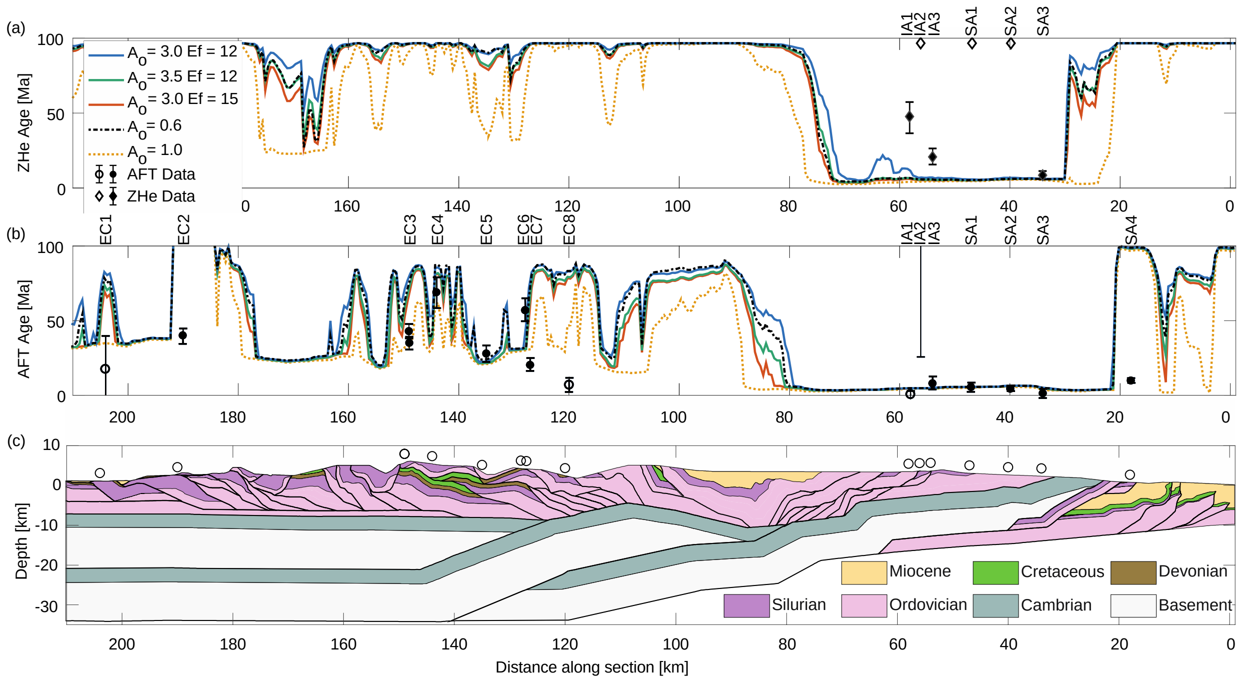


Figure 2.7. Effect of radiogenic heat production on thermal modelled ages for Model 4 kinematics and velocity are the same between thermal models. (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages, (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section 4.2.4.

2.4.2.4 Effect of Shortening Rate on Modelled Ages

We evaluated the effect of changing shortening rate on model-predicted thermochronology ages by analyzing a suite of velocity models, including constant velocity (*i*, ~5.8 mm/yr), SA fast deformation (*iii*, *vi-ix*: 6-10 mm/yr), a hiatus in EC deformation during 40-20 Ma (*ii*, ~4 mm/y), and variations on a rapid pulse of deformation (*iv*: 11.5 mm/yr, 6-2 Ma; *ii*, *x*: 8-12 mm/yr, 16-6 Ma; *v*: 10 mm/yr, 11-5 Ma, 4 mm/yr 5-0 Ma) on kinematic model 4 (Figure 2.4a, Table 2.4,Figure 2.16).The most important result is that velocity changes do not significantly alter the across-strike cooling pattern, as that is controlled by kinematics. Instead, velocity changes can shift the ages older or younger, broaden the suite of reset ages, flatten the slope of ages, and reduce (or amplify) the appearance of second-order patterns in cooling ages. In this section, we present three velocity frameworks to show the effect of shortening rate on modelled ages. The simplest velocity framework, constant deformation through all Andean deformation, is here compared to two variable velocity frameworks to test their effects on modelled ages, particularly during SA deformation. All three velocity frameworks shown in Figure 2.9 have the same initial deformation rate, ~5.8 mm/yr, at model start time. However, they differ in the shortening rate during SA deformation. Velocity *i* remains at a constant ~5.8 mm/yr during the entire SA deformation, which begins at ~10 Ma. Velocity *v* has an initial rapid (10 mm/yr) pulse of SA deformation from ~12 to ~10 Ma, and then decreases to 4 mm/yr from ~10 Ma till present. Velocity *viii* has a ~10 mm/yr shortening rate during the entire SA emplacement, which begins at ~6 Ma (Figure 2.9).

Subandes and Interandes. At the front of the SA, all velocity frameworks shown exhibit partially reset AFT ages (~75-87 Ma) due to deformation-induced exhumation consistent with Model 4 (Figure 2.9b, ~3-12 km). Both velocity frameworks with slow SA deformation, velocities *i* and *v*, show slightly younger partially reset ages across this region (~75-92 Ma, ~3-12 km) than the fast SA velocity model (*viii*) (~88-95 Ma). Between 20 and 75 km, all velocity frameworks produce fully reset AFT ages, with the youngest reset ages (2-4 Ma) corresponding to the fastest SA emplacement (velocity *viii*), and the oldest reset ages (~5-12 Ma) corresponding to the slowest SA emplacement (velocity *v*). Faster rates during entire SA deformation young the AFT cooling ages, both because the timing of deformation is younger and faster rates produce flatter cooling curves.

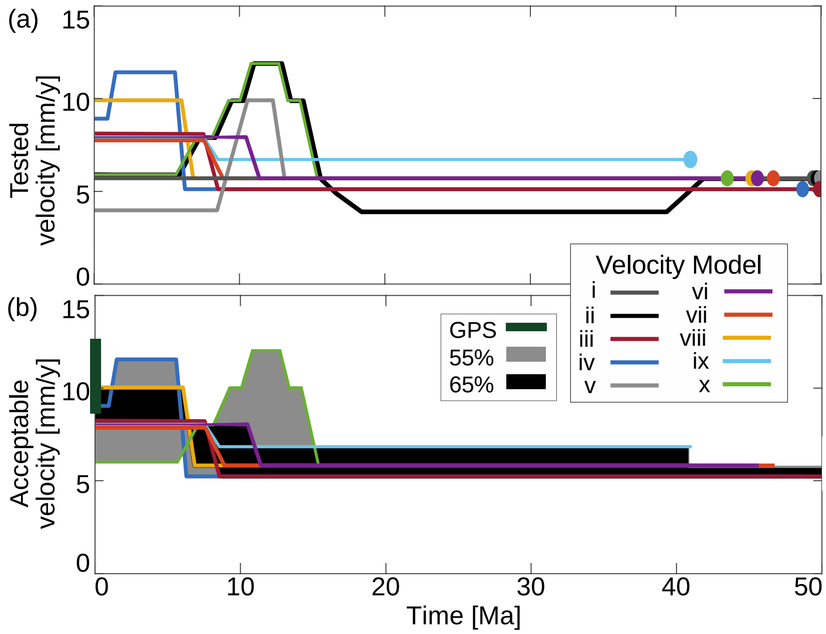


Figure 2.8. Shortening rate versus time

showing (a) suite of velocity models tested (corresponding to Table 2.4); closed dots indicate model start time. (b) acceptable velocity envelope for kinematic Model 4 defined by thermochronologic age match of >55% (gray) and >65% (black); green bar is modern GPS shortening rate.

Predicted ZHe ages are all partially reset from ~18-26 km. Slower velocities *i* and *v* (~32-73 Ma) predict ages up to 32 Myr younger than those of the fastest velocity *viii* (~64-90 Ma) because the faster rate on the OOS fault dampens the predicted age pattern (i.e. restricts exhumation) in front of it (~20 km) (Figure 2.9a). Predicted ZHe ages are completely reset west of the OOS fault at ~22 km in all frameworks. Cooling ages range from ~6 Ma (*viii*) to ~10-14 Ma (*i* and *v*). Between 33 and 75 km, model *i* predicts increasingly older ZHe partially reset ages (~75-90 Ma), while velocities *v* and *viii* remain completely reset until ~42 km, where predicted ages show a more complicated partially reset age pattern (~15-55 Ma) that has a zone of younger partially reset ages between 50 and 60 km.

Velocities *v* and *viii* have a nearly identical pattern of predicted ages between ~25 and ~70 km, offset by ~6 Myr, that do not overlap in time because the signal is set by initial SAZ fault motion at 10 mm/yr, regardless if that rate starts at ~7 Ma or at 13 Ma. These two frameworks have the exact same velocities during EC and AP emplacement, and differ only in velocity for the most recent ~42 km of shortening. This portion of the ZHe signal is collocated with samples IA1, IA3 and SA4, whose ages we argue are replicated in the signals of velocities *v* and *viii*, within error, but not in that of constant velocity *i*. Thus, the shape of the pattern of ZHe reset ages requires velocities greater than ~5.8 mm/yr (that of model *i*), and that, the timing of this fast emplacement of models *v* and *vii* within model error are equivalent, but that model *viii* is on the outer edge of this error envelope.

Eastern Cordillera. Across the EC, velocity frameworks *v* and *viii* have more fully reset AFT ages and younger partially reset ages than the constant velocity *i* particularly between 100 and 135 km; though the bulk of this pattern is controlled by kinematics, the faster velocities result in more fully reset ages by driving younger exhumation that occurs during SAZ deformation (particularly between 100 and 135 km, Figure 2.9). Velocity *viii* has EC ages ~3 Myr younger than those in *v*, as the duration of rapid shortening in the SA results in the timing of deformation in the EC being younger.

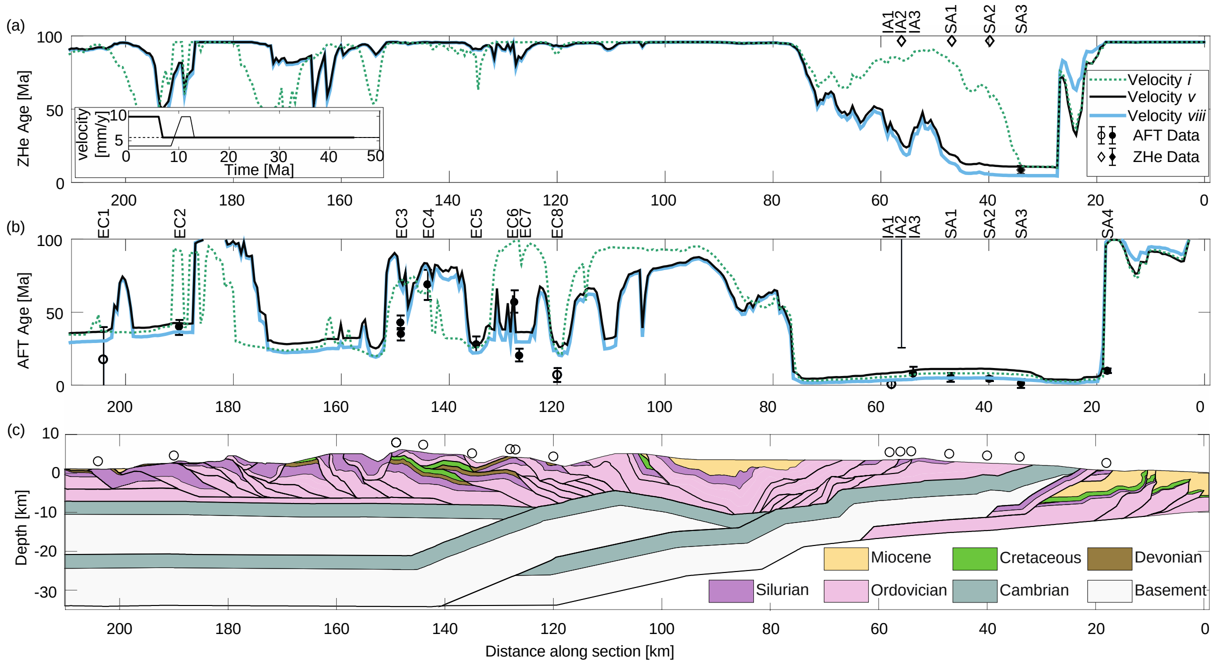


Figure 2.9. Effect of velocity on thermal modelled ages

kinematics and thermal structure are the same between models. (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages; inset with velocity structure; (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section with no vertical exaggeration; open circles indicate sample locations.

2.4.3 Best Fit Velocity Models

Modelled thermochronologic ages predicted by various velocity structures for two kinematic variations (in-sequence, model 1, and OOS, model 4) were compared based on their ability to reproduce the measured thermochronology (AFT, ZHe) ages, all of which were given equal weight. Matches between measured and modelled ages were identified if the modelled age, with a ±1 Ma and ±2 km error, fell within any portion of the measured age and its error. The total percent fit is computed by dividing the number of matches by the total number of samples (Table 2.4). For constant velocity, Model 1 (in-sequence deformation) matches better (*i*, 52%) than Model

4 (*i*, 47%) (Table 2.4). However, for all other velocity frameworks, Model 4 consistently produces better results than Model 1, by 5-19%.

Velocity Structure		SA Rate		Percent Fit	
50 Ma Start		AP + SA	SA	Model 1	Model 4
i	All @ 5.8mm/y ("Constant") *	1.8-4.6 mm/y **14.48**-11.81 Ma	5.78 mm/y 11.81-0 Ma	52.38%	47.62%
ii	Hiatus 40-20Ma, SA + AP @ 10-12mm/r *, SA 6-10mm/y	3.8-8.0 mm/y **11.53**-10.12 Ma	6-10 mm/y 10.12-0 Ma	47.62%	52.38%
iii	SA @ 8mm/r *	1.65-4.2 mm/y **11.52**-8.54 Ma	8 mm/y 8.54-0 Ma	57.14%	71.43%
iv	6-2Ma @ 11.5mm/y, 2-0Ma @ 9mm/y *	1.65-4.2 mm/y **9.24**-6.26 Ma	9-11.5 mm/y 6.26-0 Ma	52.38%	61.90%
v	11-5Ma @ 10mm/y, 5-0Ma @ 4mm/y *	1.8-4.6 mm/y **15.79**-13.11 Ma	4-10 mm/y 13.11-0 Ma	42.86%	52.38%
45 Ma Start					
vi	SA + AP @ 8mm/y *	2.4-6.4 mm/y **10.22**-8.54 Ma	8 mm/y 8.54-0 Ma	47.62%	71.43%
vii	SA @ 8mm/r *	1.8-4.6 mm/y **11.21**-8.54 Ma	8 mm/y 8.54-0 Ma	57.14%	71.43%
viii	SA @ 10mm/r *	1.8-4.6 mm/y **9.51**-6.83 Ma	10 mm/y 6.83-0 Ma	52.38%	71.43%
40 Ma Start					
ix	SA@ 8mm/y *	2.16-5.44 mm/y **10.81**-8.54 Ma	8 mm/y 8.54-0 Ma	57.14%	71.43%
x	SA + AP @ 10-12mm/y, SA 6-10mm/y *	3.8-8.0 mm/y **11.53**-10.12 Ma	6-10 mm/y 10.12-0 Ma	57.14%	61.90%

Table 2.4. Summary of velocities modelled, SA velocities and ages, and percentage fit. Bold is age of SA initiation. Modelled thermochronology error is ±1 Ma and ±2 km along section. Percent Fit is computed by number of measured samples that fall within the model error divided by the total number of measured samples. Note: the first ~11.4 km of SA shortening is interfingered with the last 11.5 km of AP shortening. The * full rate given in column is partitioned between AP and SA shortening.

Sensitivity to model start times (50, 45, 40 Ma) was tested with the same SA velocity frameworks (8mm/yr, ~8-0 Ma, Figure 2.17), but with three different rates during EC and AP deformation (5.2, 5.8, and 6.8 mm/yr). All three models match the thermo- chronology data equally well (71%, models iii, vii, ix; Table 2.4), though velocity ix lies on the outer edge of the error envelope (40 Ma start). This sensitivity to model start suggests that the EC is relatively in- sensitive to variation between ~5 and ~7 mm/yr deformation rates orage of initiation between 40 and 50 Ma, with a slightly better fit at >5.2 mm/yr and model start >40 Ma. Additionally, velocity *ii*, with EC deformation occurring at 4 mm/yr, has a lower percentage fit than velocity *x* whose EC deformation occurs at 5.7 mm/yr (52% vs 62%) and the 40 Ma start of deformation just barely captures the younger error limit for sample EC2. These observations highlight the critical velocity for EC thermochronologic fit as at least 5.2 mm/yr while the youngest permissible age of deformation is 40 Ma.

Velocity frameworks with greater than 55% fit for kinematic Model 4 were used to define an acceptable velocity envelope, shown in Figure 2.8b. The best fit velocity frameworks (65% or greater match) require the SA to be 8-10 mm/yr (Table 2.4), and EC deformation at 5.2 mm/yr or greater. Five velocities result in the same percentage fit: *iii*, *vi*, *vii*, *viii*, and *ix* with a fit of 71.43%. As velocities *iii*, *vii*, and *ix* all have the same shortening rate during SA deformation (8 mm/yr) but different deformation (50, 45, 40 Ma) start times, we compare velocities *vi*, *vii*, and *viii* (all with a 45 Ma initiation of deformation) to remove any variation in modelled thermochronology due to variation in model start time. For ZHe sample IA2, velocity *vi* provides a better fit, rather than at the edge of the model error envelope (Figure 2.10d). For ZHe sample SA3 and AFT samples IA3, SA1, SA2, and SA4, models *vi* and *vii* provide a better fit than *viii* (Figure 2.10e-g). Model *viii* only provides a better fit than *vi* or *vii* for AFT samples IA1 and SA3 (Figure 2.10f). As velocity *vi* best predicts the measured ages of the majority of SA thermochronology samples within a smaller error envelope than the other two velocities, velocity *vi* is used for modelled sedimentology ages in Figure 2.11.

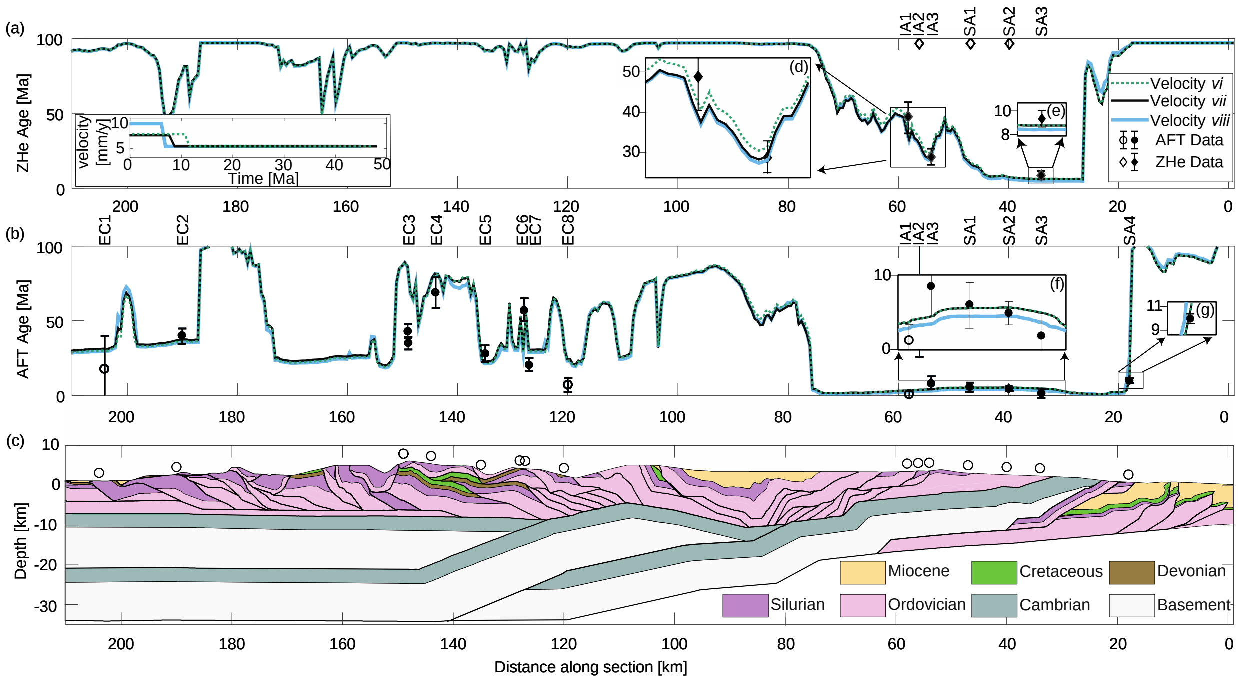


Figure 2.10. Best fit velocities vi, vii, and viii

Insets (d)-(g) highlight important variations between the models and their abilities to fit SA thermochronology data. Kinematics and thermal structure are the same between models. (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages; inset with velocity structure; (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section with no vertical exaggeration; open circles indicate sample locations; (d) ZHe samples IA1 and IA3; (e) ZHe sample SA3; (f) AFT samples IA1 through SA3; (g) AFT sample SA4.

2.4.4 Modelled Sedimentology

As flexural loading, due to the generation of new topography by crustal shortening in the kinematic model, is accounted for, the previous depositional horizon subsides creating a modelled stratigraphy. These flexurally-created foreland and hinterland basins initially form by deposition up to 0 km elevation. To continue sedimentation during and after deformation in the AP, sediment was accumulated in topographic lows up to 0.5 km per step. We compare the model-predicted age and thickness of stratigraphy in five distinct regions to their correlative published strata using our best-fit velocity model (Figure 2.11). Model-associated ages are listed on the left, while the measured formations corresponding to thickness are listed on the right (Figure 2.11 (a)-(e)).

2.4.4.1 Corque

Modelled synorogenic sedimentation in the Altiplano produced a westward thickening 1.1-3.2 km total of backbulge, forebulge, and foredeep sediments prior to thermal model initiation due to 200 km of shortening in the proto-WC FTB. These modelled strata may be correlative to the Santa Lucia, El Molino Fms, and basal Potoco (Figure 2.2, Figure 2.5a) depending on the start of EC deformation (early 50 Ma starts precludes the ~40 Ma base of the Potoco Formation as part of these early synorogenic strata). As modelled deformation in the EC begins, uplift over two basement ramps, one west and one east of the AP, provides a sediment source and loading mechanism for the AP. This shortening is transferred up these two basement ramps and into the Paleozoic section, which shortens initially eastward, and then propagates back west through the EC (Figure 2.5b-e). The Corque syncline is situated between these two loci of deformation (e.g. basement ramps) and has ~4.5 km of modelled hinterland sediment accumulated by ~36 Ma, corresponding to the Eocene Lower Potoco, and ~4 km of modelled synorogenic sediment by ~28 Ma corresponding to the Oligocene Upper Potoco (Figure 2.11a).

The modelled subsidence was restricted due to imposed isostatic uplift which slowed accumulation, resulting in only ~4 kms more of accumulated sediment by ~11.6 Ma (Figure 2.5f), and ~2 km more by 0 Ma restricted to the topographic lows in the core of the syncline only, corresponding to the lower Miocene Coniri and upper Miocene to Quaternary upper Totora and Crucero Fms (Figure 2.5h, Figure 2.11a).

2.4.4.2 Lago Poopo

Lago Poopo modelled sedimentation initiated around ~25 Ma after OOS motion allowed erosional removal of up to ~6 km of synorogenic and the Cretaceous through Devonian overburden (Figure 2.5d). Flexural-induced subsidence, due to shortening in the adjacent AP, allowed 4.8 km of synorogenic sediment accumulation between ~22 and 6.5 Ma (Figure 2.5f), with 0.4 km more between 6.5 and 0 Ma (Figure 2.11b). The restricted subsidence 6.5-0 Ma is due to the locus of deformation, and thus subsidence, shifting ~170 km towards the foreland, with a secondary component due to imposed isostatic uplift.

2.4.4.3 Morochata

The 0.3 km of modelled backbulge sedimentation corresponding to the bulk of the Santa Lucia and El Molino Fms formed prior to the initiation of thermal modelling (Figure 2.2, Figure 2.5a). At the initiation of deformation, Morochata lies in a forebulge position. As deformation initiates in the modern FTB, Morochata lies between two loci of deformation, the eastward propagating deformation in Paleozoic rocks to the east and the basement ramp located to the west (Figure 2.5b). As the isostatic response is dominated by the loading associated with the basement ramp, the remaining sedimentation occurs in the position of the foredeep/wedgetop, with 0.3 km forming by ~43 Ma and 0.2 km more by ~40 Ma (Figure 2.11c).

2.4.4.4 Along Strike with Incapampa

Although initial modelled deformation in the Paleozoic strata of the FTB starts on the eastern edge of the EC (Figure 2.5b), as the basement thrusts propagate eastward and the overlying deformation propagates westward into the EC, the EC/IAZ boundary is a region of subsidence and basin formation (Figure 2.5c, d). The modelled subsidence in the region is limited prior to ~34 Ma, as the loading is controlled by the emplacement of basement thrust sheets 1 and 2 and thus focused in the hinterland. This results in a very condensed section of sedimentation of only 0.1 km between 50 and 34 Ma (Figure 2.5b). However, as the FTB grows and the décollement deepens in response to continued, eastward-propagating shortening of the basement thrust sheets, the eastern edge of the EC transitions to a more proximal foredeep location in front of basement thrust 3 and accumulates ~2.7 km of modelled synorogenic sediment by 22 Ma (Figure 2.5d, e, and Figure 2.11d). No sediments are preserved after this time due to erosional removal from uplift and emplacement of basement thrust sheet 4 (Figure 2.5g). As described in section 4.2.2, our model preserves this sedimentary section that is not preserved in the FTB at 18°S due to insufficient exhumation over basement thrust sheets 3 and 4.

2.4.4.5 Foreland Basin

Modelled sediments in the foreland initially accumulate quite slow, with only ~0.35 km of sedimentation occurring between the model start and ~10 Ma due to the hinterland focus of shortening and weak initial lithosphere not forcing flexural subsidence far from the locus of deformation (Figure 2.5a-e). As the deformation front propagates forward and shortening is focused in the IAZ and SA, the FB transition to proximal foredeep (Figure 2.5f). The bulk of the FB sedimentation, the remaining ~4 km, occur between ~10 and 0 Ma due to flexural and imposed subsidence in the proximal foredeep of the FTB (Figure 2.5f-h, Figure 2.11e).

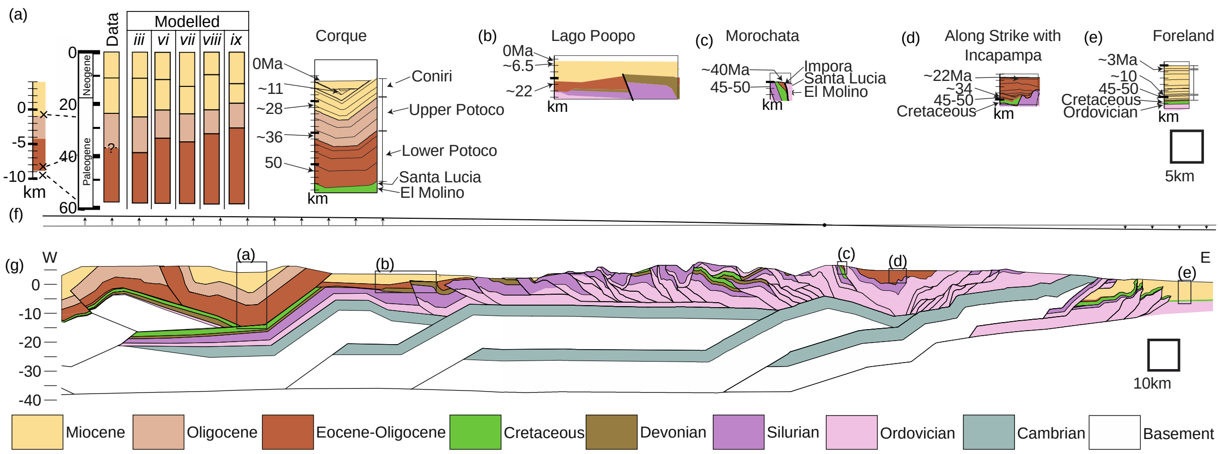


Figure 2.11. (a)-(e) Sedimentary basins and associated model ages [Ma]

(f) Depiction of Imposed Isostasy showing locations of uplift and subsidence. (g) Final flexural-kinematic model showing (a)-(e), (g) with no vertical exaggeration. Stratigraphy insets at 2x Cross-Section Scale.

2.5 Discussion

2.5.1 Basement Thrust Sheet Controls on Kinematics, Exhumation, and Cooling Signals

The pronounced changes in structural elevation between the SA, IAZ, and EC, in combination with the projected décollement depth, leaves a large space to be filled between the depth of the Paleozoic strata underneath the SA and its projection towards the hinterland and the exposed Paleozoic rocks at the surface in the IAZ and EC. These pronounced structural elevation changes have been interpreted as a function of basement thrust sheets in crustal scale cross-sections (Anderson et al., 2017; Baby et al., 1997; Kley, 1996; Kley et al., 1999; McQuarrie, 2002; McQuarrie, Barnes, et al., 2008; Müller et al., 2002; Rak et al., 2017), although the geometry of the basement thrusts may differ. While duplexing of sedimentary rocks would fill the space, it would significantly increase the magnitude of shortening (McQuarrie, 2002), and require substantial removal of crustal rocks (Eichelberger et al., 2015). Provided there is an appropriate strength contrast between strong basement rocks and weak rocks due to the brittle-ductile transition, emplacement of a long basement thrust sheet requires less work than duplexing (Hatcher & Hooper, 1992; Mitra & Boyer, 1986). The initial cross-section geometry in this study (from McQuarrie, 2002) used three basement thrust sheets (with one very long thrust sheet) as a balance between the need for structural elevation and reduction in the amount of work necessary to deform. However, the initial geometry placed the hangingwall/footwall cutoff of basement thrust sheet 2 underneath a preserved thrust of Paleozoic through Cretaceous strata (near ~300 km in Figure 2.5a). Basement thrust sheet 1 has ~33 km of displacement, and the majority of loading in the initial FTB is focused within ~50 km of this uplift (275-400 km, Figure 2.5b). As displacement propagates forward and shortening begins on thrust sheet 2, the Cretaceous strata (preserved in modern times) are located near the surface (~0 km elevation) (~260-300 km, Figure 2.5b). The initial basement thrust sheet 2 ramp was located directly below this and uplifted the previous topographic surface to >10 km in elevation, regardless of EET tested. Any viable estimate of topographic elevation resulted in erosional removal of Cretaceous through the middle of Ordovician section. In order to preserve these Cretaceous through Ordovician strata present in the modern FTB, the basement thrust sheet was split into two and the ramp location was shifted to lie under the deepest erosion level with the majority of the cross-section remaining unchanged. Due to the strong relationship between basement thrust propagation and erosion, an obvious way to test the new basement geometry is to see if its predicted ages agree with measured ages though the region. The match between predicted and measured cooling ages as well as the flexurally produced basins all support basement thrusts as a viable method for transferring slip from deeper décollement levels up to the thin-skinned surface thrusting seen in the Central Andes. The incorporation of flexural modelling of basement thrust sheets imparted a new constraint to evaluating the viability of a cross -section and cross -section kinematics and highlights the large effect basement thrusts have on the location of exhumation.

Differences in the geometry and location of basement structures, as well as the proposed kinematics and vergence, should impart notably different mineral cooling ages. Baby et al. (1997) show two cross-sections through northern (15-18°S) and southern (21-22°S) Bolivia. Both sections show four to five eastward stepping basement thrusts similar to the four eastward stepping basement thrusts we propose for our section at 17-18°S. Notable differences include: 1) the thickness of these thrust sheets, the Baby et al., (1997) basement structures involve the entire thickness of the basement (20-25 km) compared to the ~12 km thrust sheets in the cross -sections we model, and 2) the location of the proposed modern, active ramp. The southern section of Baby et al. (1997) depicts a steep, active basement ramp located near the center of the EC. This steep basement ramp would produce a modern westward -younging signal similar to the signal basement thrust sheet 2 produced at ~24-27 Ma (Figure 2.5c,d). The northern cross -section (15-18°S) has the active ramp even farther to the west. The young 0.5-8.1 Ma AFT ages between 20 and 60 km from the deformation front at 18°S (Figure 2.10f, Table 2.1) requires that the active ramp at this latitude is located near the EC-IAZ boundary (Figure 2.1).

Armijo et al. (2015) depict an orogen scale cross -section located at 21°S. In contrast to the eastward -verging basement structures proposed by Baby et al. (1997) and this study, the basement structures are predominantly west -verging (Armijo et al., 2015). The proposed westward -verging and westward -younging basement thrusts would provide uplift and exhumation in the eastern cordillera form 40 Ma to 10 Ma with the youngest ages (10 Ma) above the active ramp in the western EC. This kinematic scenario and timing are hard to reconcile with the measured thermochronology in the area (Anderson et al., 2018; Calle et al., 2018; H. Ege, Sobel, Scheuber, & Jacobshagen, 2007; Harald Ege, 2004; Tawackoli, 1999). However, Armijo et al. (2015) additionally provide surface observations of west-vergent structures west of the WC that accommodate 10's of kilometers of shortening. These west-vergent structures are not incompatible with the predominantly east-verging WC basement ramps that facilitate 100's of kilometers of shortening in the EC, IAZ, and SA that we present here.

The focused uplift imparted by basement ramps produce locally high topography and an increase in erosional exhumation (Figure 2.5). In our model, basement thrust sheets are the primary driver of uplift and exhumation, and thus impart the youngest ages above the active ramps (see Figure 2.5c). Shortening over the hinterland footwall ramp due to emplacement of basement thrust sheet 2 produces the westward-younging AFT reset signal recorded by the two furthest west EC samples (180-250 km, Figure 2.5h). Eastward propagation of this basement thrust sheet forms a crustal-scale, passive roof duplex which produces an eastward-younging cooling signal as westward propagation of the EC backthrust belt move material up and over the hangingwall ramp (Figure 2.5d). These two patterns, when combined, have the oldest ages near the center of the EC (~160-150 km), bordered by westward-younging and eastward-younging patterns such that the youngest ages are on the outer edges of the EC (Figure 2.5h). This pattern is slightly obfuscated by low exhumation (most notably at 180 km), where the pattern of basement faulting is only seen in the youngest reset ages (Figure 2.5, Figure 2.7), but can be augmented by increased heat production or velocity (Figure 2.7, Figure 2.8). Low exhumation imparts a saw tooth pattern of reset and partially reset ages with a higher frequency (shorter wavelength) that highlights the exhumation induced by deformation on individual Paleozoic structures, complementing the broad wavelength cooling signal imposed by basement thrust sheets. We argue that this combination signal is recorded in our measured AFT ages where adjacent samples can have a difference in age of 25-30 Myr (e.g. samples EC3 and EC4), a range replicated between reset and partially reset ages in our model. Further eastward propagation of deformation emplaces the final basement thrust up and over its basement ramp and cuts through the full Paleozoic through Cenozoic section where it breaks the surface at ~20 km (Figure 2.5h, Figure 2.6b). This series of ramps produces a shallowly sloped westward-younging signal where the western edge is pinned at the ramp through the lower Paleozoic section (~75 km, Figure 2.5g). Superimposed on this westward-younging signal are young ages that are a function of the OOS fault and the frontal ramp through the SA foreland basin. This combination of predicted signals matches the measured thermochronology.

2.5.2 Sedimentary Basin Formation

Sedimentary basins in retroarc systems encapsulate an important record of crustal shortening, flexure, and accumulation histories that are formed in isostatic response to the load created by FTB shortening (Angevine, Heller, & Paola, 1990; Beaumont, 1981; Horton, 2018a; Jordan, 1981, 1995; Stockmal, Beaumont, Nguyen, & Lee, 2007). The sedimentary history preserved in the Corque syncline provides a long-lived hinterland basin record of the duration of Andean mountain building, while the Chaco foreland basin contains only the youngest stage of Andean history (DeCelles & Horton, 2003; Horton, 2018a; McQuarrie et al., 2005). Incorporation of kinematic and flexural modelling allows us to investigate the structural controls on sedimentation and preservation. In our model, ramps through basement rocks double a crustal section of ~12-15 km, producing a much larger uplift signal (and thus topographic load) than shortening in the Paleozoic section, which typically repeats a ~3-5 km stratigraphic section. The basins produced due to basement loading (e.g. ~375-575 km, Figure 2.5b) are broader and deeper than those produced by shortening in the Paleozoic section (e.g. ~50-225 km, Figure 2.5b) because of the contrast in load imparted by larger ramps. Thus, the location and magnitude of crustal deflection (and thus accommodation space) is a function of the loads imparted by basement thrusts. These flexurally subsiding regions are filled with sediments shed from the FTB and elevated topography to the west and can form on both hinterland and foreland sides of the load.

Our model tracks the accumulation of sediments through time and is able to reproduce the thicknesses of synorogenic sedimentation found in Cenozoic basins throughout the region (Figure 2.1c, Figure 2.11). Sedimentation in the AP initiates as a flexurally-induced backbulge through foredeep environment prior to model initiation due to deformation in the proto-WC FTB (Figure 2.2). This initial sedimentation, of only ~1.5 km, is a small portion of the overall preserved AP sedimentation in the Corque syncline (~15 km). Depending on the age of deformation in the EC this sedimentation predates 50-40 Ma and thus precedes or coincides with the deposition of the ~40 Ma Potoco Formation. As the FTB propagates eastward and shortening in the modern FTB initiates, the AP forms a flexurally induced hinterland basin that fills with Cenozoic synorogenic strata (‘C’, Figure 2.5) sourced primarily off of the uplifted flank of the deep basement thrust sheet in the west (~615 km, Figure 2.5b), with additional sedimentation possible from the east either from basement thrust sheet 1 (in sequence basement deformation) or basement thrust sheet 2 (a potential OOS basement thrust order) (Figure 2.18). In sequence basement thrust sheets would argue that the lower 4 km of Potoco Formation preserved in the eastern limb of the Corque syncline potentially were derived from the east. However, the predicted topography over the uplifted basement thrust 1 shows a steep westward facing topographic slope with limited drainage area defined by the uplifted topography. A steep western slope and a broad gentle eastern slope may limit sediment contribution from the east to within ~ 5-10 km of the uplift (Leeder, 1995) that is then eroded as the Corque syncline deforms. Alternatively, initial motion on basement thrust 2 limits the contribution to the Corque syncline from eastern sources by moving the potential source area farther to the east (Figure 2.18), promoting western -derived sediments in the lower 4 km of the Potoco Formation (Horton et al., 2001). The thickness and age range of the Potoco Fm is reproduced by our flexural model and matches the measured rate of sedimentation, within error, when we use our best fit velocities, and the sediment sources predicted by our models do not conflict with the predominant eastward paleoflow measured in the lower 4 km of the Potoco Formation (Table 2.4, Figure 2.11) (Horton et al., 2001). The measured rate of sedimentation and sedimentary ages requires rapid subsidence and argues for more than just a foreland basin sedimentation history (Horton, 2018a) preserved in the AP. We argue that this rapid sedimentation rate and the Eocene to modern thickness of sedimentation in the AP is generated as a function of a double load imparted by two basement ramps. A double load, and thus rapid subsidence, is perhaps a unique feature of hinterland basins (Horton, 2012, 2018a).

Along the AP/EC border, thrust-induced exhumation associated with motion over the footwall of basement ramp 2 removes the Cretaceous through Silurian strata. Loading of the FTB due to continued uplift over a WC basement ramp (Figure 2.5c) and eastward emplacement of basement thrust sheets in the EC and SA, as well as AP shortening, drives subsidence of the AP/EC between 25 Ma and present ('LP', Figure 2.5e-h). Thus, our model reproduces the erosional and sedimentary history found in Lago Poopo (McQuarrie & DeCelles, 2001).

In the EC, the pre-Andean FTB in our model drove backbulge through forebulge sedimentation at the location of Morochata (Figure 2.2). As deformation propagates eastward into the modern FTB, Morochata lies between the locus of deformation on the western edge due to basement thrust sheet emplacement, and the thin-skinned shortening in the Paleozoic cover to the east. Isostatic loading and erosion from this proximal thin-skinned thrusting initiated sedimentation in the Morochata basin east (‘M’, Figure 2.5b). Initial loading from basement sheet 2 dominated the isostatic load, and the resulting sedimentation in the basin kept Morochata in a foredeep/wedgetop position. This sedimentation history of Morochata broadly matches the measured thicknesses, ages, and sedimentary environments of sediments preserved in measured sections along strike but farther to the south in the Camargo Syncline (Figure 2.11c)(DeCelles & Horton, 2003).

On the far eastern edge of the EC, our model preserves basin sedimentation not found in the section at 18°S. These sedimentary strata formed initially in a backbulge location with restricted sedimentation, but as the FTB migrated eastward and increased subsidence in the region between Oligocene and early Miocene, the bulk of the sedimentation occurred in proximal foredeep environments similar and adjacent to Morochata (‘I’, Figure 2.5b-d). Though this basin is not preserved at 18°S, similar captured Cretaceous through Miocene sedimentary basins (Incapampa (Figure 2.5c) and Tarabuco) are found along strike south of this section (DeCelles and Horton, 2003, Horton 2005). These basins are preserved in front of the hangingwall of the easternmost EC basement thrust sheet. These are separate from the Camargo, Torotoro, and Morochata basins, by a structural high of predominantly Ordovician age rocks (Horton, 2005) that place the Camargo, Torotoro, and Morochata basins behind (west) of the basement thrust sheet hangingwall cutoff (Eichelberger et al., 2013; McQuarrie, 2002). In geology of the region and in the kinematics of the original cross-section, the modelled basin remnant at the far eastern edge of the EC would have been erosionally removed though the uplift associated with fault-bend-fold hanging wall steepening (Suppe, 1983) of basement thrust 3, a deformation process *Move* is not able to model without perfectly flat décollements. The inability to reproduce this deformation, forces the basin to remain caught between the eastern basement thrust hanging wall and the SA basement thrust, a position that has preserved these basins in the fold-thrust belt farther to the south. The preservation of this basin argues for the basement ramp associated with basement thrust sheet 3 to be located further east. This would force the toe of the hangingwall of thrust sheet 3 farther under the EC/ IAZ boundary, providing structural elevation allowing for erosional removal of the basin.

For the majority of the model time, our modelled modern FB (i.e. the SA) receives little to no subsidence due to its location far from the locus of shortening ('F', Figure 2.5a-e). However, as SA shortening initiates in the Miocene, the bulk of the ~4.5 km of FB sediments form in response to a flexurally-driven foredeep driven by the SA basement thrust (Figure 2.5f-h) predicting basal foreland basin sediments that are ~ 10 Ma. This lack of Paleocene and Eocene strata, as well as the thickness of produced Miocene synorogenic sedimentation, match the measured FB strata (Figure 2.11) (Marshall & Sempere, 1991; Marshall, Sempere, & Gayet, 1993; Cornelius E. Uba et al., 2009; Cornelius Eji Uba, Heubeck, & Hulka, 2005).

The primary driver of loading and thus accommodation space throughout our model is the location and emplacement of basement thrusts. These exert a first-order control on accommodation space and the partitioning of basins.

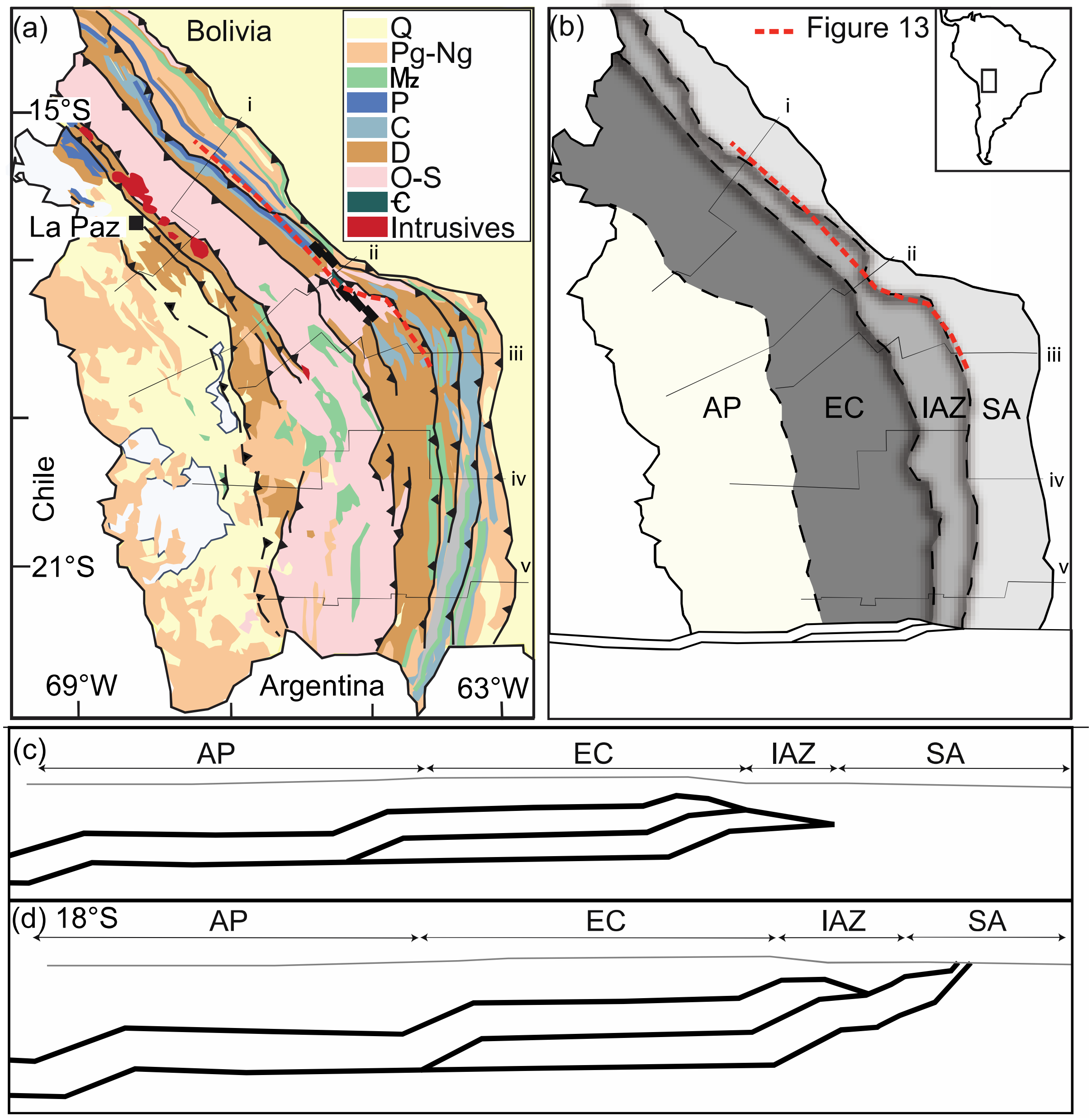


Figure 2.12. (a) Simplified geologic map of Bolivia (modified from (Eichelberger et al., 2013; McQuarrie, 2002a)); (b) Structural elevations due to basement thrusts and tectonogeomorphic zones (Paleozoic cover not shown; modified from (Kley, 1999; McQuarrie and DeCelles, 2001)); thin lines are locations of studies referenced herein (from N-S: Rak et al., 2017 and McQuarrie et a;., 2008a; McQuarrie, 2002; Eichelberger et al., 2013; McQuarrie 2002; Anderson et al., 2017, 2018); simplified cross-sections highlighting basement (thick lines) and SA (thin lines) geometry: (c) representative of the majority of Bolivia; (d) representative of 18°S.

2.5.3 Lateral Variability in Structure, Kinematics and Velocity

2.5.3.1 Structural and Stratigraphic Variability

Across the Bolivian Andes, many authors have noted lateral variability in age and location of strata exposed at the surface; as a result, they have interpreted subsurface structural changes in the geometry of the FTB to be the cause of these surface variations (Figure 2.12) (e.g. Baby et al., 1995; Kley, 1999). The first order geometry of a FTB is set by preexisting structures, crustal weaknesses, and the sedimentary material available to deform (Boyer, 1995; McQuarrie & Ehlers, 2017a; Mitra, 1997). In the Bolivian Andes, there are several pre-existing structural weaknesses that may have been reactivated or have driven significant variations is sedimentation imparting a control on FTB deformation and resulting in laterally varying map patterns: Late Permian-Jurassic rift structures (Thierry Sempere et al., 2002), Early Cretaceous rifting (Thierry Sempere et al., 1997), basement highs (Baby et al., 1995, 1994; Williams, 1995). These, in combination with changing tectonic, climatic, and transport conditions influenced the shape and contributed to the variability of the Paleozoic sedimentary basins (Baby et al., 1995; Thierry Sempere et al., 1997, 2002). Lateral variability in the sedimentary strata and basement geometry can alter what horizons act as ramps and flats.

The IAZ is broadly defined as the portion of the FTB whose rocks at the surface (primarily Devonian) require a marked change in structural elevation from the SA. This structural elevation is imparted by the easternmost, and thus most recent, basement thrust sheet (Kley, 1996, 1999; McQuarrie, 2002). As the FTB continues to deform and thrusts propagate outward to the SAZ, the rocks of the IAZ are uplifted and passively transported on this basement thrust sheet. Via a balanced cross-section, basement thrust sheets are restored to determine their ramp locations.

These basement structures, both thrust sheets and ramps, exert a first order control on the location and expression of the IAZ.

At 18°S (Figure 2.12 ii), Ordovician and Cambrian strata are at the surface in the IAZ (Figure 2.12a). Topographically, the elevations along strike are relatively consistent and cannot explain the presence of deeper stratigraphy present at the surface. To the northwest of our study area, there is a southwest-northeast trending contact, perpendicular to fault orientation, between the Ordovician-Silurian and Devonian strata (Figure 2.12 ii). This transition from Cambrian strata at the surface to Carboniferous and Permian strata at the surface ~75 km northwest illustrates a rapid decrease in structural elevation (Figure 2.12a) [and erosion level (Figure 2.13c)]. There is a similar decrease in structural elevation along strike to the southeast, where Carboniferous rocks are present again ~150 km southeast (Figure 2.13c).

As a result of a series of coarsening upward sedimentary sequences (Baby et al., 1995; T. Sempere, 1995) there are multiple décollement levels which transfer slip and accommodate shortening at various stratigraphic levels in the Bolivian Andes. Between 15-17°S (Figure 2.13b), the base of the Ordovician section acts as the décollement in both the SA and IAZ, with a secondary décollement at the bottom of the Devonian strata (McQuarrie, Barnes, et al., 2008). At 18°S, the bottom of the Ordovician and Devonian sections are still décollements in the SA, however, the bottom of the Cambrian section is also a décollement (McQuarrie, 2002). At 19°S, the primary décollement has moved to the top of the Ordovician section in the SA, as the shallow dip and FB thickness does not permit carrying the additional stratigraphy (Eichelberger et al., 2013). The SA at 18°S are narrow, likely due to the edge of the Paleozoic basin and/or the Chapare basement high (Baby et al., 1995, 1994), which limited and promoted the propagation of the FTB into the foreland (i.e. a forward lateral boundary) and likely promoted the presence of exposed Cambrian rocks at the surface. This unique presence of Cambrian rocks in the map pattern requires a change in geometry to facilitate the increase in structural elevation required by these older, deeper rocks at the surface (Figure 2.12d). The adjacent cross -sections show the basement thrust at the lowest décollement level either above or below the Ordovician (Eichelberger et al., 2013; McQuarrie, Barnes, et al., 2008; Rak et al., 2017). These constraints require lateral structures, either in the hangingwall, footwall or both, between this study and the adjacent cross-sections in order to permit the variation in map pattern and cross-section geometry.

Figure 2.13 highlights the location of the décollement and stratigraphy that is immediately above and below it along the border of the IAZ and SA (orange dashed line, Figure 2.12a,b). The figure shows variations in stratigraphic thickness as well as marked changes in the hangingwall and footwall stratigraphy (Figure 2.13a). From south to north, the décollement, between 19 and 18°S, changes from the top of a potentially thinner Ordovician section at 19°S to the top of the Silurian at 18°S A lateral footwall ramp separating these two sections can be ~1 km to ~ 3 km depending on if the thickness of the Ordovician changes gradually or abruptly (solid and dashed line between 19°S and 18°S, Figure 2.13b). Between 18 and 15-17° S the décollement steps down (~1 km) to the top of the Ordovician. While variations in hangingwall stratigraphy from 19 and 18°S are minimal with both sections showing the basement cut off immediately above the décollement, hanging wall stratigraphy between 18 to 15-17° S shows a much more dramatic change from basement at 18°S to the Devonian section 15-17°S. This marked change in stratigraphy argues for a ~5 km hangingwall lateral ramp. The lateral ramp is highlighted in the map pattern at 17°S. In this location, the trend of fold and faults are northwest, perpendicular to the shortening direction, while the stratigraphy at the surface changes from Cambrian to Carboniferous strata over a 10-20 km distance perpendicular (i.e. with contacts parallel) to the shortening direction, and 90° to expected (and ubiquitous) stratigraphic changes which are generally parallel to the transport direction. Thus, the more deeply exposed stratigraphy at 17-18°S is a function of both a hanging wall geometry carrying basement rocks farther east than sections to the north as well as a change in the footwall geometry where the décollement above thinner Ordovician strata allows for less structural uplift and less exhumation to the south.

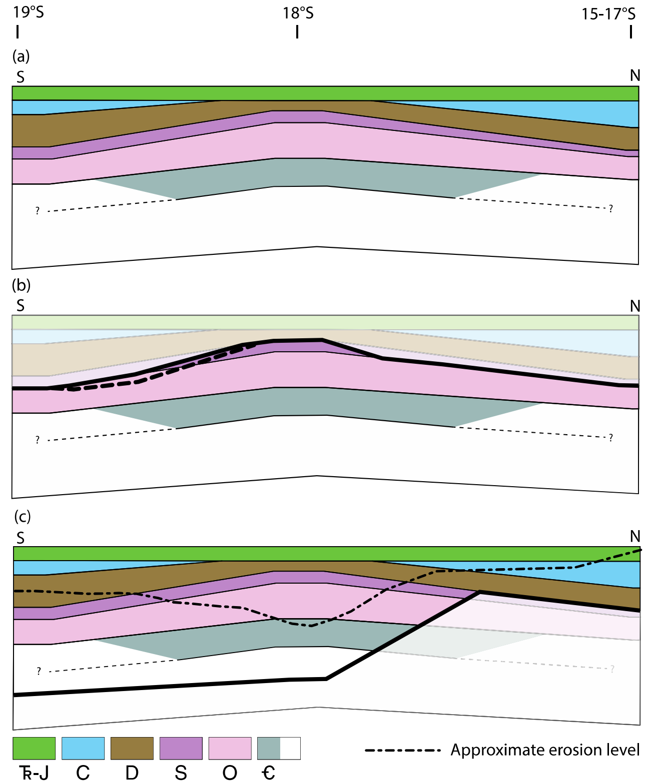


Figure 2.13. Structural and stratigraphic variation along the IAZ/SA boundary

Line of section along IAZ/SA border (Figure 2.12). (a) initial stratigraphic variation in thickness between 15 and 17°S and 19°S (thicknesses measured from McQuarrie, 2002; McQuarrie et al., 2008a and Eichelberger et al., 2013. (b) depiction of footwall structure with alternate (dashed) geometry; (c) depiction of hangingwall structure and erosion level.

2.5.3.2 Kinematic Variability

OOS faulting in this region of Bolivia was first proposed by Whipple and Gasparini (2014) based on the location and pattern of steep river channels (Ksn, a measure of channel slope normalized to drainage area). High Ksn values associated with active uplift and can be related to surface breaking thrust faults or uplift over subsurface ramps (Gasparini & Whipple, 2014; Kirby & Whipple, 2012; Wobus, Whipple, Kirby, et al., 2006). Thus, high Ksn would be expected over the active basement ramp and at the deformation front in our model (if deformation was occurring in-sequence). However, the highest Ksn values located west of the deformation front are not collocated with our basement ramp. Instead, these high Ksn values are immediately west-southwest of the thrust that breaks the surfaces at ~20 km in our model (Figure 2.10), providing support for an out-of-sequence thrust in this location. Our modelling results show that in-sequence deformation does not produce the best fit to measured data, and instead highlights the efficacy of an OOS model at reproducing the measured thermochronology (Figure 2.6, Table 2.4). Our modelled OOS thrust is permitted by the map pattern and there are several mechanistic arguments in support of OOS thrusting in this portion of the Bolivian Andes including focused erosion, either due to increased climatic effects or tectonically-controlled topographic driving orographic rainfall (Dahlen, 1990) and pre-existing structural controls limiting forward propagation of the FTB. Propagation of the thrust front is limited when more work is required to slip along the basal décollement than to ramp upsection (Boyer, 1995; Dahlen, 1990; Mitra, 1997). While it is possible that the high precipitation gradient in the region has focused erosional removal of material and promoted a hinterland increase in topography prior to further propagation, the modern-day high precipitation gradient could be an orographic response to OOS thrusting.

2.5.3.3 Variability of Velocity Along and Across Section

Many previous authors have proposed variable shortening rates in the central Andes (Anderson et al., 2018; Eichelberger et al., 2013; Elger et al., 2005; Hindle & Burkhard, 1999; Hindle, Kley, Oncken, & Sobolev, 2005; McQuarrie et al., 2005; Oncken et al., 2006; Rak et al., 2017). Spatial and temporal variation in shortening rate can be affected by convergence rate and coupling between the subducting and overriding plate (Babeyko & Sobolev, 2005; Horton, 2018a, 2018b), angle and rate of slab subduction, possibly related to full or partial mantle convection (DeCelles, Ducea, Kapp, & Zandt, 2009; Faccenna, Becker, Conrad, & Husson, 2013; Royden, 1993; Schellart, 2017), upper plate response to accumulation and loss of mantle lithosphere (DeCelles et al., 2009), and the lithologic and stratigraphic variability altering the work required to deform (Mitra, 1997). Rates of deformation have been determined from a variety of data sources, and initial velocity estimates in the Central Andes frequently relied on the San Juan del Oro surface (~10 Ma) to constrain the end of EC deformation, and the initiation of SA deformation (e.g. Gubbels et al., 1993). Sedimentation ages and rates are frequently used to constrain the timing of shortening (Echavarria et al., 2003; Espurt et al., 2008; Cornelius E. Uba et al., 2009), however, poor age control (DeCelles & Horton, 2003) and the disconnect between age of sedimentation and age of deformation (Rak et al., 2017) can affect the reliability of these shortening rates. Newer methods of determining rates of deformation incorporate exhumation ages determined by low-temperature thermochronology (Anderson et al., 2018; Barnes et al., 2012, 2006; Lease et al., 2016; McQuarrie, Barnes, et al., 2008), but require a cautious approach because the age of exhumation recorded by a given chronometer system may not be the age of deformation. This is particularly important in the central Andean FTB as exhumation over basement ramps provides a first-order control on exhumation and thus can overwrite an initial deformation history (Rak et al., 2017; this study). Young AFT samples from the IAZ and AFT (at 18°S) that record rapid exhumation at 6 ± 2 Ma have been used to argue for an initiation of SA deformation at this time (Lease et al., 2016); however, as shown in Figure 2.5f-g, exhumation in the IAZ occurs again at ~10 Ma due to uplift over basement structures, and the resulting IAZ and SA modelled ages fall within the published AFT thermochronology range of 2-9 Ma (Figure 2.10f). The period of rapid exhumation is not directly equivalent to the start of IAZ or SA deformation, as exhumation induced by the emplacement over the active basement ramp associated with SA shortening overwrites the initial IAZ ages such that the measured thermochronology ages only record the last stage of shortening.

Our model results for 18°S suggests that the shortening rate associated with initial Andean deformation was relatively stable at ~5.2-6.8 mm/yr during EC, IAZ, and the majority of AP emplacement (~50 to ~12-9 Ma), and that shortening rate increased to 8-10 mm/yr between 12-9 Ma (black, Figure 2.14b). The shortening rate during initial Andean deformation is roughly in agreement with the ~7 mm/yr approximate shortening rate predicted from balanced cross-section and low-temperature thermochronology just to the south (19°S, Eichelberger et al., 2013, Figure 2.14b). In northern Bolivia (15-17°S), the acceptable velocity envelope identified in previous studies incorporating thermokinematic modelling permits a constant 5-7 mm/yr velocity over the entire 50-55 Ma shortening history, with possible windows of increased shortening rates between 50-42 Ma (up to 11 mm/yr) and between 15-5 Ma (up to 12 mm/yr) (Figure 2.14a) (Rak et al., 2017). This pulse of rapid SA deformation starts earlier than our preferred shortening rate (white line, Figure 2.14b). To match measured cooling ages and basin data, SA shortening in northern Bolivia has to decrease to 7 mm/yr from ~5 Ma to present. In contrast, shortening rates remain at

8-10 mm/yr to the present in our study region suggesting lateral variability in shortening rates in the SA.

Authors incorporating balanced cross-section and low-temperature thermochronology, approximate the shortening rate at 19.5°S to be 9 mm/yr from ~40-15 Ma during EC, AP, and IAZ shortening (Figure 2.14c, Barnes et al., 2008) with rates decreases to ~5 mm/yr from 15 Ma to present during SA shortening (Barnes et al., 2008). However, a detailed study incorporating U-Pb dating and sedimentology in the SA between 19.5-21.5°S found that SA shortening may have initiated closer to ~13-10 Ma at rates increasing up to ~11 mm/yr between 5-3 Ma (Cornelius E. Uba et al., 2009). In far south Bolivia (20.5-22°S), authors proposed a FTB shortening rate incorporating low-temperature thermochronology exhumation ages, sedimentation, and balanced cross-section deformation estimates (Anderson et al., 2018; Oncken et al., 2006). The proposed rate by Oncken et al. (2006) increases from ~0 mm/yr at 50 Ma, to ~7±2 mm/yr from 33-10 Ma, and finally to 13±4 mm/yr between 10 Ma and present (Figure 2.14c). Anderson et al. (2018) proposed that deformation occurred in a series of pulses, with a background shortening rate of ~3-5 mm/yr, and pulses of 9-13 mm/yr, 15-27 mm/yr and 9-16 mm/yr during 28-18 Ma, 11-7 Ma, and 2-0 Ma, respectively (Figure 2.14c). The proposed shortening rates are strongly influenced by the San Juan del Oro paleosurface and synchronous IAZ populations of AFT and AHe ages (10-20 Ma, Anderson et al., 2018). Due to these factors, the proposed timing pulses may overpredict shortening rate for ~10 Ma to present compared to 15-17°S and 18°S. Synchronous populations of thermochronometers from multiple thermochronometer systems in the IAZ at 21-22°S require that a large exhumation event occurred (McQuarrie and Ehlers, 2015; 2017) at ~10 Ma and potentially as early as 20 Ma, likely due to the initiation of motion on the most recent basement ramp (Rak et al., 2017; this study)with associated uplift and exhumation (Rak et al., 2017; this study). A 13-10

Ma initiation of SAZ deformation would produce 4-8 mm/yr rate of deformation, with possible faster rates from 2 Ma to present.

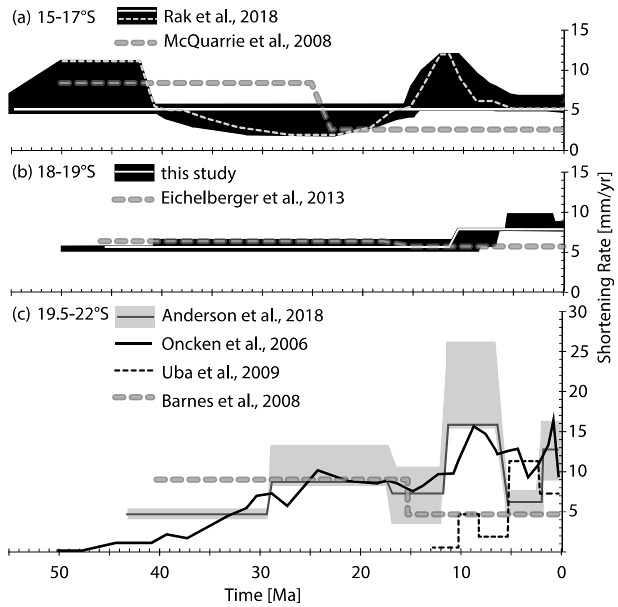


Figure 2.14. Proposed shortening rates across Bolivia

(a: Rak et al., 2017; b: this study) determined from coupled thermokinematic modelling, White line (b) is our preferred velocity model, and (c) light gray (Anderson et al., 2018) and black line (Oncken et al., 2006), from thermochronology, sedimentology, and kinematic restoration. Note: different vertical axis limits on (c).

2.6 Conclusions

The research presented here highlights the importance of thermal, flexural, and kinematic modelling as additional constraints to test the validity of balanced cross-sections. While low exhumation amounts can present additional challenges, the methodology presented herein allows a mensurable link between balanced cross-sections, synorogenic sedimentation, and thermal histories. Derivation of particle paths from detailed flexural and kinematic models allow us to quantitatively link balanced cross-sections with modelled thermal ages and the formation of sedimentary basins through time which they produce. The location of a basement thrust ramp proposed by McQuarrie (2002) resulted in over-erosion of the sedimentary cover, while relocation of this ramp successfully replicated surface geology.

Significantly low AP elevations predicted by initial flexural modelling required the incorporation of a long-wavelength, imposed uplift to prevent excess subsidence and raise the AP to its modern elevations. Inadequately thick SA deposits additionally argue for imposed subsidence to increase the accommodation space available. The requirement for imposed subsidence in the foreland may suggest the impartation of a signal due to viscous coupling between the mantle wedge and subducting oceanic plate (DeCelles, 2012). The need for imposed uplift of the AP could be representative of mantle delamination (Garzione et al., 2006), Airy isostasy attainment (Beck et al., 1996), and/or lower crustal flow (Eichelberger et al., 2015; Isacks, 1988; S. Lamb, 2011), and is consistent with arguments to maintain elevation in the absence of deformation -induced uplift (Rak et al., 2017).

Mismatch between modelled and published ages informs us as to how cross-section kinematics and deformation rates can be revised to create a more accurate solution to known constraints. In-sequence kinematics and constant (~5.8 mm/yr) deformation rates resulted in a mismatch between published and modelled ZHe data, while increasing SA deformation rates and revising the SA kinematic order to incorporate OOS thrusting provides a necessary pause in uplift and exhumation over doubled SA ramps in order to match published ZHe data. While the geometry and kinematics set the pattern of permissible cooling ages, changes in velocity controls the absolute ages recorded and changes to radiogenic heat production alter which patterns of cooling are recorded in which thermochronometer. Our model is insensitive to model start times tested (50, 45, 40 Ma), and requires at least 5.2 mm/yr shortening rate during EC emplacement. While a peak velocity of up to 12 mm/yr during initial SA emplacement, slowing to ~6 mm/yr at present, provides an acceptable fit (>55% fit), the best fit (>65% fit) has deformation rates of 8-10 mm/yr during the entire SA emplacement.

2.7 Acknowledgments

The authors would like to thank Guest Editor Christoph Von Hagke and reviewers Joel Saylor and Ryan Anderson for their comments and suggestions that have greatly improved this manuscript. Supporting data are included in the supporting information file; any additional data or information are available through contacting VMB Parks (email: vmbparks@gmail.com). This work was supported by NASA headquarters under the NASA Earth and Space Science Fellowship program - Grant 80NSSC17K0388. We acknowledge and thank Midland Valley and Petex for support and use of the program Move. This manuscript benefitted from thoughtful discussions over the years with Paul Eizenhöfer, Mary Braza, Chloë Glover, and Brad Parks. T.A. Ehlers and Willi Kappler (University of Tübingen) are thanked for support with conducting modelling simulations and providing the modified version of Pecube that can be coupled to 2D Move restoration files.

2.8 Supplementary Information

Supplementary video showing best-fit model kinematics and modelled thermochronology age sequential development is available here: https://youtu.be/rjdN-1PkCk8

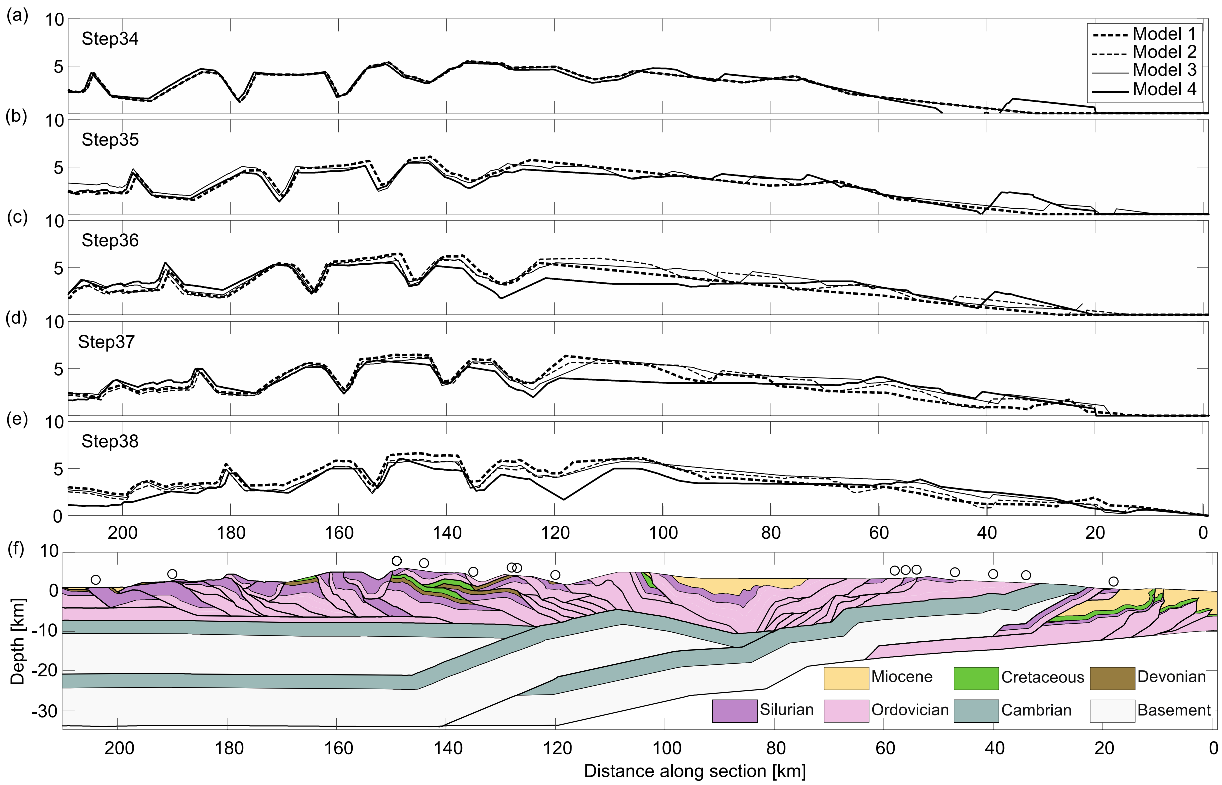


Figure 2.15. SA topographic evolution

(a)-(e) SA/IAZ/eastern EC topographyic evolution in Model Steps; (f) cross-section with no vertical exaggeration

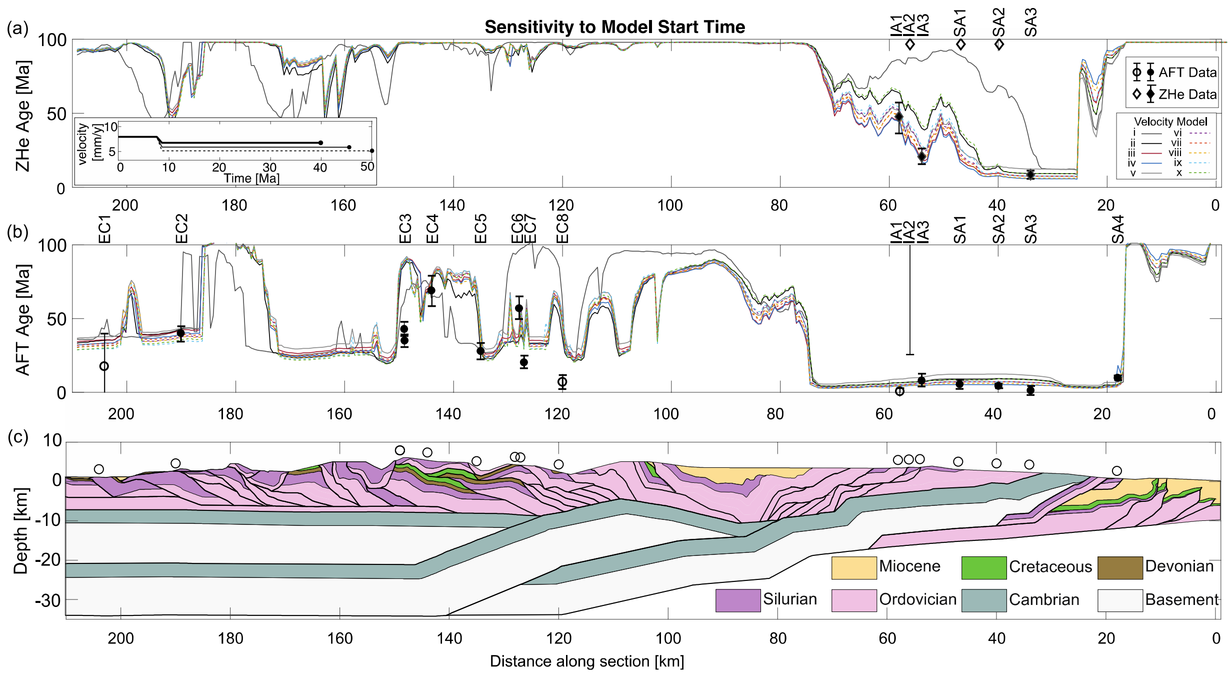


Figure 2.16. Modelled thermal ages for all tested velocities (model 4 kinematics). (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages; (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section with no vertical exaggeration; open circles indicate sample locations; inset with velocity structure (modified from Figure 2.8).

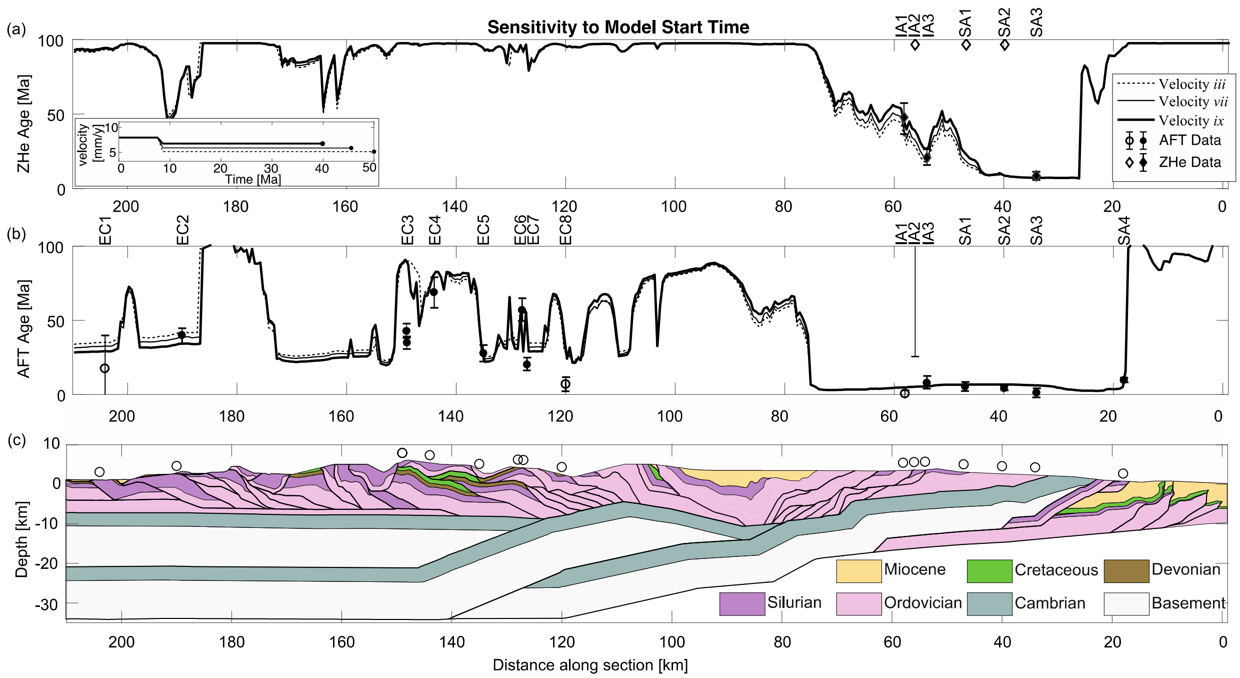


Figure 2.17. Sensitivity to model start time for Model 4 kinematics

thermal parameters are the same between velocity models. (a) ZHe modelled and measured ages; open diamonds indicate locations of samples with mixed/partial reset ages; inset with velocity structure; (b) AFT modelled and measured ages; open markers indicate samples with poor data resolution, (c) cross-section with no vertical exaggeration; open circles indicate sample locations.

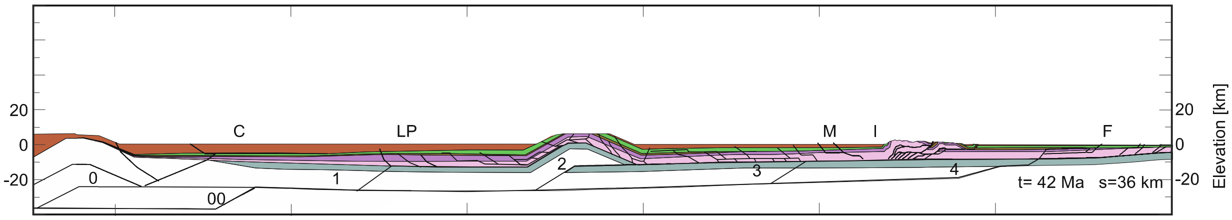
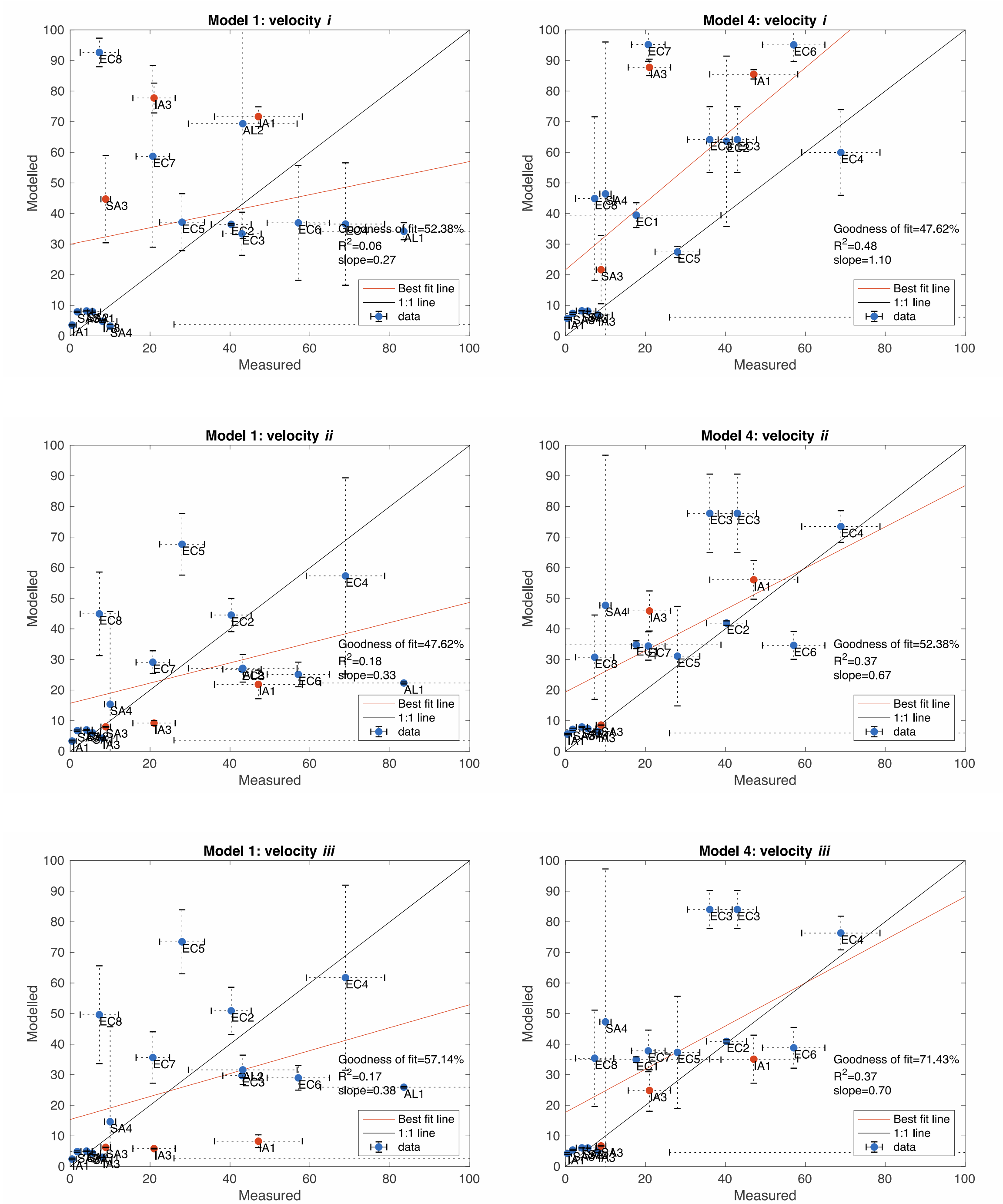
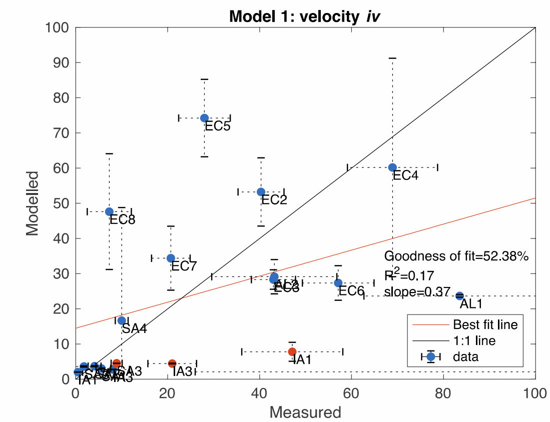
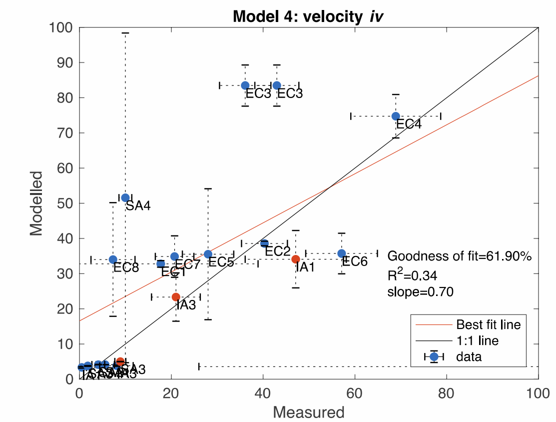


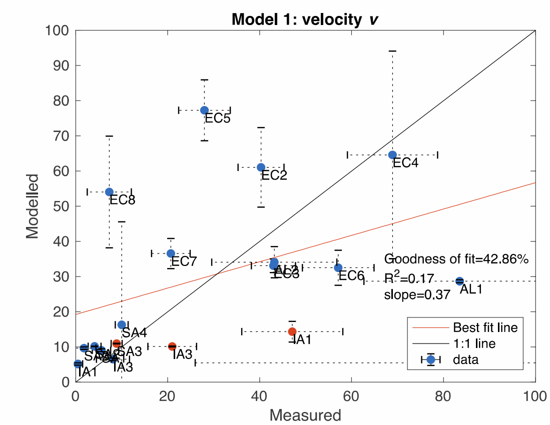
Figure 2.18. Alternative basement ramp kinematic sequence

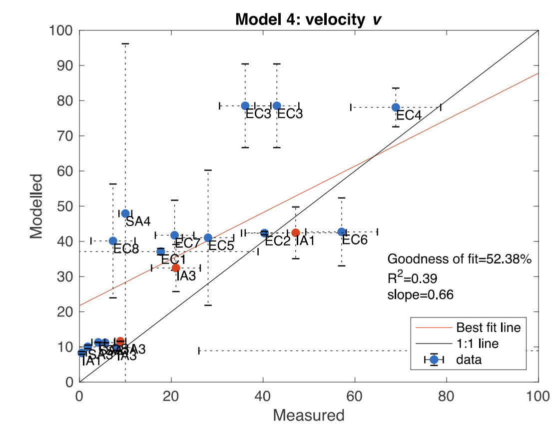
This time step corresponds to Figure 2.5b.

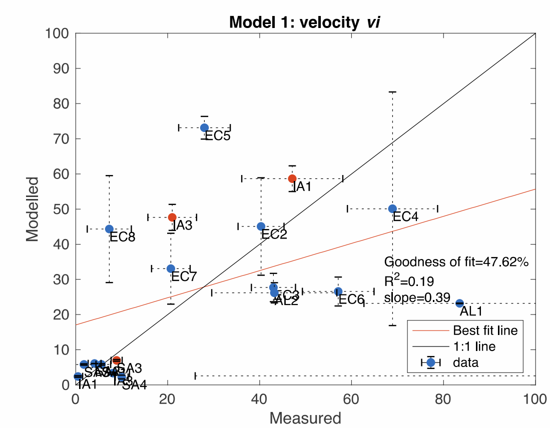

Model 1: velocity *i*
Goodness of fit=52.38%
R²=0.06
slope=0.27
Model 4: velocity *i*
Goodness of fit=47.62%
R²=0.48
slope=1.10
Model 1: velocity *ii*
Goodness of fit=47.62%
R²=0.18
slope=0.33
Model 4: velocity *ii*
Goodness of fit=52.38%
R²=0.37
slope=0.67
Model 1: velocity *iii*
Goodness of fit=57.14%
R²=0.17
slope=0.38
Model 4: velocity *iii*
Goodness of fit=71.43%
R²=0.37
slope=0.70
Best fit line
1:1 line
data
Measured
Modelled

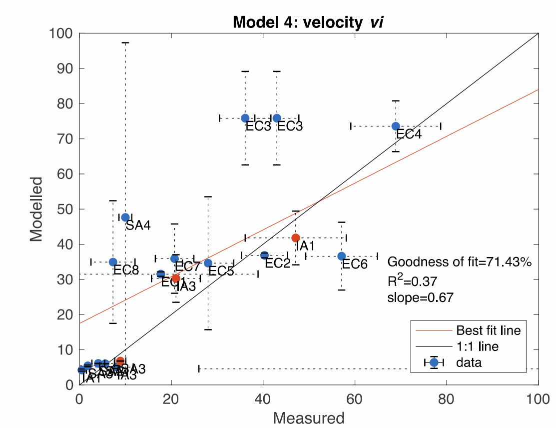

Model 1: velocity *iv*
Modelled
Measured
Goodness of fit=52.38%
R^2=0.17
slope=0.37
Best fit line
1:1 line
data
EC5
EC4
EC2
EC8
EC7
EC6
AL1
SA4
IA1
IA3
SA3

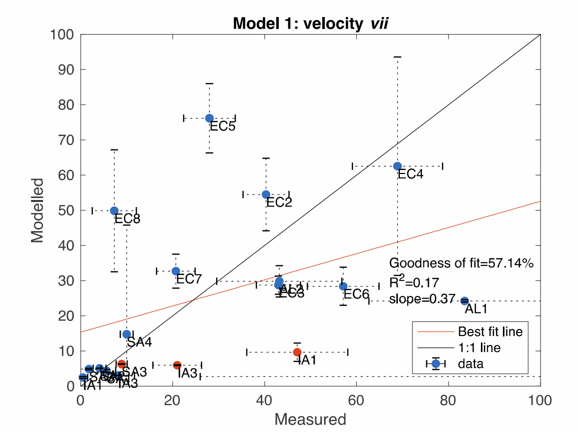

Model 4: velocity *iv*
Modelled
Measured
Goodness of fit=61.90%
R^2=0.34
slope=0.70
Best fit line
1:1 line
data
EC3
EC3
EC4
SA4
EC8
EC1
EC7
EC5
EC2
IA1
EC6
IA3
SA3

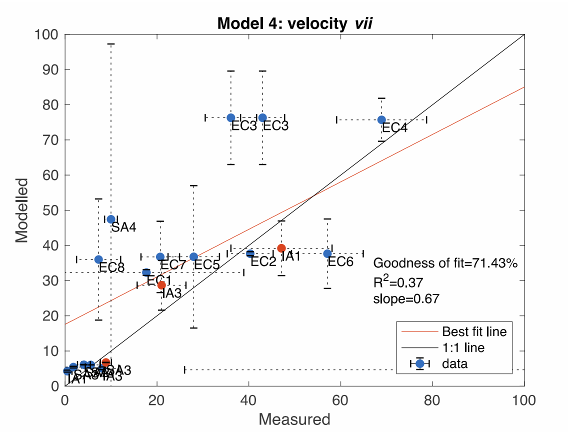

Model 1: velocity *v*
Modelled
Measured
Goodness of fit=42.86%
R^2=0.17
slope=0.37
Best fit line
1:1 line
data
EC5
EC4
EC2
EC8
EC7
EC6
AL1
SA4
IA1
IA3
SA3

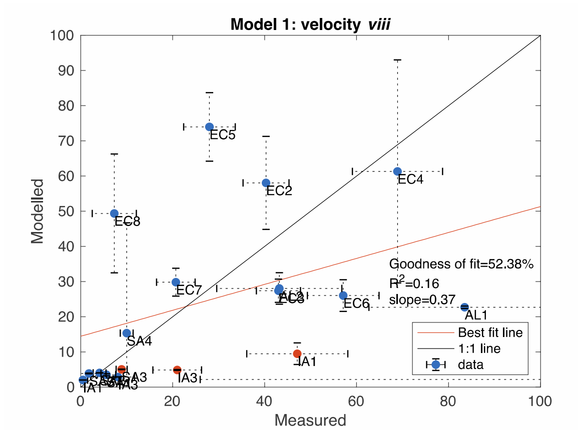

Model 4: velocity *v*
Modelled
Measured
Goodness of fit=52.38%
R^2=0.39
slope=0.66
Best fit line
1:1 line
data
EC3
EC3
EC4
SA4
EC8
EC1
EC7
EC5
EC2
IA1
EC6
IA3
SA3

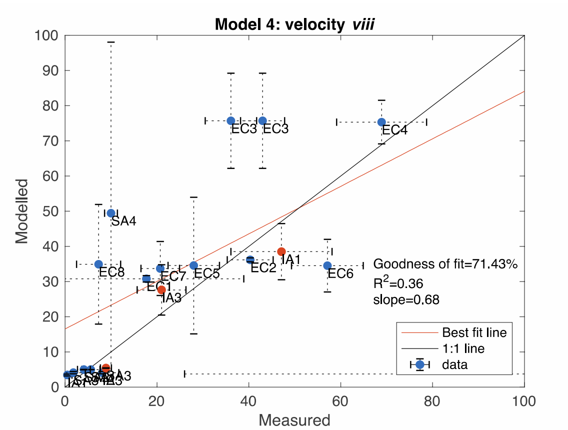

Model 1: velocity *vi*
Modelled
Measured
Goodness of fit=47.62%
R^2=0.19
slope=0.39
Best fit line
1:1 line
data
EC5
IA1
IA3
EC4
EC2
EC8
EC7
EC6
AL2
AL1
SA3
SA4

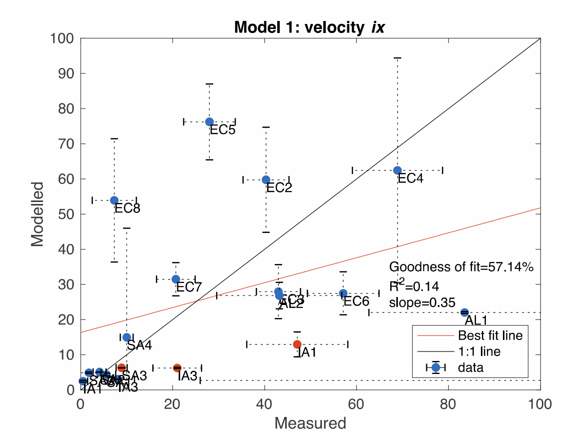

Model 4: velocity *vi*
Modelled
Measured
Goodness of fit=71.43%
R^2=0.37
slope=0.67
Best fit line
1:1 line
data
EC3
EC3
EC4
SA4
IA1
EC8
EC1
EC7
EC5
EC2
EC6
IA3
SA3

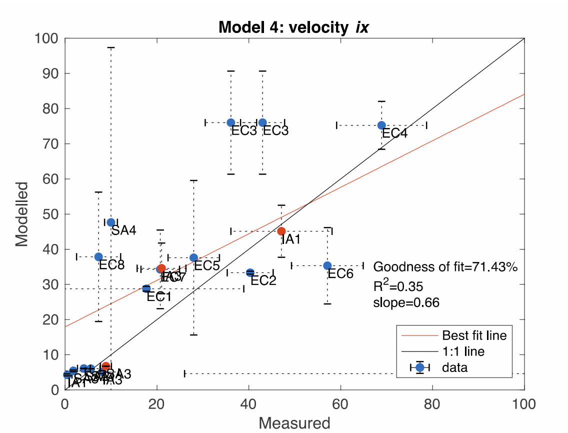

Model 1: velocity *vii*
Goodness of fit=57.14%
R^2=0.17
slope=0.37
Best fit line
1:1 line
data
Measured
Modelled


Model 4: velocity *vii*
Goodness of fit=71.43%
R^2=0.37
slope=0.67
Best fit line
1:1 line
data
Measured
Modelled


Model 1: velocity *viii*
Goodness of fit=52.38%
R^2=0.16
slope=0.37
Best fit line
1:1 line
data
Measured
Modelled


Model 4: velocity *viii*
Goodness of fit=71.43%
R^2=0.36
slope=0.68
Best fit line
1:1 line
data
Measured
Modelled


Model 1: velocity *ix*
Goodness of fit=57.14%
R^2=0.14
slope=0.35
Best fit line
1:1 line
data
Measured
Modelled


Model 4: velocity *ix*
Goodness of fit=71.43%
R^2=0.35
slope=0.66
Best fit line
1:1 line
data
Measured
Modelled

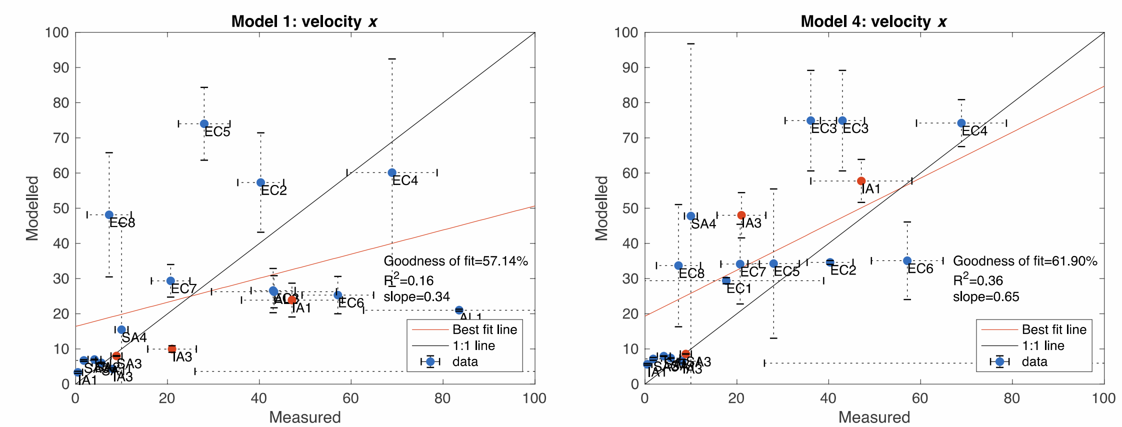


Figure 2.19. Cross plots for all model variations

(all velocities and kinematic Models 1 and 4). Best fit line based on full modelled and measured age variation, and slope of best fit line displayed. R2 from how well measured data is captured by best fit line. Goodness of fit computed by number of measured smaples that fall within model error (±1 Ma and ±2 km) divided by the total number of measured samples.

Supplementary Information for Figure 13 + Discussion on velocities across Bolivia INCLUDING AP SHORTENING (except for Eichelberger)

Source of all data is stated in the box with the associated study. The shortening rate in Figure 13 has been modified from this published data to remove AP shortening (as it is not available for all studies).
= data plotted in figure 13.
= time range where AP is removed

Barnes et al., 2008

time	rate
40	8.72
15	8.72
15	4.466666667
0	4.466666667

Barnes et al., 2008, states that EC, IAZ, and AP shortening is accommodated between 40 and 15 Ma, and that SA shortening is accommodated between 15 and 0 Ma. McQuarrie, 2002 states that there is 259 km of shortening in EC + IAZ + AP, and there is 67 km of shortening in the SA.

McQuarrie et al., 2008

50	8.4
25	8.4
23	2.64
0	2.64

McQuarrie et al., 2008 (Geology) states EC+IAZ+AP shortening (210 km) occurs ~50-25 Ma, and SA shortening (of 66 km) occurs between 23-0 Ma.

Eichelberger et al., 201

time	rate [km
46	7.857
18	7.857
15	5.733
0	5.733

Eichelberger et al., 2013 (p1444, sections 5.1 and 5.2) states that EC and IAZ shortening (of 179 km) is accommodated 46 to ~18 Ma, and SA shortening (86 km-in the abstract) began at 15 Ma.

21°S

Anderson et al., 2018

Removing Computed AP Shortening

PUBLISHED DATA pg 3597 Time [Ma]	Rate [mm/yr]	+	-	PUBLISHED DATA fig 3598, fig 10A Time	Cumulative AP sht	Computed AP rate	wo AP Rate
43	4.5	0.9	0.5		0	0	4.5
35	4.5	0.9	0.5			0	4.5
29	4.5	0.9	0.5	32.5	0	3.125	1.375
28.5	8.4	4.5	0.5	28.6	12.5	3.125	5.275
27	8.4	4.5	0.5	28.5		0	8.4
22	8.4	4.5	0.5				8.4
18	8.4	4.5	0.5	19	12.5	0	3.64
17	8.4	4.5	0.5	18.9		4.759615385	3.64
16.5	7	3.5	3.5	2.33			2.24
11.5	7	3.5	3.5				2.24
11	15.4	10.5	0.3				10.64
10	15.4	10.5	0.3	5			10.64
8	15.4	10.5	0.3	8.5	62	4.759615385	15.28
7.25	15.4	10.5	0.3	8.4		0.119047619	15.28
6	15.4	10.5	0.3				15.28
5	6	1.7	1.7				5.881
1.75	6	1.7	1.7				5.881
1.5	12.4	3.6	3.6				12.28
0	12.4	3.6	3.6	0	63	0.119047619	12.28

In Figure 13, data from Anderson et al., 2018 (p3597, and p3598 Figure 10A) was modified to remove ~7km of AP shortening between 30-27Ma, and ~55km of AP shortening between 19-8Ma.

This Study			
Ages	min velc	Ages	max velocity
45.02		45.64	
43.67	5.78	44.29	5.78
42.29	5.78	42.90	5.78
41.12	5.78	41.73	5.78
39.87	5.78	40.48	5.78
38.75	5.78	39.36	5.78
37.28	5.78	37.89	5.78
35.91	5.78	36.52	5.78
34.96	5.78	35.57	5.78
33.98	5.78	34.60	5.78
33.02	5.78	33.63	5.78
31.65	5.78	32.26	5.78
30.06	5.78	30.67	5.78
28.67	5.78	29.29	5.78
27.45	5.78	28.06	5.78
25.97	5.78	26.59	5.78
24.60	5.78	25.21	5.78
23.38	5.78	23.99	5.78
22.25	5.78	22.87	5.78
20.98	5.78	21.59	5.78
19.44	5.78	20.05	5.78
17.94	5.78	18.55	5.78
16.90	5.78	17.51	5.78
14.83	5.78	15.44	5.78
13.37	5.78	13.98	5.78
12.07	5.78	12.69	5.78
10.78	5.78	11.39	5.78
9.51	5.78	10.47	8.00
8.12	5.78	9.47	8.00
6.83	5.78	8.54	8.00
6.03	10.00	7.54	8.00
5.13	10.00	6.41	8.00
4.28	10.00	5.35	8.00
3.43	10.00	4.29	8.00
2.70	10.00	3.37	8.00
1.93	10.00	2.41	8.00
1.33	10.00	1.66	8.00
0.83	10.00	1.04	8.00
0.00	10.00	0.00	8.00

Rak et al., 2017						
		Rak et al., 2017 Max Var		Rak et al., 2017 Constant Rate		
Shortening (	Cumulative :	Age	Rate	Age	Rate	
0.00	0.00	49.93	11.1	54.81	5.20	This data sourced from Rak et al., 2017 and personal communication.
9.60	9.60	49.06	11.1	52.96	5.20	
10.00	19.60	48.16	11.1	51.04	5.20	
7.30	26.90	47.50	11.1	49.63	5.20	
7.80	34.70	46.80	11.1	48.13	5.20	
12.30	47.00	45.69	11.1	45.77	5.20	
13.10	60.10	44.51	11.1	43.25	5.20	
8.00	68.10	43.79	11.1	41.71	5.20	
8.20	76.30	43.05	11.1	40.13	5.20	
7.60	83.90	42.37	11.1	38.67	5.20	
8.00	91.90	41.22	7	37.13	5.20	
8.00	99.90	39.66	5.1	35.60	5.20	
8.30	108.20	38.03	5.1	34.00	5.20	
9.90	118.10	35.61	4.1	32.10	5.20	
9.40	127.50	32.48	3	30.29	5.20	
10.30	137.80	27.33	2	28.31	5.20	
9.50	147.30	22.58	2	26.48	5.20	
10.00	157.30	19.25	3	24.56	5.20	
10.50	167.80	17.15	5	22.54	5.20	
10.00	177.80	15.33	5.5	20.62	5.20	
9.50	187.30	14.14	8	18.79	5.20	
10.20	197.50	13.12	10	16.83	5.20	
10.00	207.50	12.29	12	14.90	5.20	
10.00	217.50	11.45	12	12.98	5.20	
10.00	227.50	10.34	9	11.06	5.20	
10.00	237.50	8.74	6.25	9.13	5.20	
10.00	247.50	7.14	6.25	7.21	5.20	
10.00	257.50	5.24	5.25	5.29	5.20	
10.10	267.60	3.31	5.25	3.35	5.20	
9.00	276.60	1.60	5.25	1.62	5.20	
8.40	285.00	0.00	5.25	0.00	5.20	

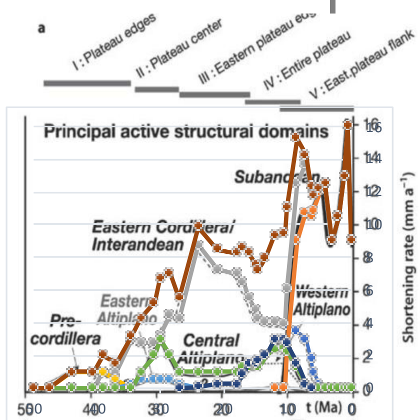
19.5-22°S	
Uba et al 2009	
time	rate
12.4	0.4
10	0.4
9.9	4.5
8	4.5
7.9	1.7
5	1.7
4.9	11
2	11
1.9	7
0	7

Uba et al., 2009 data from Figure 8, p 143. This data is only SA shortening.

Elger et al., 2005; Onkcen et al., 2006

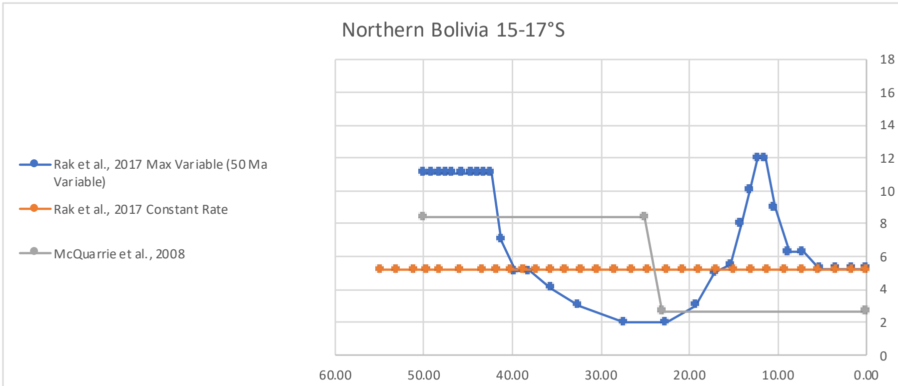
	EC/IAZ	SA	EC+IAZ+SA	Precord	EAP	CAP	WAP	ALL
50	0		0	0		0		0
47.5	0		0	0		0		0
44	0		0	1		0		1
40.5	0		0	1		0		1
39	1		1	1		0		2
37	1		1	0.5	0	0		1.5
34.5	3		3	0	0.25	0		3.25
33	2.8		2.8		0.5	1		4.3
31	2.75		2.75		0.5	2		5.25
30	3.25		3.25		0.5	3		6.75
28.5	4.5		4.5		0.5	2		7
27	4.25		4.25		0.25	1	0	5.5
24	8.75		8.75		0	1	0.1	9.85
21	7.25		7.25			1	0.25	8.5
18	7		7			1	0.25	8.25
17	6.5		6.5			1	0.85	8.6
16	5.5		5.5			1	1.5	8.25
15	4.8		4.8			1	1.5	7.55
14.5	4.25		4.25			1	1.75	7.25
13.5	4		4			2	2	8
12	4	0	4			2	3	9.4
10.5	3.95	0	3.95			3	3	9.45
10	3.8	2.25	6.05			2	2.75	11.05
8.5	3.5	9	12.5			1	1.5	15.25
7.25	3	10.8	13.75			0	0.25	14.25
6.25	1.8	10.5	12.3			0	0	12.3
6	1	10.8	11.8			0		11.8
5	0.25	12	12.25			0		12.25
4	0	12.5	12.5			0		12.5
3		9	9			0		9
2		10.5	10.5			0		10.5
1		13	13			0		13
0.5		16	16			0		16
0		9	9			0		9

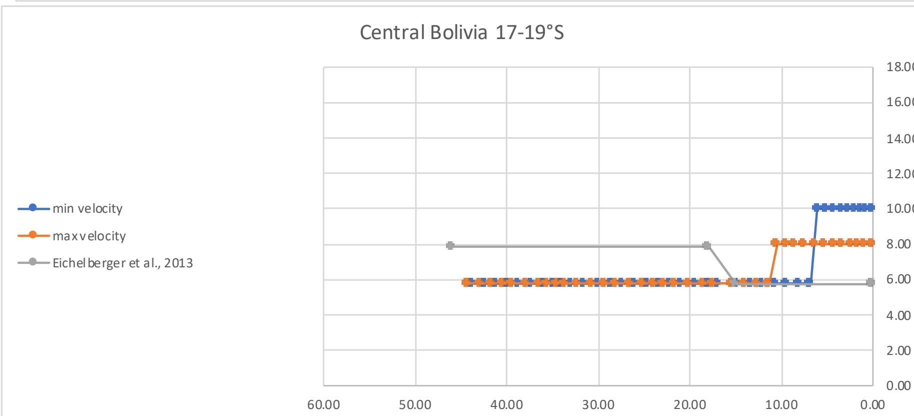
Data extracted from Figure 1.5a, p10 Oncken et al., 2006. The AP shortening rate was removed based on the plot of Fig 1.5a (hidden behind the excel plot below).



<table>
<tr><th>Cross-Section</th><th>Location</th><th>Shortening Amount</th><th colspan="3">Shortening Timing and/or Stated Velocity</th><th>Data Used</th></tr>
<tr><td rowspan="4">15-17°S</td><td>AP</td><td>39[8]</td><td colspan="2" rowspan="3">EC + IAZ + AP 50-25 Ma[2,8]</td><td rowspan="4">Ma[10] | mm/yr[10]
55-50 | 5-7
50-42 | 5-11
42-32 | 2.5-6
32-23 | 2-6
23-15 | 2.5-6
15-5 | 5-12
5-0 | 5-7</td><td>Thermochronology[2,8]</td></tr>
<tr><td>EC</td><td>123[8]</td><td></td></tr>
<tr><td>IAZ</td><td>48[8]</td><td></td></tr>
<tr><td>SA</td><td>75[10]</td><td colspan="2">SA 23-0 Ma[2,8]</td><td>Thermokinematic modelling[10]</td></tr>
<tr><td rowspan="4">17.5°S</td><td>AP</td><td>47[7]</td><td colspan="2" rowspan="3">EC + IAZ + AP 50-17±2 Ma[3,6]</td><td rowspan="4">Ma[12] | mm/yr[12]
50-42 | 5.2-5.9
41-11 | 5.2-6.8
11-7 | 6-8
7-0 | 8-10</td><td>Thermochronology[3,6]</td></tr>
<tr><td>EC</td><td>142[7]</td><td></td></tr>
<tr><td>IAZ</td><td>39[7]</td><td></td></tr>
<tr><td>SA</td><td>72[7]</td><td colspan="2">SA 17±2 - 0 Ma[3,6]</td><td>Thermokinematic modelling[12]</td></tr>
<tr><td rowspan="4">18.5°S</td><td>AP</td><td>Not listed</td><td colspan="2" rowspan="3">EC + IAZ + AP 44 - 14±7 Ma[5]</td><td rowspan="3"></td><td>Thermochronology[5]</td></tr>
<tr><td>EC</td><td>136[5]</td><td></td></tr>
<tr><td>IAZ</td><td>43[5]</td><td></td></tr>
<tr><td>SA</td><td>86[5]</td><td colspan="2">SA 14±7 -0 Ma[5]</td><td>15-0Ma[5] | 5.7-8.5 mm/y[5]</td><td></td></tr>
<tr><td rowspan="4">19°S</td><td>AP</td><td>41[7]</td><td colspan="2" rowspan="3">EC + IAZ + AP 40-15 Ma[2,3]</td><td rowspan="4"></td><td>Thermochronology[2,3]</td></tr>
<tr><td>EC</td><td>122[7]</td><td></td></tr>
<tr><td>IAZ</td><td>96[7]</td><td></td></tr>
<tr><td>SA</td><td>67[7]</td><td colspan="2">SA 15-0 Ma[2,3]</td><td></td></tr>
<tr><td rowspan="4">21°S</td><td>AP</td><td>65[1]</td><td rowspan="3"></td><td>Ma[9] | mm/yr[9]
35-25 | 1-3
25-5 | 1.5-3.7</td><td rowspan="4">Ma[1] | mm/yr[1]
43-36+11.9/-6.1 | 4.5+0.9/-0.5
28-17 | 8.4+4.5/-0.4
17-11 | 7±3.5
11-6.3 | 15.4+10.5/-0.3
5-1.5 | 6±1.7
1.5-0 | 12.4±3.6</td><td>Thermochronology[1]</td></tr>
<tr><td>EC</td><td>120[1]</td><td rowspan="2">40-24 | 0-7
24-15 | 8.8-5
15-10 | 4-4.8
10-5 | 0-3.8</td><td>Sedimentology[9,11]</td></tr>
<tr><td>IAZ</td><td>70[1]</td><td></td></tr>
<tr><td>SA</td><td>82[1]</td><td>Ma[11] | mm/yr[11]
12.4-10 | 0.4
10-8 | 4.5
8-5 | 1.7
5-2 | 11
2-0 | 7</td><td>10-8.5 | 2-9
7-5 | 11
5-4 | 12-13
4-2 | 9-13
2-0 | 9-16</td><td></td></tr>
</table>

Sources: [1]Anderson et al., 2017, 2018; [2]Barnes et al., 2008; [3]Barnes et al., 2012; [4]Echavarria 03; [5]Eichelberger et al., 2013; [6]Lease and Ehlers, 2016; [7]McQuarrie, 2002; [8]McQuarrie et al., 2008
[9]Elger et al., 2005; Oncken et al., 2006; [10]Rak et al., 2017; [11]Uba et al., 09; [12]this paper
Barnes 2006?





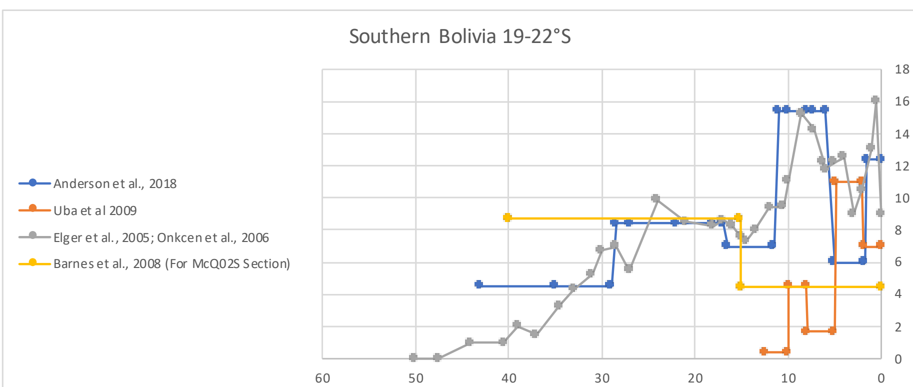


Table 2.5. Supplementary velocity information.

for Figure 2.14 and Discussion section 2.5.3.3

2.9 References

Almendral, A., Robles, W., Parra, M., Mora, A., Ketcham, R. A., & Raghib, M. (2015). FetKin: Coupling kinematic restorations and temperature to predict thrusting, exhumation histories, and thermochronometric ages. AAPG Bulletin, 99(8), 1557–1573. https://doi.org/10.1306/07071411112

Anderson, R. B., Long, S. P., Horton, B. K., Calle, A. Z., & Ramirez, V. (2017). Shortening and structural architecture of the Andean foldthrust belt of southern Bolivia (21°S): Implications for kinematic development and crustal thickening of the central Andes. Geosphere, 13(2), 538–558. https://doi.org/10.1130/GES01433.1

Anderson, R. B., Long, S. P., Horton, B. K., Thomson, S. N., Calle, A. Z., & Stockli, D. F. (2018). Orogenic wedge evolution of the central Andes, Bolivia (21°S): Implications for Cordilleran cyclicity. Tectonics, 37(10), 3577–3609. https://doi.org/10.1029/2018TC005132

Angevine, C. L., Heller, P. L., & Paola, C. (1990). Quantitative Sedimentary Basin Modeling, Continuing Education, Course Note Series #32. AAPG.

Armijo, R., Lacassin, R., Coudurier-Curveur, A., & Carrizo, D. (2015). Coupled tectonic evolution of Andean orogeny and global climate. Earth-Science Reviews, 143, 1–35. https://doi.org/10.1016/j.earscirev.2015.01.005

Ashwal, L. D., Morgan, P., Kelley, S. A., & Percival, J. A. (1987). Heat production in an Archean crustal profile and implications for heat flow and mobilization of heat-producing elements. Earth and Planetary Science Letters, 85(4), 439–450. https://doi.org/10.1016/0012-821X(87)90139-7

Babeyko, A. Y., & Sobolev, S. V. (2005). Quantifying different modes of the late Cenozoic shortening in the central Andes. Geology, 33(8), 621–624. https://doi.org/https://doi.org/1130/G21126.1

Baby, P., Moretti, I., Guillier, B., Limachi, R., Mendez, E., Oller, J., & Specht, M. (1995). Petroleum System of the Northern and Central Bolivian sub-Andean Zone. In A. J. Tankard, R. S. S., & H. J. Welsink (Eds.), Petroleum basins of South America: AAPG Memoir 62 (pp. 445–458). Retrieved from http://search.datapages.com/data/specpubs/memoir62/22baby/0445.htm

Baby, P., Rochat, P., Mascle, G., & Herail, G. (1997). Neogene shortening contribution to crustal thickening in the back arc of the Central Andes. Geology, 25(10), 883–886. https://doi.org/10.1130/0091-7613(1997)025<0883:NSCTCT>2.3.CO;2

Baby, P., Specht, M., Oller, J., Montemurro, G., Colletta, B., & Letouzey, J. (1994). The Boomerang-Chapare transfer zone (recent oil discovery trend in Bolivia): Structural Interpretation and Experimental Approach. In F. Roure, N. Ellouz, V. S. Shein, & I.

Skvortsov (Eds.), Geodynamic Evolution of Sedimentary Basins, International Symposium, Moscow (Editions T, pp. 203–218.). Paris.

Barnes, J. B. (2012). Linking orography, climate, and exhumation across the central Andes: GSA Data Repository item 2012328.

Barnes, J. B., & Ehlers, T. A. (2009). End member models for Andean Plateau uplift. Earth-Science Reviews, 97(1–4), 105–132. https://doi.org/10.1016/j.earscirev.2009.08.003

Barnes, J. B., Ehlers, T. A., Insel, N., McQuarrie, N., & Poulsen, C. J. (2012). Linking orography, climate, and exhumation across the central Andes. Geology, 40(12), 1135–1138. https://doi.org/10.1130/G33229.1

Barnes, J. B., Ehlers, T. A., McQuarrie, N., O'Sullivan, P. B., & Pelletier, J. D. (2006). Eocene to recent variations in erosion across the central Andean fold-thrust belt, northern Bolivia: Implications for plateau evolution. Earth and Planetary Science Letters, 248(1–2), 103–118. https://doi.org/10.1016/j.epsl.2006.05.018

Barnes, J. B., Ehlers, T. A., McQuarrie, N., O'Sullivan, P. B., & Tawackoli, S. (2008). Thermochronometer record of central Andean Plateau growth, Bolivia (19.5°S). Tectonics, 27(3). https://doi.org/10.1029/2007TC002174

Beaumont, C. (1981). Foreland Basins. Royal Astronomical Society Geophysical Journal, 65, 291–329. https://doi.org/10.1111/j.1365-246X.1981.tb02715.x

Beck, S. L., Zandt, G., Myers, S. C., Wallace, T. C., Silver, P. G., & Drake, L. (1996). Crustal thickness variations in the central Andes. Geology, 24(5), 407–410. https://doi.org/https://doi.org/10.1130/0091-7613(1996)024%3C0407:CTVITC%3E2.3.CO;2

Boyer, S. E. (1995). Sedimentary basin taper as a factor controlling the geometry and advance of thrust belts. American Journal of Science. https://doi.org/10.2475/ajs.295.10.1220

Boyer, S. E., & Elliot, D. (1982). Thrust Systems. American Asociation Petrological Geological Bulletin, 66(9), 1196–1230.

Braun, J. (2002). Quantifying the effect of recent relief changes on age-elevation relationships. Earth and Planetary Science Letters, 200(3–4), 331–343. https://doi.org/10.1016/S0012-821X(02)00638-6

Braun, J. (2003). Pecube: A new finite-element code to solve the 3D heat transport equation including the effects of a time-varying, finite amplitude surface topography. Computers and Geosciences, 29(6), 787–794. https://doi.org/10.1016/S0098-3004(03)00052-9

Brooks, B. A., Bevis, M., Whipple, K. X., Ramon Arrowsmith, J., Foster, J., Zapata, T., … Smalley, R. J. (2011). Orogenic-wedge deformation and potential for great earthquakes in the central Andean backarc. Nature Geoscience, 4(6), 380–383. https://doi.org/10.1038/ngeo1143

Calle, A. Z., Horton, B. K., Limachi, R., Stockli, D. F., Uzeda-Orellana, G. V., Anderson, R. B., & Long, S. P. (2018). Cenozoic provenance and depositional record of the Subandean foreland basin during growth and advance of the central Andean fold-thrust belt, southern Bolivia. In & V. R. G. Zamora, K. McClay (Ed.), Petroleum basins and hydrocarbon potential of the Andes of Peru and Bolivia (pp. 459–506). American Association of Petroleum Geologists Memoir Vol. 117. https://doi.org/10.1306/13622132M1173777

Castelluccio, A., Andreucci, B., Zattin, M., Ketcham, R. A., Jankowski, L., Mazzoli, S., & Szaniawski, R. (2015). Coupling sequential restoration of balanced cross sections and low-temperature thermochronometry: The case study of the Western Carpathians. Lithosphere, 7(4), 367–378. https://doi.org/10.1130/L436.1

Catuneanu, O. (2004). Retroarc foreland systems—Evolution through time. Journal of African Earth Sciences, 38(3), 225–242. https://doi.org/https://doi.org/10.1016/j.jafrearsci.2004.01.004

Chapman, A. D. (2017). The Pelona–Orocopia–Rand and related schists of southern California: a review of the best-known archive of shallow subduction on the planet. International Geology Review, 59(5–6), 664–701. https://doi.org/10.1080/00206814.2016.1230836

Dahlen, F. A. (1990). Critical Taper Model of Fold-And-Thrust Belts and Accretionary Wedges. Annual Review of Earth and Planetary Sciences, 18(1), 55–99. https://doi.org/10.1146/annurev.ea.18.050190.000415

DeCelles, P. G. (2012). Foreland basin systems revisited: Variations in response to tectonic setting. In C. Busby & A. Azor (Eds.), Tectonics of Sedimentary Basins: Recent Advances (1st ed., pp. 405–426). Oxford, UK: Blackwell. https://doi.org/https://doi.org/10.1002/9781444347166.ch20

DeCelles, P. G., Ducea, M. N., Kapp, P., & Zandt, G. (2009). Cyclicity in Cordilleran orogenic systems. Nature Geoscience, 2(4), 251–257. https://doi.org/10.1038/ngeo469

DeCelles, P. G., & Horton, B. K. (2003). Early to middle Tertiary foreland basin development and the history of Andean Early to middle Tertiary foreland basin development and the history of Andean crustal shortening in Bolivia. Geological Society of America Bulletin, (May 2009), 58–77. https://doi.org/10.1130/0016-7606(2003)115

Donelick, R. A., O'Sullivan, P. B., & Ketcham, R. A. (2005). Apatite Fission-Track Analysis. In P. W. Reiners & T. A. Ehlers (Eds.), Low-Temperature Thermochronolgy: Techniques, Interpretations, and Applications: Reviews in Mineralogy & Geochemistry,v. 58 (pp. 49–94). https://doi.org/10.2138/rmg.2005.58.3

Echavarria, L., Hernandez, R., Allmendinger, R., & Reynolds, J. (2003). Subandean thrust and fold belt of northwestern Argentina: Geometry and timing of the Andean evolution. American Asociation Petrological Geological Bulletin, 87(6), 965–985. https://doi.org/https://doi.org/10.1306/01200300196

Ege, H., Sobel, E. R., Scheuber, E., & Jacobshagen, V. (2007). Exhumation history of the southern Altiplano plateau (southern Bolivia) constrained by apatite fission track thermochronology. Tectonics, 26(1), 1–24. https://doi.org/10.1029/2005TC001869

Ege, Harald. (2004). Exhumations-und Hebungsgeschichte der zentralen Anden in Südbolivien durch Spaltspur-Thermochoronolgie an Apatit.

Ehlers, T. A. (2005). Crustal Thermal Processes and the Interpretation of Thermochronometer Data. In Peter W. Reiners & T. A. Ehlers (Eds.), Low-Temperature Thermochronolgy: Techniques, Interpretations, and Applications: Reviews in Mineralogy & Geochemistry, v. 58 (pp. 315–350). https://doi.org/https://doi.org/10.2138/rmg.2005.58.12

Ehlers, T. A., & Farley, K. A. (2003). Apatite (U-Th)/He thermochronometry: Methods and applications to problems in tectonic and surface processes. Earth and Planetary Science Letters, 206(1–2), 1–14. https://doi.org/10.1016/S0012-821X(02)01069-5

Eichelberger, N., McQuarrie, N., Ehlers, T. A., Enkelmann, E., Barnes, J. B., & Lease, R. O. (2013). New constraints on the chronology, magnitude, and distribution of deformation within the central Andean orocline. Tectonics, 32(5), 1432–1453. https://doi.org/10.1002/tect.20073

Eichelberger, N., McQuarrie, N., Ryan, J., Karimi, B., Beck, S. L., & Zandt, G. (2015). Evolution of crustal thickening in the central Andes, Bolivia. Earth and Planetary Science Letters, 426, 191–203. https://doi.org/10.1016/j.epsl.2015.06.035

Elger, K., Oncken, O., & Glodny, J. (2005). Plateau-style accumulation of deformation: Southern Altiplano. Tectonics, 24(TC4020), 1–19. https://doi.org/10.1029/2004TC001675

Erdös, Z., Van Der Beek, P., & Huismans, R. S. (2014). Evaluating balanced section restoration with thermochronology data: A case study from the Central Pyrenees. Tectonics, 33(5), 617–634. https://doi.org/10.1002/2013TC003481

Espurt, N., Brusset, S., Baby, P., Hermoza, W., Bolaños, R., Uyen, D., & Déramond, J. (2008). Paleozoic structural controls on shortening transfer in the Subandean foreland thrust sytem, Ene and southern Ucayali basins, Peru. Tectonics, 27(3), 1–21. https://doi.org/10.1029/2007TC002238

Faccenna, C., Becker, T. W., Conrad, C. P., & Husson, L. (2013). Mountain building and mantle dynamics. Tectonics, 32(1), 80–93. https://doi.org/10.1029/2012TC003176

Garzione, C. N., Auerbach, D. J., Jin-Sook Smith, J., Rosario, J. J., Passey, B. H., Jordan, T. E., & Eiler, J. M. (2014). Clumped isotope evidence for diachronous surface cooling of the Altiplano and pulsed surface uplift of the Central Andes. Earth and Planetary Science Letters, 393, 173–181. https://doi.org/10.1016/j.epsl.2014.02.029

Garzione, C. N., Hoke, G. D., Libarkin, J. C., Withers, S., MacFadden, B. J., Eiler, J., … Mulch, A. (2008). Rise of the Andes. Science, 320(5881), 1304–1307. https://doi.org/10.1126/science.1148615

Garzione, C. N., McQuarrie, N., Perez, N. D., Ehlers, T. A., Beck, S. L., Kar, N., … Horton, B. K. (2017). Tectonic Evolution of the Central Andean Plateau and Implications for the Growth of Plateaus. Annual Review of Earth and Planetary Sciences, 45(1), 529–559. https://doi.org/10.1146/annurev-earth-063016-020612

Garzione, C. N., Molnar, P., Libarkin, J. C., & MacFadden, B. J. (2006). Rapid late Miocene rise of the Bolivian Altiplano: Evidence for removal of mantle lithosphere. Earth and Planetary Science Letters, 241(3–4), 543–556. https://doi.org/10.1016/j.epsl.2005.11.026

Gasparini, N. M., & Whipple, K. X. (2014). Diagnosing climatic and tectonic controls on topography: Eastern flank of the northern Bolivian Andes. Lithosphere, 6(May), 230–250. https://doi.org/10.1130/L322.1

Gillis, R. J., Horton, B. K., & Grove, M. (2006). Thermochronology, geochronology, and upper crustal structure of the Cordillera Real: Implications for Cenozoic exhumation of the central Andean plateau. Tectonics, 25(6), 1–22. https://doi.org/10.1029/2005TC001887

Gilmore, M. E., Mcquarrie, N., Eizenhöfer, P. R., & Ehlers, T. A. (2018). Testing the effects of topography , geometry and kinematics on modeled thermochronometer cooling ages in the eastern Bhutan Himalaya. Solid Earth, 9(May), 599–627. https://doi.org/https://doi.org/10.5194/se-9-599-2018

Gonfiantini, R., Roche, M., Olivry, J., Fontes, J., & Maria, G. (2001). The altitude effect on the isotopic composition of tropical rains. Chemical Geology, 181, 147–167. https://doi.org/https://doi.org/10.1016/S0009-2541(01)00279-0

Gotberg, N., McQuarrie, N., & Caillaux, V. C. (2010). Comparison of crustal thickening budget and shortening estimates in southern Peru (12-14S): Implications for mass balance and rotations in the "Bolivian orocline." Bulletin of the Geological Society of America, 122(5–6), 727–742. https://doi.org/10.1130/B26477.1

Gubbels, T. L., Isacks, B. L., & Farrar, E. (1993). High-level surfaces, plaeau uplift, and foreland development, Bolivian central Andes. Geology, 21(August), 695–698. https://doi.org/10.1130/0091-7613(1993)021<0695:HLSPUA>2.3.CO;2

Guenthner, W. R., Reiners, P. W., Ketcham, R. A., Nasdala, L., & Giester, G. (2013). Helium diffusion in natural zircon: radiation damage, anisotropy, and the interpretation of zircon (U-TH)/He thermochronology. American Journal of Science, 313(3), 145–198. https://doi.org/10.2475/03.2013.01

Gurnis, M. (1993). Depressed continental hypsometry behind oceanic trenches: a clue to subduction controls on sea-level change. Geology, 21(1), 29–32. https://doi.org/10.1130/0091-7613(1993)021<0029:DCHBOT>2.3.CO;2

Hatcher, R. D. J., & Hooper, R. J. (1992). Evolution of crystalline thrust sheets in the internal parts of mountain chains. In Thrust Tectonics (pp. 217–233). Chapman & Hall.

Heier, K. S., & Adams, J. A. S. (1965). Concentration of radioactive elements in deep crustal material. Geochimica et Cosmochimica Acta, 29(1), 53–61. https://doi.org/10.1016/0016-7037(65)90078-5

Henry, S. G., & Pollack, H. N. (1988). Terrestrial heat flow above the Andean Subduction Zone in Bolivia and Peru. Journal of Geophysical Research: Solid Earth, 93(B12), 15153–15162. https://doi.org/10.1029/jb093ib12p15153

Hindle, D., & Burkhard, M. (1999). Strain, displacement and rotation associated with the formation of curvature in fold belts; the example of the Jura arc. Journal of Structural Geology, 21(8–9), 1089–1101. https://doi.org/10.1016/S0191-8141(99)00021-8

Hindle, D., Kley, J., Oncken, O., & Sobolev, S. (2005). Crustal balance and crustal flux from shortening estimates in the Central Andes. Earth and Planetary Science Letters, 230(1–2), 113–124. https://doi.org/10.1016/j.epsl.2004.11.004

Horton, B. K. (2005). Revised deformation history of the central Andes: Inferences from Cenozoic foredeep and intermontane basins of the Eastern Cordillera, Bolivia. Tectonics, 24, n/a-n/a. https://doi.org/10.1029/2003TC001619

Horton, B. K. (2012). Cenozoic Evolution of Hinterland Basins in the Andes and Tibet. In Cathy Busby & A. Azor (Eds.), Tectonics of Sedimentary Basins: Recent Advances (pp. 427–444). Blackwell Publishing Ltd. https://doi.org/10.1002/9781444347166.ch21

Horton, B. K. (2018a). Sedimentary record of Andean mountain building. Earth-Science Reviews, 178(March), 279–309. https://doi.org/10.1016/j.earscirev.2017.11.025

Horton, B. K. (2018b). Tectonic Regimes of the Central and Southern Andes: Responses to Variations in Plate Coupling During Subduction. Tectonics, 37(2), 402–429. https://doi.org/10.1002/2017TC004624

Horton, B. K., Hampton, B. A., Lareau, B. N., & Baldellon, E. (2002). Tertiary provenance history of the northern and central Altiplano (Central Andes, Bolivia); a detrital record of plateau-margin tectonics. Journal of Sedimentary Research, 72(5), 711–726. https://doi.org/10.1306/020702720711

Horton, B. K., Hampton, B. A., & Waanders, G. L. (2001). Paleogene synorogenic sedimentation in the Altiplano plateau and implications for initial mountain building in the Central Andes. Bulletin of the Geological Society of America, 113(11), 1387–1400. https://doi.org/10.1130/0016-7606(2001)113<1387:PSSITA>2.0.CO;2

Isacks, B. L. (1988). Uplift of the Central Andean Plateau and bending of the Bolivian orocline. Journal of Geophysical Research, 93(B4), 3211–3231. https://doi.org/https://doi.org/10.1029/JB093iB04p03211

Jaupart, C., & Mareschal, J. C. (2011). Heat Generation and Transport in the Earth. New York: Cambridge University Press.

Jaupart, C., Mareschal, J. C., & Iarotsky, L. (2016). Radiogenic heat production in the continental crust. Lithos, 262, 398–427. https://doi.org/10.1016/j.lithos.2016.07.017

Jordan, T. E. (1981). Thrust Loads and Foreland Basin Evolution, Cretaceous, Western United States. American Asociation Petrological Geological Bulletin, 65(12 (December)), 2506–2520. https://doi.org/https://doi.org/10.1306/03B599F4-16D1-11D7-8645000102C1865D

Jordan, T. E. (1995). Retroarc Foreland and Related Basins. In R. V. Ingersoll & C. J. Busby (Eds.), Tectonics of Sedimentary Basins (1st ed., pp. 331–362). Blackwell Science.

Kennan, L., Lamb, S., & Rundle, C. (1995). K-Ar dates from the Altiplano and Cordillera Oriental of Bolivia: Implicaitons for Cenozoic stratigraphy and tectonics. Journal of South American Earth Sciences, 8(2), 163–186. https://doi.org/https://doi.org/10.1016/0895-9811(95)00003-X

Ketcham, R. A. (1996). Distribution of heat-producing elements in the upper and middle crust of southern and west central Arizona: Evidence from the core complexes. Journal of Geophysical Research: Solid Earth, 101(B6), 13611–13632. https://doi.org/10.1029/96JB00664

Kirby, E., & Whipple, K. X. (2012). Expression of active tectonics in erosional landscapes. Journal of Structural Geology, 44, 54–75. https://doi.org/10.1016/j.jsg.2012.07.009

Kley, J. (1996). Transition from basement-involved to thin-skinned thrusting in the Cordillera Oriental of southern Bolivia. Tectonics, 15(4), 763. https://doi.org/10.1029/95TC03868

Kley, J. (1999). Geologic and geometric constraints on a kinematic model of the Bolivian orocline. Journal of South American Earth Sciences, 12(2), 221–235. https://doi.org/10.1016/S0895-9811(99)00015-2

Kley, J., Monaldi, C. R., & Salfity, J. A. (1999). Along-strike segmentation of the Andean foreland: Causes and consequences. Tectonophysics, 301(1–2), 75–94. https://doi.org/10.1016/S0040-1951(98)90223-2

Lamb, S. (2011). Did shortening in thick crust cause rapid Late Cenozoic uplift in the northern Bolivian Andes? Journal of the Geological Society, 168(5), 1079–1092. https://doi.org/10.1144/0016-76492011-008

Lamb, Simon, & Hoke, L. (1997). Origin of the high plateau in the Central Andes, Bolivia, South America. Tectonics, 16(4), 623–649. https://doi.org/10.1029/97TC00495

Lease, R. O., & Ehlers, T. A. (2013). Incision into the eastern Andean Plateau during Pliocene cooling. Science, 341(6147), 774–776. https://doi.org/10.1126/science.1239132

Lease, R. O., Ehlers, T. A., & Enkelmann, E. (2016). Large along-strike variations in the onset of Subandean exhumation: Implications for Central Andean orogenic growth. Earth and Planetary Science Letters, 451, 62–76. https://doi.org/10.1016/j.epsl.2016.07.004

Leeder, M. R. (1995). Continental Rifts and Proto-Oceanic Rift Troughs. In R. V. Ingersoll & C. J. Busby (Eds.), Tectonics of Sedimentary Basins (1st ed., pp. 119–148). Blackwell Science.

Leier, A. L., McQuarrie, N., Horton, B. K., & Gehrels, G. E. (2010). Upper Oligocene Conglomerates of the Altiplano, Central Andes: The Record of Deposition and Deformation Along the Margin of a Hinterland Basin. Journal of Sedimentary Research, 80(8), 750–762. https://doi.org/10.2110/jsr.2010.064

Leier, A., McQuarrie, N., Garzione, C. N., & Eiler, J. (2013). Stable isotope evidence for multiple pulses of rapid surface uplift in the central Andes, Bolivia. Earth and Planetary Science Letters, 371–372, 49–58. https://doi.org/10.1016/j.epsl.2013.04.025

Mareschal, J. C., & Jaupart, C. (2013). Radiogenic heat production, thermal regime and evolution of continental crust. Tectonophysics, 609, 524–534. https://doi.org/10.1016/j.tecto.2012.12.001

Marshall, L. G., & Sempere, T. (1991). The Eocene to Pleistocene Vertebrates of Bolivia and Their Stratigraphic Context: A Review. In R. Suarez-Suruco (Ed.), Fosiles y facies de Bolivia Vertebrados. (pp. 631–652). Rev. Tec. Yac. Petrol. Fis. Bolivia. Retrieved from http://www.documentation.ird.fr/hor/fdi:010024181

Marshall, L. G., Sempere, T., & Gayet, M. (1993). The Petaca (Late Oligocene–Middle Miocene) and Yecua (Late Miocene) formations of the Subandean Chaco basin, Bolivia and their tectonic significance. Documents Du Laboratoire de Geology de Lyon, 125, 291–301. Retrieved from http://www.documentation.ird.fr/hor/fdi:010024184

McQuarrie, N. (2002). The kinematic history of the central Andean fold-thrust belt, Bolivia: Implications for building a high plateau. Bulletin of the Geological Society of America, 114(8), 950–963. https://doi.org/10.1130/0016-7606(2002)114<0950:TKHOTC>2.0.CO;2

McQuarrie, N., Barnes, J. B., & Ehlers, T. A. (2008). Geometric, kinematic, and erosional history of the central Andean Plateau, Bolivia (15-17°S). Tectonics, 27(3). https://doi.org/10.1029/2006TC002054

McQuarrie, N., & Davis, G. H. (2002). Crossing the several scales of strain-accomplishing mechanisms in the hinterland of the central Andean fold-thrust belt, Bolivia. Journal of Structural Geology, 24(10), 1587–1602. https://doi.org/10.1016/S0191-8141(01)00158-4

McQuarrie, N., & DeCelles, P. G. (2001). Geometry and structural evolution of the central Andean. Tectonics, 20(5), 669–692. https://doi.org/https://doi.org/10.1029/2000TC001232

McQuarrie, N., & Ehlers, T. A. (2015). Influence of thrust belt geometry and shortening rate on thermochronometer cooling ages: Insights from thermokinematic and erosion modeling of the Bhutan Himalaya. Tectonics, 34(1), 1–25. https://doi.org/10.1002/2014TC003783

McQuarrie, N., & Ehlers, T. A. (2017). Techniques for understanding fold-thrust belt kinematics, and thermal evolution. In R. D. Law, J. R. Thigpen, A. J. Merschat, & H. H. Stowell (Eds.), Linkages and Feedbacks in Orogenic Systems: Geological Society of America Memoir 213 (Vol. 1213, pp. 1–30). https://doi.org/10.1130/2017.1213(02)

McQuarrie, N., Ehlers, T. A., Barnes, J. B., & Meade, B. J. (2008). Temporal variation in climate and tectonic coupling in the central Andes. Geology, 36(12), 999–1002. https://doi.org/10.1130/G25124A.1

McQuarrie, N., Horton, B. K., Zandt, G., Beck, S. L., & DeCelles, P. G. (2005). Lithospheric evolution of the Andean fold–thrust belt, Bolivia, and the origin of the central Andean plateau. Tectonophysics, 399(1–4), 15–37. https://doi.org/10.1016/j.tecto.2004.12.013

Mitra, G. (1997). Evolution of salients in a fold- and-thrust belt: the effects of sedimentary basin geometry, strain distribution and critical taper. In S. Sengupta (Ed.), Evolution of Geological Structures in Micro- to Macro-scales (pp. 59–90). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-011-5870-1_5

Mitra, G., & Boyer, S. E. (1986). Energy balance and deformation mechanisms of duplexes. Journal of Structural Geology, 8(3–4), 291–304. https://doi.org/10.1016/0191-8141(86)90050-7

Mitrovica, J. X., Beaumont, C., & Jarvis, G. T. (1989). Tilting of continental interiors by the dynamical effects of subduction. Tectonics, 8(5), 1079–1094. https://doi.org/10.1029/TC008i005p01079

Mora, A., Casallas, W., Ketcham, R. A., Gomez, D., Parra, M., Namson, J., … Ghorbal, B. (2015). Kinematic restoration of contractional basement structures using thermokinematic models: A key tool for petroleum system modeling. AAPG Bulletin, 99(8), 1575–1598. https://doi.org/10.1306/04281411108

Müller, J. P., Kley, J., & Jacobshagen, V. (2002). Structure and Cenozoic kinematics of the Eastern Cordillera, southern Bolivia (21°S). Tectonics, 21(5), 1-1-1–24. https://doi.org/10.1029/2001TC001340

Oncken, O., Boutelier, D., Dresen, G., & Schemmann, K. (2012). Strain accumulation controls failure of a plate boundary zone: Linking deformation of the Central Andes and lithosphere mechanics. Geochemistry, Geophysics, Geosystems, 13(12), 1–22. https://doi.org/10.1029/2012GC004280

Oncken, O., Hindle, D., Kley, J., Elger, K., Victor, P., & Schemmann, K. (2006). Deformation of the central Andean upper plate system—Facts, fiction, and constraints for plateau models. In et al. O. Oncken (Ed.), The Andes—Active Subduction Orogeny (pp. 3–27). Berlin: Springer- Verlag. https://doi.org/https://doi.org/10.1007/978-3-540-48684-8_1

Perez, N. D., Horton, B. K., McQuarrie, N., Stubner, K., & Ehlers, T. A. (2016). Andean shortening, inversion and exhumation associated with thin- and thick-skinned deformation in southern Peru. Geological Magazine, 1–29. https://doi.org/10.1017/S0016756816000121

Rak, A. J., McQuarrie, N., & Ehlers, T. A. (2017). Kinematics, exhumation, and sedimentation of the north-central Andes (Bolivia): An integrated thermochronometer and thermokinematic modeling approach. Tectonics, 36, 1–31. https://doi.org/10.1002/2016TC004440

Reiners, Peter W. (2005). Zircon (U-Th)/He Thermochronometry. In Peter W. Reiners & T. A. Ehlers (Eds.), Low-Temperature Thermochronolgy: Techniques, Interpretations, and Applications: Reviews in Mineralogy & Geochemistry, v. 58 (pp. 151–179). https://doi.org/10.2138/rmg.2005.58.6

Reiners, Peter W., Farley, K. A., & Hickes, H. J. (2002). He diffusion and (U–Th)/He thermochronometry of zircon: initial results from Fish Canyon Tuff and Gold Butte. Tectonophysics, 349, 297–308. https://doi.org/10.1016/S0040-1951(02)00058-6

Reiners, Peter W., Spell, T. L., Nicolescu, S., & Zanetti, K. A. (2004). Zircon (U-Th)/He thermochronometry: He diffusion and comparisons with 40Ar/39Ar dating. Geochimica et Cosmochimica Acta, 68(8), 1857–1887. https://doi.org/10.1016/j.gca.2003.10.021

Royden, L. H. (1993). The tectonic expression slab pull at continental convergent boundaries. Tectonics, 12(2), 303–325. https://doi.org/https://doi.org/10.1029/92TC02248

Ryan, J., Beck, S. L., Zandt, G., Wagner, L., Minaya, E., & Tavera, H. (2015). Central Andean crustal structure from receiver function analysis. Tectonophysics, 682, 120–133. https://doi.org/10.1016/j.tecto.2016.04.048

Saylor, J. E., & Horton, B. K. (2014). Nonuniform surface uplift of the Andean plateau revealed by deuterium isotopes in Miocene volcanic glass from southern Peru. Earth and Planetary Science Letters, 387, 120–131. https://doi.org/10.1016/j.epsl.2013.11.015

Schellart, W. P. (2017). Andean mountain building and magmatic arc migration driven by subduction-induced whole mantle flow. Nature Communications, 8(1), 1–13. https://doi.org/10.1038/s41467-017-01847-z

Sempere, T. (1995). Phanerozoic evolution of Bolivia and adjacent regions. In A. J. Tankard, R. Suarez, & H. J. Welsink (Eds.), Petroleum Basins of South America: AAPG Memoir 62 (pp. 207–230).

Sempere, Thierry, Butler, R. F., Richards, D. R., Marshall, L. G., Sharp, W., & Swisher, C. C. (1997). Stratigraphy and chronology of Upper Cretaceous-lower Paleogene strata in Bolivia and northwest Argentina. Bulletin of the Geological Society of America, 109(6), 709–727. https://doi.org/10.1130/0016-7606(1997)109<0709:SACOUC>2.3.CO;2

Sempere, Thierry, Carlier, G., Soler, P., Fornari, M., Carlotto, V., Jacay, J., … Jiménez, N. (2002). Late Permian-Middle Jurassic lithospheric thinning in Peru and Bolivia, and its bearing on Andean-age tectonics. Tectonophysics, 345(1–4), 153–181. https://doi.org/10.1016/S0040-1951(01)00211-6

Springer, M. (1999). Interpretation of heat-flow density in the Central Andes. Tectonophysics, 306(3–4), 377–395. https://doi.org/10.1016/S0040-1951(99)00067-0

Stockmal, G. S., Beaumont, C., Nguyen, M., & Lee, B. (2007). Mechanics of thin-skinned fold-and-thrust belts : Insights from numerical models. Geological Society of America Special Paper, 433, 63–98. https://doi.org/10.1130/2007.2433(04).

Sundell, K. E., Saylor, J. E., Lapen, T. J., & Horton, B. K. (2019). Implications of variable late Cenozoic surface uplift across the Peruvian central Andes. Scientific Reports, (February), 1–12. https://doi.org/10.1038/s41598-019-41257-3

Suppe, J. (1983). Geometry and kinematics of fault-bend folding. American Journal of Science, 283(September), 684–721. https://doi.org/https://doi.org/10.2475/ajs.283.7.684

Tawackoli, S. (1999). Andine Entwicklung der Ostkordillere in der Region Tupiza (Südbolivien). Selbstverlag Fachbereich Geowissenschaften, FU Berlin, 203, 1–116.

Turcotte, D. L., & Schubert, G. (1982). Geodynamics. New York: John Wiley.

Uba, Cornelius E., Kley, J., Strecker, M. R., & Schmitt, A. K. (2009). Unsteady evolution of the Bolivian Subandean thrust belt: The role of enhanced erosion and clastic wedge progradation. Earth and Planetary Science Letters, 281(3–4), 134–146. https://doi.org/10.1016/j.epsl.2009.02.010

Uba, Cornelius Eji, Heubeck, C., & Hulka, C. (2005). Facies analysis and basin architecture of the Neogene Subandean synorogenic wedge, southern Bolivia. Sedimentary Geology, 180(3–4), 91–123. https://doi.org/10.1016/j.sedgeo.2005.06.013

Whipp, D. M., Ehlers, T. A., Braun, J., & Spath, C. D. (2009). Effects of exhumation kinematics and topographic evolution on detrital thermochronometer data. Journal of Geophysical Research: Earth Surface, 114(4), 1–20. https://doi.org/10.1029/2008JF001195

Williams, K. E. (1995). Tectonic Subsidence Analysis and Paleozoic Paleogeography of Gondwana. In A. J. Tankard, R. Suarez S., & H. J. Welsink (Eds.), Petroleum Basins of South America: AAPG Memoir 62 (pp. 79–100).

Wobus, C. W., Whipple, K. X., Kirby, E., Snyder, N., Johnson, J., Spyropolou, K., … Sheehan, D. (2006). Tectonics from topography: procedurses, promise, and pitfalls. In Willett, S.D., Hovius, N., Brandon, M.T., and Fisher, D.M., eds., Tectonics, Climate, and Landscape Evolution: Geological Society of America Special Paper 398, Penrose Conference Series (pp. 55–74). https://doi.org/10.1130/2006.2398(04).

Wolfe, M. R., & Stockli, D. F. (2010). Zircon (U-Th)/He thermochronometry in the KTB drill hole, Germany, and its implications for bulk He diffusion kinetics in zircon. Earth and Planetary Science Letters, 295(1–2), 69–82. https://doi.org/10.1016/j.epsl.2010.03.025

3.0 Timing and Drivers of Exhumation and Canyon Incision in the Eastern Peruvian Andes: Insights from Thermokinematic Modelling

3.1 Introduction

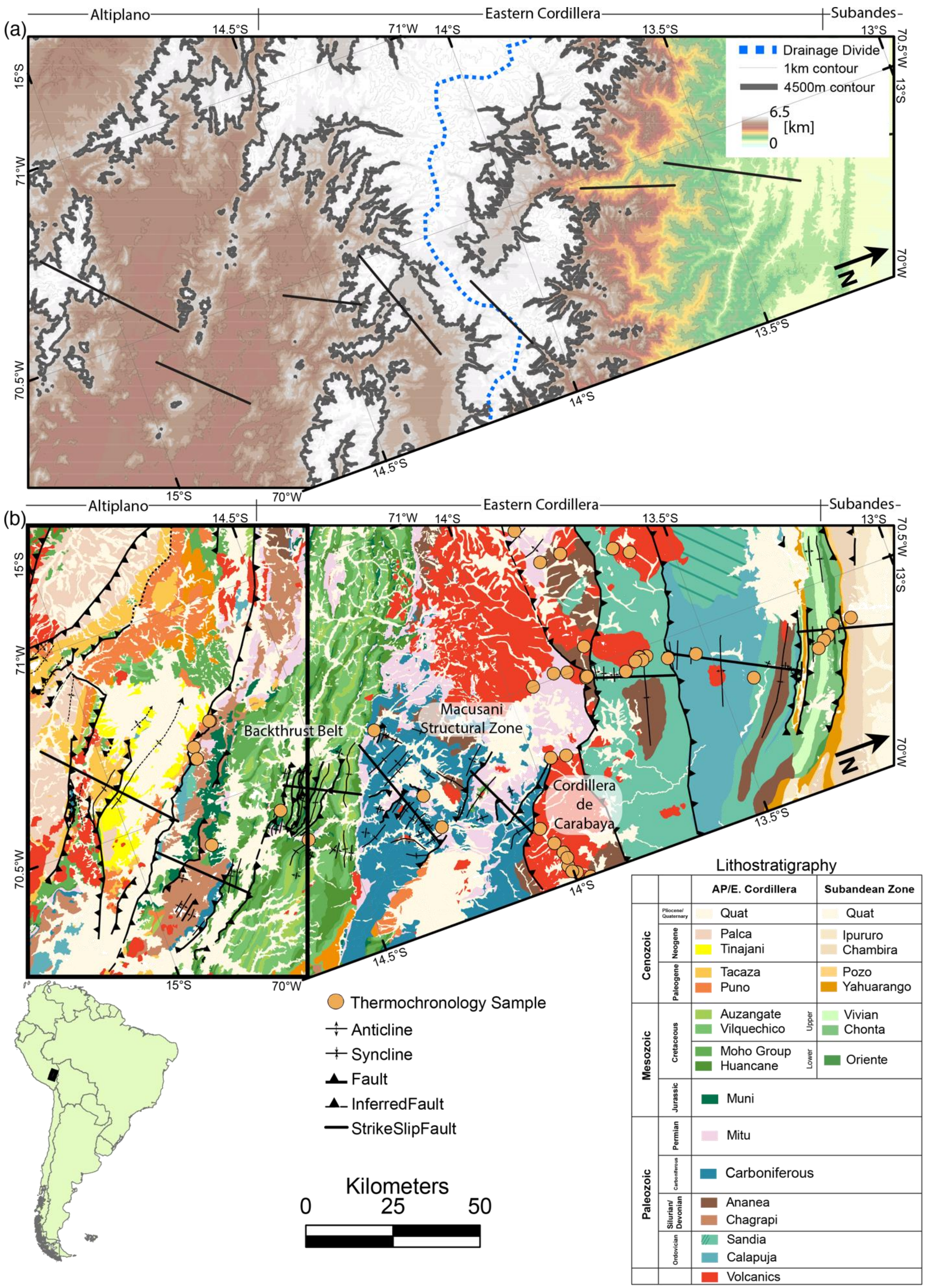
The southern Peruvian Andes are home to some of the deepest incised canyons in the Central Andes, reaching nearly ~3+ km of relief. In southeastern Peru, a unique set of thermochronology samples located along strike at both low, river canyon elevations and high, interfluve elevations present a unique opportunity to evaluate the age, location, and geometry of structural uplift as well as the timing of canyon incision. Quantifying the age of and differentiating the drivers of canyon incision in the relies on a multi-system approach. Incorporating balanced cross-sections, sedimentary basin data, and thermochronology data in a flexurally and thermokinematically modelled section, we evaluate the influence on age and rate of both structural uplift and canyon incision on modelled thermokinematic ages and adjacent interfluve ages. We tested an average, long-term shortening rate for the southern Peruvian Andes (3-4 mm/y), a suite of shortening rates that vary only during Subandean shortening (0.25-2.5 mm/y), and no shortening post 12 Ma. Our best fit model replicates basin ages and depositional contacts, and requires a background shortening rate of 3-4 mm/y until ~12 Ma, when the shortening rate drops to 0.25-0.5 mm/y and shortening continues until present. Age of initiation of incision, independent of local structural uplift, had no measured effect on cooling ages, though incision level (e.g. interfluve, mean, or canyon elevations) was critical to matching the full suite of available thermochronologic data.

Of great debate in the geologic community are the roles and balance between climate and tectonics in shaping morphology, particularly in actively deforming regions. Differentiating the relative roles between climate and tectonics and identifying the strongest influence on morphology is limited by our ability to identify and quantify unique signals imparted from climate or tectonic processes (e.g. Whipple, Kirby, & Brocklehurst, 1999; Whipple & Meade, 2006). In the southern Peruvian Andes, high, broad interfluves, that appear to be extensions of the northern Andean plateau themselves, are separated by deep canyons have been carved into both the eastern and western edges of the plateau. Previous studies have argued that this incision is independent of active deformation, and either driven by marked changes in climate (Jeffery, Ehlers, Yanites, & Poulsen, 2013; Lease & Ehlers, 2013), or by large scale plateau uplift via loss of dense lithosphere or addition of lower crustal material (Garzione et al., 2017; Hoke et al., 2007; Schildgen, Hodges, Whipple, Reiners, & Pringle, 2007). Both drivers for incision are well argued in the literature, and many authors have argued for plateau surface uplift, ranging from 1 – 3.5 km, since 10 Ma in Bolivia (Barke & Lamb, 2006; Garzione et al., 2006; Kennan, Lamb, & Hoke, 1997; Simon Lamb & Hoke, 1997) and ~2.5 km between 9 and 5 Ma in southern Peru (Kar et al., 2016). However, both dynamically driven plateau uplift, as well as climatically enhanced incision, can occur in the presence of structurally driven uplift. This leaves three proposed drivers of incision: (1) enhanced climate, (2) plateau uplift, or (3) active shortening along faults that can either independently or potentially combine to produce the recorded exhumation age and magnitude.

In active cordilleran systems, shortening on surface breaking frontal faults is accompanied by uplift over larger and deeper ramps in the hinterland (Allmendinger & Zapata, 2000; Buford Parks & McQuarrie, 2019; DeCelles & Coogan, 2006; Kley, 1996; McQuarrie, Barnes, et al., 2008; McQuarrie & Ehlers, 2017a; Rak et al., 2017). Thus ongoing shortening on low elevation frontal structures are commonly connected to active uplift and enhanced erosion over ramps in the hinterland (McQuarrie et al., 2019) and could be potential drivers of increased canyon incision. In addition, surface shortening is matched by equal amounts of shortening in the lower crust. Accumulation of lower crustal material in the hinterland can thicken the crust and facilitate surface uplift (Eichelberger et al., 2015; Husson & Sempere, 2003). In contrast, the accumulation of lower crust and mantle lithosphere can also develop into a dense, lithospheric root, that may slow shortening rates (DeCelles et al., 2009; Rak et al., 2017) and lower surface elevations (Houseman, Mckenzie, & Molnar, 1981; Krystopowicz & Currie, 2013; Sobolev & Babeyko, 2005). When this dense root is lost by delamination, surface elevations rise to a new, higher isostatic equilibrium. This broadscale plateau uplift would potentially drive incision on the boundaries of the newly elevated plateau. Any of these processes may have occurred during documented Pliocene intensification of climate (Fedorov et al., 2013, 2006) which is argued to have delivered increased moisture flux to the high and dry eastern plateau edge facilitating canyon incision (Lease & Ehlers, 2013). Climate enhanced incision could have occurred independent of, or in the presence of active uplift.

Thermochronometer cooling ages are a direct result of the timing, age, and magnitude of exhumation. This exhumation is a composite of the fault geometry and rate of deformation, which controls location and rate of uplift, as well as climatically driven exhumation and incision. Due to these complexities, the exact thermochronometer cooling age may not reflect an erosional or deformational event at that time, but rather an integrated signal of several uplift and/or erosional periods. This composite history permits that deformation could pre-date a cooling age and subsequent erosion brought the rocks/sample to the surface, or that some portion of the deformation and exhumation could post-date the cooling age but was not of a sufficient magnitude to fully exhume a sample through its closure temperature (Buford Parks & McQuarrie, 2019; McQuarrie & Ehlers, 2017a; Rak et al., 2017). Because of this composite history, cooling ages record both vertical and lateral motion along faults due to their impact on timing and magnitude of exhumation (Gilmore et al., 2018; McQuarrie & Ehlers, 2015, 2017a). In compressional orogens, subsurface fault ramps and surface breaking faults, control the locations of structural uplift (Lock & Willett, 2008; McQuarrie & Ehlers, 2017a). The large, ~10 km thick, basement thrust sheets proposed throughout the Central Andes are the first-order driver of structural uplift with resulting erosional exhumation of up to 8 km. Through this mechanism, the large basement thrust sheets set the primary cooling signal recorded by thermochronometers (Buford Parks & McQuarrie, 2019; McQuarrie & Ehlers, 2017a; Rak et al., 2017). Basement structures have been proposed across the southern Peruvian Andes (Gotberg et al., 2010; Perez, Horton, McQuarrie, et al., 2016), presenting the possibility that these structures also exert a first order control on uplift and exhumation. Comparing thermokinematic model-predicted cooling ages to measured thermochronologic ages can be used to constrain the location, magnitude, and age of these basement thrusts, as well as evaluate the timing and mechanisms of exhumation.

We aim to link the timing, rate, and location of structural shortening to measured cooling ages in order to evaluate proposed causes of incision in southern Peru by presenting a sequentially-deformed, flexural-kinematic balanced cross-section integrated with a thermal model. This methodology allows us to compare published thermochronology and basin ages to modelled ages that are a result of balanced cross-section geometry and kinematics, and to evaluate if thermochronologic ages can be replicated solely with local structural uplift (and the resultant exhumation), determine if shortening continues at constant rates, and gauge if additional climate-driven incision or large-scale plateau uplift is needed.



Altiplano
Eastern Cordillera
Subandes
(a)
14.5°S
71°W
14°S
13.5°S
13°S
15°S
71°W
70.5°W
70°W
13°S
13.5°S
14°S
14.5°S
15°S
70°W
Drainage Divide
1km contour
4500m contour
6.5
[km]
0
N
(b)
Backthrust Belt
Macusani Structural Zone
Cordillera de Carabaya
Thermochronology Sample
Anticline
Syncline
Fault
InferredFault
StrikeSlipFault
Kilometers
0 25 50
Lithostratigraphy
AP/E. Cordillera
Subandean Zone
Cenozoic
Pliocene/Quaternary
Quat
Quat
Neogene
Palca
Tinajani
Ipururo
Chambira
Paleogene
Tacaza
Puno
Pozo
Yahuarango
Mesozoic
Cretaceous
Upper
Auzangate
Vilquechico
Vivian
Chonta
Lower
Moho Group
Huancane
Oriente
Jurassic
Muni
Paleozoic
Permian
Mitu
Carboniferous
Carboniferous
Silurian/Devonian
Ananea
Chagrapi
Ordovician
Sandia
Calapuja
Volcanics

Figure 3.1. Study area

(a) digital elevation model with bolded 4500m contour and drainage divide between systems that drain northward and southward, (b) Geologic map (modified from INGEMMET and Perez et al., (2016b))with thermochronology sample locations (inset corresponds to Figure 3.4)

3.2 Background

3.2.1 Geologic Framework

The central Andes in Southern Peru are a 400 km wide, NW- trending orogen that consists of a subduction trench, Neogene Western Cordillera (WC) magmatic arc, largely internally drained plateau (the Altiplano - AP), Eastern Cordillera (EC), Subandean Zone (SA), and modern foreland basin. Southern Peru marks the northern extent of the Central Andean Plateau, defined as the >3.5 km elevation region encompassing the AP, EC, and WC. The low relief AP contains 4-6 km and locally up to 10 km, of synorogenic Cenozoic sediment in Peru (Figure 3.1) (Carlotto, 2013; Horton, 2018a; Horton, Perez, Fitch, & Saylor, 2015; Perez & Horton, 2014; Sundell et al., 2018). In the north, the Peruvian AP has been breached by the Pachachaca, Apurimac, and Urubamba rivers sometime after 7.4 Ma (Sundell et al., 2018). The EC, a hinterland fold-and-thrust belt with surface exposures of Paleozoic-Mesozoic strata, ecompasses several distinct structural zones (Figure 3.1, Figure 3.2). The backthrust belt primarily exposes Cretaceous strata in Peru in a series of tight folds and faults. The Macusani structural zone, which has an oblique orientation of faults and folds in Carboniferous-Permian strata, has been attributed to Andean-age re-activation of late Paleozoic rift structures (A. H. Clark et al., 1990; Perez, Horton, & Carlotto, 2016; Thierry Sempere et al., 2002). Finally, the Cordillera de Carabaya, a suite of Permo-Triassic plutons, and broadly folded Ordovician and Devonian exposures mark the NE extent of the EC. The SA is the thin-skinned frontal belt with the most recent deformation, exposing Cretaceous through Cenozoic strata.

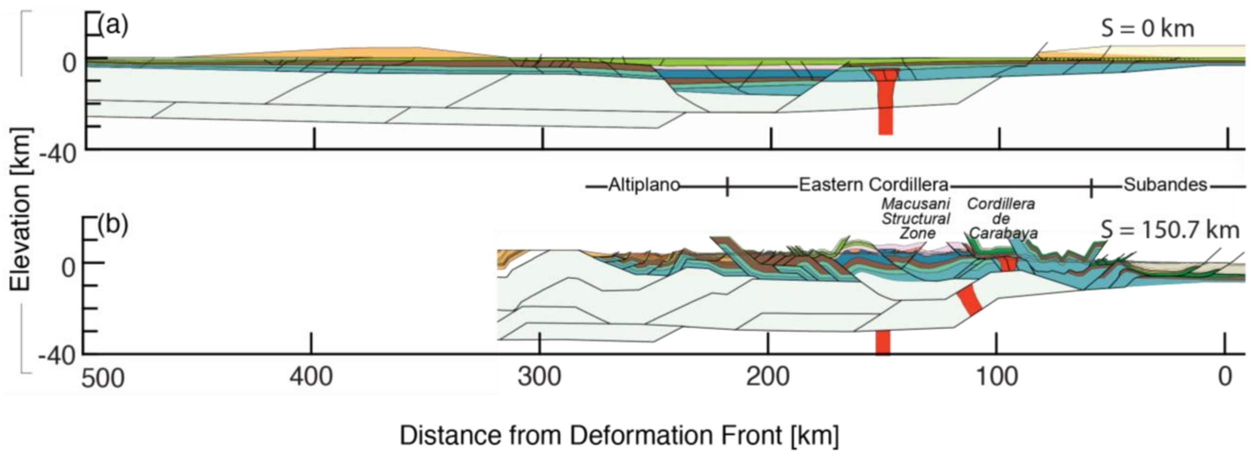


Figure 3.2. Balanced cross-section modified from Perez et al. (2016b) (a) restored and (b) deformed

3.2.2 Regional Thermochronology Record

Published thermochronology data throughout southern Peru show reset or partially reset ages in several low-temperature thermochronology systems, ranging from Biotite Ar-Ar through apatite helium in a suite of samples across the Eastern Cordillera through the Subandes, with nearly ~80% of the data concentrated in the ~2 km deep canyons that dissect the plateau edge (A. H. Clark et al., 1990; Falkowski et al., n.d.; Farrar, Clark, Kontak, & Archibald, 1988; Kontak, Clark, Farrar, Archibald, & Baadsgaard, 1987; Kontak, Farrar, Clark, & Archibald, 1990; Lease & Ehlers, 2013; Perez, 2015; Perez, Horton, McQuarrie, et al., 2016).

We limit the thermochronometer data used in this study to within 50 km of the cross-section to include all potentially relevant samples and exclude lateral variability in structural history. These data are projected along structure to the cross-section line (Figure 3.1, Figure 3.3) (Falkowski et al., n.d.; Kontak, Clark, Farrar, Archibald, & Baadsgaard, 1990; Kontak, Farrar, et al., 1990; Lease & Ehlers, 2013; Perez, 2015; Perez, Horton, McQuarrie, et al., 2016). Apatite and zircon (U-Th)/He (AHe, ZHe), apatite and zircon fission track (AFT, ZFT), and muscovite and biotite $^{40}Ar/^{39}Ar$ (MAr, BAr) ages are plotted as mean or pooled ages when the grain ages represent a unique population, or as individual grain ages, when the individual grain ages are distinct and do not represent a cohesive population. All ages are plotted with 2σ error (Figure 3.3). Although plotted to highlight locations and age of early exhumation, BAr ages are not thermally modeled as the measured ages are younger than ZFT ages, indicating Ar loss.

Previously published $^{40}Ar/^{39}Ar$, AFT, and ZFT thermochronology ages have been used to argue for deformation-induced exhumation in southern Peru beginning in the Eastern Cordillera between 45 and 38 Ma (Benjamin, Johnson, & Naeser, 1987; Farrar et al., 1988; Gillis et al., 2006; Kontak et al., 1987; Perez, 2015; Perez, Horton, McQuarrie, et al., 2016). Young AHe and ZHe cooling ages found in the SA and northeastern EC are attributed to either continued deformation, climate change induced exhumation, or a combination thereof (Lease & Ehlers, 2013; Perez, Horton, McQuarrie, et al., 2016).

Of interest are a suite of samples collected in the Rio San Gaban canyon in southern Peru (Falkowski et al., n.d.; Lease & Ehlers, 2013; Perez, Horton, McQuarrie, et al., 2016), and recently published samples collected on adjacent interfluves (Falkowski et al., n.d.). Interfluve samples are from elevations of 3700 to 5042 m, and are within ~750 m of the max elevation along strike (within a 50 km swath). The canyon samples are collected as close to the Rio San Gaban as possible, ranging in elevation from 484 to ~3300 m, and are at least 750 m lower in elevation than any along-strike interfluve sample. The separation of interfluve and canyon samples ends when the incision of the canyon-interfluve relief drops to less than 1.5 km; after this, all samples are considered to be mean elevations.

From the SA to the EC, fully reset AHe cooling ages at low, canyon elevations (~0.3-3.3 km) are 3-4 Ma, increasing southwestward to 6 Ma, while measured AHe ages from high, interfluve elevations (3.5-5.1 km) increase from ~15 to ~30 Ma to the southwest (Falkowski et al., n.d.; Lease & Ehlers, 2013; Perez, 2015; Perez, Horton, McQuarrie, et al., 2016). In the frontal most portion of the SA, there are two AHe canyon ages that are anomalously old (47.1 ± 2.4, 28.8 ±2.8 Ma). AFT canyon cooling ages are relatively consistent, at ~9 ± 1 Ma in the canyon and one age at the southwestern extent of the canyon ~33 Ma, while interfluve samples are 15-30 Ma (Falkowski et al., n.d.; Kontak, Clark, et al., 1990; Kontak, Farrar, et al., 1990). We consider the canyon AFT sample dated 63.1 ± 31.2 Ma to be an outlier, and not included in these groupings. ZHe reset cooling ages increase southwestward from ~13 to ~22 Ma in the canyon, with likely partially reset outliers of 94.3 ± 22.1 Ma and 75.5 ±17.7 Ma, while interfluve ZHe cooling ages are ~20 to ~40 Ma (Falkowski et al., n.d.; Lease & Ehlers, 2013; Perez, 2015; Perez, Horton, McQuarrie, et al., 2016).

In the EC backthrust belt, AHe cooling ages range from 2.7 Ma (an anomalously young, single grain age) to 24 Ma, with a median age of ~22 Ma (Falkowski et al., n.d.; Perez, 2015). The single AFT age in the backthrust belt is 44 ± 22.2 Ma (Falkowski et al., n.d.). The two ZHe samples in the backthrust belt have two distinct age populations based on single grains: a likely partially reset population around ~80 Ma (83.9 ± 3.7, 76.1 ± 2.7, 87.3 ± 2.4 Ma) and a likely reset population around ~25 Ma (24.5 ± 0.9, 25.6 ±1 Ma).

ZFT cooling ages are currently being analyzed.

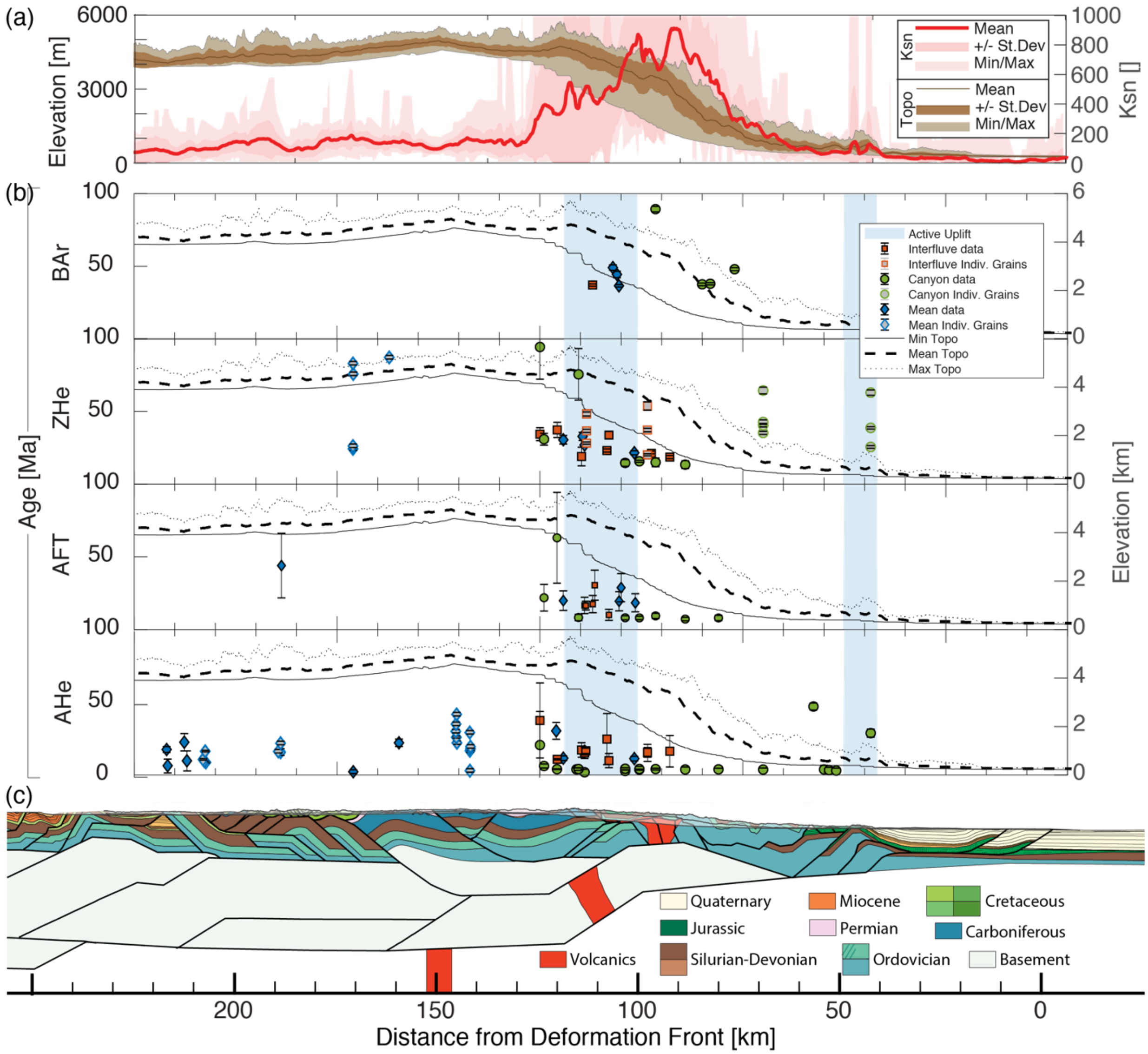


Figure 3.3. Cross-sectional projection of thermochronology in our study area

(a) topography and Ksn from a 50 km swath, (b) regional thermochronology data for BAr, ZHe, AFT, and AHe; grayed markers identify individual grain ages where no mean age has been computed; (c) balanced cross-section

3.2.3 Regional Sedimentology

Synorogenic sedimentation in the Altiplano of southern Peru is preserved in several segregated basins, that show components of a shared depocenter history as well as unique depozones with respect to age, thickness, and composition (Callot, Sempere, Odonne, & Robert, 2008; Carlotto, 2013; Horton, 2018a; Horton et al., 2015; Jaillard et al., 1993; Kar et al., 2016; Lecaros, Moncayo, Vilchez, & Fernándex, 1999; Noblet, Leonardi, Taquet, Marocco, & Cordova, 1995; Perez & Horton, 2014; Saylor & Horton, 2014; Sigé, Sempere, Butler, Marshall, & Crochet, 2004; Sundell et al., 2018). The Ayaviri Basin, which at full width reaches 35 km in the northeastern portions between the Pucapuca-Sorapata Fault and the Ayaviri Fault (Horton et al., 2015), become two sub-basins further south: the Llalli and Tinajani sub-basins (Figure 3.4). The Llalli sub-basin runs ~30 km from the Pucapuca-Sorapata Fault to the Pasani Fault, while the Tinajani Basin runs ~20 km from the Pasani Fault to the Ayaviri Fault (Figure 3.4). These two sub-basins have distinct sedimentation histories, and are separated by the Pasani fault that dies out in a plunging anticline into the Ayaviri Basin proper (Figure 3.1, Figure 3.4). In the southern Peru Llalli sub-basin of the Ayaviri basin, the oldest deposits are the western-derived Eocene-Oligocene San Jeronimo (Puno) Group, of which >3-5 km is preserved unconformably overlain on Cretaceous strata, and is dated from 37 to 28 Ma in the southwestern portions of the AP near Macari and Llalli (Horton et al., 2015). Further east, in the Tinajani sub-basin, the ~1.3 km thick Puno group is dated between ~29 and 24 Ma, and is in direct contact in the east with the Jurassic-Cretaceous Muni Formation (Perez & Horton, 2014). Overlain on the Puno Group in the Tinajani sub-basin is the Neogene Tinajani Formation, which accumulated >1.1 km thickness between 24 and 17 Ma (Perez & Horton, 2014). The Tacaza formation, which includes andesites, volcanic breccias, and rhyolitic tuffs, erupted ~23.5 Ma and unconformably covers portions of all three basins (A. H. Clark et al., 1990; Klinck, Ellison, Hawkins, & Compilers, 1986). There are several angular unconformities preserved in exposures in the Ayaviri Basin, Llalli, and Tinajani sub-basins that highlight differential uplift throughout the basin. Near the northeastern margin of the Tinajani sub-basin, surface exposures of lower Cretaceous (possibly Jurassic) Muni and the Silurian/Devonian Chagrapi Formations are unconformably overlain by the Puno group, while on the southwestern side of the Tinajani sub-basin, the Tinajani Formation is in contact with the lower Ordovician San Jose Formation (Perez & Horton, 2014). The Tinajani sub-basin is also bounded on both sides by faults and dated growth structures: on the southwestern side is the northeast verging Pasani Fault that places the Cretaceous Ayavacas Formation on the Neogene Tinajani Formation, with growth structures dating the last preserved motion between 18-16 Ma (Perez & Horton, 2014). On the northeastern side of the Tinajani sub-basin is the southwest verging Ayaviri Fault, which places middle Ordovician strata on the Oligocene Puno Group, and is dated with growth structures to have had its last preserved motion between 28-26 Ma (Perez & Horton, 2014).

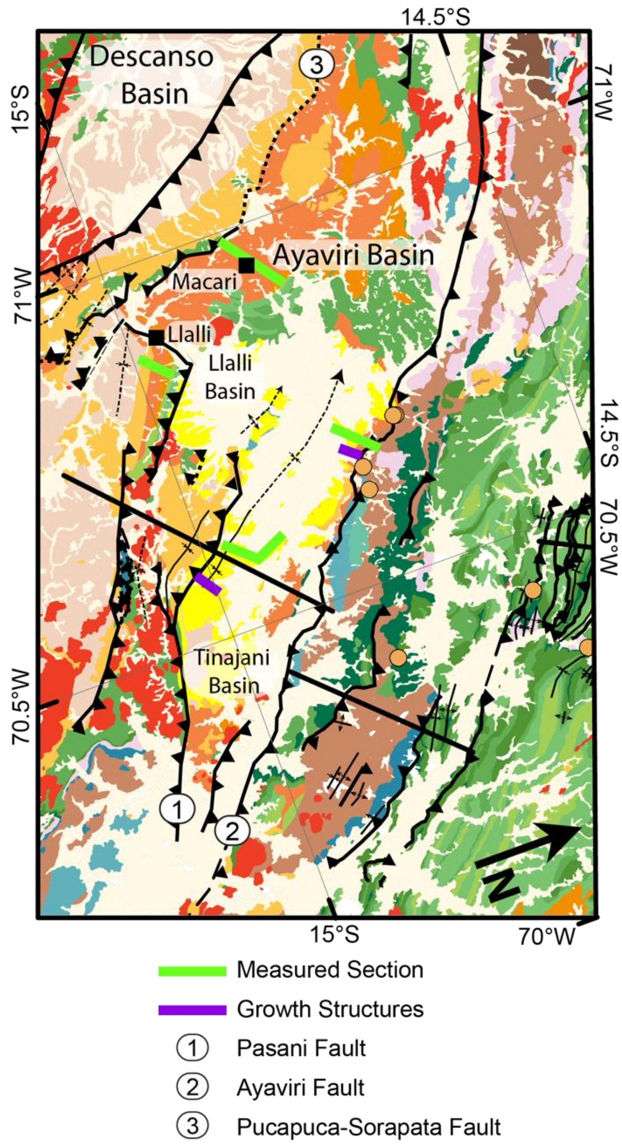


Figure 3.4. Geologic map of the Altiplano in Southern Peru

(modified from INGEMMET and Perez et al., (2016b).

3.2.4 Balanced Cross-Section

Perez et al. (2016) published a balanced cross-section through southern Peru at 13-15°S. This section spans the Eastern Cordillera and Subandean Zone and constrained ~130 km (38%) of Cenozoic orogen Andean shortening (Figure 3.2). The SA portion of the cross section shows 3 horses comprised of Ordovician through Devonian strata that transfer shortening into overlying Cretaceous through Tertiary age strata. Equivalent amounts of shortening (~36 km) in deeper rocks is accommodated by motion of a basement thrust sheet over a ramp located under the high peaks of the EC. Motion on the SA basement fault uplifts folded and faulted Ordovician through Devonian strata and Triassic-age intrusions of the Cordillera de Carabaya. Directly west of the high peaks is the Macusani structural zone. Accumulation of ~5.5 km of the Carboniferous Ambo and Tarma, Permian Copacabana, and Permo-Triassic age, volcaniclastic Mitu formation was facilitated by proposed coeval extension and graben formation (Figure 3.2). This basement-involved graben was inverted during Andean-age deformation forming the modern day, gently folded and faulted late Paleozoic strata. The southwestern portion of the Eastern Cordillera is comprised of folded and faulted Cretaceous strata of the backthrust belt (Figure 3.1, Figure 3.2). Shortening in predominantly Cretaceous age rocks at the surface are balanced by duplexing in Ordovician through Devonian strata. The Eastern Cordillera ends in the south at the Ayaviri Fault, which places Ordovician age rocks over 24-17 Ma Tertiary strata.

3.2.5 Exhumation History

Both the eastern and western sides of the Central Andes are incised by large (1.5-3 km in relief) canyons draining the plateau. These canyons have been interpreted to be the result of recent (post 10 Ma) plateau uplift (Hoke et al., 2007; Schildgen et al., 2007; Whipple & Gasparini, 2014) or the result of climate change either independent of, or associated with changing elevation with time (Jeffery et al., 2013; Lease & Ehlers, 2013). Lease and Ehlers (2013) present a suite of AHe and ZHe canyon ages taken between ~1.5 and ~4 km in elevation in the bottom of the San Gaban Canyon. All six of their AHe ages are young, at ~4 Ma, while their ZHe ages are ~15 Ma with two partially reset older ages (75.5 and 94.3 Ma). They argue that the ZHe ages suggest a cessation of active shortening in the northern EC by 12-15 Ma, and the young AHe ages are interpreted to represent 1.5-2 km of exhumation independent of shortening or uplift. Schildgen et al. (2007, 2009) present a suite of AHe and ZHe canyon ages in vertical transects from the western edge of the Andes, in the Ocoña river canyon. When plotted with respect to a once horizonal, low-elevation paleo-surface, their AHe ages range from 72.5 to 5.3 Ma and show a steep decrease in ages, ~15-5 Ma, between 2-3 km below the paleosurface. They interpret this age/ depth relationship to represent a minimum of 1.0 km and up to 3.0 km of incision starting between 7 and 11 Ma and ending between 2 and 5.5 Ma as a result of plateau uplift (Schildgen et al., 2009). Jeffery et al. (2013) used a Monte Carlo search approach to evaluate the sensitivity of incision to uplift history and resulting changes in precipitation. They argue that the Ocoña River incision from 11-2 Ma is consistent with both steady or punctuated uplift of 1.5-3.5 km since 16 Ma and highlight accelerated incision is due to increase precipitation resulting from both regional and orographic changes in climate (Jeffery et al., 2013).

Uplift driven incision is either a function of uniform uplift of the plateau from ~2 km elevation at ~10 Ma to its modern elevation by ~5 Ma (e.g. Kar et al., 2017; Schildgen et al., 2007), or alternatively, the eastern and western edges of the plateau are locations of ~10 Ma to present uplift on active structures (e.g. Rak et al., 2017). In both cases, incision is a direct function of uplift. Climate driven incision for the eastern plateau edge argues that deformation within the EC had ceased and at least the eastern portion of the plateau reached modern elevations by 12 Ma. In this scenario, ~3-4 Ma canyon incision history recorded in AHe thermochronometers is solely a response to well-documented increased Pliocene climate (Fedorov et al., 2013, 2006; Lease & Ehlers, 2013).

3.3 Methods

3.3.1 Sequential Deformation and Isostasy

In order to quantitatively link the balanced cross-section deformational history with synorogenic sedimentation and compute thermal ages, we use methods that build on the flexural-kinematic methodology from previous work (Buford Parks & McQuarrie, 2019; Ghoshal et al., 2020; Gilmore et al., 2018; Olsen, McQuarrie, & Robinson, 2019; Rak et al., 2017). Using the 2D kinematic module in Petroleum Export's (previously Midland Valley) Move software, the section was iteratively deformed using fault-parallel flow and shear algorithms followed by flexural isostatic loading and unloading. The section was overlaid with a 0.5 × 0.5 km grid of unique points and was displaced in ~10 km increments of shortening. This grid was used to produce vectors of displacement at each grid point that are converted to velocity fields by assigning time to each deformation step. After each ~10 km shortening step, the flexural isostatic thrust load, with a density of ρ_{thrust} = 2500 kg/m^3, was calculated from the difference between the deformed topography and previously undeformed topographic surfaces. The elastic thickness (EET) used in flexural modeling was systematically altered during multiple model iterations to strike the best balance between preserving the geology at the surface and matching basin depths. In our final model, EET increases through time, from a starting EET of 25 km to a maximum of 90 km. As our best EET match is still unable to reproduce surface elevations in the southwestern backthrust belt and the full modern foreland basin depth, uplift and subsidence in the shape of a sinusoidal curve was imposed over the last 70 km of shortening in the model, which elevated the hinterland and subsided the foreland (Buford Parks and McQuarrie, 2019; Rak et al., 2017). A total amount of 7 km of imposed uplift under the AP/ southwestern EC and 2.5 km of imposed subsidence under the

SA were added. The imposed uplift/ subsidence occurs after isostatic loading but before the new topographic surface is estimated.

After each increment of isostatic loading and imposed uplift/ subsidence, we estimated a new topographic surface, filling the basins that have subsided below 0 km, and following the pre-existing surface except when uplift exceeds a set, westward increasing slope (α), typically 1-3°. Erosional removal occurs everywhere above the new topographic surface. In areas that did not experience structural uplift, the new topographic surface follows the existing, isostatically loaded topography. Maximum elevation limits are applied in the EC (6.5 km) and AP (3-4 km), and westward facing slopes are limited to 30°. After isostatic unloading of the eroded strata, we apply an additional load due to sedimentation in the newly created basin increment, with a density of $\rho_{sed} = 2100$ kg/m^3. Throughout the model, we track surface topography through time as a method of calculating isostatic load, tracking basin sedimentation, and identifying surface geology. During the bulk of the modeling process, we track the mean topography across the section, but for the last deformation steps, we tracked three topographies through time, interfluve, canyon and mean. During these steps, we use the mean topography to compute isostasy, but tracking the hypothesized topographies representative of interfluve and canyon elevations during the last of the Subandean shortening allows us to evaluate the sensitivity of modeled cooling ages to differential topography and when that differential topography was established. This method allows us to compute the predicted thermochronology ages for mean, canyon, and interfluve samples.

3.3.2 Thermal Model

The grid of unique points and surface topography from each step in the flexural-kinematic model were used, in combination with thermal parameters, as the input for a modified version of the advection-diffusion thermal modelling code *Pecube* (Braun, 2002, 2003; McQuarrie & Ehlers, 2015; D. M. Whipp et al., 2009; David M. Whipp et al., 2007). This version of *Pecube* solves the three-dimensional heat transport equation to simulate the evolving crustal thermal field based on the input deformation grids and thermal parameters in order to derive the time-temperature history of exhumed rocks based on their transport paths (McQuarrie & Ehlers, 2015; Rak et al., 2017). To calculate model-predicted ages in *Pecube*, published thermochronometer kinetics combined with the rock cooling rate, rather than nominal closure temperatures, are used. These kinetics are as follows: for AHe (Farley, 2000), AFT (Crowley, Cameron, & Schaefer, 1991), ZHe (Peter W. Reiners et al., 2004), and ZFT (Braun, 2003; Ehlers, 2005; Ehlers et al., 2005)(natural, radiation damaged (Brandon, Roden-Tice, & Carver, 1998)).To thermally model interfluve and canyon ages, the same velocities and thermal parameters were used, but the input surface topographies used to compute the model-predicted ages were altered to reflect either canyon or interfluve topography for that thermal model run. The modelled ages from Pecube are directly compared to measured thermochronology ages; matches between measured and modelled ages were identified if the modelled age, with a ±1 Ma and ± 2km error, fell within any portion of the measured age and its error. For samples with individual grain ages, the modelled age was considered a match if it fell within the range of any of the grain ages. For samples across the portion of section with multiple topographies (e.g. the eastern EC and SA), we also take sample elevation into account. Low elevation samples are compared to the predicted ages using the canyon topography. Similarly, higher elevation samples are compared to the predicted ages generated using the mean, or interfluve topographies depending on the sample elevation.

The *Pecube* thermal model extends to a base depth of 110 km with a temperature of 1300°C, and up to the surface, where the temperature at sea level is 23°C (Puerto Maldonado yearly average), and decreases at 4.0°C/km, the mean lapse rate measured in southern Peru (Navarro-Serrano et al., 2020). The model holds the temperature at the surface and base constant. The thermal model is permitted 50 My to equilibrate crustal temperatures prior to initiation of Andean shortening. For the models presented here we use 50 Ma as the initiation of deformation to be older than 37- 50 Ma BAr ages in the Zongo San Gaban Structural Zone and Coasa Pluton (Farrar et al., 1988; Kontak, Clark, et al., 1990; Kontak, Farrar, et al., 1990) to evaluate the sensitivity of model results to age that deformation initiates, we also evaluated initiation at 40 and 45 Ma. We evaluated a suite of predicted ages with varied thermal structures based on surface radiogenic heat production values that range from (A_o=2.0–3.0μW/m^3) and that decreases exponentially with depth using a prescribed e-folding (ef) depth of 12 km (Ghoshal et al., 2020; McQuarrie & Ehlers, 2017a).

3.4 Results

3.4.1 Balanced Cross-Section

The existing Perez et al. (2016b) section only covered the SA through the EC, and stops at the boundary between the EC backthrust belt and the AP at the northeast dipping Ayaviri thrust Fault, which places Paleozoic strata onto synorogenic Oligocene-Miocene strata, and is part of a larger fault system interpreted to be continuous for ~400 km along strike from 14 to 18°S (Carlotto, 2013; Perez & Horton, 2014; Perez, Horton, McQuarrie, et al., 2016; T. Sempere, Herail, Oller, & Bonhomme, 1990). We continued the Perez et al. (2016b) section southwestward by constructing the geometry and amount of shortening through the AP based on the available sedimentologic constraints and mapped geologic strata (Figure 3.2, Figure 3.4). Due to the age differences in the basal Puno Fm between the Llalli (basal age ~37 Ma) and Tinajani (basal age ~29 Ma) sub-basins, the Ayaviri Basin must have been sub-divided in the location of our section. Fault generated uplift is required at ~30 Ma to limit Puno Fm. sedimentation and/or remove the basal portion of the Puno Fm. to the northeast in the Tinajani sub-basin (Figure 3.1, Figure 3.2). Additionally, on the northeastern side of the Tinajani sub-basin, the ~29 Ma lower Puno Formation is directly deposited on the Jurassic-Cretaceous Muni Formation, which supports erosional removal of the upper Cretaceous and younger synorogenic sedimentation prior to 29 Ma, a minimum of ~1 km of strata. Furthermore, on the southwestern side of the Tinajani sub-basin, the presence of surface exposures of the lower Ordovician San Jose Formation in direct contact with the 24-17Ma Tinajani Formation requires uplift and erosional removal of nearly the entire Paleozoic stratigraphic section including any accumulated Mesozoic through Cenozoic strata (at minimum ~ 6+ km of strata) prior to initiation of deposition of the Tinajani Formation at 24 Ma. (Figure 3.2, Figure 3.4).

The separation between the Tinajani and Llalli sub basins as well as the erosional removal of Paleozoic through Tertiary strata is accomplished through motion on a basement fault moving over a ramp located between the two sub-basins (Figure 3.4). The location of the footwall ramp of this northeastern basement thrust is constrained by the mapped unconformity between Tinajani and lower Ordovician on the southwestern border of the Tinajani sub-basin. A second southwestern basement fault is interpreted southwest of the Llalli sub-basin to both provide the uplift needed to define the edge of the basin as well as distribute shortening to the series of surface and subsurface faults that repeat the Paleozoic through Cretaceous section and partition deformation in the sub-basins (e.g. Horton et al., 2015). A lower level basement fault system is proposed to maintain western uplift and a western derived sediment source to the basins (Figure

3.5). The last component of this uplift initiates at 19-17 Ma, consistent with paleoelevation estimates indicating rapid uplift of the Western Cordillera at this time (Saylor & Horton, 2014).

The surface breaking structures in the AP, primarily the Pasani and Llali faults, along with all other faults in the AP, total 16 km of shortening and are balanced at depth via shortening along the two basement thrusts. Our balanced AP section has a break in the section line along strike of significant structures to permit the cross-section to follow the best surface exposures and mapped geology. The AP component of the section added a deformed length of 58 km that equals an undeformed length of 74 km, producing 16 km of shortening (22%) in the AP. The final section covers the SA, EC, and AP at 13-15°S and constrains a total of ~146 km (35%) shortening.

3.4.2 Sequential Deformation and Flexure

Modelled deformation initiates in the northeastern portion of the Eastern Cordillera in the west-verging duplex in the Zongo San Gaban Zone (~140 km, Figure 2.5b), and is balanced at depth by shortening along the basement thrust sheet beneath the EC. The Zongo San Gaban Zone has the oldest reset K-Ar and ^{40}Ar-^{39}Ar thermochronology ages (~45-37 Ma) (Farrar et al., 1988; Perez, Horton, McQuarrie, et al., 2016) and are part of a larger set of published cooling ages that range from ~47 to 38 Ma in Bolivia (Benjamin et al., 1987; Gillis et al., 2006). Modelled shortening propagates southwestward into the Macusani structural zone. The modelled deformation has produced two synorogenic flexural basins, one on either side of the uplifted, deformed topography (540-950 km, 130-20 km, Figure 2.5b). The flexural basin is deeper on the southwestern side, reaching a maximum of ~5.75 km depth (versus ~2.0 km on the northeastern side), driven by the uplift over the deeper basement structures.

As deformation proceeds into the backthrust belt, the exposed Cretaceous strata in the backthrust belt shorten separately from the Devonian, as the two stratal packages are separated by a décollement along the top of the Devonian. The shortening in the backthrust belt is balanced at depth with basement thrust sheets beneath the AP. These basement thrust sheets under the AP partitioned the AP flexural basin that initially developed as a contiguous system (~490 km, Figure 3.5b, c), into two separate basins (the Llalli and Tinajani sub-basins) with independent sedimentation histories. These active basement ramps also uplift lower Paleozoic strata to produce an angular unconformity (Figure 3.5c, d), in accordance with sedimentology and stratigraphy data (Horton et al., 2015; Perez and Horton, 2014). These basement thrusts remain active through the remainder of shortening in the EC backthrust belt and the AP. The shortening in the AP elevates surface structures to modern elevations (~4 km). At the end of AP shortening, an additional ~2 km of sediment has accumulated in the northeastern flexural basin, creating a total of up to 4.5 km of synorogenic sedimentation prior to SA deformation initiation (~15 Ma).

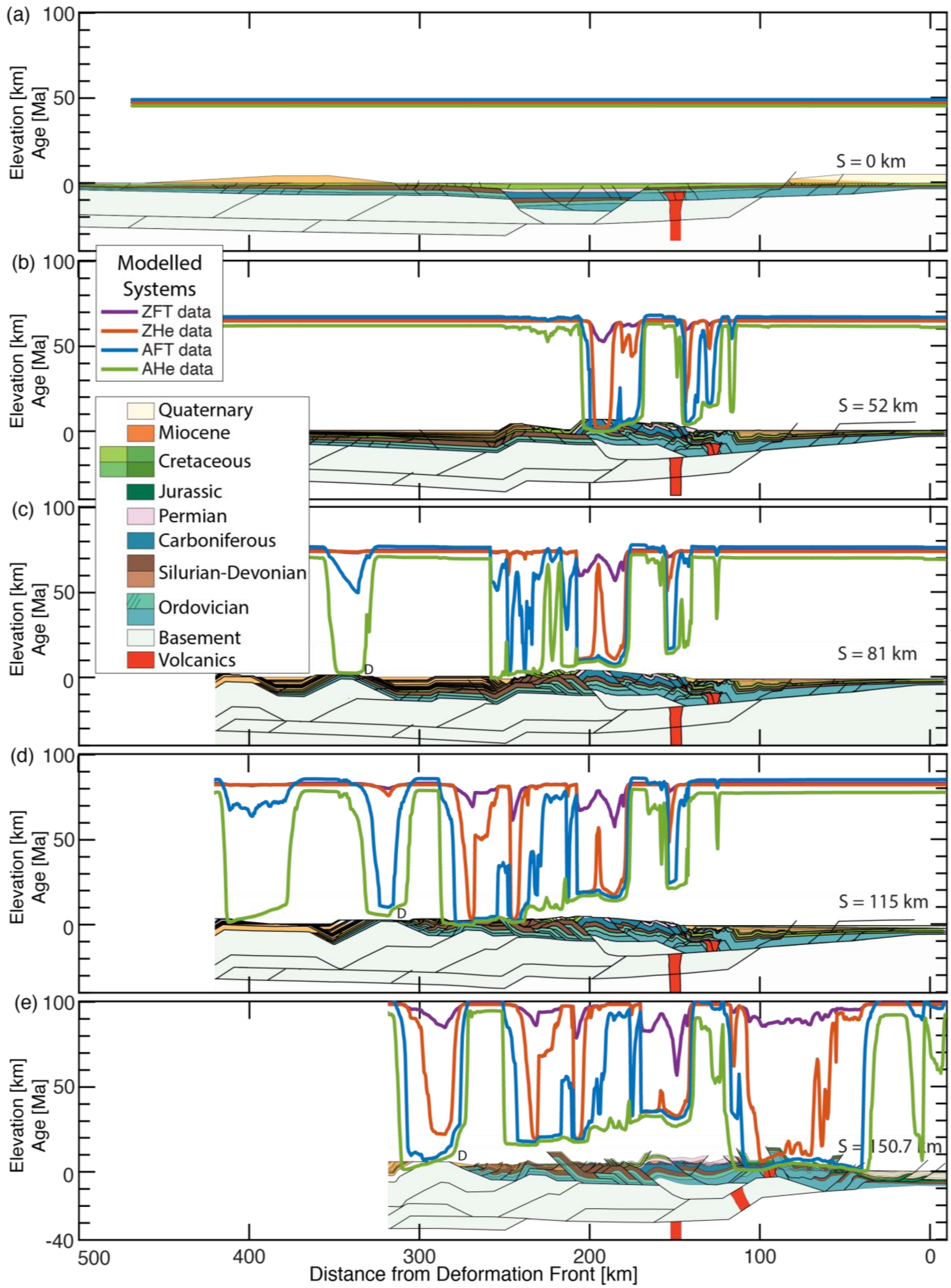

(a)
(b)
(c)
(d)
(e)
Elevation [km]
Age [Ma]
S = 0 km
S = 52 km
S = 81 km
S = 115 km
S = 150.7 km
Modelled Systems
ZFT data
ZHe data
AFT data
AHe data
Quaternary
Miocene
Cretaceous
Jurassic
Permian
Carboniferous
Silurian-Devonian
Ordovician
Basement
Volcanics
D
Distance from Deformation Front [km]

Figure 3.5. Sequential deformation and modelled thermal ages

Shortening moves from the AP and propagates through the SA from southwest to northeast. Motion in the easternmost basement thrust lifts the EC and transfers shortening to faults that repeat the Paleozoic section in a northeastward propagating duplex. This shortening is balanced by a Cretaceous duplex that uplifts and folds upper Cretaceous and younger rocks at the surface. There is a small amount (2 km) of OOS motion on the first westward verging fault in the EC that accompanies initial motion in the SA. Final synorogenic sedimentation in the SA adds up to ~3 km of sediment in the northeastern most reaches of the modelled basin, to a total basin depth of ~4.5 km.

Flexural models using constant values for EET were not able to simultaneously preserve the necessary geology at the surface (particularly the Mitu Fm. in the southwestern Macusani Structural Zone), produce sufficient basin depths in the SA, or preserve modern AP elevations. Thus, our final model relies on an initially low EET (25 km), corresponding to the thin crust at the time, and preserving the Mitu strata in the western limb of the graben, and gradually increasing in EET (to 90 km) to promote subsidence in the foreland basin throughout the remainder of the model. This change in EET was not enough to match modern elevations in the AP and western EC, nor provide sufficient SA basin depths, thus 7 km of imposed uplift under the AP/ western EC and 2.5 km of imposed subsidence under the SA were added. Imposed uplift and subsidence were applied in two separate windows of time. The first period of imposed uplift limited subsidence in the Tinajani sub-basin during Ayaviri Fault shortening. Over this period, 1.8 km of uplift was imposed in 0.6 km increments over ~100 km in width centered on the Tinajani sub-basin during the ~18 km of Ayaviri Fault shortening. There was no imposed subsidence applied during these deformation steps. The second period, coeval with the remaining ~52 km of shortening, allowed for achieving and maintaining modern elevations in the AP and western EC, and achieving the foreland basin depth. This final portion of the uplift (~5.2 km) and subsidence (~2.5 km) in the model was applied in ~0.7 km increments of uplift and ~0.3 km increments of subsidence. The uplift was focused under the AP with a wavelength of ~200 km. The hinge point of no subsidence or uplift was collocated with the active SA ramp. The maximum subsidence occurred in the northern extent of the SA.

Parameter	Input Value
Crustal Volumetric Heat Production (A_o)	2.0-3.0 $\mu W/m^3$
e-folding depth (Ef)	12 km
Thermal Conductivity	2.5 W/m/K
Specific Heat	800 J/kg/K
Model Base	110 km
Temperature at base	1300°C
Temperature at surface	23°C
Atmospheric Lapse Rate	4.0°C/km
Kinematic grid spacing	0.5 km x 0.5 km
Displacement Increment	~8-10 km
Model Domain	600 km x 110 km x 5 km
Horizontal node spacing	0.5 km
Vertical node spacing	1.0 km
Model start time (thermal initiation)	100 Ma

Table 3.1. Thermal parameters for Pecube modelling.

The initial velocity model evaluated in Pecube is one where shortening initiates at 50 Ma and is constant through time using a surface heat production value of $A_o = 2.7 \mu W/m^3$, with ef-12 km. As *Pecube* is allowed ~50 My to thermally equilibrate and remove boundary effects, the modelled ages at the surface just prior to the initiation of shortening is ~50 Ma (Figure 3.5a). As shortening initiated in the Zongo San Gaban structural zone (Figure 3.5b), modelled AHe, AFT, and ZHe experience enough uplift-driven exhumation to be fully reset over the active basement ramp, showing characteristic U-shaped cooling pattern, with the youngest ages at the ramp (~200 km Figure 3.5a), increasing northeastward from ~0 Ma to ~10 Ma between ~230 and ~190 km where there is a break in modelled age to unreset ages. The surface structure deformation in the Zongo San Gaban structural zone provides enough structural uplift to drive exhumation and fully reset modelled AHe, AFT and partially reset ZHe ages between ~165 and ~140 km (Figure 3.5b).

Shortening across the EC backthrust belt exhumes the synorogenic basin deposited prior to 28 Ma, which fully resets modelled AHe ages and partially resets modelled AFT ages between ~285 and 250 km (Figure 3.5c). Modelled ages that were reset to 0 Ma in the previous step (Figure 3.5b) have increased in age from 0-10 Ma to 10-20 Ma due to the passage of model time (~235-195 km, Figure 3.5c). Shortening in the lower basement levels in the WC (at ~410-400 km, Figure 3.5c), partially reset AHe and provides a western source of sedimentation to the AP. Shortening on the southwestern basement thrust sheet underneath the AP (~370-350 km) fully resets the modelled AHe and partially resets the AFT ages, while the northern basement thrust sheet under the AP (~300-280) has not yet had enough shortening reset any modelled ages (Figure 3.5c).

At the end of AP shortening (Figure 3.5d), the characteristic U-shaped cooling signal from the lower basement level active in the WC (~410-400 km, Figure 3.5d) now extends to ~380 km before a break in modelled age to unreset. The width of this signal is dependent on the amount of uplifted and laterally translated topography over this WC basement structure. The shortening in the AP, both on surface and subsurface structures, have provided enough uplift-driven exhumation to reset AHe and AFT across nearly the entire AP, with a break in ages between ~325-310 km as this region is caught between the uplift fields imposed by the two AP basement ramps that provide uplift. The modelled reset AHe pattern across the AP merges with that of the EC, and the boundary between them is not visible in the AHe pattern. However, at ~255-250, there is a break in modelled AFT ages to partially reset and in modelled ZHe to unreset that shows a spatial gap in exhumation magnitude between the AP and EC in this location (Figure 3.5d).

At the end of SA shortening (Figure 3.5f), the modelled AHe ages increase from ~0 Ma over the active SA basement ramp (~130-120 km Figure 3.5f) to 5 Ma at 100 km (midway up the basement thrust sheet hangingwall cutoff), and then young again to ~0 Ma at ~70 km before a break in ages at 480 km to unreset. The modelled AFT ages are fully reset over a narrower window (~130-80 km) when compared to AHe ages, and are slightly older (~2-7 Ma), and are partially reset between 80-60 km. ZHe across the entire SA region is partially reset, from the youngest modelled ages of ~15 Ma over the SA basement ramp (130-120 km) to unreset at ~80 km (Figure 3.5f).

3.4.3 Variable Incision Effect on Modelled Ages in the SA

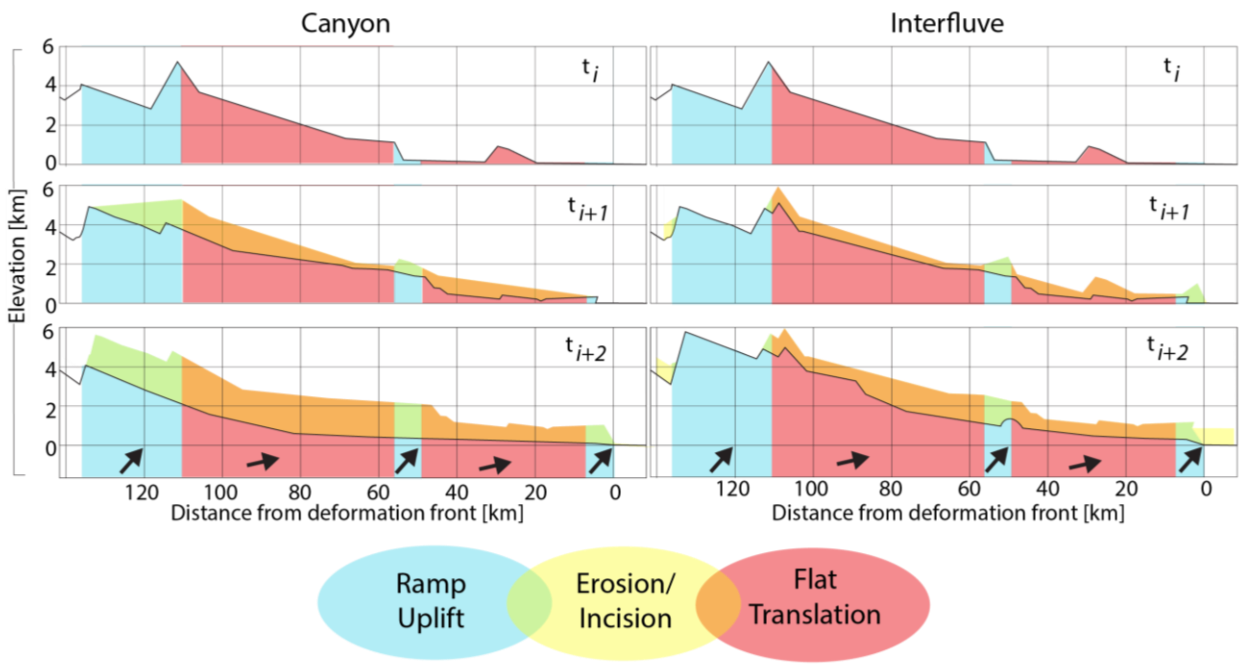


Figure 3.6. Schematic depicting evolution through time for canyon incision (left column) and interfluve preservation (right column)

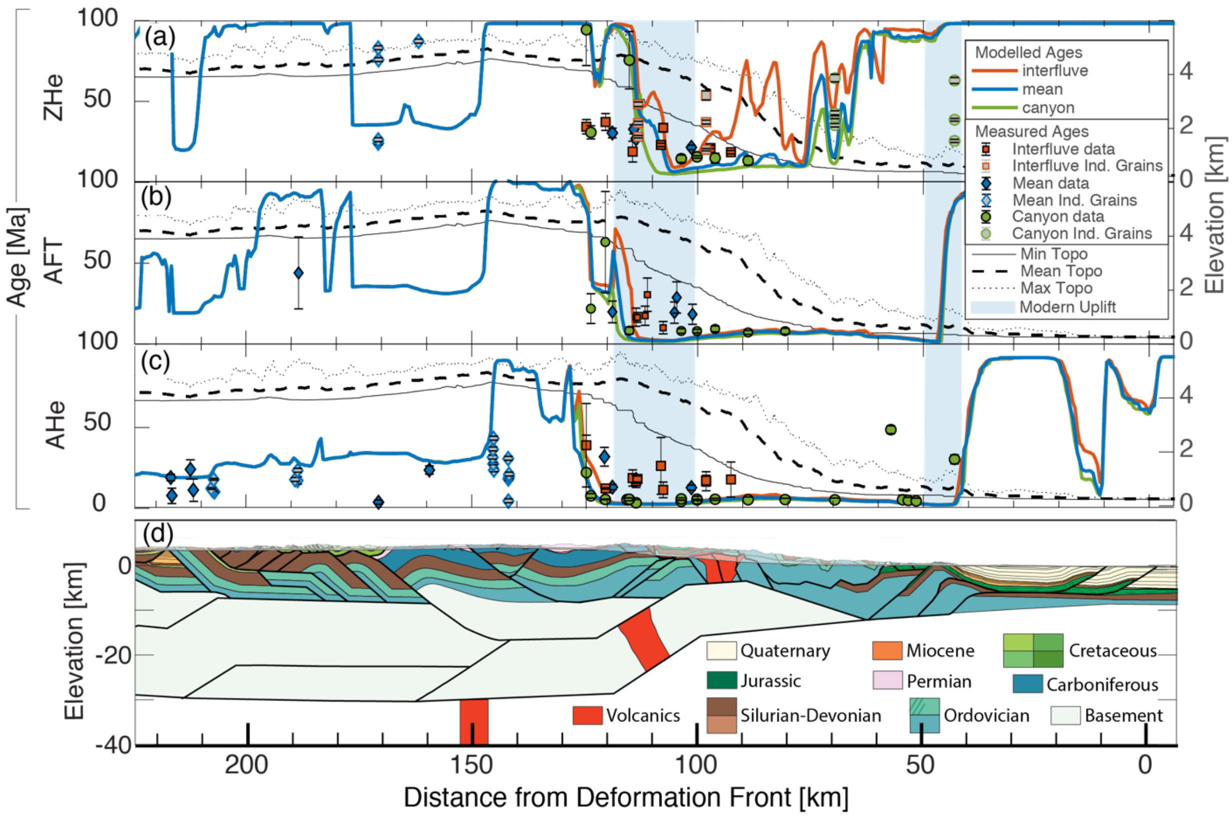


Figure 3.7. Effect of incision level on modelled thermal ages

We tracked mean, canyon, and interfluve topographies over the last ~17 km of shortening (Figure 3.6, Figure 3.7) to evaluate different potential periods of canyon incision occurring over the last 17, 4, 2, 1, 0.5 km of shortening). Here, we evaluate the effects of varying incision level and timing of incision on the resultant modelled ages, holding velocity and heat production constant. Comparing predicted canyon, mean, and interfluve ages where incision occurred over the last 17 km of shortening (incision initiating at 5.67 Ma, shortening at a constant rate of 3.0 mm/y, and radiogenic heat production ef-12 km, A_o = 2.7 μW/m^3), we see the same pattern of cooling ages located between 115 and 50 km for AHe and AFT, regardless of incision level. The only chronometer system that responds to a varying level of incision (e.g. interfluve vs canyon model topographies) over a broad region is ZHe, in the region between 120-50 km (Figure 3.7).

The modelled canyon and interfluve AHe and AFT ages (Figure 3.7) show the same reset pattern as the mean ages (Figure 3.5, Figure 3.7), with youngest modelled AHe reset ages of ~0 Ma over the active SA basement ramp (130-120 km) and the most recently active Paleozoic thrust (~60 km), with ages in the middle (~100 km) being the oldest fully reset AHe ages (Figure 3.7c). At 45 km, the modelled ages for AHe break sharply to unreset for all topographies, while at 120-130 km, there is a slight difference in the location that predicted ages change from fully to partially reset ages that differentiate the exhumation experienced by the three topographies. The modelled ZHe cooling ages (Figure 3.7a) show the greatest sensitivity to the variation in modelled incision, as the canyon, mean, and interfluve ages show distinctly different age patterns. The mean and canyon ZHe pattern shares the same shape of the modelled ages seen in the predicted AHe and AFT patterns, with youngest reset ages at ~110 km, gradually getting old to the east where the oldest fully reset ages are located at ~75 The modelled interfluve ZHe ages are only fully reset over the ramp (~110 km) and at ~75 km, but are partially reset over a majority of the SA (115-65 km). At ~75 km, the modelled ZHe ages for all topographies are partially reset, increasing in age northeastward until they are unreset at ~60-55 km (Figure 3.7a). As shown in Figure 3.7, at this rate of shortening (3 mm/yr) only ZHe is sensitive to erosion level to produce variations in modelled ages across the three topographies. However, the modelled ages for ZHe are too young for the canyon and mean ages. Additionally, the predicted AFT canyon, mean, and interfluve ages are too young to match the measured ages, particularly over the region of ramp uplift (~140-120 km). While the modeled canyon AHe ages match the measured ages, the modelled interfluve AHe ages are equally as young creating a 10- 15 Ma difference between predicted and measured ages from ~140-110 km, showing that incision at constant shortening rates is not sufficient to achieve the AHe, AFT, and ZHe measured ages.

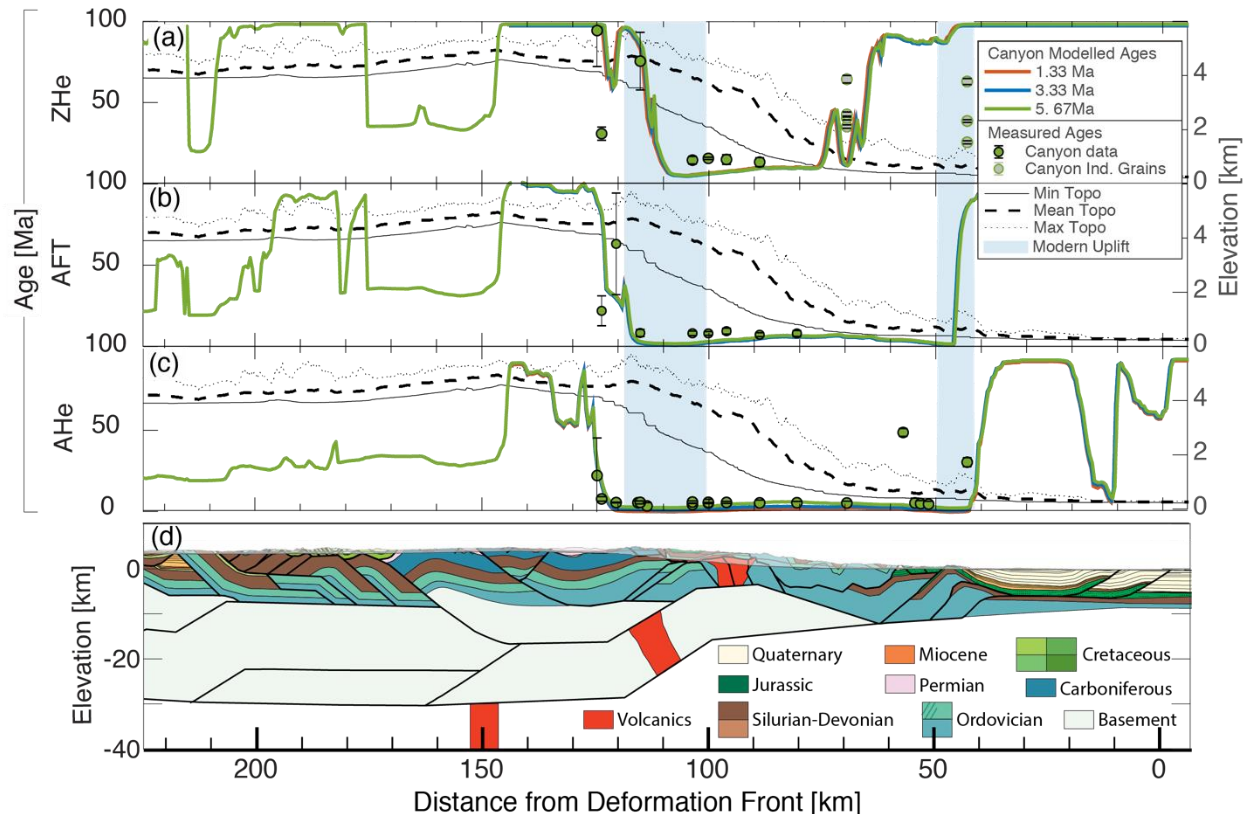


Figure 3.8. Effect of timing of SA canyon incision on modelled thermal ages

Next, we evaluated the effect of altering the age of canyon incision from 5.67, 3.33, and finally 1.33 Ma on the resultant modelled thermochronology ages (Figure 3.8) with a constant shortening rate (3 mm/y) and radiogenic heat production ef-12 km, A_o = 2.7 μW/m^3. As shown in Figure 3.8, the modelled canyon AHe, AFT, and ZHe thermochronology ages for varying ages of incision are virtually indistinguishable. The only chronometer system that responds to the age of canyon incision is AHe, and only in the region between 105-90 km by 1-2 Ma, where at constant rate the AHe ages are the oldest fully reset ages, and are not actively undergoing uplift and exhumation. This shows that because exhumation is great enough and fast enough in this model, that the amount of canyon incision is not sufficient to produce the spread in measured canyon and interfluve ages when different incision ages are applied.

3.4.4 Effect of Radiogenic Heat Production on Modelled Ages

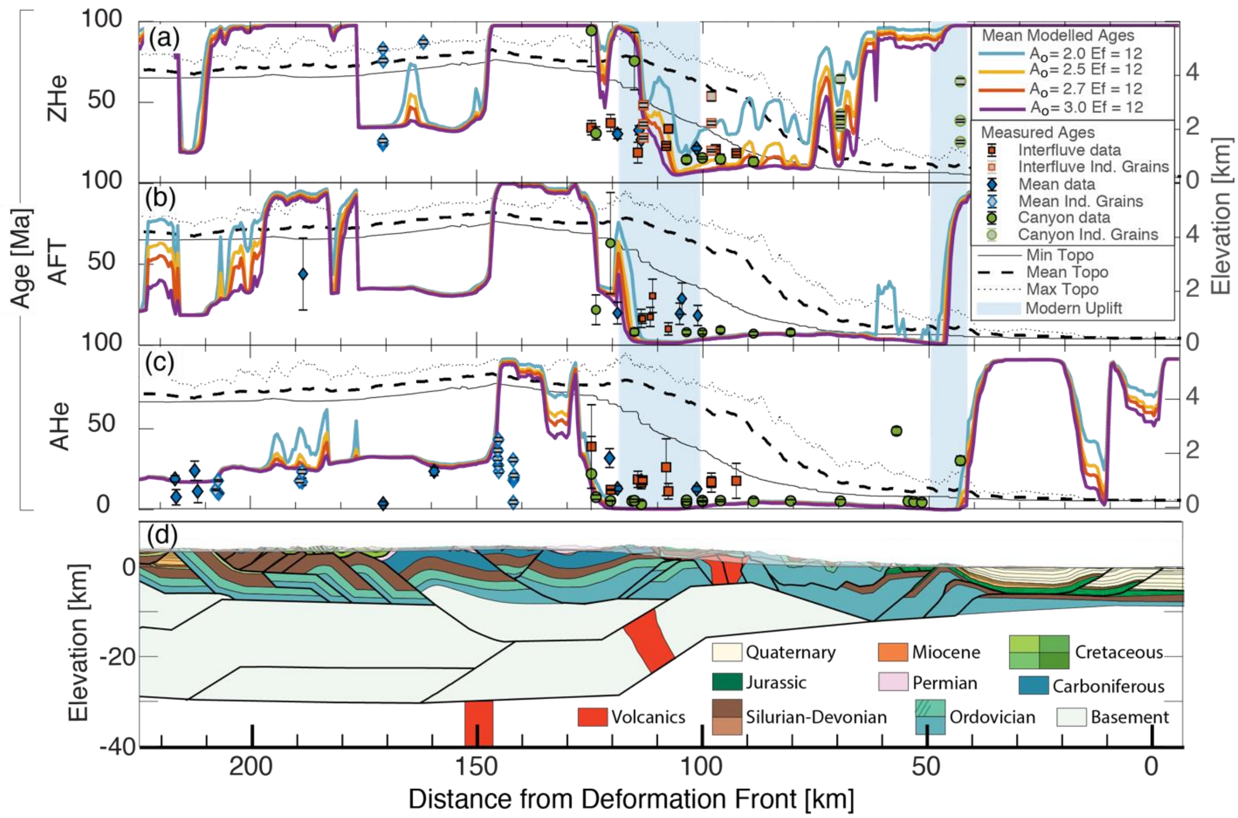


Figure 3.9. Effect of radiogenic heat production on modelled ages

Depending on the chronometer system, exhumation rate, and radiogenic heat production, predicted cooling ages may be sensitive to small change in modeled heat production (Buford Parks & McQuarrie, 2019; Gilmore et al., 2018; McQuarrie & Ehlers, 2015). We test the effect of upper crustal radiogenic heat production on the predicted cooling ages to see if either AFT or AHe are sensitive to changes in heat production and if lower or higher heat production values allow for a greater differentiation between predicted interfluve and canyon ages. We evaluate changing crustal radiogenic heat production on the same model shown in Figure 3.8, with velocity held constant and using mean topography. We use a heat production model that applies an initial surface radiogenic heat production ($A_o = 2.0\text{-}3.0\ \mu W/m^3$) that exponentially decreases with a characteristic scaling depth (ef = 12 km). Modelled ZHe cooling ages are the most sensitive to changes in radiogenic heat production (Figure 3.9). Cooler heat production values (e.g. ef-12 km, $A_o = 2.0\mu W/m^3$) produces partially reset ages across 120-90 km, while warmer heat production values (e.g. ef-12 km, $A_o = 3.0\mu W/m^3$) produces fully reset ages across this same region (Figure 3.9a). Variation in heat production produces no change in modelled ages for AHe across ~135-60km or AFT between 130-80 km. On the edges of the reset signals, where the change in exhumation is most apparent, there is a spread in the age of partially reset modelled AFT ages, from a maximum age of ~75 Ma at ~132 km with the coolest radiogenic heat production ($A_o = 2.0\mu W/m^3$), compared to a minimum age of 45 Ma in the same location with the warmest radiogenic heat production ($A_o = 3.0\mu W/m^3$) (Figure 3.9b). A similar, but less drastic, fanning of ages between the warmest and coolest radiogenic heat production values is present in the AHe signal between ~145 and ~135 km (Figure 3.9b). At all of the crustal radiogenic heat production values evaluated, only ZHe produced notably different predicted cooling ages, when varying topographies (canyon, mean, and interfluve) were applied.

3.4.5 Effect of Shortening Rate on Modelled Ages (SA, and EC)

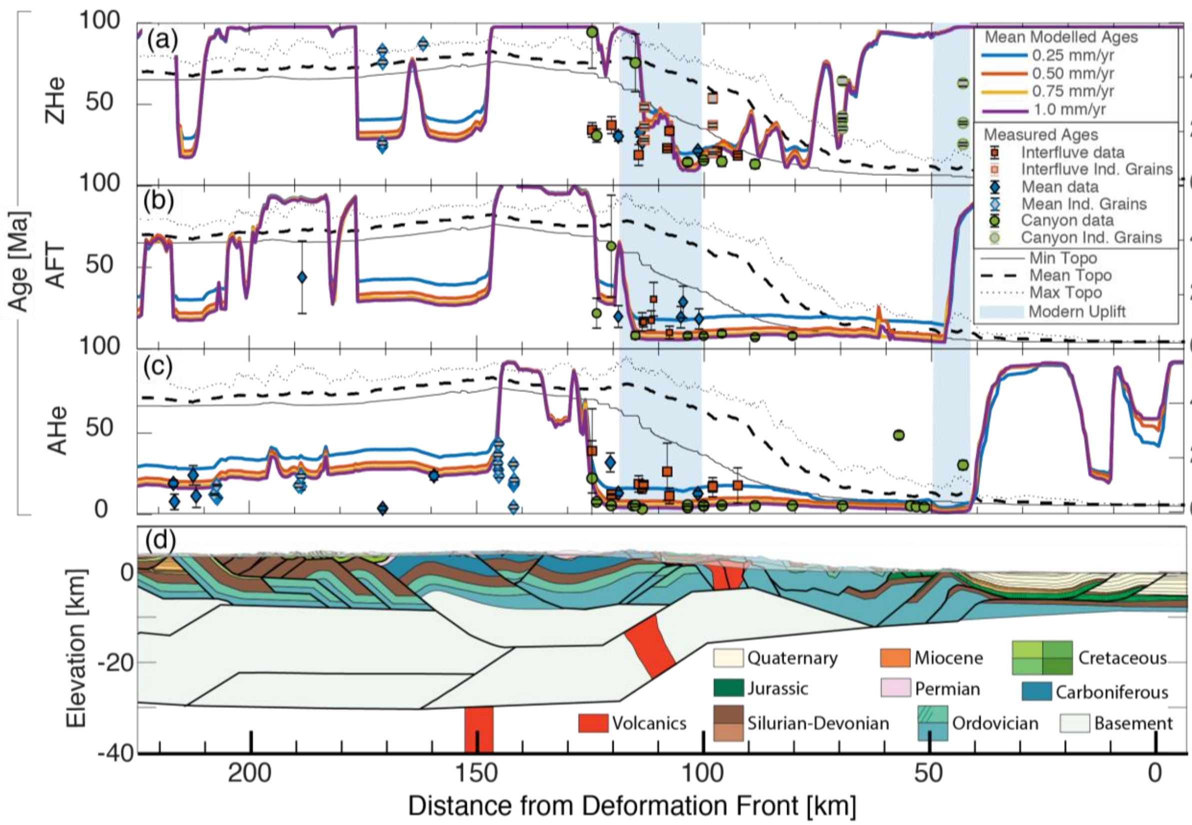


Figure 3.10. Effect of SA shortening rate on modelled ages

We evaluated a suite of shortening rates to evaluate if velocity had the ability to differentiate AHe and AFT modelled ages for canyon and interfluve topographies. The velocities tested slow SA shortening rates (0.25, 0.5, 0.75, 1.0, and 2.5 mm/y). These changes in velocity, decreased from our previous constant model rate of 3.0 mm/y, have the greatest difference in modelled ages between ~120 and 60 km in the AHe and AFT thermochronometer systems (Figure 3.12). The fastest rate shown, 1.0 mm/y, produces similarly young modelled AHe ages (purple line, Figure 3.12) as the constant rate shown previously (e.g. Figure 3.7), but slowing the rate down further to a minimum of 0.25 mm/y (blue line, Figure 3.10) increases the modelled AHe and AFT ages across the rest portions of the Subandes (~120-60 km) by ~15 My to ~18 Ma, creating a spread in model ages and matching measured interfluve ages between 120-90 km.

As discussed in previous sections, changes in incision age or radiogenic heat production alone are not sufficient to reproduce the measured ages, while a slow down in the shortening rate to between 0.25 and 0.5 mm/y during SA shortening is required to reproduce the interfluve ages. Because earlier work has argued for a cessation of EC uplift and exhumation by 12 Ma (Lease & Ehlers, 2013) with only incision driving exhumation from ~ 4 Ma to present, We tested this hypothesis, and found that while all shortening happening by 12 Ma (at 4 mm/y) is sufficient to produce the interfluve ZHe, AFT, and AHe ages, incision at any proposed time (8, 4, or 2 Ma) in this scheme (e.g. in the absence of any additional shortening) is unable to match canyon AHe ages, particularly over the active Subandean basement ramp at 100-120 km (Figure 3.11).

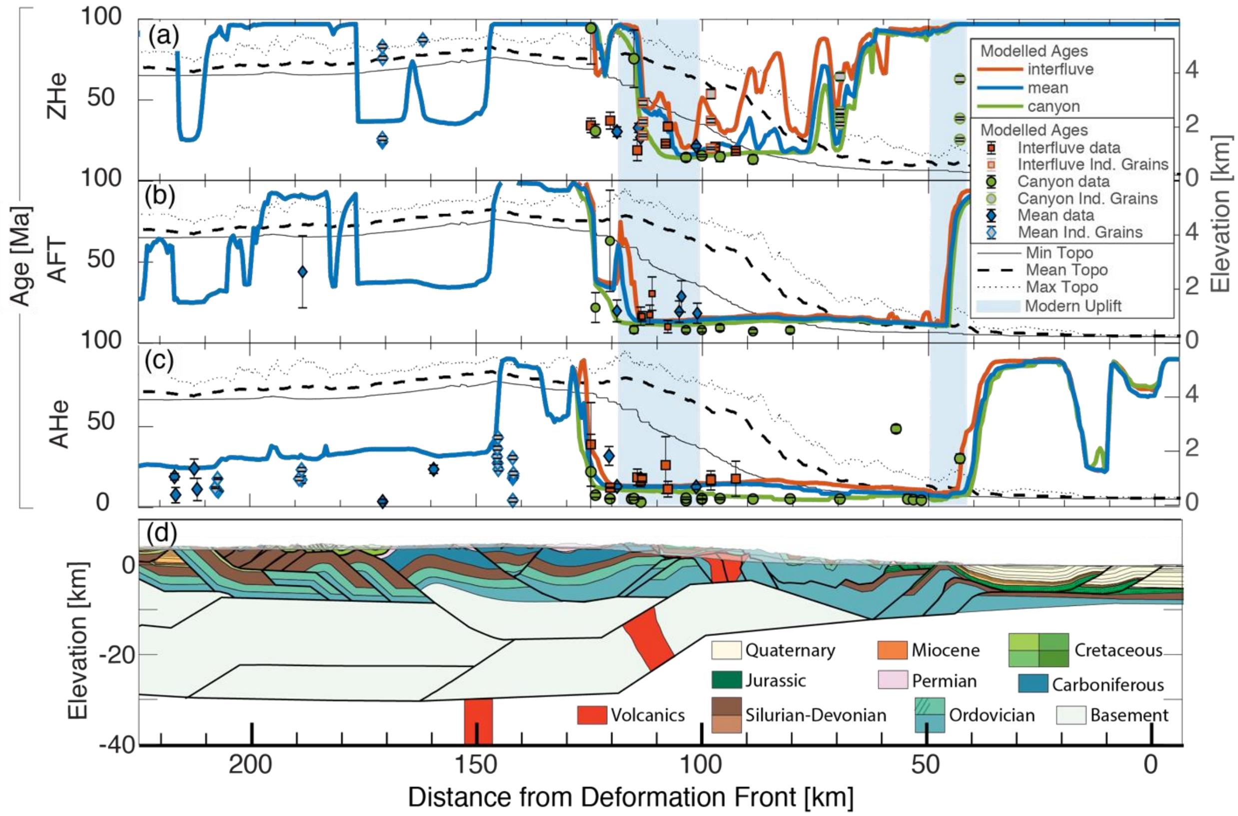


Figure 3.11. Thermal modelled ages for all shortening occurring by 12 Ma, with canyon incision beginning at 4 Ma

3.4.6 Best Fit Velocity Models

Modelled thermochronologic ages predicted by all velocity structures were compared based on their ability to reproduce the measured ages. If the modelled age, including a model error of ±1 Ma and ±2 km, was within the measured age with error, then the sample was considered a match. Individual grain ages were weighted by the number of grains measured (e.g. if 1 grain matches and three were measured, it is counted as 1/3 match). These comparisons also took into account the modelled elevation and sample elevation, such that canyon ages were only compared to modelled canyon thermal ages. Then, the matches for canyon, mean, and interfluve elevations were averaged to compute the overall goodness of fit (GOF) for that model. Overall GOF values ranged from 40% to 61% (Table 3.2).The overall GOF values were broken into percentile ranges, where the 50th percentile produce a match of at least 50.8% of the data. The 75th percentile matched 56.8% of the data (Figure 3.12). For the models in the 75th percentile and above, we also performed a X^2 analysis that agreed with our GOF metric (Table 3.3).

The best fit velocity models (Figure 3.13) slows down from a background shortening rate of 4.0 mm/y to 2.4 mm/y over the last 6-3 km of shortening that drives initial exhumation on the interfluves, then slows down even further to 0.35 mm/y for the majority of canyon incision (3 km of shortening). The last 1 km of shortening occurs at 0.5 mm/y (Figure 3.13). This velocity structure is virtually indistinguishable from several others with similar velocity variations, highlighting that the best fit occurs in models with shortening rates over the last 10 km between 0.6 and 1.0 mm/y for essentially any incision age between 8 and 2 Ma, although a higher number of viable scenarios show incision initiating between 4-5 Ma (Figure 3.14). In addition, while the total shortening rate over the last 10 km averages out to 0.5-1 mm/yr, best fit velocities require a period of significantly slower shortening (~ 0.3 mm/yr) between ~11 and ~1 Ma (Figure 3.12).

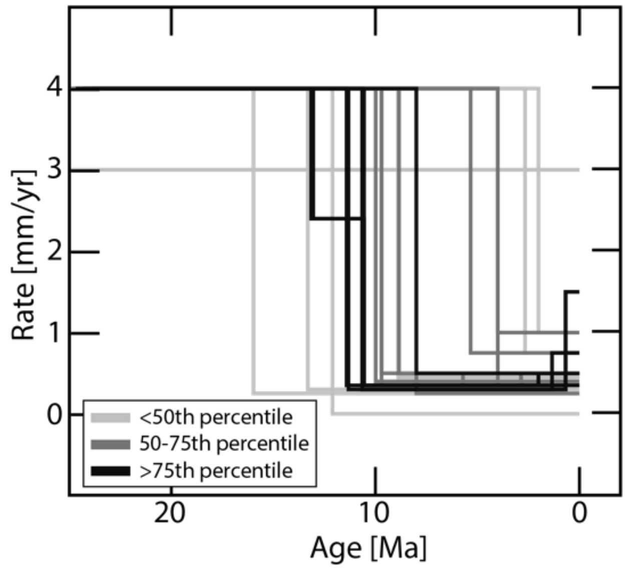


Figure 3.12. Testing shortening rates classified by goodness of fit

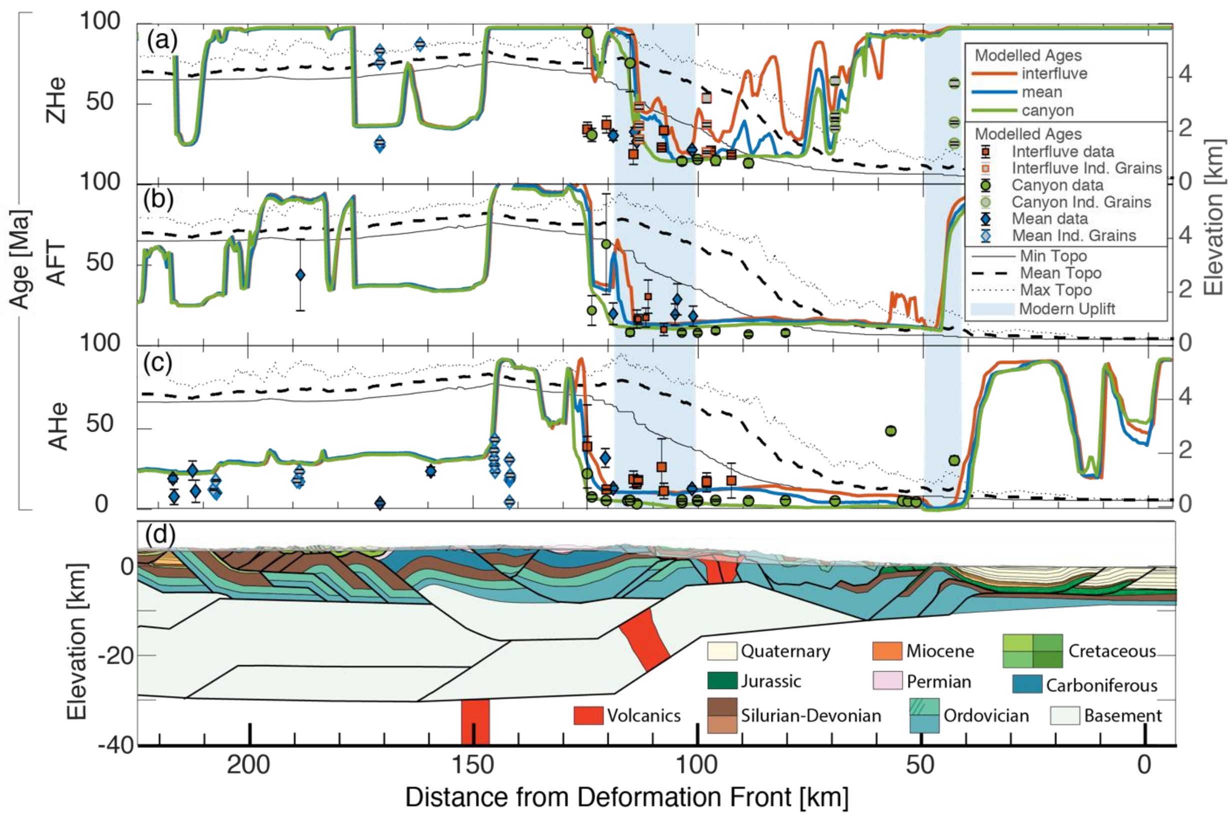


Figure 3.13. Best fit velocity with variation in topographies

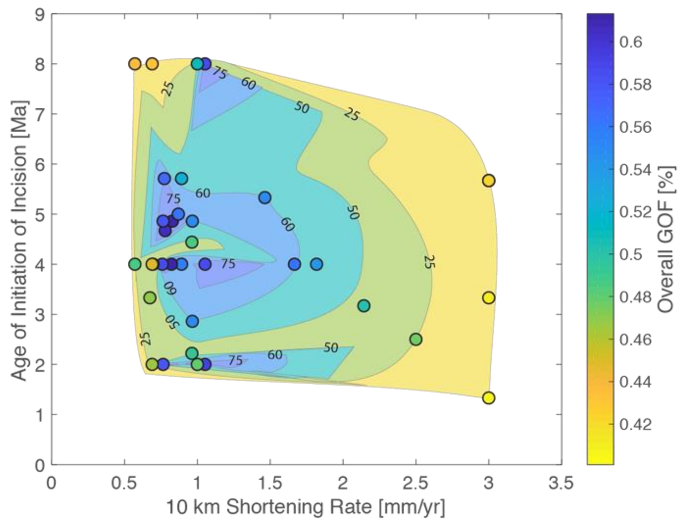


Figure 3.14. Goodness of fit plot for all tested velocities and incision ages. Contours are in percentiles.

3.5 Discussion

3.5.1 Altiplano Basin Ages Controls on Age and Location (Rate) of Deformation

Basin Sedimentology. The age and thickness of synorogenic strata exhumed during Andean deformation and deposited in the AP is well constrained in the regions of the Tinajani and Llalli sub-basins. The Eocene-Oligocene Puno group, which unconformably overlies the Cretaceous Muni Formation, is dated in two separate measured sections, one near each sub-basin. In the Tinajani sub-basin, the ~1.3 km thick Puno formation is dated on the northeastern margin between 29 and 24 Ma (Perez and Horton, 2014). Further southwestward, near the Llalli sub-basin, the Puno group is dated between 37 and 28, with >4 km accumulated at the measured section in Macari, and ~3 km accumulated in Llalli. These two measured section provide distinct controls on the location of deformation between 38 and 24 Ma. As the basal accumulation of the Puno group to the northwest in the Tinajani sub-basin is much younger, the basin history across the region had to have separated prior to 29 Ma, the oldest age of the Puno section measured in the Tinajani sub-basin. The two basins could have been connected and equally collecting sedimentation at earlier times, but by 29 Ma, the lower Puno formation must have been removed in the Tinajani sub-basin. Additionally, the ~1.1 km thick Tinajani Formation, dated between 24 and 17 Ma, is present only in the Tinajani sub-basin where it conformably overlays the Puno Formation (Perez and Horton, 2014).

In our best fit velocity model, the Puno formation has accumulated ~3 km in the Llalli region by ~35 Ma, in the southwestern flexural basin created by initial shortening in the northern Eastern Cordillera and the Macusani structural zone consistent with accumulation in the basin between 37 and 28 Ma. Initial deposition in the Tinajani basin (deposited from ~50 - 28 Ma) is erosionally removed by basement uplift between 28.5 and 25.4 Ma, with subsequent deposition of ~1 km of Puno strata between 25.4-23.8 Ma. The ~1 km Tinajani formation is deposited in the Tinajani sub-basin between ~19-16.5 Ma. Thus, our modelled basin accumulation and model produced erosional contacts match well to the published sedimentology in the region. The age of the basal Puno in the Tinajani sub-basin is young compared to measured ages, however, it is controlled by the basement deformation. The basement thrust that exhumes the Tinajani sub-basin between 28.5-25.4 Ma is required to kinematically balance the shortening on the Ayaviri Fault, and could not have started earlier because of the geometry constraints on the amount of shortening on this basement thrust sheet and the controls on the basement ramp's location.

Growth Structures. The thrust faults that bound the Tinajani sub-basin on the northeast and southwestern sides are both accompanied by growth structures. Growth strata measured on the northeastern side of the Tinajani sub-basin dates the Ayaviri Fault, which places middle Ordovician on the Oligocene Puno group, between 28-26 Ma (Perez & Horton, 2014). Growth strata measured on the southwestern side of the Tinajani sub-basin dates the Pasani Fault, which places Cretaceous Ayavacas Formation on the Neogene Tinajani, between 18-16 Ma (Perez & Horton, 2014). These growth structures, particularly those associated with the Pasani Fault, show that the exhumational controls that sub-divide the Tinajani and Llalli sub-basins are separate from those that deform the surface structures. In our model with best fit velocity, the Ayaviri Fault is active between 28.5-24 Ma, and the Pasani Fault is active 24-22 Ma. The timing of the Ayaviri Fault in our model matches well, but may be slightly young compared to available growth strata dating. However, there is a possibility that younger Ayaviri Fault activity was not recorded in growth strata, or that strata recording its younger motion was erosionally removed. The age of final Pasani Fault motion is slightly too old. Currently, the shortening in the Altiplano occurs in two deformation steps, where 8 km of shortening, including the Pasani Fault, occurs immediately after the Ayaviri Fault shortening ceases. The subsequent 8 km of shortening in the second deformation step of AP shortening includes shortening further south, including the Llalli region and shortening on the faults bounding the Descanso Basin. Rearranging the faulting sequence in the Altiplano to allow for the youngest faulting in the AP to be on the Pasani Fault would allow shortening at 20-18 Ma, which is at the older age limit of the measured growth strata.

3.5.2 Persistent Difference Between Measured and Modelled ages and Implications for Deformation and Exhumation History

On the northern edge of the Macusani structural zone, immediately southwest of the active basement ramp, measured reset ZHe ages are 30-37 Ma. Predicted ages in this region are only partially reset and thus as old as ~100 Ma (110-130 km, Figure 3.13a). However, the southern structures in the Macusani zone do produce ZHe cooling ages that are ~ 30 Ma (150-170 km, Figure 3.13a). The northern boundary of the zone is actually the first to deform in our sequential model, during a time when EET values are kept low to minimize exhumation and not over erode the stratigraphy that is clearly seen at the surface. Due to the mismatch between these modelled and measured ZHe ages on the northeastern edge of the Macusani structural zone (110-130 km, Figure 3.13a), we hypothesize that there is a lack of sufficient exhumation on the eastern side of the Macusani structural zone necessary to reproduce the reset/partially reset ZHe ages when the rocks are being uplifted and exhumed. To compensate for this, we would need to exhume more, but since the section as modelled matches the geology at the surface, the samples would need to be buried deeper by thickening the Mitu prior to the initiation of Andean deformation, particularly along the northeastern limb of the graben. If the thickness of the Mitu was increased across the graben, it would help prevent over erosion in the flexural modelling process, allowing a higher EET at the beginning, and thus allowing more and older erosion across the entire graben region from 30-40 Ma (Figure 3.5b, 150-200 km) creating reset ZHe cooling ages (~30 Ma) and matching measured ages in the region.

3.5.3 Model Predicted Timing and Rates of Shortening

Perez et al. (2016b) suggested initiation of shortening prior to 37 Ma based on K-Ar ages, and that protracted exhumation in the EC, from Bolivia through Peru initiated in Eocene time. They suggest that the ZHe age population between 34-18 and the older AHe age population of 30-26 in the Coasa Pluton require initial deformation associated with surface structures and shortening along the reactivated normal fault on the northeastern side of the Mitu Graben to occur at this time.

The younger AHe population of 17-11 Ma was associated with exhumation driven by motion up and over the SA basement ramp initiating at ~ 18 Ma. Based on fully reset canyon AHe ages of 5.8 and 5.7, Perez et al. (2016b) require SA deformation at least by ~6Ma, but suggested SA initiation early as ~15 Ma (based on a partially reset AHe sample at ~15Ma, and the connection between SA deformation and motion along the SA basement ramp). These age groupings match our best fit models, however, we argue for a long-term shortening rate of 3-4 mm/y until 11 Ma when a marked slowdown to ~2.6-0.35 mm/y occurs, continuing until present. This is in contrast with the hypothesized Perez et al. (2016) rate of ~2-2.5 mm/y from 15 Ma until present.

Espurt et al. (2011) report a balanced cross-section through the Subandes at 12°S with 53 km (39%) total shortening and AFT ages between ~5.9 and 24 Ma. They suggest that the last 23 km of motion has happened since 6 Ma, at a rate of ~3.8mm/y and hypothesize that the rate held constant for the duration of SA shortening, which initiated at ~14 Ma. Their youngest AFT age of 5.9 Ma is of similar age to our AHe ages of 5.7 and 5.8 Ma, and their SA shortening estimate since 14 Ma (43 km) is similar to our total Subandean shortening (~37 km / 38%). We suggest that similarly linked SA shortening combined with coeval basement thrust induced uplift can account for recent uplift and exhumation ~ 200 km to the northwest in the Camisea Subandean zone (Espurt et al., 2011) and the high peaks of the Abancay Embayment (Machu Picchu region) to the south producing the measured 5-1 Ma AFT and AHe ages along the Urubamba river (Gérard et al., 2020).

3.5.4 Proposed Mechanisms and Timing of Incision

Proposed drivers of incision in northeastern Peru include climate only driven incision (Lease & Ehlers, 2013) incision due to dynamically driven plateau uplift (Garzione et al., 2017; Kar et al., 2016), or local, structurally driven uplift. Lease and Ehlers (2013) propose EC shortening ceased in the middle Miocene at ~ 12 Ma following interpreted deformation driven exhumation and associated cooling from ~ 25-15 Ma based on modeling of BAr and ZHe cooling ages. They argued young (~ 4 Ma) measured AHe were then a function of ~1.0-1.6 km of canyon incision occurring at ~4.1-2.8 Ma, in the absence of uplift and due to an increase in Pliocene climate. While our model supports canyon incision at ~4 Ma as the mechanism to produce young AHe cooling ages, it requires incision to accompany active, shortening induced uplift between 12 Ma and present. Incision alone (initiating at 8, 4, or 2 Ma) is insufficient to produce the young reset AHe or AFT ages present in the canyon (Figure 3.11). Just as important, incision driven only by plateau uplift may match the measured mean and interfluve AHe ages, but would not produce the exhumation needed to match the young, 4 Ma AHe ages in the canyon. Our results require incision to be enhanced through an outside mechanism such as a marked change in climate or plateau uplift, but to occur in the presence of ongoing structurally driven uplift.

Frequently in geodynamic models, both when incision occurs, and the drivers of it, can be convoluted. Most of our models are a combination of plateau-scale uplift (imposed uplift), local structurally driven uplift (the active surface and subsurface structures), and an independently driven incision, consistent with an increase in climate driven precipitation (modelled as rapid incision in the last steps of our model). Imposed uplift is included in our models for several separate reasons: the first period of imposed uplift limits flexural subsidence and facilitates the exhumation in the AP, while the second period maintains high elevations in the AP and western EC. The second period of imposed uplift in our model is applied continuously, after AP shortening has ceased (at ~18 Ma), in ~0.7 km increments over the last ~52 km of shortening, in order to maintain AP elevations at close to modern.

Paleoelevation estimates (and associated uplift histories) for Peru are located in the Condoroma district along the eastern edge the WC (Saylor & Horton, 2014) and in the Descanso Yauri Basin of the AP (Kar et al., 2016) The southernmost basin in our model (~275-280 km, Figure 3.5d), along strike with the Descanso Yauri Basin, has a cyclic elevation history. It is initially lifted to ~3.5 km passively as the southernmost basement thrust sheet is active between ~32-28.5 Ma and then gradually sinks back to ~1 km in elevation at ~24 Ma, until it is lifted again to ~4+ km in elevation when the AP shortening is active between 24 and 18 Ma. This basin's elevation of >3 km is maintained until model finish by our imposed uplift. Paleoelevation proxies combined with sediment accumulation from ~ 18-9 Ma are consistent with the northernmost Altiplano basin at low elevations ~1-2 km until late Miocene (~9 Ma), followed by a period of geodynamic uplift to reached modern elevations of ~4 km by 5.4 Ma (Kar et al., 2016). This elevation history is also permitted by our modeling; instead of continuous ~0.7 km uplift at each deformation step to maintain elevations from ~ 18 Ma to present, we allowed the basin to naturally subside to ~1-2 km elevation until ~9 Ma, and then applied ~2-3 km of uplift by ~5 Ma, and subsequent uplift of ~1 km to maintain the elevations until modern. This change does not affect our modelled thermochronology ages, as the absolute elevation does not appreciably affect the modelled ages.

Although imposed uplift (to replicate geodynamic processes such as lower crustal flow or mantle delamination) is required to maintain elevations in the AP and westernmost EC, the uplift history in the northeastern portion of the EC as well as uplift of the southwestern WC are independent of the imposed uplift or subsidence. For the EC, Once the SA basement ramp is active, it controls the elevation history of rocks that have been transported over the ramp (and lifted to ~4.5+ km in elevation) separating ramp driven uplift (and exhumation) from the elevation history of the backthrust belt or Altiplano Plateau to the southwest. This basement ramp is the main control on cooling ages in the eastern EC. Imposed on this first order signal is an incision signal that is partially driven by uplift, but mainly driven by incision that occurs in the last 4 Ma. Since uplift due to the active SA basement ramp has been ongoing from 18 Ma to present generating elevations of > 4 km since ~13 Ma, uplift alone cannot be called upon to propagate an incision signal into the EC (as the timing of elevation gain due to uplift is inconsistent with the proposed timing of incision). What remains is either a climate signal independent of what is going on in the plateau to the southwest (i.e. its elevation), or an incision response to a marked decrease in shortening rate (and associated decrease in uplift rate) that propagates a knickpoint through the canyon that initiates when the system slows at ~ 11 Ma. The timing of EC incision is younger than that proposed for the western edge of the plateau which is hypothesized to be ~1-3 km between ~11 Ma and ~2 Ma (Schildgen et al., 2009, 2007) or ~1.5-3.5 km since 16 Ma (Jeffery et al., 2013). This longer window of incision can more readily be linked to the ongoing structural driven uplift of the WC over the same time window (Figure 3.5d-e) in conjunction with changing climate (Jeffery et al., 2013).

Although both the EC and WC have independent drivers of structural uplift that can readily account for their modern elevation, continued shortening and thickening of Andean crust should be accompanied by accumulation of mantle lithosphere and eclogitic lower crust. The accumulation of dense material under the orogen has the potential to uniformly lower the entire plateau (e.g. Garzione et al., 2017). Changing elevations due to the accumulation or loss of dense lithosphere is a fundamental component in the hypothesis of cordilleran cyclicity (DeCelles et al., 2009), wherein active shortening slows as the mantle root thickens and deepens, dragging all surface elevations low (~2 km)(DeCelles et al., 2009; Rak et al., 2017). After delamination this dense mantle root is lost and the plateau uplifts to a new isostatic higher elevation, driving incision into the plateau edge.

The slow down in velocity (~11 – 1 Ma) that best reproduces the measured cooling ages could be a function of mantle lithosphere accumulation and lower crust eclogitization. In the models we evaluated, the slow velocities persist till 1.3-0.5 Ma. If the loss of this eclogitic root is seen in plateau uplift and the associated increase in SA velocity (Garzione et al., 2006; Rak et al., 2017), it suggest that the plateau uplift has been very recent, as our slow velocities continue until ~1.3-0.5 Ma. This ~ 1 Ma age for plateau uplift is notably younger than the paleoelevation estimates from the Descanso Yauri Basin (9-5 Ma; Kar et al., 2016). In contrast to the potential loss of the entire mantle lithosphere under the plateau, the Moho is imaged at ~60-70 km depth in Peru with a likely present, mantle root that is either small (already partially removed) or in the process of removal (due to delamination or ablation) (Ward, Zandt, Beck, Wagner, & Tavera, 2016). Missing support for mantle loss suggests lower crustal flow as the probable driver of AP elevation change at 9-4 Ma (Garzione et al., 2017). Additional support for ongoing crustal thickening is present in trace element ratios of crustal melts which suggest AP/ western EC thickening over the past 5 Ma, likely due to lower crustal flow (Garzione et al., 2017).

3.6 Conclusions

Our thermokinematically modelled balanced cross-section reproduces measured thermochronology ages from AHe, AFT, and ZHe systems. Our model permits an initial age of deformation in the northernmost Eastern Cordillera between 40-50 Ma, with background shortening rates between 3.0 and 4.0 mm/y. Ages and rates of deformation of the Eastern Cordillera and Altiplano are primarily constrained by patterns of synorogenic sedimentation in hinterland basins. Sedimentation occurs in the Altiplano region for the duration of the model, but is both interrupted and exhumed during periods of deformation, simulating measured strata in the Ayaviri Basin. Out model replicates the measured angular unconformities in the Altiplano by interrupting sedimentation with a basement uplift that separates the Llali and Tinajani sub-basins and imparts different sedimentation histories. Early (50-29 Ma) sedimentation in the Tinajani basin is removed between 28.5 and 25.4 Ma by this basement uplift, followed by the subsequent deposition of the Puno formation on this unconformity in conjunction with motion on the Ayaviri Fault. Ages and rates of deformation in the SA are limited by measured cooling ages. Our thermal model highlights the need for a slow down in shortening rates from background rates of 4.0 mm/y to between 0.25 and 0.5 mm/y from ~12 Ma to present in order to match measured canyon and interfluve thermochronologic ages. A complete cessation of shortening by 12 Ma is not sufficient to match measured young AHe ages. Canyon incision, which is independent in age from structurally driven uplift, is necessary to reproduce the measured canyon AHe ages in combination with slow shortening rates. While many of the best fit models are consistent with incision ages of ~ 4, Ma, permissible age of canyon incision ranges from 8-2 Ma.

3.7 Acknowledgments

Supporting data are included in the supporting information file; any additional data or information are available through contacting VMB Parks (email: vmbparks@gmail.com). This work was supported by NASA headquarters under the NASA Earth and Space Science Fellowship program - Grant 80NSSC17K0388. We acknowledge and thank Midland Valley and Petex for support and use of the program Move. This manuscript benefitted from thoughtful discussions over the years with Mary Braza, Todd. Ehlers, Paul Eizenhöfer, Sarah Falkowski, Chloë Glover, Brad Parks and Kelin Whipple. T.A. Ehlers and Willi Kappler (University of Tübingen) are thanked for support with conducting modelling simulations and providing the modified version of Pecube that can be coupled to 2D Move restoration files.

3.8 Supplementary Info

This supplementary information section contains Supplementary Tables 1 and 2.

Supplementary video: Best fit pecube-kinematics model (VV29): https://youtu.be/PgtwTu4HgFQ

Supplementary Table 1 contains all modelled velocity runs and the corresponding goodness-of-fit values for canyon, mean, and interfluve ages. This compares the measured age, within published 2-sigma error to the modelled age, with a ±2 km and ±1 Ma model error.

Supplementary Table 2 contains models that are 75th percentile and above in goodness-of-fit, with all corresponding data from Supplementary Table 1, and added χ^2 analysis. This analysis computed this equation, $\chi^2 = \sum_i \frac{(age_{measured} - age_{modelled})^2}{(error_{measured})^2}$. For each modelled sample point within model error, the χ^2 value was computed, and the minimum for the sample was selected to go into the model χ^2 sum. The $p = 0.05$ value associated with this model's χ^2 is less than or equal to 119, wherein models with $\chi^2 \leq 119$ have statistical significance.

Model	GOF overall	GOF Can+I	GOFMean	GOF Can	GOF Ind	10 km Rate	4 km Rate	2 km Rate	Age of Incision	Rate @ incisio	Rate during incis
SP3_const_ef-12_ao-2.0_canyon@5.67	0.40199495	0.491667	0.22265152	0.65	0.333333333	3	3	3	5.67	3	3
SP3_const_ef-12_ao-2.5_canyon@5.67	0.4523064	0.524306	0.30830808	0.5625	0.486111111	3	3	3	5.67	3	3
SP3_const_ef-12_ao-2.7_canyon@1.33	0.40094276	0.411806	0.37921717	0.4625	0.361111111	3	3	3	1.33	3	3
SP3_const_ef-12_ao-2.7_canyon@3.33	0.40927609	0.424306	0.37921717	0.4875	0.361111111	3	3	3	3.33	3	3
SP3_const_ef-12_ao-2.7_canyon@5.67	0.42594276	0.449306	0.37921717	0.5375	0.361111111	3	3	3	5.67	3	3
SP3_const_ef-12_ao-3.0_canyon@5.67	0.42196128	0.433333	0.39921717	0.45	0.416666667	3	3	3	5.67	3	3
SP4_VV10_ef-12_ao-2.7_canyon@4Ma	0.58374579	0.640972	0.46929293	0.7125	0.569444444	1.05263158	0.5	0.5	4	0.5	0.5
SP4_VV10_ef-12_ao-2.7_canyon@8Ma	0.58374579	0.640972	0.46929293	0.7125	0.569444444	1.05263158	0.5	0.5	8	0.5	0.5
SP4_VV11_ef-12_ao-2.7_canyon	0.54776094	0.599306	0.44467172	0.7125	0.486111111	1.46341463	0.75	0.75	5.33	0.75	0.75
SP4_VV12_ef-12_ao-2.7_canyon	0.43631734	0.439931	0.42909091	0.26875	0.611111111	0.57142857	0.25	0.25	8	0.25	0.25
SP4_VV13_ef-12_ao-2.7_canyon	0.54346801	0.570139	0.49012626	0.6125	0.527777778	1.81818182	1	1	4	1	1
SP4_VV14_ef-12_ao-2.7_canyon	0.50889731	0.515972	0.49474747	0.4625	0.569444444	1	0.470588235	0.25	8	2.125	0.25
SP4_VV15_ef-12_ao-2.7_canyon	0.52059764	0.611806	0.33818182	0.6125	0.611111111	0.89171975	0.411764706	0.35	5.71	0.35	0.425
SP4_VV16_ef-12_ao-2.7_canyon	0.56013468	0.595139	0.49012626	0.6625	0.527777778	1.66666667	0.888888889	0.5	4	0.5	0.5
SP4_VV17_ef-12_ao-2.7_canyon	0.50069024	0.515972	0.47012626	0.5875	0.444444444	2.14285714	1.263157895	0.75	3.17	4	2.375
SP4_VV18_ef-12_ao-2.7_canyon	0.47430135	0.465972	0.4909596	0.4875	0.444444444	2.5	1.6	1	2.5	1	2.5
SP5_VV10_ef-12_ao-2.7_canyon	0.59207912	0.653472	0.46929293	0.7375	0.569444444	1.05263158	0.5	0.5	2	0.5	0.5
SP5_VV12_ef-12_ao-2.7_canyon	0.48631734	0.514931	0.42909091	0.41875	0.611111111	0.57142857	0.25	0.25	4	0.25	0.25
SP5_VV14_ef-12_ao-2.7_canyon@2Ma	0.47973064	0.472222	0.49474747	0.375	0.569444444	1	0.470588235	0.25	2	0.25	0.25
SP5_VV19_ef-12_ao-2.7_canyon@2.86Ma	0.54686027	0.628472	0.38363636	0.6875	0.569444444	0.96551724	0.451612903	0.4117647	2.86	0.35	0.425
SP5_VV19_ef-12_ao-2.7_canyon@4.86Ma	0.54686027	0.628472	0.38363636	0.6875	0.569444444	0.96551724	0.451612903	0.4117647	4.86	0.5	0.463
SP5_VV20_ef-12_ao-2.7_canyon	0.55393098	0.661806	0.33818182	0.7125	0.611111111	0.89171975	0.411764706	0.5	4	0.35	0.5
SP5_VV21_ef-12_ao-2.7_canyon	0.6131229	0.695139	0.44909091	0.7375	0.652777778	0.82840237	0.378378378	0.4117647	4.86	0.35	0.425
SP5_VV22_ef-12_ao-2.7_canyon	0.56059764	0.661806	0.35818182	0.7125	0.611111111	0.86956522	0.4	0.4	5	0.4	0.4
SP5_VV23_ef-12_ao-2.7_canyon@2.22Ma	0.49351431	0.571181	0.33818182	0.53125	0.611111111	0.96256684	0.45	0.45	2.22	0.45	0.45
SP5_VV23_ef-12_ao-2.7_canyon	0.48518098	0.558681	0.33818182	0.50625	0.611111111	0.96256684	0.45	0.45	4.44	0.45	0.45
SP5_VV24_ef-12_ao-2.7_canyon	0.46851431	0.533681	0.33818182	0.45625	0.611111111	0.6741573	0.3	0.3	3.33	0.3	0.3
SP5_VV25_ef-12_ao-2.7_canyon	0.56811027	0.650347	0.40363636	0.60625	0.694444444	0.77348066	0.35	0.35	5.71	0.35	0.35
SP5_VV26_ef-12_ao-2.7_canyon	0.60352694	0.703472	0.40363636	0.7125	0.694444444	0.77922078	0.352941176	0.4285714	4.67	0.3	0.525
SP5_VV27_ef-12_ao-2.7_canyon	0.61103956	0.692014	0.44909091	0.73125	0.652777778	0.82191781	0.375	0.5	4	0.3	0.9
SP5_VV28_ef-12_ao-2.7_canyon@2Ma	0.57795875	0.687847	0.35818182	0.68125	0.694444444	0.76502732	0.378378378	0.4117647	2	0.5	0.5
SP5_VV28_ef-12_ao-2.7_canyon	0.57795875	0.687847	0.35818182	0.68125	0.694444444	0.76502732	0.378378378	0.4117647	4.86	0.35	0.425
SP5_VV29_ef-12_ao-2.7_canyon	0.58629209	0.700347	0.35818182	0.70625	0.694444444	0.75949367	0.375	0.5	4	0.3	0.9
SP5_VV30_ef-12_ao-2.7_canyon	0.46699916	0.508681	0.38363636	0.40625	0.611111111	0.69589422	0.310800311	0.1616815	4	0	0.0417
SP5_VV31_ef-12_ao-2.7_canyon@2Ma	0.46699916	0.508681	0.38363636	0.40625	0.611111111	0.68965517	0.307692308	0.16	2	0	0
SP5_VV31_ef-12_ao-2.7_canyon@4Ma	0.45033249	0.483681	0.38363636	0.35625	0.611111111	0.68965517	0.307692308	0.16	4	0	0
SP5_VV31_ef-12_ao-2.7_canyon@8Ma	0.43366582	0.458681	0.38363636	0.30625	0.611111111	0.68965517	0.307692308	0.16	8	0	0
MAX	0.6131	0.7035	0.4947	0.7375	0.6944						
Standard Dev.	0.06489415	0.091631	0.06298261	0.1324594	0.102411731						
						GOFOverall					
	Min		Min		25th percentil	0.46699916					
	75th %'ile		75th %'ile		50th percentil	0.50889731					
	Max		Max		75th percentil	0.56811027					

Table 3.2. Tabulated goodness of fit for all model runs, including mean, canyon, interfluve, and overall fits.

Model	GOF overall	GOF Can+I	GOFMean	GOF Can	GOF Ind	Chi2
SP4_VV10_ef-12_ao-2.7_canyon@4Ma	0.58374579	0.640972	0.46929293	0.7125	0.569444444	113.663682
SP4_VV10_ef-12_ao-2.7_canyon@8Ma	0.58374579	0.640972	0.46929293	0.7125	0.569444444	113.357292
SP5_VV10_ef-12_ao-2.7_canyon	0.59207912	0.653472	0.46929293	0.7375	0.569444444	115.814356
SP5_VV21_ef-12_ao-2.7_canyon	0.6131229	0.695139	0.44909091	0.7375	0.652777778	109.787435
SP5_VV25_ef-12_ao-2.7_canyon	0.56811027	0.650347	0.40363636	0.60625	0.694444444	112.354857
SP5_VV26_ef-12_ao-2.7_canyon	0.60352694	0.703472	0.40363636	0.7125	0.694444444	112.351197
SP5_VV27_ef-12_ao-2.7_canyon	0.61103956	0.692014	0.44909091	0.73125	0.652777778	111.497372
SP5_VV28_ef-12_ao-2.7_canyon@2Ma	0.57795875	0.687847	0.35818182	0.68125	0.694444444	113.463487
SP5_VV28_ef-12_ao-2.7_canyon	0.57795875	0.687847	0.35818182	0.68125	0.694444444	113.234389
SP5_VV29_ef-12_ao-2.7_canyon	0.58629209	0.700347	0.35818182	0.70625	0.694444444	115.981024

Min		Min
75th %'ile		75th %'ile
Max		Max

Table 3.3. Tabulated models with overall GOF >75th percentile and corresponding chi-squared values.

Values less than 119 (p=0.05) are statistically significant.

3.9 References

Allmendinger, R.W., Zapata, T.R., 2000. The footwall ramp of the Subandean decollement, northernmost Argentina, from extended correlation of seismic reflection data. Tectonophysics 321, 37–55. https://doi.org/10.1016/S0040-1951(00)00077-9

Barke, R., Lamb, S., 2006. Late Cenozoic uplift of the Eastern Cordillera, Bolivian Andes. Earth Planet. Sci. Lett. 249, 350–367. https://doi.org/10.1016/j.epsl.2006.07.012

Benjamin, T., Johnson, N.M., Naeser, C.W., 1987. Recent rapid uplift in the Bolivian Andes: Evidence from fission-track dating. Geology 15, 680–683. https://doi.org/10.1130/0091-7613(1987)15<680:RRUITB>2.0.CO;2

Brandon, M.T., Roden-Tice, M.K., Carver, J.I., 1998. Late Cenozoic exhumation of the Cascadia accretionary wedge in the Olympic Mountains, northwest Washington State. Bull. Geol. Soc. Am. 110, 985–1009. https://doi.org/https://doi.org/10.1130/0016-7606(1998)110%3C0985:LCEOTC%3E2.3.CO;2

Braun, J., 2003. Pecube: A new finite-element code to solve the 3D heat transport equation including the effects of a time-varying, finite amplitude surface topography. Comput. Geosci. 29, 787–794. https://doi.org/10.1016/S0098-3004(03)00052-9

Braun, J., 2002. Quantifying the effect of recent relief changes on age-elevation relationships. Earth Planet. Sci. Lett. 200, 331–343. https://doi.org/10.1016/S0012-821X(02)00638-6

Buford Parks, V.M., McQuarrie, N., 2019. Kinematic, flexural, and thermal modelling in the Central Andes: Unravelling age and signal of deformation, exhumation, and uplift. Tectonophysics 766, 302–325. https://doi.org/10.1016/j.tecto.2019.06.008

Callot, P., Sempere, T., Odonne, F., Robert, E., 2008. Giant submarine collapse of a carbonate platform at the Turonian-Coniacian transition: The Ayabacas Formation, southern Peru. Basin Res. 20, 333–357. https://doi.org/10.1111/j.1365-2117.2008.00358.x

Carlotto, V., 2013. Paleogeographic and tectonic controls on the evolution of Cenozoic basins in the Altiplano and Western Cordillera of southern Peru. Tectonophysics 589, 195–219. https://doi.org/10.1016/j.tecto.2013.01.002

Clark, A.H., Farrar, E., Kontak, D.J., Langridge, R.J., Arenas F., M.J., France, L.J., McBride, S.L., Woodman, P.L., Wasteneys, H.A., Sandeman, H.A., Archibald, D.A., 1990. Geologic and geochronologic constraints on the metallogenic evolution of the Andes of southeastern Peru. Econ. Geol. 85, 1520–1583. https://doi.org/10.2113/gsecongeo.85.7.1520

Crowley, K.D., Cameron, M., Schaefer, R.I., 1991. Experimental studies of annealing of etched fission tracks in fluorapatite Author links open overlay panel. Geochim. Cosmochim. Acta 55, 1449–1465. https://doi.org/https://doi.org/10.1016/0016-7037(91)90320-5

DeCelles, P.G., Coogan, J.C., 2006. Regional structure and kinematic history of the Sevier fold-and-thrust belt, central Utah. Bull. Geol. Soc. Am. 118, 841–864. https://doi.org/10.1130/B25759.1

DeCelles, P.G., Ducea, M.N., Kapp, P., Zandt, G., 2009. Cyclicity in Cordilleran orogenic systems. Nat. Geosci. 2, 251–257. https://doi.org/10.1038/ngeo469

Ehlers, T.A., 2005. Crustal Thermal Processes and the Interpretation of Thermochronometer Data, in: Reiners, P.W., Ehlers, T.A. (Eds.), Low-Temperature Thermochronolgy: Techniques, Interpretations, and Applications: Reviews in Mineralogy & Geochemistry, v. 58. pp. 315–350. https://doi.org/https://doi.org/10.2138/rmg.2005.58.12

Ehlers, T.A., Chaudhri, T., Kumar, S., Fuller, C.W., Willett, S.D., Ketcham, R.A., Brandon, M.T., Belton, D.X., Kohn, B.P., Gleadow, A.J.W., Dunai, T.J., Fu, F.Q., 2005. Computational tools for low-temperature thermochronometer interpretation. Rev. Mineral. Geochemistry 58, 589–622. https://doi.org/10.2138/rmg.2005.58.22

Eichelberger, N., McQuarrie, N., Ryan, J., Karimi, B., Beck, S.L., Zandt, G., 2015. Evolution of crustal thickening in the central Andes, Bolivia. Earth Planet. Sci. Lett. 426, 191–203. https://doi.org/10.1016/j.epsl.2015.06.035

Espurt, N., Barbarand, J., Roddaz, M., Brusset, S., Baby, P., Saillard, M., Hermoza, W., 2011. A scenario for late Neogene Andean shortening transfer in the Camisea Subandean zone (Peru, 12°S): Implications for growth of the northern Andean Plateau. Bull. Geol. Soc. Am. 123, 2050–2068. https://doi.org/10.1130/B30165.1

Falkowski, S., Ehlers, T.A., McQuarrie, N., Buford Parks, V.M., Glover, C., Cadenas, J., n.d. Along-strike variations in the timing of exhumation in the eastern Peruvian Andes.

Farley, K.A., 2000. Helium diffusion from apatite: general behavior as illustrated by Duragno fluoroapatite. J. Geophys. Res. Solid Earth 105, 2903–2914. https://doi.org/https://doi.org/10.1029/1999JB900348

Farrar, E., Clark, A.H., Kontak, D.J., Archibald, D.A., 1988. Zongo-San Gabán zone: Eocene foreland boundary of the Central Andean orogen, northwest Bolivia and southeast Peru. Geology 16, 55–58. https://doi.org/10.1130/0091-7613(1988)016<0055:ZSGNZE>2.3.CO;2

Fedorov, A. V., Brierley, C.M., Lawrence, K.T., Liu, Z., Dekens, P.S., Ravelo, A.C., 2013. Patterns and mechanisms of early Pliocene warmth. Nature 496, 43–49. https://doi.org/10.1038/nature12003

Fedorov, A. V., Dekens, P.S., McCarthy, M., Ravelo, A.C., DeMenocal, P.B., Barreiro, M., Pacanowki, R.C., Philander, S.G., 2006. The Pliocene Paradox (Mechanisms for a Permanent El Niño). Science (80-.). 312, 1485–1489. https://doi.org/10.1126/science.1122666

Garzione, C.N., McQuarrie, N., Perez, N.D., Ehlers, T.A., Beck, S.L., Kar, N., Eichelberger, N., Chapman, A.D., Ward, K.M., Ducea, M.N., Lease, R.O., Poulsen, C.J., Wagner, L.S., Saylor, J.E., Zandt, G., Horton, B.K., 2017. Tectonic Evolution of the Central Andean Plateau and Implications for the Growth of Plateaus. Annu. Rev. Earth Planet. Sci. 45, 529–559. https://doi.org/10.1146/annurev-earth-063016-020612

Garzione, C.N., Molnar, P., Libarkin, J.C., MacFadden, B.J., 2006. Rapid late Miocene rise of the Bolivian Altiplano: Evidence for removal of mantle lithosphere. Earth Planet. Sci. Lett. 241, 543–556. https://doi.org/10.1016/j.epsl.2005.11.026

Gérard, B., Audin, L., Robert, X., Gautheron, C., Van, P., Beek, D., Bernet, M., Benavente, C., Delgado, F., 2020. Abbreviated title : Pliocene incision of the northern Altiplano. https://doi.org/10.1144/jgs2020-100/5170484/jgs2020-100.pdf

Ghoshal, S., McQuarrie, N., Robinson, D.M., Adhikari, D.P., Morgan, L.E., Ehlers, T.A., 2020. Constraining Central Himalayan (Nepal) Fault Geometry Through Integrated Thermochronology and Thermokinematic Modeling. Tectonics 33. https://doi.org/10.1029/2020TC006399

Gillis, R.J., Horton, B.K., Grove, M., 2006. Thermochronology, geochronology, and upper crustal structure of the Cordillera Real: Implications for Cenozoic exhumation of the central Andean plateau. Tectonics 25, 1–22. https://doi.org/10.1029/2005TC001887

Gilmore, M.E., Mcquarrie, N., Eizenhöfer, P.R., Ehlers, T.A., 2018. Testing the effects of topography , geometry and kinematics on modeled thermochronometer cooling ages in the eastern Bhutan Himalaya. Solid Earth 9, 599–627. https://doi.org/https://doi.org/10.5194/se-9-599-2018

Gotberg, N., McQuarrie, N., Caillaux, V.C., 2010. Comparison of crustal thickening budget and shortening estimates in southern Peru (12-14S): Implications for mass balance and rotations in the “Bolivian orocline.” Bull. Geol. Soc. Am. 122, 727–742. https://doi.org/10.1130/B26477.1

Hoke, G.D., Isacks, B.L., Jordan, T.E., Blanco, N., Tomlinson, A.J., Ramezani, J., 2007. Geomorphic evidence for post-10 Ma uplift of the western flank of the central Andes 18°30′ - 22°S. Tectonics 26, 1–17. https://doi.org/10.1029/2006TC002082

Horton, B.K., 2018. Sedimentary record of Andean mountain building. Earth-Science Rev. 178, 279–309. https://doi.org/10.1016/j.earscirev.2017.11.025

Horton, B.K., Perez, N.D., Fitch, J.D., Saylor, J.E., 2015. Punctuated shortening and subsidence in the Altiplano Plateau of southern Peru: Implications for early Andean mountain building. Lithosphere 7, 117–137. https://doi.org/10.1130/L397.1

Houseman, G.A., Mckenzie, D.P., Molnar, P., 1981. Convectiv instability of a thickened boundary layer and its relavance for the thermal evolution of convergent belts. J. Geophys. Res. 86, 6115–6132. https://doi.org/https://doi.org/10.1029/JB086iB07p06115

Husson, L., Sempere, T., 2003. Thickening the Altiplano crust by gravity-driven crustal channel flow. Geophys. Res. Lett. 30, 1–4. https://doi.org/10.1029/2002gl016877

Jaillard, E., Cappetta, H., Ellenberger, P., Feist, M., Grambast-Fessard, N., Lefranc, J.P., Sigé, B., 1993. Sedimentology, palaeontology, biostratigraphy and correlation of the late cretaceous vilquechico group of southern Peru. Cretac. Res. 14, 623–661. https://doi.org/10.1006/cres.1993.1044

Jeffery, M.L., Ehlers, T.A., Yanites, B.J., Poulsen, C.J., 2013. Quantifying the role of paleoclimate and Andean Plateau uplift on river incision. J. Geophys. Res. Earth Surf. 118, 852–871. https://doi.org/10.1002/jgrf.20055

Kar, N., Garzione, C.N., Jaramillo, C., Shanahan, T., Carlotto, V., Pullen, A., Moreno, F., Anderson, V., Moreno, E., Eiler, J., 2016. Rapid regional surface uplift of the northern Altiplano plateau revealed by multiproxy paleoclimate reconstruction. Earth Planet. Sci. Lett. 447, 33–47. https://doi.org/10.1016/j.epsl.2016.04.025

Kennan, L., Lamb, S.H., Hoke, L., 1997. High-altitude palaeosurfaces in the Bolivian Andes: Evidence for late Cenozoic surface uplift. Geol. Soc. Spec. Publ. 120, 307–323. https://doi.org/10.1144/GSL.SP.1997.120.01.20

Kley, J., 1996. Transition from basement-involved to thin-skinned thrusting in the Cordillera Oriental of southern Bolivia. Tectonics 15, 763. https://doi.org/10.1029/95TC03868

Klinck, B.A., Ellison, R.A., Hawkins, M.P., Compilers, 1986. The geology of the Cordillera Occidental and Altiplano west of Lake Titicaca, southern Peru. British Geol. Survey Inst. Geologico Minero Metalurgico, Lima, Peru.

Kontak, D.J., Clark, A.H., Farrar, E., Archibald, D.A., Baadsgaard, H., 1990a. Late Paleozoic-early Mesozoic magmatism in the Cordillera de Carabaya, Puno, southeastern Peru: Geochronology and petrochemistry. J. South Am. Earth Sci. 3, 213–230. https://doi.org/10.1016/0895-9811(90)90004-K

Kontak, D.J., Clark, A.H., Farrar, E., Archibald, D.A., Baadsgaard, H., 1987. Scientific communications: Geochronological data for tertiary granites of the southeast Peru segment of the central Andean tin belt. Econ. Geol. 82, 1611–1618. https://doi.org/10.2113/gsecongeo.82.6.1611

Kontak, D.J., Farrar, E., Clark, A.H., Archibald, D.A., 1990b. Eocene tectono-thermal rejuvenation of an upper Paleozoic-lower Mesozoic terrane in the Cordillera de Carabaya, Puno, southeastern Peru, revealed by KAr and 40Ar/39Ar dating. J. South Am. Earth Sci. 3, 231–246. https://doi.org/10.1016/0895-9811(90)90005-L

Krystopowicz, N.J., Currie, C.A., 2013. Crustal eclogitization and lithosphere delamination in orogens. Earth Planet. Sci. Lett. 361, 195–207. https://doi.org/10.1016/j.epsl.2012.09.056

Lamb, S., Hoke, L., 1997. Origin of the high plateau in the Central Andes, Bolivia, South America. Tectonics 16, 623–649. https://doi.org/10.1029/97TC00495

Lease, R.O., Ehlers, T.A., 2013. Incision into the eastern Andean Plateau during Pliocene cooling. Science (80-.). 341, 774–776. https://doi.org/10.1126/science.1239132

Lecaros, W.L., Moncayo, O.P., Vilchez, L.V., Fernándex, A.S., 1999. Boletín No. 136 (Serie A): Memoria Explicativa del mapa geológico del Perú- Escala 1:1 000 000, Pengaruh Harga Diskon Dan Persepsi Produk Terhadap Nilai Belanja Serta Perilaku Pembelian Konsumen. Instituto Geológico Minero Y Metalúrgico (INGEMMET), Lima, Peru.

Lock, J., Willett, S.D., 2008. Low-temperature thermochronometric ages in fold-and-thrust belts. Tectonophysics 456, 147–162. https://doi.org/10.1016/j.tecto.2008.03.007

McQuarrie, N., Barnes, J.B., Ehlers, T.A., 2008. Geometric, kinematic, and erosional history of the central Andean Plateau, Bolivia (15-17°S). Tectonics 27. https://doi.org/10.1029/2006TC002054

McQuarrie, N., Ehlers, T.A., 2017. Techniques for understanding fold-and-thrust belt kinematics and thermal evolution, in: Law, R.D., Thigpen, J.R., Merschat, A.J., Stowell, H.H. (Eds.), Linkages and Feedbacks in Orogenic Systems. Geological Society of America, pp. 1–30. https://doi.org/10.1130/2017.1213(02)

McQuarrie, N., Ehlers, T.A., 2015. Influence of thrust belt geometry and shortening rate on thermochronometer cooling ages: Insights from thermokinematic and erosion modeling of the Bhutan Himalaya. Tectonics 34, 1–25. https://doi.org/10.1002/2014TC003783

McQuarrie, N., Eizenhöfer, P.R., Long, S.P., Tobgay, T., Ehlers, T.A., Blythe, A., Morgan, L., Gilmore, M.E., Dering, G., 2019. The influence of foreland structures on hinterland cooling: evaluating the drivers of exhumation in the eastern Bhutan Himalaya. Tectonics 38, 1–29. https://doi.org/10.1029/2018TC005340

Navarro-Serrano, F., López-Moreno, J.I., Domínguez-Castro, F., Alonso-González, E., Azorin-Molina, C., El-Kenawy, A., Vicente-Serrano, S.M., 2020. Maximum and minimum air temperature lapse rates in the Andean region of Ecuador and Peru. Int. J. Climatol. 1–19. https://doi.org/10.1002/joc.6574

Noblet, C., Leonardi, G., Taquet, P., Marocco, R., Cordova, E., 1995. Another discovery of Dinosaur foot-prints in the Red Beds Formation (Cuzco-Sicuani basin, Southern Peru): stratigraphic and tectonic implications. Comptes Rendus Hebd. des Séances l'Académie des Sci. Série IIA (Sciences la Terre des planètes) 320, 785–791.

Olsen, J.E.S., McQuarrie, N., Robinson, D.M., 2019. Determining kinematic order and relative age of faulting via flexural-kinematic restoration: a case study in far western Nepal. Basin Res. https://doi.org/doi: 10.1111/bre.12362

Perez, N.D., 2015. Cenozoic deformation history of the Andean plateau in southern Peru : Stratigraphic, structural, and geochronologic constraints. The University of Texas at Austin.

Perez, N.D., Horton, B.K., 2014. Oligocene-miocene deformational and depositional history of the andean hinterland basin in the northern altiplano Plateau, Southern Peru. Tectonics 33, 1819–1847. https://doi.org/10.1002/2014TC003647

Perez, N.D., Horton, B.K., Carlotto, V., 2016a. Structural inheritance and selective reactivation in the central Andes: Cenozoic deformation guided by pre-Andean structures in southern Peru. Tectonophysics 671, 264–280. https://doi.org/10.1016/j.tecto.2015.12.031

Perez, N.D., Horton, B.K., McQuarrie, N., Stubner, K., Ehlers, T.A., 2016b. Andean shortening, inversion and exhumation associated with thin- and thick-skinned deformation in southern Peru. Geol. Mag. 1–29. https://doi.org/10.1017/S0016756816000121

Rak, A.J., McQuarrie, N., Ehlers, T.A., 2017. Kinematics, exhumation, and sedimentation of the north-central Andes (Bolivia): An integrated thermochronometer and thermokinematic modeling approach. Tectonics 36, 1–31. https://doi.org/10.1002/2016TC004440

Reiners, P.W., Spell, T.L., Nicolescu, S., Zanetti, K.A., 2004. Zircon (U-Th)/He thermochronometry: He diffusion and comparisons with 40Ar/39Ar dating. Geochim. Cosmochim. Acta 68, 1857–1887. https://doi.org/10.1016/j.gca.2003.10.021

Saylor, J.E., Horton, B.K., 2014. Nonuniform surface uplift of the Andean plateau revealed by deuterium isotopes in Miocene volcanic glass from southern Peru. Earth Planet. Sci. Lett. 387, 120–131. https://doi.org/10.1016/j.epsl.2013.11.015

Schildgen, T.F., Ehlers, T.A., Whipp, D.M., Van Soest, M.C., Whipple, K.X., Hodges, K. V., 2009. Quantifying canyon incision and Andean Plateau surface uplift, southwest Peru: A thermochronometer and numerical modeling approach. J. Geophys. Res. Earth Surf. 114, 1–22. https://doi.org/10.1029/2009JF001305

Schildgen, T.F., Hodges, K. V., Whipple, P.W., Reiners, P.W., Pringle, M.S., 2007. Uplift of the western margin of the Andean plateau revealed from canyon incision history, southern Peru. Geology 35, 523–526. https://doi.org/10.1130/G23532A.1

Sempere, T., Carlier, G., Soler, P., Fornari, M., Carlotto, V., Jacay, J., Arispe, O., Néraudeau, D., Cárdenas, J., Rosas, S., Jiménez, N., 2002. Late Permian-Middle Jurassic lithospheric thinning in Peru and Bolivia, and its bearing on Andean-age tectonics. Tectonophysics 345, 153–181. https://doi.org/10.1016/S0040-1951(01)00211-6

Sempere, T., Herail, G., Oller, J., Bonhomme, M.G., 1990. Late Oligocene-Early Miocene major tectonic crisis and related basins in Bolivia. Geology 18, 946–949. https://doi.org/10.1130/0091-7613(1990)018<0946:LOEMMT>2.3.CO;2

Sigé, B., Sempere, T., Butler, R.F., Marshall, L.G., Crochet, J.Y., 2004. Age and stratigraphic reassessment of the fossil-bearing Laguna Umayo red mudstone unit, SE Peru, from regional stratigraphy, fossil record, and paleomagnetism. Geobios 37, 771–794. https://doi.org/10.1016/j.geobios.2003.06.006

Sobolev, S. V., Babeyko, A.Y., 2005. What drives orogeny in the Andes? Geology 33, 617–620. https://doi.org/10.1130/G21557.1

Sundell, K.E., Saylor, J.E., Lapen, T.J., Styron, R.H., Villarreal, D.P., Usnayo, P., Cárdenas, J., 2018. Peruvian Altiplano Stratigraphy Highlights Along-Strike Variability in Foreland Basin Evolution of the Cenozoic Central Andes. Tectonics 37, 1876–1904. https://doi.org/10.1029/2017TC004775

Ward, K.M., Zandt, G., Beck, S.L., Wagner, L.S., Tavera, H., 2016. Lithospheric structure beneath the northern Central Andean Plateau from the joint inversion of ambient noise and earthquake-generated surface waves. J. Geophys. Res. Solid Earth 121, 8217–8238. https://doi.org/10.1002/2016JB013237

Whipp, D.M., Ehlers, T.A., Blythe, A.E., Huntington, K.W., Hodges, K. V., Burbank, D.W., 2007. Plio-Quaternary exhumation history of the central Nepalese Himalaya: 2. Thermokinematic and thermochronometer age prediction model. Tectonics 26, 1–23. https://doi.org/10.1029/2006TC001991

Whipp, D.M., Ehlers, T.A., Braun, J., Spath, C.D., 2009. Effects of exhumation kinematics and topographic evolution on detrital thermochronometer data. J. Geophys. Res. Earth Surf. 114, 1–20. https://doi.org/10.1029/2008JF001195

Whipple, K.X., Gasparini, N.M., 2014. Tectonic control of topography, rainfall patterns, and erosion during rapid post-12 Ma uplift of the Bolivian Andes. Lithosphere 6, 251–268. https://doi.org/10.1130/L325.1

Whipple, K. X., Kirby, E., & Brocklehurst, S. H. (1999). Geomorphic limits to climate-induced increases in topographic relief. *Nature*, *401*(6748), 39–43. https://doi.org/10.1038/43375

Whipple, K. X., & Meade, B. J. (2006). Orogen response to changes in climatic and tectonic forcing. *Earth and Planetary Science Letters*, *243*(1–2), 218–228. https://doi.org/10.1016/j.epsl.2005.12.022

4.0 Preservation and Translation Of Topography in Convergent Orogens: Landscape Response in the Central Bolivian Andes over Geologic Time

4.1 Introduction

Convergent orogens provide a key study area to investigate the balance between tectonic drivers and erosional processes on the formation and preservation of the modern landscape. Orogen-scale models of convergence and landscape evolution typically assume a critically tapered wedge where shortening and rock uplift occurs everywhere within the wedge, while mapped and seismically imaged structures highlight shortening by horizontal and vertical displacement along discrete fault planes where rock uplift is generated by motion over ramps, out-of-sequence thrusting, and/or a frontal surface breaking fault. To evaluate the balance between uplift, translation and erosion of topography in central Bolivia, we integrate a detailed balanced cross-section with a physics-based surface processes model. Fault ramps are the main driver of uplift and generate elevated topography, while fault décollements and flats translate the elevated topography in the direction of transport. Lateral translation of topography requires a critical balanced between rock strength, precipitation, and rate of shortening (translation). Slow shortening rates in the Bolivian Andes (8 mm/y) compared to other regions historically studied using landscape evolution (e.g. Himalaya at >25 mm/y), require limited erosion to maintain elevated landscapes and produce translated topography. Rock erodibility was critical for promoting, or inhibiting erosion and provides a primary control on translated topography and morphology. The role that precipitation has, while also strong, is notably different from that of rock strength. Changes in global precipitation values strongly influenced absolute elevations of model generated topography, while rock strength more strongly influenced the width of elevated topography. Imposed orographic precipitation further decreases elevations and increases variability in channel and interfluve morphology due to the variability in precipitation magnitude and locality. An essential part in producing the translated topography that was critical to replicating modern plateau morphology was the forward propagation of the fault system at ~4 Ma which lengthened the décollement. Following this time (4 -0 Ma), all models show an eastward migration of the drainage divide. For models with low rock strength this migration is limited, ~ 5-20 km (depending on precipitation magnitude). For high rock strength the divide migrates 30-35 km. When high rock strength is combined with higher/variable precipitation, the drainage divide may migrate back west, but elevated topography retains its translated signature of 30-35 km.

Detailed study of the evolution of deformation, erosion, and the resulting landscape across compressional settings is critical to our understanding of the balance between tectonically driven displacement and erosion. In tectonically active areas, uplift along active faults exerts a first-order control on the resulting morphology of the mountain range (Stockmal et al., 2007; Willett, Slingerland, & Hovius, 2001). Structural uplift is called upon to explain both elevated low relief surfaces, (Barke & Lamb, 2006; M. K. Clark et al., 2006; Spotila, Farley, & Sieh, 1998; Whipple & Gasparini, 2014) and increased structural relief via river incision (e.g. DiBiase & Whipple, 2011; Wager, 1933; Whipple & Tucker, 1999). Rock uplift in convergent tectonic settings is generated by tectonic displacement over fault ramps and is reflected in fluvial channels as a steepening or shallowing of slopes in response to this change in rock uplift rate (Whipple & Tucker, 1999; Wobus, Whipple, & Hodges, 2006). The fluvial channels provide the boundary conditions to which hillslope processes respond (Whipple & Tucker, 1999). This correlation between uplift, relief and channel steepness has led to the interpretation that in absence of strong changes in rock strength and precipitation, broad regions of elevated relief, slope and channel steepness require equally broad mechanisms of active uplift (Gasparini & Whipple, 2014; Kirby & Whipple, 2012; Whipple, Shirzaei, Hodges, & Ramon Arrowsmith, 2016; Wobus, Whipple, Kirby, et al., 2006). However, in these compressional orogens, locations of rock uplift along ramps are separated by fault flats (or décollements), and that displacement along these flats drive lateral translation of topography (Eizenhöfer et al., 2019), the migration of drainage divides (Willett & Brandon, 2002; Willett et al., 2001), and distributed elevated normalized channel steepness (Ksn) (Eizenhöfer et al., 2019; Miller et al., 2007). We argue that the morphology of mountains is a function of ongoing or former uplift and translation. Sequentially restored cross sections are a model of how fault location, geometry and kinematics combine to reproduce the geology exposed at the earth's surface and include estimates of vertical uplift and horizontal translation. Because uplift of topographic surfaces and the creation of elevated relief, slope and channel steepness is intrinsically connected to subsurface structures that impart vertical motion, we posit that viable models of fault location, geometry and kinematics should also combine to reproduce the shape of modern topography and associated geomorphic metrics.

The central Andes are a location of ongoing debate over the drivers of elevated low relief surfaces, incision, steep slopes, relief and high Ksn values (Gasparini & Whipple, 2014; Gubbels et al., 1993; Lease & Ehlers, 2013; Norton & Schlunegger, 2011; Safran, Blythe, & Dunne, 2006; Schlunegger, Norton, & Zeilinger, 2011; Whipple & Gasparini, 2014) In northern Bolivia, geomorphology studies have hypothesized that elevated Ksn values that gradually decrease along tens of kilometers in the direction of transport to the northwest of the high peaks of the eastern Cordillera are due to either a gradual decline in rock uplift rates across the region (Safran et al., 2005; Whipple & Gasparini, 2014) or the distribution of orographic rainfall (Schlunegger et al.,

2011) Perched, low relief surfaces, such as the San Juan del Oro surfaces of southern Bolivia (e.g. Gubbels et al., 1993), are commonly interpreted to record a landscape morphology prior to a change in condition (such as uplift) that lead to river incision (Barke & Lamb, 2006; Kirby & Whipple, 2012; Whipple & Gasparini, 2014). In Bolivia these surfaces have been interpreted as once contiguous with the low elevation foreland, and thus represent 1.5-3 km of uplift since ~ 9-10 Ma (Barke & Lamb, 2006; Hoke & Garzione, 2008; Whipple & Gasparini, 2014). The central Bolivian Andes (Figure 1) are particularly intriguing with respect to these debates due to the concentration of perched, low relief surfaces, elevated Ksn values and maximum mean annual rainfall all within a 50 km distance. Across the Bolivian Andes, spatial and temporal changes in shortening amount and geometry due to evolving structures are well-documented through a suite of studies combining structural geology and low-temperature thermochronology (Anderson et al., 2017; Barnes et al., 2006; Benjamin et al., 1987; Eichelberger, 2014; McQuarrie, Barnes, et al., 2008; McQuarrie, Ehlers, et al., 2008; Rak et al., 2017) including central Bolivia (Barnes et al., 2012; Buford Parks & McQuarrie, 2019). Fault location, geometry and kinematics constrained by thermo-kinematic models that can reproduce both mapped surface geology and low temperature thermochronometer ages in Central Bolivia (e.g. Buford Parks & McQuarrie, 2019) provide an opportunity to evaluate the predicted topographic and geomorphic metrics created from discrete structures and locations of uplift combined with horizontal translation. We assess if the cross-section geometry and associated kinematics is viable by forward modelling the structural kinematics in Midland Valley/Petroleum Exports' 2D Move, and then incorporating these kinematics into the surface processes model CASCADE. Our goal is to determine the predicted topographic evolution implied by the proposed sequence of deformation and compare the modeled topography and topographic metrics (slope, relief, and Ksn values) to observed topography and metrics. In doing so, we also evaluate the impact of horizontal translation on the formation of perched, low relief surfaces.

4.2 Background

4.2.1 CASCADE

Paleotopography of mountain belts is extremely difficult to determine through inverse methods (Olen, Ehlers, & Densmore, 2012; Valla, Herman, van der Beek, & Braun, 2010). However, a forward model of topographic evolution both requires a known or testable history of both structural uplift and erosional exhumation. Simple topographic estimates of increased elevation in response to structural uplift histories (Buford Parks & McQuarrie, 2019; Gilmore et al., 2018; McQuarrie & Ehlers, 2017a) do not take into account physically-based erosional surface processes. Predictions of topography need to include location and rate of uplift as well as physics-based descriptions of surface erosion and sediment transport (Beaumont, Fullsack, & Hamilton, 1992; Chase, 1992; Eizenhöfer et al., 2019; Seidl & Dietrich, 1992; Seidl, Dietrich, & Kirchner, 1994; Whipple & Tucker, 1999; Willett, 1999). To facilitate integration of structural uplift with physics-based descriptions of surface erosion, We are using a modified version of the surface process model CASCADE (Braun & Sambridge, 1997; Braun, Zwartz, & Tomkin, 1999); this model is ideal for modelling long-term landscape evolution because CASCADE inherently allows vertical and lateral translation of nodes in response to input kinematic vectors. This modified version of CASCADE is coupled to accept the kinematic grids modelled in the structural geology software Move (Petroleum Exports/Midland Valley) (Eizenhöfer et al., 2019).

Precipitation in CASCADE is applied as either a constant, global precipitation value that is spatially and temporally invariant and based on assigned atmospheric moisture and minimum and maximum sea level temperatures, or a suite of elevation-dependent orographic precipitation values determined from inputted climate parameters (Yanites & Ehlers, 2012, 2016). In a majority of the world, particularly when modelling large-scale convergent orogens, constant precipitation does not represent the real world precipitation pattern. Global satellite data records modern orographically-induced precipitation focused on the windward side of mountains (e.g. Bookhagen & Burbank, 2006; Bookhagen & Strecker, 2008; Nesbitt & Anders, 2009; Smith et al., 2005) Additionally, modelling orogen-scale precipitation requires a temporal and spatial element that encompasses a suite of well-documented variations in precipitation and temperature in response to orography (Insel, Poulsen, & Ehlers, 2010) and global climate change events (Jeffery et al., 2013; Salzmann et al., 2011). Particularly, in central Bolivia, there is a strong mountain-front orographic precipitation gradient where precipitation decreases from ~4 m/y in the front of the system to ~0.35 m/y in the hinterland. The majority of this change in precipitation occurs over ~75 km.

CASCADE's in-built orographic precipitation is based on dynamic orography, wherein a forced rise in air creates dynamic instability due to relative humidity and causes rainout. However, a main driver of precipitation in South America is convective precipitation, where in addition to the dynamic instabilities, there is also heating of the rising air mass which drives convective storms (Carvalho, Jones, & Liebmann, 2002; Insel et al., 2010). This causes the air to rise faster, and leads to heavier storms, creating a different pattern of rain out than when there is only dynamic orography (Bechtold et al., 2004). This means that CASCADE's in-built orographic precipitation module will struggle to replicate the modern Andean orographic gradient. The importance of incorporating both orographic precipitation and regional climate change through time lead to the creation and incorporation of a module that imposes precipitation modelled externally from a global climate model. Precipitation patterns from the paleoclimate model ECHAM5 are added as input into CASCADE. ECHAM5 is a global atmospheric general circulation model that is ground-truthed with both paleoclimate proxy data and thermochronology (Mutz et al., 2018)

CASCADE computes the change in elevation for nodes in the network based on fluvial geomorphology and empirically derived relationships between rock uplift, lateral translation, and erodibility properties as follows:

$$\frac{dh}{dt} = v_u(x) - v_h(x)\frac{dh}{dt} - \mathrm{K}A(x)^m S(x)^n, (1)$$

where $v_h(x)$[L/t] represents the lateral advection rate (Miller et al., 2007; Willett et al., 2001), and $v_u(x)$ [L/t] represents the rock uplift rate, A [L^2] is the drainage area, and S [] is the channel slope (Whipple, 2001; Whipple & Tucker, 1999). Bulk erodibility, K [$t^{-1}L^{1-2m}$], is dependent on rock erodibility $\mathrm{K_f}$[], precipitation p [L/t], erosion-deposition length scale l_f[L], and a proportionality constant c[$(t/L)^{1/2}$] used as an input parameter in this version of CASCADE, such that

$$\mathrm{K} = \frac{\mathrm{K}_f\sqrt{p}}{l_f c}. (2)$$

Equation 1 highlights an inverse relationship between velocity and rock strength/ erodibility. If erodibility is much higher than velocity (v_h), topography created by uplift remains proximal to the structures producing that uplift, and topography reaches steady state (Eizenhöfer et al., 2019). Conversely if erodibility is small compared to the horizontal component of velocity, uplifted topography is translated in the direction of lateral advection, creating a legacy landscape (Eizenhöfer et al., 2019).

4.2.2 Andean Geomorphology

The central Andes is composed of several distinct tectonogeomorphic zones. These zones progress from west to east as follows: the Altiplano Plateau, a low-relief, internally drained basin of ~3.7 km elevation located between the volcanic peaks of the Western Cordillera and the faulted and folded high peaks of the Eastern Cordillera. The Eastern Cordillera (EC) and Interandean Zone (IAZ) host a thick (~15 km) continuous succession of Paleozoic marine siliclastic rocks and a discontinuous section (2-4 km) of non-marine Carboniferous through Cretaceous rocks (T. Sempere, 1995) deformed by faults and narrow folds. These two zones compose a bivergent thrust belt that reaches ~6.4 km in elevation in the EC and decreases in elevation toward the east through the IAZ and Subandes (SA). Both the EC and IAZ are uplifted due to basement thrust faults (Kley, 1996; Kley et al., 1999; McQuarrie, 2002). The SA is the actively deforming portion of the Andean fold-and-thrust belt (FTB). In Central Bolivia focused uplift and erosion of the SA has exposed Cambrian through Cretaceous faulted strata and up to 5-7 km of Tertiary foreland basin sediments (Baby et al., 1995; McQuarrie, 2002).

Previous studies have hypothesized that perched low relief surfaces identified across the eastern Cordillera of Bolivia (Barke & Lamb, 2006; Hoke & Garzione, 2008; Whipple & Gasparini, 2014) suggest broad, plateau-wide uplift across the Andean Plateau of 1.5-2 km (Barke & Lamb, 2006; Hoke & Garzione, 2008). In addition to this potential orogen-wide uplift, marked variations in the elevation of these surfaces from northwest to southeast exist along the frontal portion of the Eastern Cordillera, as do hanging valleys and variations in patterns of channel steepness. Based on these variations, Whipple & Gasparini, (2014) argue for marked along strike changes in structural uplift (locations of ramps and/or out-of-sequence faults).

Channel steepness (Ksn) values, a measure of channel slope normalized by drainage area, are elevated across a 200 km long (strike parallel) 30-50 km wide (strike perpendicular) swath in northern Bolivia, directly to the east of the high peaks of the EC (Whipple and Gaspirini, 2014; Gaspirini and Whipple, 2014; Eizenhöfer et al., 2019) between 15-17°S. In contrast, in central Bolivia near 18°S, the elevated Ksn values are ~50-100 km closer to the front of the system, and span an across strike distance of 20-25 km (Figure 4.1) (Whipple & Gasparini, 2014).

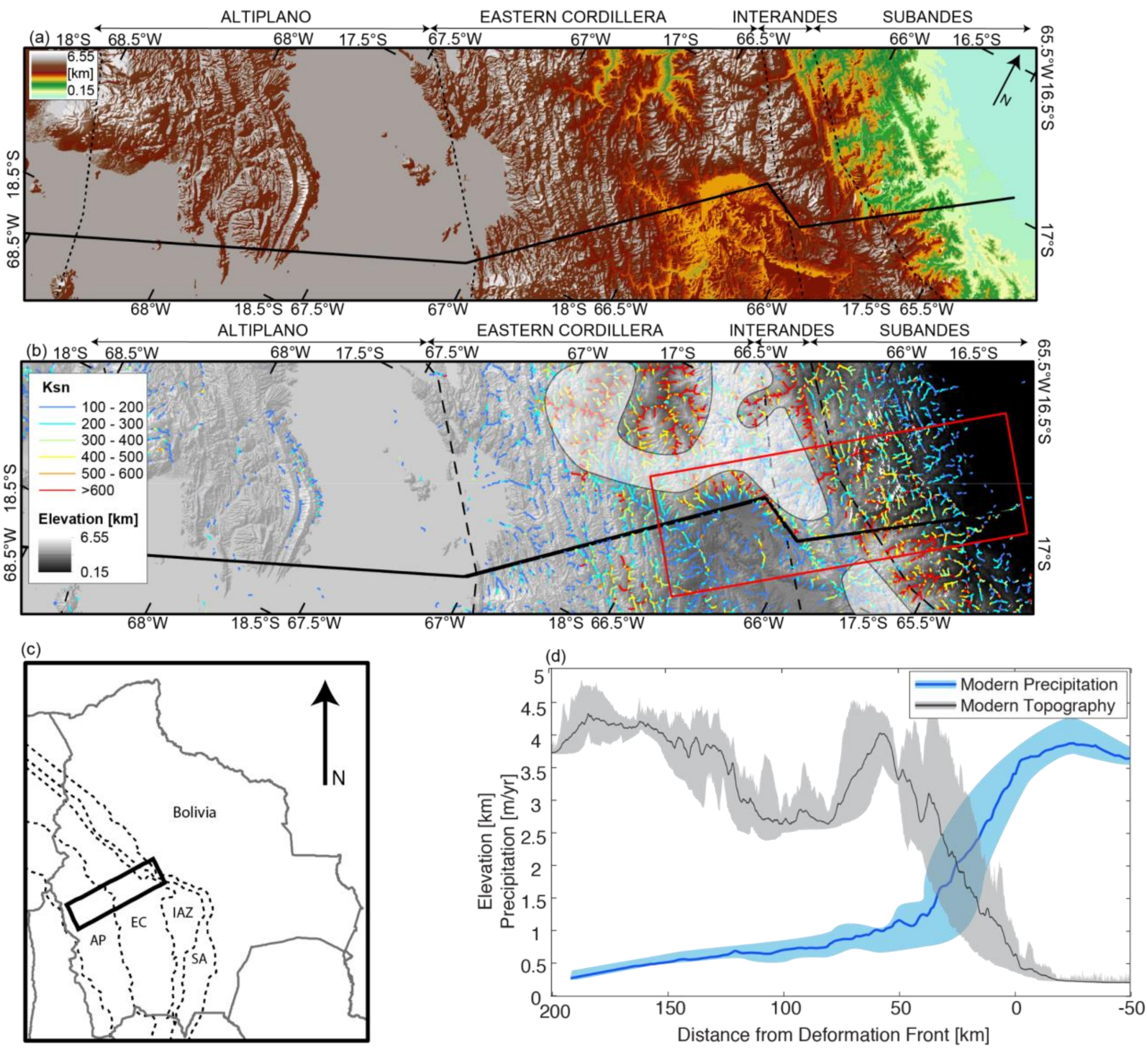


Figure 4.1. Area map, reference topography and precipitation

(a) Combined ASTER and TanDEM-X DEM of the study area, solid black line is the cross-section line. (b) Hillshade DEM with classified channel steepness (Ksn) with reference concavity of 0.5. and (gray transparency) low relief surfaces (c) Reference map with bounding tectonogeomorphic zones (d) 50 km swath averaged topography (ASTER and TanDEM-X) and precipitation *(Fick & Hijmans, 2017)* along the cross-section. (modified from Buford Parks and McQuarrie, 2019)

4.3 Methods

4.3.1 Kinematics

Using the kinematics from Buford Parks and McQuarrie (2019) (Figure 4.2), we use the balanced cross-section that was sequentially deformed in *Move* (Petroleum Exports/Midland Valley 2015), and input it into CASCADE. When the cross-section was sequentially deformed in *Move*, we followed established procedure for deformation, isostasy, and erosion (Buford Parks & McQuarrie, 2019; McQuarrie & Ehlers, 2015, 2017b) and exported the steps from *Move* after every ~7.5 km of shortening. *Move*-generated topography forms due to uplift and translation along proscribed fault geometries, and is eroded above a set, westward-increasing angle from the deformation front and below which the uplifted topography is preserved, and where sedimentation occurs below 0 km. Figure 4.2 highlights a model of the sequential deformation of the central Andes over the last 12 Myr, the location of the deformation front through time, and the generation of topography in *Move* (Supplementary Video 1: https://youtu.be/K_uIfU917RY). This kinematic sequence relies heavily on two out-of-sequence (OOS) faults in the last two deformation steps (3.2- 0 Ma) which total ~15 km of OOS motion on the faults on the boundary between the IAZ and SA (hollow arrows, Figure 4.2e-f). The shortening rate used in this study is 8 mm/y for the duration of shortening, and was determined to be the best fit through thermal modelling and comparison to measured thermochronometer data in the region (Buford Parks & McQuarrie, 2019).

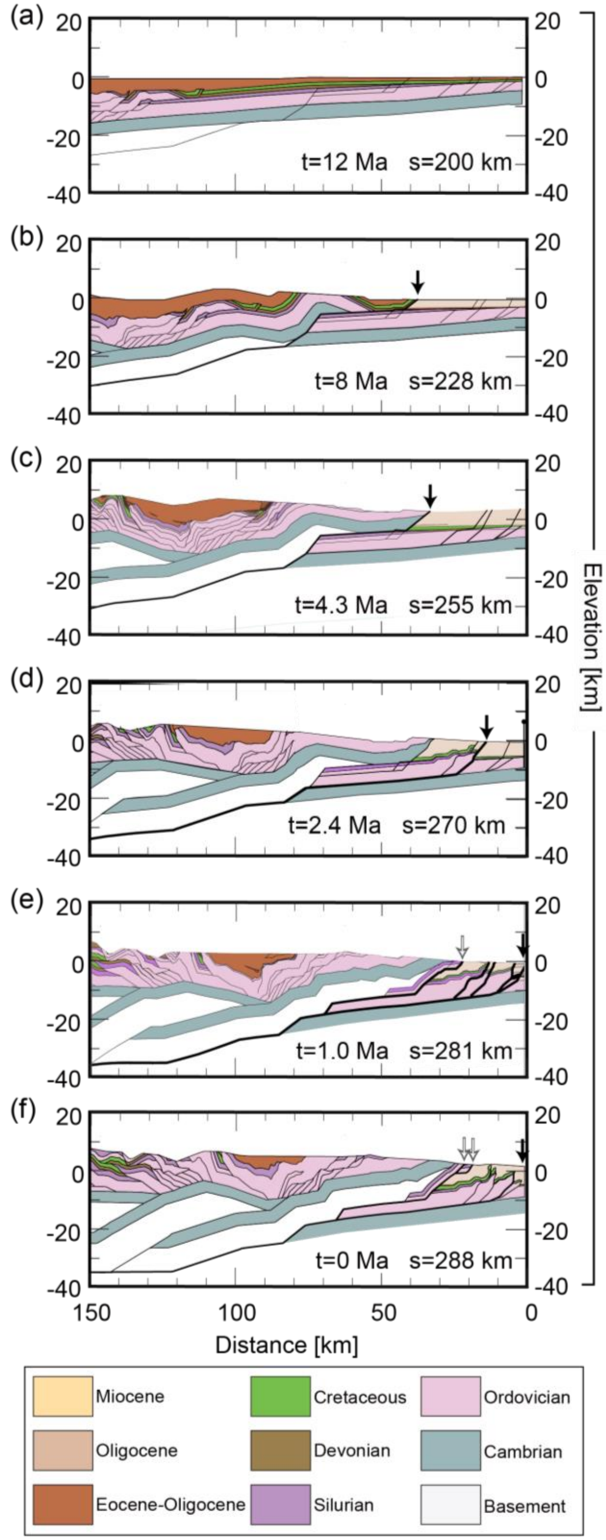


Figure 4.2. Sequence used for fault kinematics

Solid black arrow denotes furthest deformation front propagation; hollow black arrows highlight out-of-sequence thrusts. (modified from Buford Parks and McQuarrie, 2019)

4.3.2 CASCADE

The version of CASCADE (Braun & Sambridge, 1997; Braun et al., 1999) we use was rewritten in Fortran90, including modification to file input/output, precipitation (Yanites & Ehlers, 2012, 2016; this study), and Move-linked kinematics (Eizenhöfer et al., 2019). Our CASCADE model domain spans 150 km in the direction of transport, with the furthest propagation of the deformation front located at ~10 km in model space. The CASCADE model domain is 50 km wide, with an initial average node spacing of 1.0 km. The model runs for the duration Subandean shortening (~91 km) over 11.5 My, starting from a flat, foreland basin (Figure 4.2a) through the activation of the Subandean basement and surface structures. We allow an initial 0.1 My with no deformation for *Cascade* to equilibrate and form initial fluvial networks. The resultant displacement vectors from the incremental Move steps are assigned to the surface nodes in the CASCADE triangulated irregular network (TIN). These nodes move according to the displacement vectors from Move, both laterally along fault flats where there is little vertical uplift, and vertically up fault ramps, where lateral advection decreases (Eizenhöfer et al., 2019).

In CASCADE, we evaluated changes in erosion by independently changing rock strength and precipitation parameters. Rock strength was changed by modifying the parameter K_f (bulk erodibility) in the input file. We tested a range of rock strengths over an order of magnitude, from 10^{-4} to 10^{-5}. We used two different methods of applying precipitation to the model space: (1) a constant, global precipitation that does not change temporally or spatially throughout the CASCADE model, and (2) a spatially and temporally dependent 'imposed' precipitation. The constant, global precipitation is defined by input minimum and maximum sea level temperatures and atmospheric moisture. The spatially and temporally 'imposed' precipitation is an orographic precipitation gradient, but is not calculated through the orographic module in CASCADE. We created the module 'rainmaker_imposed' in Fortran90, which allows for user input of spatially and temporally variate precipitation grids in order to leverage published paleoclimate simulations. The input requires a plain text file denoting the ASCII precipitation grid and the corresponding age at which to apply this grid. The module then interpolates the input precipitation grid to the CASCADE TIN network. A user-designated time interval will initiate re-interpolation of the input precipitation grid, which will either apply the next precipitation grid if it is time for it, or will correct for the TIN network shifting the precipitation grid due to the prescribed kinematics.

We use published paleoclimate simulation data from the ECHAM5 global atmospheric general circulation model (Mutz et al., 2018), which leverages a statistical cluster analysis of climate over different orogens for several distinct time periods. As the ECHAM5 model is available only at a large resolution (~80 km x 80 km), and downscaling the climatic simulations to our field area would be an extremely time-intensive effort, we instead average the precipitation along our cross-section line by searching for the modelled nodes within 70 km (Figure 4.3, left). We extracted ECHAM5 climate data for modern, last glacial maximum (LGM), and Pliocene times (Figure 4.3, right top). At this time, mid- and late-Miocene datasets are not available, so we apply the Pliocene precipitation over >3 Ma. As climate models tend to overpredict rainfall, we compared the extracted modern precipitation to measured modern precipitation from the ~1 km x 1 km WorldClim2 database, which uses >10000 weather stations, MODIS, and SRTM data to spatially interpolate climate estimates for global land surfaces (Fick & Hijmans, 2017) to determine a percent change model inaccuracy . Then, we assumed that the model inaccuracies remain constant through time, which is not necessarily true, but is a best estimate, and applied this percent change to all precipitation grids we used (Figure 4.3, right bottom). This produces the initial LGM and Pliocene precipitation grids that we impose on our CASCADE models. As ECHAM5 is based on the modern Andes topography, and thus does not take into account the evolution and the eastward propagation of elevated topography inherent in convergent orogens, we adjusted the spatial locations in the direction of transport to match the location of the ECHAM5 adjusted orographic gradient with the mountain front at that time (Figure 4.4, Figure 4.11). We created an initial lateral translation amount based on Move generated topographies (Figure 4.4,Figure 4.11). Keeping the modern relationship between topography and orographic precipitation (Figure 4.1), where the precipitation gradient decrease and elevation increase cross at ~2 km elevation, we used Move-generated topography to identify when and how far the precipitation grids should shift. This is a first-order approximation of the location of topography generation, prior to any processes in CASCADE running.

Several modeling studies have evaluated the impact of topographic elevation on precipitation in the central Andes (Ehlers & Poulsen, 2009; Insel et al., 2010; Jeffery, Poulsen, & Ehlers, 2012; Poulsen, Ehlers, & Insel, 2010). These studies emphasize that modern precipitation values are reached when the Andes exceed 75 % of their modern elevation. Figure 4 highlights that at the start of our model time (11.4 Ma) structural uplift was sufficient to support modern elevations in the Eastern Cordillera (Buford Parks & McQuarrie, 2019).

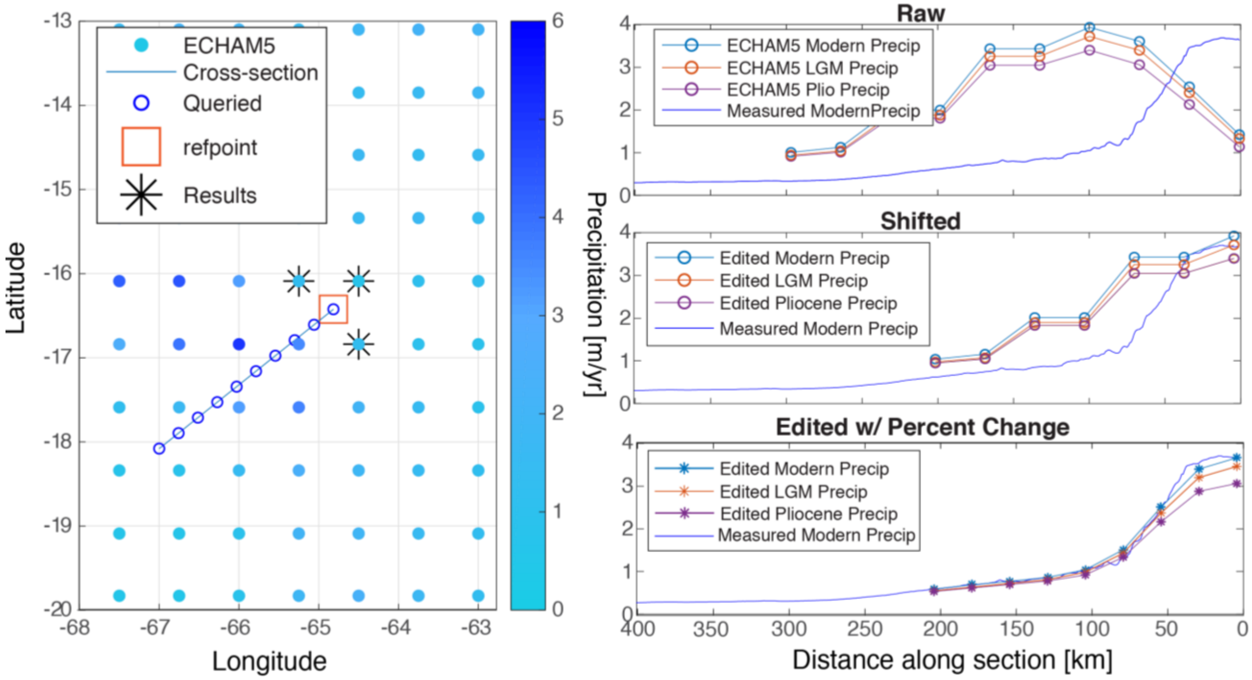


Figure 4.3. Interpolated and adjusted data from ECHAM5 global climate model (left) filled circles: raw ECHAM5 data points; open blue circles denote queried points along cross-section line (blue line); asterisk show averaged points for the reference point. (right, top) ECHAM5 raw data interpolated to cross-section line for each time, (right, middle) ECHAM5 data laterally shifted to align with measured modern precipitation, (right, bottom) ECHAM5 data modified with percent change.

4.3.3 Ksn Analysis

Ksn was extracted from both the DEM and CASCADE output in MATLAB using TopoToolBox (Schwanghart & Kuhn, 2010; Schwanghart & Scherler, 2014). Values for Ksn were extracted from fluvial channels with a minimum upstream drainage area of 4 km^2 and 10 km^2 to exclude hillslope processes with reference concavity of $\theta_{ref} = 0.5$ and $\theta_{ref} = 0.45$ for the natural landscape (e.g. DEM) and simulated landscape (e.g. CASCADE output), and both were averaged over 2 km stream channel segments. The simulated landscape was converted from the native triangulated irregular network (TIN) to a regularly gridded network with a spacing of 0.5 km, of which a width of 40 km and length of 150 km was analyzed to exclude boundary effects.

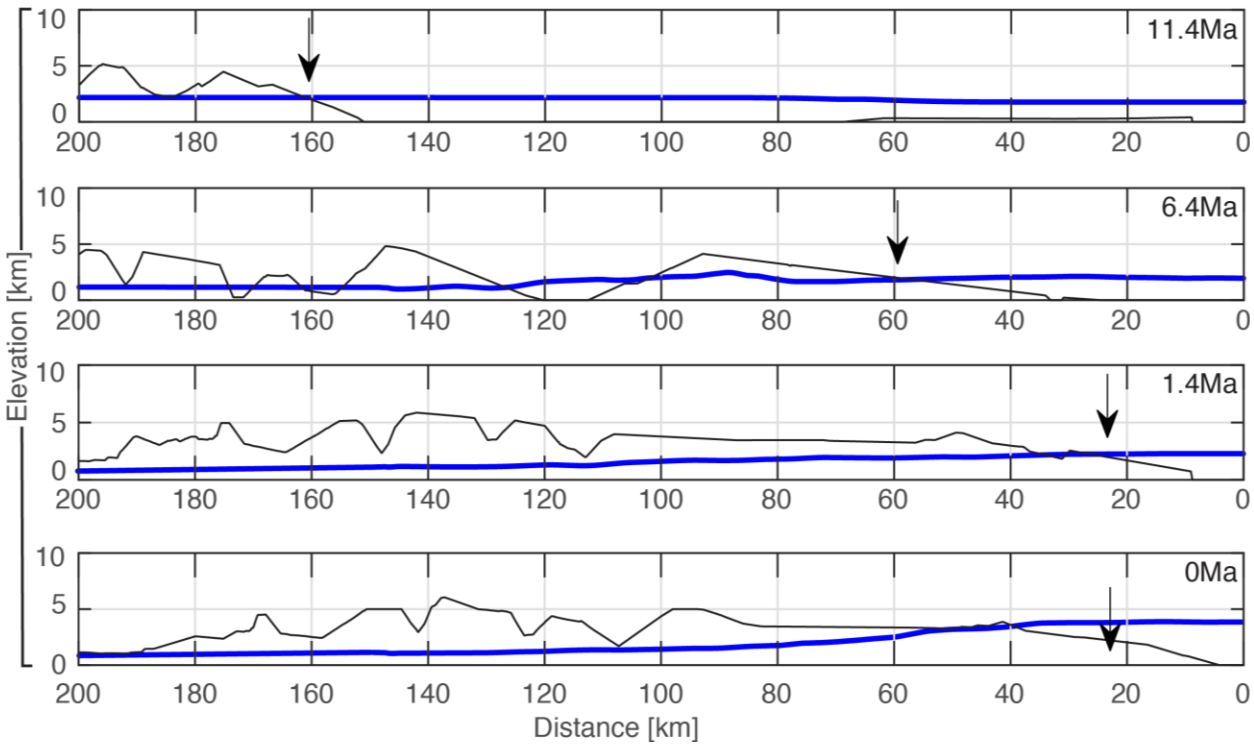


Figure 4.4. Move-generated topography (black) example timesteps with corresponding adjusted precipitation grid inputs (blue)

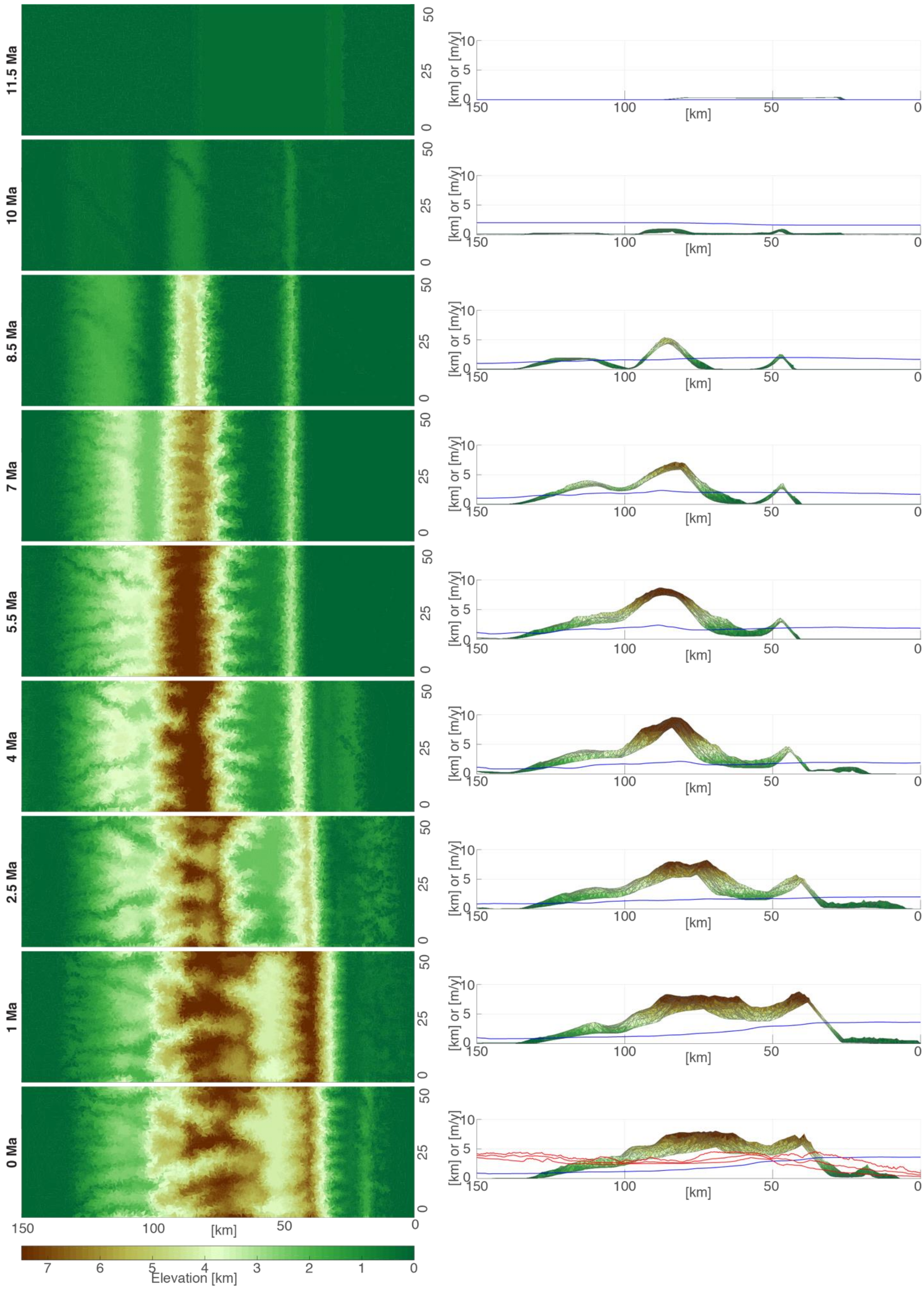
See Figure 4.11 for Move topographies for all deformation steps.

4.4 Results

4.4.1 Topographic Evolution

Our model starts at t = 11.5 Ma with no shortening or displacement to allow the initial landscape to equilibrate an initial fluvial network (Figure 4.5, top; Supplementary Video 2:

https://youtu.be/LXuABMIFub8). This model (Figure 4.5) uses our best fit velocity (8 mm/y), $K_F = 1.0 * 10^{-4}$, and our best-fit imposed precipitation model. From t = 11.4 Ma onwards, deformation initiates along a basement ramp (125-100 km) and stratigraphic ramp (85-70 km) system (Figure 4.2b) with a surface breaking fault at 40 km (Figure 4.5, t = 10 Ma). This creates initial topography across 40-55 and 70-100 km based on the uplift generated over these ramps, and initiates an eastward dipping drainage system off the flanks of the active uplift. As model time passes (compare t = 11.5, 10, 8.5 Ma), the uplifted topography generated over the basement ramp increases in elevation and widens in extent as it is laterally translated forward (see Figure 4.5, t = 9.0 Ma vs t = 5.5 Ma). High uplift rates (due to a large 12 km high ramp) increases stream power and facilitates a westward retreat of the drainage divide (from 7-4 Ma) offsetting lateral translation. Even with the increased river incison, high uplift combined with limited erosion over the region of focused uplift results in unrealistically high elevations (>6.0 km, see Figure 4.5 t = 5.5 Ma, 80-95 km). At t = 4.0Ma, the deformation propagates forward (Figure 4.2d), with structural uplift generated at the stationary basement ramp (125-100 km), a stratigraphic ramp that is half the original height at 80-85 km and along the flat across the top of the Cambrian (Figure 4.2d). The final ramp to the surface is at ~20 km (Figure 4.5, t = 4 Ma). The newly created ~ 60 km décollement and a more moderate ramp enhances lateral translation of topography (Figure 4.5, t=4 Ma to t=2.5 Ma). OOS motion provides uplift at 30-40 km from t ~1.5 Ma to t = 0 Ma (Figure 4.5, t = 1,0 Ma). The SA ramp located at 80-90 kmis the key driver of uplift and the generator of high topography for the majority of the model (e.g. ramp at 80-90 km). The high topography that is generated at this ramp, is then translated along the décollement towards the foreland. The preservation and lateral translation of topography is key to the creation of the broad, high elevation, low relief topography that to a first order shares similar features to modern topography, although is 2-3 km higher (Figure 4.5) (Supplementary Video 2).



11.5 Ma
10 Ma
8.5 Ma
7 Ma
5.5 Ma
4 Ma
2.5 Ma
1 Ma
0 Ma
[km]
[km] or [m/y]
Elevation [km]

Figure 4.5. Time series landscape formation using CASCADE model, Bolivian kinematics, and ECHAM5-adjusted precipitation

shortening occurs at 8 mm/y, $K_F = 1.0 * 10^{-4}$

(left) Map view landscape evolution (right) cross-section view evolution, , blue lines are modelled precipitation, red lines indicate modern-day measured 50 km swath topography

4.4.2 Global Precipitation

To compare the effects of different magnitudes of global precipitation that is spatially and temporally invariant on the evolution of topography, we performed simulations with wet and dry precipitation values (Figure 4.6) using an erodibility value of $K_F = 8.5 * 10^{-5}$ and the best fit velocity of 8 mm/y. The first precipitation value tested, of ~1.25 m/y, is a seeming balance between the modern ~4 m/y in the frontal portion of the system and ~0.35 m/y in the hinterland. This generates no excessively high topography (max elevation achieved in the model is ~6 km), but final elevations in the region between 45-75 km are 0.25-1 km in elevation, significantly below modern elevations of 3-4 km (Figure 4.6, bottom). This low elevation basin caught between two regions of active uplift is a result of too much erosion and thus limited preservation of uplifted, legacy topography. Topography generated over the basement ramp at ~80-90 km remains at 80 km (Figure 4.6, bottom). The high precipitation (1.25 m/y) model highlights the need for an elevated and translated legacy landscape to replicate modern topography between 45-75 km where no mechanism for active uplift is present. To evaluate the difference that precipitation magnitude imparts on the preservation of legacy topography, we tested a lower global precipitation value (0.5 m/y). However, as shown in Figure 4.5 (top), there is not enough erosion, resulting in an unrealistically high >8 km peak at ~35-50 and a 7-8 km plateau between 70-100 km. Although not as low, or as wide as produced with higher precipitation, the region between 45-65 km is still 3-5 km lower than the adjacent mountains. Uplifted and translated topography extends to 70 km compared to 80 km in the landscape produced with higher precipitation.

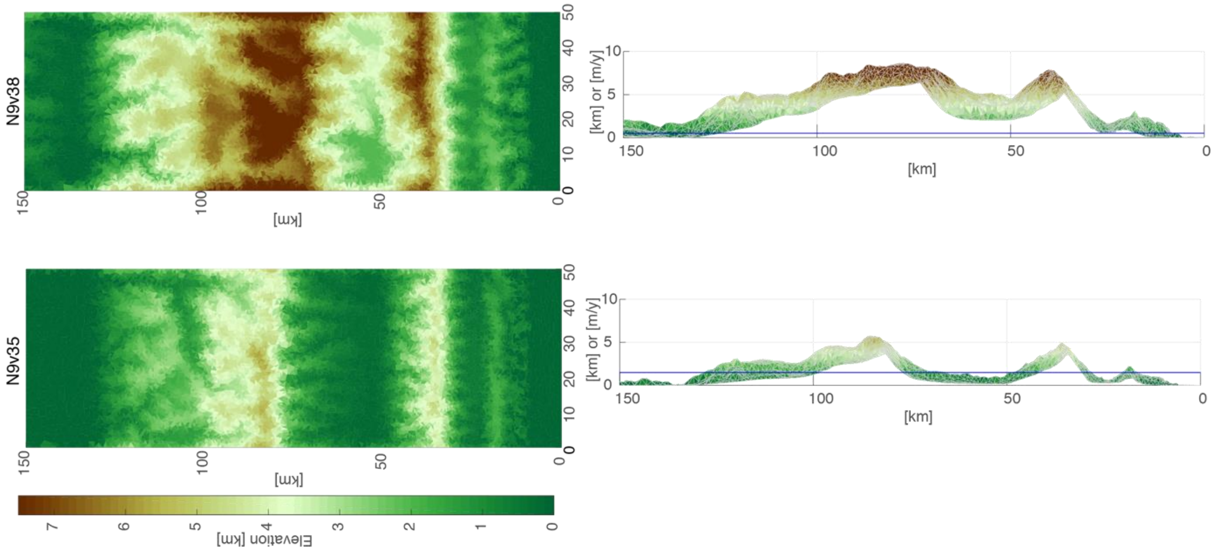


Figure 4.6. Effect of wetter climate in global precipitation models

(N9v35, bottom, wetter ~1.25m/y, versus N9v38, top, drier ~0.5 m/y)

4.4.3 Rock Erodibility

We evaluated the topographic response to changes in modelled rock erodibility. In Figure 4.7, we hold the global precipitation (~1.25 m/y) and velocity (8 mm/y) constant, and change the rock erodibility throughout the entire CASCADE model run. We compared of ¼ of an order of magnitude higher rock erodibility to the values used in section 4.5.2 (e.g. , $K_f = 8.5 * 10^{-5}$ -weaker rock strength, versus , $K_f = 2 * 10^{-4}$-stronger rock strength, for top and bottom of Figure

4.7, respectively). As highlighted in section 4.5.2, to the effects of too much precipitation, models with weaker rock strength had trouble preserving elevated topography at 45-70 km (Figure 4.7, top). High erodibility also produces lower total elevations over the ramp by ~3 km (at ~40 and ~80 km, as well as facilitates quicker removal of material from the system in front of ramps. The main effect of lowering rock erodibility can be seen in the region of the décollement (between 40-80 km) where decreasing erodibility prevents excess erosion and preserves translated topography, generating a plateau. In detail, the lower rock erodibility allowed for the translation of the drainage divide from 90 km at 4 Ma, to 55 km at 0 Ma (Figure 4.7, bottom). Low rock erodibility is critical for the preservation of these laterally advected landscapes.

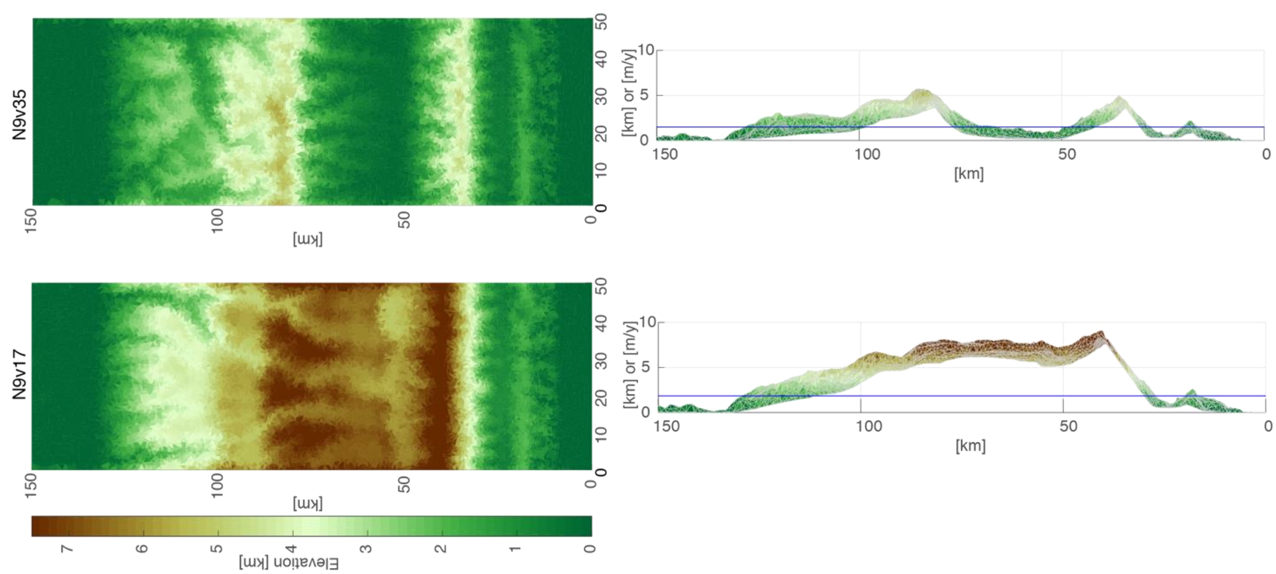


Figure 4.7. Effects of rock strength in constant precipitation (~1.25 m/y) models (N9v17,-bottom, weaker Kf=8.5e-5, versus N9v35, top, stronger Kf = 2.0e-4)

4.4.4 Imposed Precipitation

Our best-fit imposed precipitation model (Figure 4.8, Figure 4.9; Supplementary Video 2) is the result of >300 model simulations where in rock strength, climate parameters, velocity, and magnitude and location of precipitation were varied (within the bounds permitted by the climate model). This model relies on our adjusted ECHAM5-based Pliocene precipitation over the largest duration of model time (~11.5 Ma-2.4 Ma), ECHAM5-based last glacial maximum precipitation between 2.4-2.0 Ma, and modern precipitation between 2.0-0 Ma. This model retains some critical components of the low erodibility model from section 4.5.3, notably a migrating drainage divide from 4-1 Ma before increased precipitation relaxes it back towards the west (Supplementary Video 2). The translated topography is more moderate, than Figure 4.7 (bottom) but still too high from 90-60 km. Modern elevations are obtained between 60 and 45 km. Elevations above the OOS faults (35-45 km) are high and the elevations in front of the OOS faults (at ~25 km) are slightly depressed. Despite some of these elevation differences, the overall shape of the topography follows that of the modern quite well, with lateral drainages, low relief in the hinterland, and a drainage divide that sits relatively far forward (~40 km).

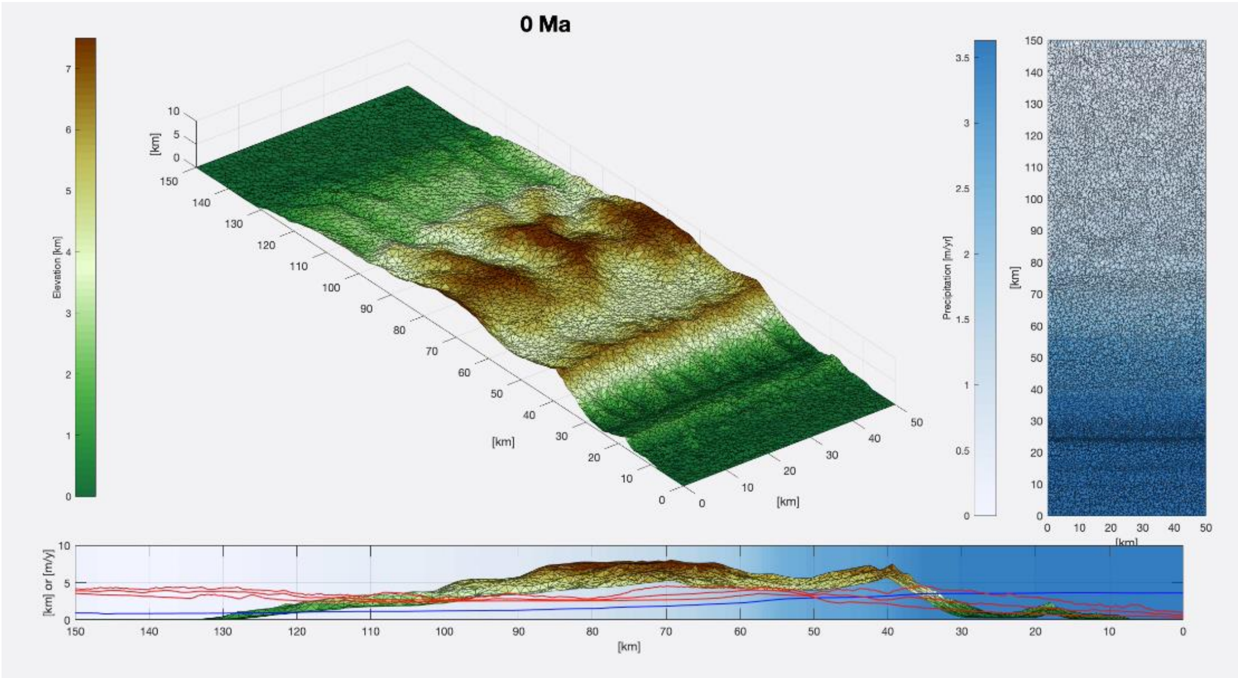


Figure 4.8. Best fit imposed precipitation model (N9v146)

(Top left) 3D view of landscape evolution. (Right) map view of precipitation pattern. (Bottom) Cross-section view of topography (greens through browns), precipitation (variant blue background) and mean precipitation (solid blue line) with measured modern 50 km swath topography showing minimum, mean, and maximum values (red lines).

Despite our best efforts to find a solution solely altering the magnitude and location of precipitation with a spatially and temporally invariant rock strength, elevations gained over the ramp at ~90 km, were still too high for a large portion of the model time (~7-4 Ma, >7 km) (Supplementary Video 2) . While elevations of 5-6 km over the ramp are necessary for the preservation of legacy topography between ~40-80 km, elevations in excess of 7-8 km over a 20 km distance may not be realistic for earth topographies. CASCADE does not currently allow users to provide a spatially variant rock strength (e.g. for foreland basin sediments versus strongly lithified Paleozoic rocks). To replicate the higher erodibility of rapidly uplifted weak foreland basin rocks (Figure 4.2a-c), we increased rock erodibility for a time subset (~6-4 Ma) to both increase the magnitude of erosion and decrease maximum elevations generated in this time period (Figure 4.9, a,c). This reduced maximum elevations achieved by ~0.5-1 km, and enhanced erosion of the elevated terrain (compare with and without erodibility changes Figure 4.9, a vs b, c vs d). The increase in rock erodibility from 6-4 Ma produced a final topography that is closer to the modern topography (Figure 4.9g versus h).

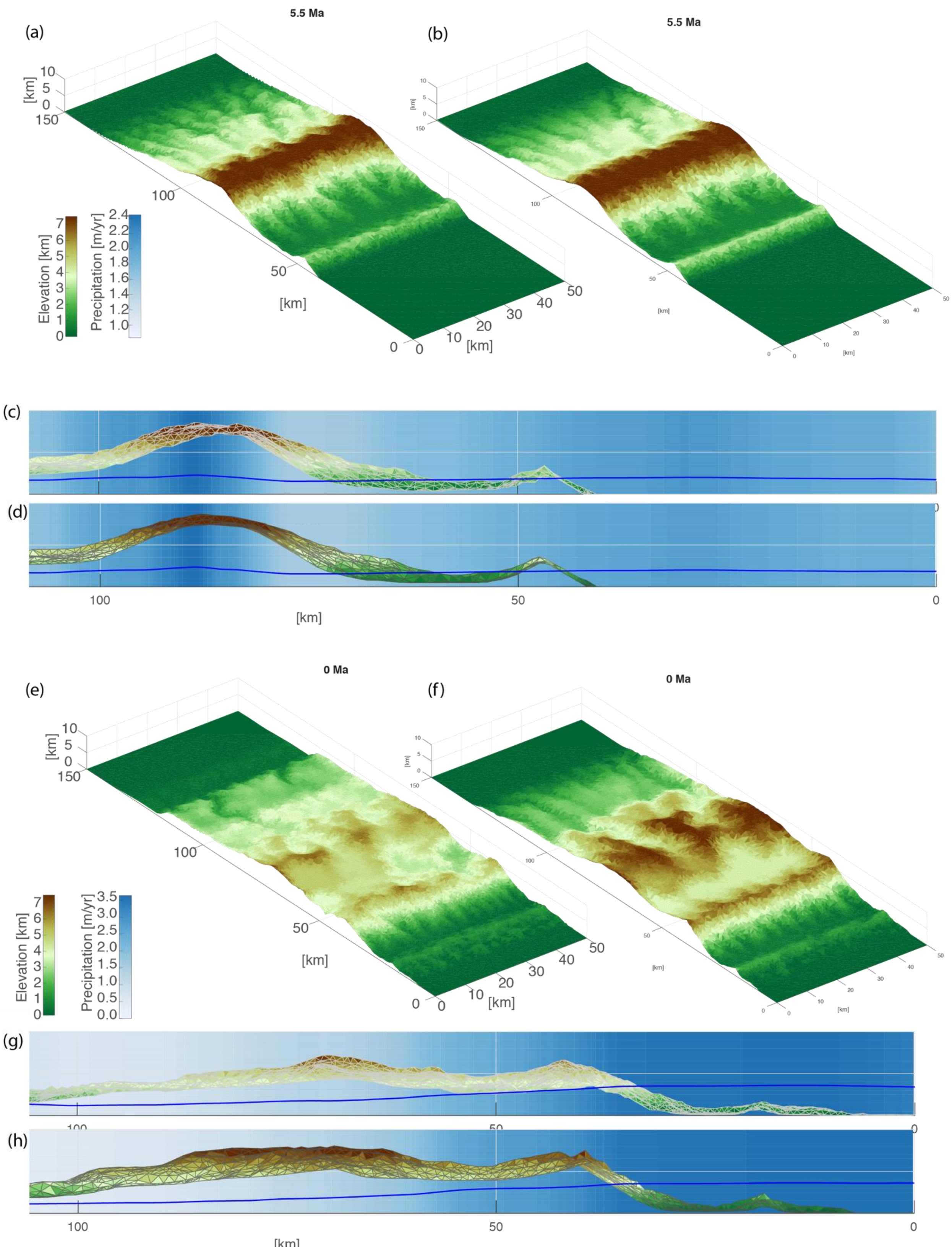


Figure 4.9. Comparison of models with and without increase in rock erodibility between 6-4 Ma

(a-b,e-f) 3D View and (c-d,g-h) Cross-section view. (a-d) 5.5 Ma (a,c) with increased Kf = 1.25 6-4 Ma (b,d) Xkf=1.0e-4 the entire model. (e-h) 0 Ma (e,g) with increased Kf = 1.25 6-4 Ma (f,h) Xkf=1.0e-4 the entire model.

4.4.5 Topographic Analysis

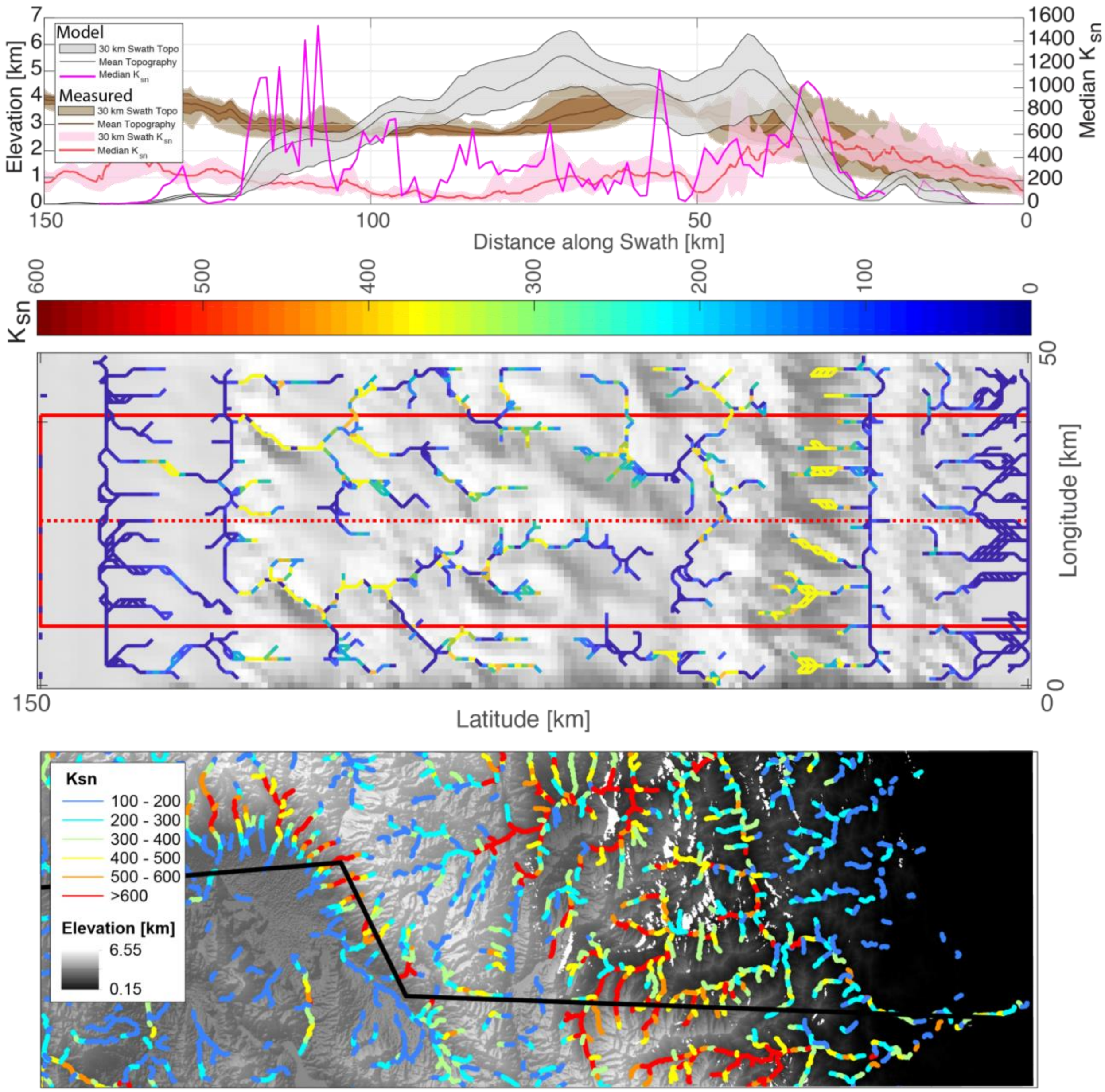


Figure 4.10. Ksn analysis

(top) Cross- Section view of measured and modelled topography and ksn. (middle) Simulated topography from CASCADE output of best-fit model with extracted Ksn values overlain on a gray hillshade map. Solid red line shows 40 km wide box used for Ksn swath in top, with dashed line as the projection line. (bottom) modern DEM hillshade map overlain with extracted Ksn values.

The most pronounced difference between the measured and modeled topography and resulting Ksn values is the sharp decline in topography and Ksn values between 40 and 25 km (Figure 4.10). Translation of topography and elevated Ksn values is a function of a geometric transition from a ramp to a decollement combined with sufficient horizontal translation. This translation of topography is seen in figure 4.5 (4-1 Ma) and in supplementary videos while translation of elevated Ksn was shown by Eizenhöfer et al. (2019). The lack of lateral translation of topography and elevated Ksn values between ~35 km and ~20 km is a function of the age of OOS motion near ~25 km, which limits translation of the uplifted terrain to the east along the décollement. A revised kinematic order that allows for ~10 km of SA shortening on the frontal most faults (Figure 4.2) following most of the motion on the OOS fault may facilitate this translation of both topography and Ksn values. The tapered decrease in model-extracted Ksn values behind this ramp (between ~35 and 50 km) agrees with the modern Ksn values. The background values in the hinterland past ~100 km should be ignored, as they are within the model domain that does not have accurate kinematic input. The elevation for this portion of the model is dependent on the lateral (150 km) boundary condition and has been laterally translated towards the foreland. As CASCADE reads in only an initial topography, any portion at the back of the model that has been translated forward throughout the whole model does not accurately represent the Move generated uplift experienced outside the model domain.

4.5 Discussion

4.5.1 Sequential Evolution of Topography

In the sequential evolution series (Figure 4.5, Supplementary Video 2) we can see that topography is generated along the ramps and translated along the flats (Figure 4.5, t = 4 through t = 1 Ma). This evolution from the physics-based hillslope and fluvial-transport erosion calculations in CASCADE has both similarities, as well as pronounced differences, from the more simplified mode of predicting topography in the kinematic models. It is well known in the geologic community, that, based on empirical measurements, topography in fold-and-thrust belts around the world increases at an angle typically between 1-3°. In southern Bolivian Andes, the angle is ~1°, while in northern Bolivia it is ~2° (Horton, 1999; Masek, Isacks, Gubbels, & Fielding, 1994), and in central Bolivia, the elevation in the front increases fairly steeply at ~2.8-3° (Figure 4.1). In our flexural-kinematic modelling, we replicate this gradual increase by requiring erosion at a set angle (Figure 4.2c). This means that uplifted topography (over a ramp, Figure 4.2c, ~80-100 km) is where erosion is focused, and then this eroded topographic surface is automatically shifted in the direction of transport during the next increment of shortening. This eastward migration of Move topography can also be seen in Figure 4.4. Although translation also occurs in CASCADE, erosion over the ramp is less, and there is more erosion of the translated topography. This can be seen by comparing Figure 4.2, t = 6.5 Ma and Figure 4.5, t = 7 and t = 5.5. Ma. The peak topography has been translated farther east in the CASADE model but elevations decline much more rapidly so that by 60 km, the Move model would predict elevations of ~ 2 km, while the Cascade model has elevation that barely reach 1 km. In our Move model, the majority of large translation occurs post 4 Ma when the deformation front has propagated forward and there is a longer décollement

(Figure 4.2d-f, 25-75 km). This is when the majority of elevated topography is translated in the CASCADE model (Figure 4.5 t = 4 Ma through t = 0 Ma), particularly when rock erodibility is lowered. In our Move model, there is a small basin that is created by the forward fault propagation (Figure 4.2), but as shortening continues, topography translates into this region. Much of the preserved legacy topography in the Cascade model, particularly between 60-40 km, is a function of a decrease in ramp magnitude, combined with a lengthening of the décollement, (e.g. longer décollement) at ~ 4 Ma (Figure 4.2) .

Our best fit model (with imposed precipitation) also has rock strength lowered between 6-4 Ma to mimic spatially variant rock strength in rocks exposed at the surface. At the start of SA deformation, there is a significant thickness (4-6 km) of foreland basin sediments deposited on top of the Paleozoic strata (Figure 4.2a,b) that comprises the remainder of the section. As CASCADE produces less erosion over the ramp compared to our Move models (which do not allow elevation above 5-6 km), these notably weaker rocks are elevated and not eroded until they are translated into river dominated systems to the east (Supplementary Video 2). Enhanced erosion of the uplifted topography generated over the ramp is expected in the 4-6 km of foreland basin sediments, which have a lower rock strength (and thus higher K_f) compared to that of the firmly compacted Paleozoic rocks. CASCADE provides cumulative erosion amounts for each node within the TIN, and for our best fit model with constant K_f (Figure 4.5, Figure 4.8, Figure 4.9), only an average of ~310 meters of total erosion has occurred by 5.5 Ma at ~100 km (Figure 4.5), and an average of ~6.5 km of total erosion by 4 Ma. Increasing the rock erodibility from $\mathrm{K}_f = 1.0 * 10^{-4}$ to $\mathrm{K}_f = 1.25 * 10^{-4}$ (between 6 and 4 Ma) increased the cumulative erosion at 5.5 Ma to ~1 km and at 4 Ma to ~7.5 km. This increased rock erodibility functions as a mechanism to quickly erode away the weaker foreland basin strata over the ramp.

4.5.2 Importance of Legacy and Translated Landscapes

The majority of our tested parameters highlight the delicate balance between the preservation and translation of uplift legacy landscapes necessary to reproduce the morphology of the central Andes. Models where erosion was limited lead to the creation and translation of highly elevated topography and scenarios where the final ~9-12 km high topography is a composite of multiple periods of uplift compounded over successive ramps. Conversely, high erodibility and/or high precipitation lead to a large basin between regions of active uplift in contrast to modern high elevations in this location. Translation of legacy topography is a result of the interplay of horizontal advection and erodibility (McQuarrie et al., 2019). With high erodibility and limited advection, erosion can keep pace with uplift limiting elevated topography to just regions above the structure driving uplift (Figure 4.6 bottom, high erosion). For translated topography, horizontal advection must be more efficient (faster) than erosion. The preservation of translated topography is particularly of concern here in the Andes because the convergence rate is slow (~8 mm/y). In simplified models of uplift over a ramp, legacy landscape that included translation of the drainage divide and creation of a plateau emerged at rates =>20 mm/y) (Eizenhöfer et al., 2019). Slower rates (10 mm/yr) lead to translated interfluves but a fixed drainage divide. Eizenhöfer et al. (2019) hypothesized that the formation of plateaus, which are essentially translated high topography, occurs when horizontal translation is more efficient than the system working to erode it. Our model runs that best replicate the plateau geometry of the central Bolivian Andes requires that the drainage divide migrates in the direction of transport creating an elevated legacy landscape behind it (90- ~60 km, Figure 4.5-9). The elevated region between ~60 and 45 km are elevated translated interfluves caught between two zones of active uplift. Because horizontal advection is slow, (8 mm/yr) rock erodibility must be low to create the legacy landscape and translated drainage divide.

This low erodibility is crucial between 4 Ma and present, but the rock erodibility necessary to produce the translated topography from 4 Ma to present generates excessive elevations between 4-6 Ma, suggesting that there is a marked change in rock strength at the surface at ~4 Ma.

4.5.3 Low Relief Surfaces

There is much debate about the formation of elevated low relief surfaces, as several hypothesized mechanisms abound. Firstly, they could form at low elevations and then undergo subsequent uplift (e.g. Clark et al., 2005, 2006; Kirby & Whipple, 2012; Miller et al., 2012; Schoenbohm et al., 2004; Spotila et al., 1998). Second, they could have formed at high elevations due to river network disruption (e.g. Yang et al., 2015), or third, they could be an in-situ production due to sediment filling related to back-tiliting in the hinterland (Adams, Whipple, Hodges, & Heimsath, 2016). The Bolivian low relief surfaces (e.g. the San Juan del Oro surface in the EC of southern Bolivia (Barke & Lamb, 2006; Kennan et al., 1997) and the Cangalli Fm. in the SA (Fornari, Herail, Viscara, Laubacher, & Argollo, 1987; Mosolf, Horton, Heizler, & Matos, 2011) in northern Bolivia are often interpreted to have formed as low elevation surfaces, graded towards low elevations in the foreland (east in an Andean perspective), that have subsequently been uplift by varying amounts to their current elevations. The problem with this interpretation is that it does not take into account the lateral translation of these surfaces (60-70 km) that had to occur over the last 9-10 Ma since their formation (Gubbels et al., 1993; Kennan et al., 1997; Mosolf et al., 2011). Animations of translated topography (Supplementary Videos 2-6, Eizenhöfer et al., 2019) highlight both a hinterland orogen and the tens of kilometers of lateral translation these low elevation surfaces would undergo.

In contrast to the San Juan del Oro surface and contiguous surfaces in the Eastern Cordillera, a smaller suite of the eastern-most low relief surfaces (Figure 4.1) and the related Cangalli Fm. are associated with pronounced hanging valleys (Whipple and Gaspirini, 2014). These low relief surfaces and hanging valleys are also adjacent to or directly west of mapped elevated Ksn values. In central Bolivia where our modelling is focused, the co-location of elevated Ksn values and young cooling ages suggested the presence of an OOS fault (Whipple & Gasparini, 2014), that when thermally modelled, produce the best match between measured and modelled cooling ages (Buford Parks & McQuarrie, 2019). The eastern-most central Bolivia surfaces associated with hanging valleys may be the best example of in-situ uplift on an active fault, however, the elevated surfaces associated with the hanging valleys still must have been translated significantly from the west.

4.5.4 Andean Uplift

Whipple and Gasparini (2014) hypothesized that the elevated low relief surfaces (e.g. San Juan del Oro surface) experienced significant (>2 km) surface uplift post-10 Ma along the Eastern Cordillera. Hoke and Garzione (2008) connected the EC surfaces to the AP and argued for 1.5-2 km of uplift if the entire plateau post 10 Ma. While many authors conducting paleoelevation studies adjacent to our study area have argued for various elevation histories of the AP either connected to (Garzione et al., 2014, 2006; Hoke & Garzione, 2008) or disconnected from (Eichelberger et al., 2015) the AP, the presence of active drivers of Andean uplift in the east (along the Eastern Cordillera, e.g. Figure 4.8) acts as a barrier between hinterland elevations and associated elevation histories and the foreland.

4.6 Conclusions

Landscape evolution models are vital to linking modern geomorphology and the structural evolution in the region. Topography is initially generated over ramp structures and translated along flats. The balance between rock uplift, controlled by fault geometry and shortening rate, and erosion, controlled by precipitation and rock erodibility, require a delicate equilibrium to preserve elevated topography, e.g. legacy landscapes. These legacy landscapes are necessary to reproduce the morphology of the region. Low relief surfaces initially form in equilibrium with the hinterland and are subsequently uplifted over active Subandean basement structures to form elevated low-relief surface. Translated legacy topography, initially uplifted over ramps, are preserved through a balance between rock erodibility and precipitation. Using a detailed balanced cross-section integrated with a physics-based surface process model, we evaluate the topographic evolution produced by the kinematics proscribed in our section, and the impact of legacy topography. In central Bolivia, the balance between preservation of uplifted and translated legacy topography is difficult to achieve due to both low (8 mm/yr) shortening rates, documented changes in rock strength (resistant lower Paleozoic rocks compared to weak foreland basin strata), and the variability in the magnitude and location of precipitation. Rock erodibility was critical for promoting, or inhibiting erosion, and provides a primary control on translated topography. Propagation of the fault system at ~4Ma was critical to replicating modern plateau morphology by lengthening the décollement, promoting eastward propagation of the drainage divide. In models with low rock strength this migration is limited to ~5-20 km (depending on precipitation), while for models with high rock strength the elevated topography can translate up to ~35 km.

4.7 Acknowledgments

Supporting data are included in the supporting information file; supplementary videos are available on the Tectonics YouTube page here: https://www.youtube.com/playlist?list=PLttn6LSpDKTGGfgj0scoC_sH7m9P0IccG. Any additional data or information are available through contacting VMB Parks (email: vmbparks@gmail.com). This work was supported by NASA headquarters under the NASA Earth and Space Science Fellowship program - Grant 80NSSC17K0388. We acknowledge and thank Midland Valley and Petex for support and use of the program Move. This manuscript benefitted from thoughtful discussions over the years with Paul Eizenhöfer, Mary Braza, Chloë Glover, and Brad Parks. T.A. Ehlers, Paul Eizenhöfer, and Willi Kappler (University of Tübingen) are thanked for support with conducting modelling simulations and providing the modified version of CASCADE that can be coupled to 2D Move restoration files, as well as providing key support for the creation of the new CASCADE module.

4.8 Supplementary Information

Included in the supplementary information is:

Supplementary Figure 1.

Supplementary Video 1: Subandean kinematics modelled in Move: https://youtu.be/K_uIfU917RY

Supplementary Video 2: Best fit model with constant rock strength time series video (N9v146): https://youtu.be/LXuABMIFub8

Supplementary Video 3: N9v35 constant precipitation – wetter (~1.25 m/y); rock strength stronger K_F=2.0e-4: https://youtu.be/p4RSgA1WkXw

Supplementary Video 4: N9v38 constant precipitation – drier (~0.5 m/y); rock strength stronger K_F=2.0e-4: https://youtu.be/UZqd4ixtgH0

Supplementary Video 5: N9v17 constant precipitation – wetter (~1.25 m/y); rock strength weaker K_F=8.5e-5: https://youtu.be/brqos2SQRvo

Supplementary Video 6: Overall best fit landscape evolution (N9v166) model: https://youtu.be/cETOkqNnc7w

Supplementary Video 7: Best fit Pecube-kinematics model: https://youtu.be/PgtwTu4HgFQ

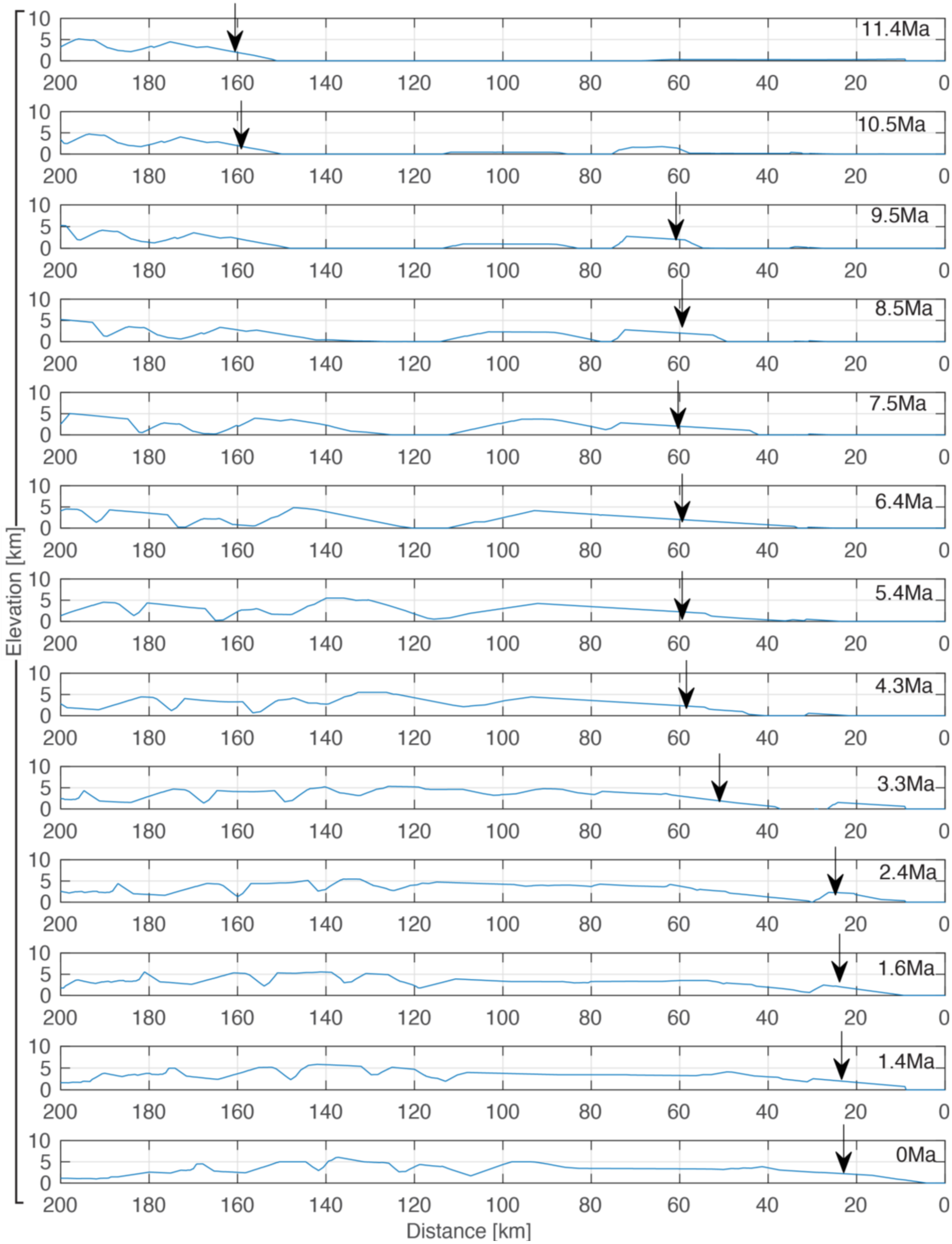


Figure 4.11.All Move generated topographies for time steps used in CASCADE modelling

Black arrow indicates location where Move topography crosses 2 km in elevation.

4.9 References

Adams, B. A., Whipple, K. X., Hodges, K. V., & Heimsath, A. M. (2016). In situ development of high-elevation, low-relief landscapes via duplex deformation in the Eastern Himalayan hinterland, Bhutan. Journal of Geophysical Research: Earth Surface, 121, 294–319. https://doi.org/doi:10.1002/2015JF003508

Anderson, R. B., Long, S. P., Horton, B. K., Calle, A. Z., & Ramirez, V. (2017). Shortening and structural architecture of the Andean fold thrust belt of southern Bolivia (21°S): Implications for kinematic development and crustal thickening of the central Andes. Geosphere, 13(2), 538–558. https://doi.org/10.1130/GES01433.1

Baby, P., Moretti, I., Guillier, B., Limachi, R., Mendez, E., Oller, J., & Specht, M. (1995). Petroleum System of the Northern and Central Bolivian sub-Andean Zone. In A. J. Tankard, R. S. S., & H. J. Welsink (Eds.), Petroleum basins of South America: AAPG Memoir 62 (pp. 445–458). Retrieved from http://search.datapages.com/data/specpubs/memoir62/22baby/0445.htm

Barke, R., & Lamb, S. (2006). Late Cenozoic uplift of the Eastern Cordillera, Bolivian Andes. Earth and Planetary Science Letters, 249(3–4), 350–367. https://doi.org/10.1016/j.epsl.2006.07.012

Barnes, J. B., Ehlers, T. A., McQuarrie, N., O’Sullivan, P. B., & Pelletier, J. D. (2006). Eocene to recent variations in erosion across the central Andean fold-thrust belt, northern Bolivia: Implications for plateau evolution. Earth and Planetary Science Letters, 248(1–2), 103–118. https://doi.org/10.1016/j.epsl.2006.05.018

Barnes, J. B., Ehlers, T. A., Insel, N., McQuarrie, N., & Poulsen, C. J. (2012). Linking orography, climate, and exhumation across the central Andes. Geology, 40(12), 1135–1138. https://doi.org/10.1130/G33229.1

Beaumont, C., Fullsack, P., & Hamilton, J. (1992). Erosional control of active compressional orogens. In Thrust Tectonics (pp. 1–18). Springer, Dordrecht.

Bechtold, P., Chaboureau, J. P., Beljaars, A., Betts, A. K., Köhler, M., Miller, M., & Redelsperger, J. L. (2004). The simulation of the diurnal cycle of convective precipitation over land in a global model. Quarterly Journal of the Royal Meteorological Society, 130 C(604), 3119–3137. https://doi.org/10.1256/qj.03.103

Benjamin, T., Johnson, N. M., & Naeser, C. W. (1987). Recent rapid uplift in the Bolivian Andes: Evidence from fission-track dating. Geology, 15(July), 680–683. https://doi.org/10.1130/0091-7613(1987)15<680:RRUITB>2.0.CO;2

Bookhagen, B., & Burbank, D. W. (2006). Topography, relief, and TRMM-derived rainfall variations along the Himalaya. Geophysical Research Letters, 33(8), 1–5. https://doi.org/10.1029/2006GL026037

Bookhagen, B., & Strecker, M. R. (2008). Orographic barriers, high-resolution TRMM rainfall, and relief variations along the eastern Andes. Geophysical Research Letters, 35(6), 1–6. https://doi.org/10.1029/2007GL032011

Braun, J., & Sambridge, M. (1997). Modelling landscape evolution on geological time scales: A new method based on irregular spatial discretization. Basin Research, 9(1), 27–52. https://doi.org/10.1046/j.1365-2117.1997.00030.x

Braun, J., Zwartz, D., & Tomkin, J. H. (1999). A new surface-processes model combining glacial and fluvial erosion. Annals of Glaciology, 28, 282–290.

Buford Parks, V. M., & McQuarrie, N. (2019). Kinematic, flexural, and thermal modelling in the Central Andes: Unravelling age and signal of deformation, exhumation, and uplift. Tectonophysics, 766(June), 302–325. https://doi.org/10.1016/j.tecto.2019.06.008

Carvalho, L. M. V., Jones, C., & Liebmann, B. (2002). Extreme precipitation events in southeastern South America and large-scale convective patterns in the South Atlantic covergence zone. Journal of Climate, 15(17), 2377–2394. https://doi.org/10.1175/1520-0442(2002)015<2377:EPEISS>2.0.CO;2

Chase, C. G. (1992). Fluvial landsculpting and the fractal dimension of topography. Geomorphology, 5(1–2), 39–57.

Clark, M. K., House, M. A., Royden, L. H., Whipple, K. X., Burchfiel, B. C., Zhang, X., & Tang, W. (2005). Late Cenozoic Uplift of Southeastern Tibet. Geology, 33(6), 525–528. https://doi.org/https://doi.org/10.1130/G21265.1

Clark, M. K., Royden, L. H., Whipple, K. X., Burchfiel, B. C., Zhang, X., & Tang, W. (2006). Use of a regional, relict landscape to measure vertical deformation of the eastern Tibetan Plateau. Journal of Geophysical Research: Earth Surface, 111(3), 1–23. https://doi.org/10.1029/2005JF000294

DiBiase, R. A., & Whipple, K. X. (2011). The influence of erosion thresholds and runoff variability on the relationships among topography, climate, and erosion rate. *Journal of Geophysical Research: Earth Surface*, *116*(4), 1–17. https://doi.org/10.1029/2011JF002095

Ehlers, T. A., & Poulsen, C. J. (2009). Influence of Andean uplift on climate and paleoaltimetry estimates. Earth and Planetary Science Letters, 281(3–4), 238–248. https://doi.org/10.1016/j.epsl.2009.02.026

Eichelberger, N. (2014). Structural Development of the Bolivian Orocline and Implications for Evolution of the Altiplano Plateau. Princeton.

Eichelberger, N., McQuarrie, N., Ryan, J., Karimi, B., Beck, S. L., & Zandt, G. (2015). Evolution of crustal thickening in the central Andes, Bolivia. Earth and Planetary Science Letters, 426, 191–203. https://doi.org/10.1016/j.epsl.2015.06.035

Eizenhöfer, P. R., McQuarrie, N., Shelef, E., & Ehlers, T. A. (2019). Landscape Response to Lateral Advection in Convergent Orogens Over Geologic Time Scales. Journal of Geophysical Research: Earth Surface, 124, 2056–2078. https://doi.org/10.1029/2019JF005100

Fick, S. E., & Hijmans, R. J. (2017). WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas. International Journal of Climatology, 37(12), 4302–4315. https://doi.org/10.1002/joc.5086

Fornari, M., Herail, G., Viscara, G., Laubacher, G., & Argollo, J. (1987). Sédimentation et structure du bassin de Ti-puani-Mapiri: un témoin de l'évolution du front amazonien des Andes du nord de la Bolivie. Comptes Rendus de l'Académie Des Sciences. Série 2, Mécanique, Physique, Chimie, Sciences de l'univers, Sciences de La Terre, 305(16), 1303–1308. Retrieved from https://www.researchgate.net/publication/32983766_Sedimentation_et_structure_de_bassin_de_Tipuani-Mapiri_un_temoin_de_l'evolution_du_front_amazonien_des_Andes_du_nord_de_la_Bolivie

Garzione, C. N., Molnar, P., Libarkin, J. C., & MacFadden, B. J. (2006). Rapid late Miocene rise of the Bolivian Altiplano: Evidence for removal of mantle lithosphere. Earth and Planetary Science Letters, 241(3–4), 543–556. https://doi.org/10.1016/j.epsl.2005.11.026

Garzione, C. N., Auerbach, D. J., Jin-Sook Smith, J., Rosario, J. J., Passey, B. H., Jordan, T. E., & Eiler, J. M. (2014). Clumped isotope evidence for diachronous surface cooling of the Altiplano and pulsed surface uplift of the Central Andes. Earth and Planetary Science Letters, 393, 173–181. https://doi.org/10.1016/j.epsl.2014.02.029

Gasparini, N. M., & Whipple, K. X. (2014). Diagnosing climatic and tectonic controls on topography: Eastern flank of the northern Bolivian Andes. Lithosphere, 6(May), 230–250. https://doi.org/10.1130/L322.1

Gilmore, M. E., Mcquarrie, N., Eizenhöfer, P. R., & Ehlers, T. A. (2018). Testing the effects of topography , geometry and kinematics on modeled thermochronometer cooling ages in the eastern Bhutan Himalaya. Solid Earth, 9(May), 599–627. https://doi.org/https://doi.org/10.5194/se-9-599-2018

Gubbels, T. L., Isacks, B. L., & Farrar, E. (1993). High-level surfaces, plaeau uplift, and foreland development, Bolivian central Andes. Geology, 21(August), 695–698. https://doi.org/10.1130/0091-7613(1993)021<0695:HLSPUA>2.3.CO;2

Hoke, G. D., & Garzione, C. N. (2008). Paleosurfaces, paleoelevation, and the mechanisms for the late Miocene topographic development of the Altiplano plateau. Earth and Planetary Science Letters, 271(1–4), 192–201. https://doi.org/10.1016/j.epsl.2008.04.008

Horton, B. K. (1999). Erosional control on the geometry and kinematics of thrust belt development in the central Andes. Tectonics, 18(6), 1292–1304. https://doi.org/10.1029/1999TC900051

Insel, N., Poulsen, C. J., & Ehlers, T. A. (2010). Influence of the Andes Mountains on South American moisture transport, convection, and precipitation. Climate Dynamics, 35(7), 1477–1492. https://doi.org/10.1007/s00382-009-0637-1

Jeffery, M. L., Poulsen, C. J., & Ehlers, T. A. (2012). Impacts of Cenozoic global cooling, surface uplift, and an inland seaway on South American paleoclimate and precipitation δ18O. Bulletin of the Geological Society of America, 124(3–4), 335–351. https://doi.org/10.1130/B30480.1

Jeffery, M. L., Ehlers, T. A., Yanites, B. J., & Poulsen, C. J. (2013). Quantifying the role of paleoclimate and Andean Plateau uplift on river incision. Journal of Geophysical Research: Earth Surface, 118(2), 852–871. https://doi.org/10.1002/jgrf.20055

Kennan, L., Lamb, S. H., & Hoke, L. (1997). High-altitude palaeosurfaces in the Bolivian Andes: Evidence for late Cenozoic surface uplift. Geological Society Special Publication, 120(January 1997), 307–323. https://doi.org/10.1144/GSL.SP.1997.120.01.20

Kirby, E., & Whipple, K. X. (2012). Expression of active tectonics in erosional landscapes. Journal of Structural Geology, 44, 54–75. https://doi.org/10.1016/j.jsg.2012.07.009

Kley, J. (1996). Transition from basement-involved to thin-skinned thrusting in the Cordillera Oriental of southern Bolivia. Tectonics, 15(4), 763. https://doi.org/10.1029/95TC03868

Kley, J., Monaldi, C. R., & Salfity, J. A. (1999). Along-strike segmentation of the Andean foreland: Causes and consequences. Tectonophysics, 301(1–2), 75–94. https://doi.org/10.1016/S0040-1951(98)90223-2

Lease, R. O., & Ehlers, T. A. (2013). Incision into the eastern Andean Plateau during Pliocene cooling. Science, 341(6147), 774–776. https://doi.org/10.1126/science.1239132

Masek, J. G., Isacks, B. L., Gubbels, T. L., & Fielding, E. J. (1994). Erosion and tectonics at the margins of continental plateaus. Journal of Geophysical Research, 99, 941–956. https://doi.org/https://doi.org/10.1029/94JB00461

McQuarrie, N. (2002). The kinematic history of the central Andean fold-thrust belt, Bolivia: Implications for building a high plateau. Bulletin of the Geological Society of America, 114(8), 950–963. https://doi.org/10.1130/0016-7606(2002)114<0950:TKHOTC>2.0.CO;2

McQuarrie, N., & Ehlers, T. A. (2015). Influence of thrust belt geometry and shortening rate on thermochronometer cooling ages: Insights from thermokinematic and erosion modeling of the Bhutan Himalaya. Tectonics, 34(1), 1–25. https://doi.org/10.1002/2014TC003783

McQuarrie, N., & Ehlers, T. A. (2017). Techniques for understanding fold-thrust belt kinematics, and thermal evolution. In R. D. Law, J. R. Thigpen, A. J. Merschat, & H. H. Stowell (Eds.), Linkages and Feedbacks in Orogenic Systems: Geological Society of America Memoir 213 (Vol. 1213, pp. 1–30). https://doi.org/10.1130/2017.1213(02)

McQuarrie, N., Barnes, J. B., & Ehlers, T. A. (2008). Geometric, kinematic, and erosional history of the central Andean Plateau, Bolivia (15-17°S). Tectonics, 27(3). https://doi.org/10.1029/2006TC002054

McQuarrie, N., Ehlers, T. A., Barnes, J. B., & Meade, B. J. (2008). Temporal variation in climate and tectonic coupling in the central Andes. Geology, 36(12), 999–1002. https://doi.org/10.1130/G25124A.1

McQuarrie, N., Eizenhöfer, P. R., Long, S. P., Tobgay, T., Ehlers, T. A., Blythe, A., et al. (2019). The influence of foreland structures on hinterland cooling: evaluating the drivers of exhumation in the eastern Bhutan Himalaya. Tectonics, 38, 1–29. https://doi.org/10.1029/2018TC005340

Miller, S. R., Slingerland, R. L., & Kirby, E. (2007). Characteristics of steady state fluvial topography above fault-bend folds. Journal of Geophysical Research: Earth Surface, 112(4), 1–21. https://doi.org/10.1029/2007JF000772

Miller, S. R., Baldwin, S. L., & Fitzgerald, P. G. (2012). Transient fluvial incision and active surface uplift in the Woodlark rift of eastern Papua New Guinea. Lithosphere, 4(2), 131–149. https://doi.org/10.1130/L135.1

Mosolf, J. G., Horton, B. K., Heizler, M. T., & Matos, R. (2011). Unroofing the core of the central Andean fold-thrust belt during focused late Miocene exhumation: Evidence from the Tipuani-Mapiri wedge-top basin, Bolivia. Basin Research, 23(3), 346–360. https://doi.org/10.1111/j.1365-2117.2010.00491.x

Mutz, S. G., Ehlers, T. A., Werner, M., Lohmann, G., Stepanek, C., & Li, J. (2018). Estimates of late Cenozoic climate change relevant to Earth surface processes in tectonically active orogens. Earth Surface Dynamics, 6(2), 271–301. https://doi.org/10.5194/esurf-6-271-2018

Nesbitt, S. W., & Anders, A. M. (2009). Very high resolution precipitation climatologies from the Tropical Rainfall Measuring Mission precipitation radar. Geophysical Research Letters, 36(15), 1–5. https://doi.org/10.1029/2009GL038026

Norton, K., & Schlunegger, F. (2011). Migrating deformation in the Central Andes from enhanced orographic rainfall. Nature Communications, 2(1). https://doi.org/10.1038/ncomms1590

Olen, S. M., Ehlers, T. A., & Densmore, M. S. (2012). Limits to reconstructing paleotopography from thermochronometer data. Journal of Geophysical Research: Earth Surface, 117(1), 1–20. https://doi.org/10.1029/2011JF001985

Poulsen, C. J., Ehlers, T. A., & Insel, N. (2010). Onset of convective rainfall during gradual late miocene rise of the central andes. Science, 328(5977), 490–493. https://doi.org/10.1126/science.1185078

Rak, A. J., McQuarrie, N., & Ehlers, T. A. (2017). Kinematics, exhumation, and sedimentation of the north-central Andes (Bolivia): An integrated thermochronometer and thermokinematic modeling approach. Tectonics, 36, 1–31. https://doi.org/10.1002/2016TC004440

Safran, E. B., Blythe, A., & Dunne, T. (2006). Spatially variable exhumation rates in orogenic belts: an Andean example. The Journal of Geology, 114, 665–681. https://doi.org/10.1086/507613

Salzmann, U., Williams, M., Haywood, A. M., Johnson, A. L. A., Kender, S., & Zalasiewicz, J. (2011). Climate and environment of a Pliocene warm world. Palaeogeography, Palaeoclimatology, Palaeoecology, 309(1–2), 1–8. https://doi.org/10.1016/j.palaeo.2011.05.044

Schlunegger, F., Norton, K. P., & Zeilinger, G. (2011). Climatic forcing on channel profiles in the Eastern Cordillera of the Coroico Region, Bolivia. Journal of Geology, 119(1), 97–107. https://doi.org/10.1086/657407

Schwanghart, W., & Kuhn, N. J. (2010). TopoToolbox: A set of Matlab functions for topographic analysis. Environmental Modelling and Software, 25(6), 770–781. https://doi.org/10.1016/j.envsoft.2009.12.002

Schwanghart, W., & Scherler, D. (2014). Short Communication: TopoToolbox 2 - MATLAB-based software for topographic analysis and modeling in Earth surface sciences. Earth Surface Dynamics, 2(1), 1–7. https://doi.org/10.5194/esurf-2-1-2014

Schoenbohm, L. M., Whipple, K. X., Burchfiel, B. C., & Chen, L. (2004). Geomorphic constraints on surface uplift, exhumation, and plateau growth in the Red River region, Yunnan Province, China. Bulletin of the Geological Society of America, 116(7–8), 895–909. https://doi.org/10.1130/B25364.1

Seidl, M. A., & Dietrich, W. E. (1992). The Problem of Channel Erosion into Bedrock. Functional Geomorphology: Catena Supplement 23, 101–124.

Seidl, M. A., Dietrich, W. E., & Kirchner, J. W. (1994). Longitudinal profile development into bedrock: an analysis of Hawaiian channels. Journal of Geology, 102(4), 457–474. https://doi.org/10.1086/629686

Sempere, T. (1995). Phanerozoic evolution of Bolivia and adjacent regions. In A. J. Tankard, R. Suarez, & H. J. Welsink (Eds.), Petroleum Basins of South America: AAPG Memoir 62 (pp. 207–230).

Smith, R. B., Barstad, I., & Bonneau, L. (2005). Orographic precipitation and Oregon's climate transition. Journal of the Atmospheric Sciences, 62(1), 177–191. https://doi.org/10.1175/JAS-3376.1

Spotila, J. A., Farley, K. A., & Sieh, K. (1998). Uplift and erosion of the San Bernardino Mountains associated with transpression along the San Andreas Fault, California, as constrained by radiogenic helium thermochronometry. Tectonics, 17(3), 360–378. https://doi.org/10.1029/98TC00378

Stockmal, G. S., Beaumont, C., Nguyen, M., & Lee, B. (2007). Mechanics of thin-skinned fold-and-thrust belts : Insights from numerical models. Geological Society of America Special Paper, 433, 63–98. https://doi.org/10.1130/2007.2433(04).

Valla, P. G., Herman, F., van der Beek, P. A., & Braun, J. (2010). Inversion of thermochronological age-elevation profiles to extract independent estimates of denudation and relief history - I: Theory and conceptual model. Earth and Planetary Science Letters, 295(3–4), 511–522. https://doi.org/10.1016/j.epsl.2010.04.033

Wager, L. R. (1933). The Rise of the Himalaya. *Nature*, *132*(28), 28. https://doi.org/https://doi.org/10.1038/132028a0

Whipple, K. X. (2001). Fluvial landscape response time: How plausible is steady-state denudation? American Journal of Science, 301(4–5), 313–325. https://doi.org/10.2475/ajs.301.4-5.313

Whipple, K. X., & Gasparini, N. M. (2014). Tectonic control of topography, rainfall patterns, and erosion during rapid post-12 Ma uplift of the Bolivian Andes. Lithosphere, 6(4), 251–268. https://doi.org/10.1130/L325.1

Whipple, K. X., & Tucker, G. E. (1999). Dynamics of the stream-power river incision model: Implications for height limits of mountain ranges, landscape response timescales, and research needs. Journal of Geophysical Research: Solid Earth, 104(B8), 17661–17674. https://doi.org/10.1029/1999JB900120

Whipple, K. X., Shirzaei, M., Hodges, K. V., & Ramon Arrowsmith, J. (2016). Active shortening within the Himalayan orogenic wedge implied by the 2015 Gorkha earthquake. Nature Geoscience, 9(9), 711–716. https://doi.org/10.1038/ngeo2797

Willett, S. D. (1999). Orogeny and orography: The effects of erosion on the structure of mountain belts. Journal of Geophysical Research: Solid Earth, 104(B12), 28957–28981. https://doi.org/10.1029/1999jb900248

Willett, S. D., & Brandon, M. T. (2002). On steady states in mountain belts. Geology, 30(2), 175–178. https://doi.org/10.1130/0091-7613(2002)030<0175:OSSIMB>2.0.CO;2

Willett, S. D., Slingerland, R. L., & Hovius, N. (2001). Uplift, shortening, and steady state topography in active mountain belts. American Journal of Science, 301(4–5), 455–485. https://doi.org/10.2475/ajs.301.4-5.455

Wobus, C. W., Whipple, K. X., & Hodges, K. V. (2006). Neotectonics of the central Nepalese Himalaya: Constraints from geomorphology, detrital 40 Ar/ 39 Ar thermochronology, and thermal modeling. Tectonics, 25(4), 1–18. https://doi.org/10.1029/2005TC001935

Wobus, C. W., Whipple, K. X., Kirby, E., Snyder, N., Johnson, J., Spyropolou, K., et al. (2006). Tectonics from topography: procedurses, promise, and pitfalls. In Willett, S.D., Hovius, N., Brandon, M.T., and Fisher, D.M., eds., Tectonics, Climate, and Landscape Evolution: Geological Society of America Special Paper 398, Penrose Conference Series (pp. 55–74). https://doi.org/10.1130/2006.2398(04).

Yang, R., Willett, S. D., & Goren, L. (2015). In situ low-relief landscape formation as a result of river network disruption. Nature, 520(7548), 526–529. https://doi.org/10.1038/nature14354

Yanites, B. J., & Ehlers, T. A. (2012). Global climate and tectonic controls on the denudation of glaciated mountains. Earth and Planetary Science Letters, 325–326, 63–75. https://doi.org/10.1016/j.epsl.2012.01.030

Yanites, B. J., & Ehlers, T. A. (2016). Intermittent glacial sliding velocities explain variations in long-timescale denudation. Earth and Planetary Science Letters, 450, 52–61. https://doi.org/10.1016/j.epsl.2016.06.022

5.0 Appendix 1: Integrating Pecube and Move: A Brief Runthrough (Tutorial)

Compiled by Victoria Buford

Vmb21@pitt.edu

Updated: October 30, 2018

Overview: This writeup discusses the following:

- Creation of point cloud in MATLAB
- Import, deformation, and export of point cloud in MOVE
- Exporting of topo lines and structure lines in MOVE
- Folder & File Structure for Pecube & Plotting in MATLAB
- Setting up all Pecube files
- Connecting to the server using Fetch (Ftp program) and Terminal/Command Prompt
- Running Pecube model
- Downloading & Analyzing the Pecube Model in MATLAB

Summary of files:

- Required input files for Pecube
    - Grids (*.dat)
        - Tab or space separated (space is best)
        - No header
        - In kilometers
        - [X Y Z   Colour ID UID]
    - Topos (*.dat)
        - Tab or space separated (space is best)
        - No header
        - In kilometers
        - [X Y Z   ID]
    - Velocity (*.txt)
        - 4 header lines followed by ages and grid files
    - Pecube.in
        - Edited to reflect your model timing, thermal parameters, topography file names, and which thermochronometers to model (and more)
    - Run_pecube.sbatch
        - Doesn't need to be edited (other than to put your email address on it the first time), unless you want to run on a specific node
- Pecube output files produced
    - Vel_new
        - Download to analyze if the velocity produced is correct. Can help identify problems with Pecube or problems with the grid itself
    - Vel_top

        - We don't really use this one
    - Pecube_errors
        - Will be 0kb if nothing is wrong, but if the model crashes, will list a (sometimes cryptic) reason for the crash
    - **.log
        - will grow the entire time the model is running, detailing what pecube is computing when. Lists the time to complete when the model is done. Is helpful for understanding where in the model run pecube crashed.
    - temps_tec**.dat
        - for each model step, this lists the temperatures for the full modeled space along the cross-section (so both x and z)
    - ages_tec**.dat
        - for each model step, this lists the ages for the modeled thermochronometers and along-section (x) coordinates; the spacing of these points is determined by the pecube spacing you selected, not by your Move grid spacing.
- Required files for MATLAB Plotting with DLG script
    - Vel_new (*.dat downloaded from Pecube)
    - Ages_tec (*.dat downloaded from Pecube)
    - Temps_tec (*.dat downloaded from Pecube)
    - Structure lines (*.dat exported from Move)
        - Space-separated
        - In kilometers
        - With a Header row
        - [XYZID]
- Optional Files for MATLAB plotting with DLG script
    - Thermochronometer data (*.txt)
        - Tab-separated
        - With header which contains: (Method Age Error xdist)
            - With that capitalization
            - Methods available: AFT, AHe, ZFT, ZHe, Mar
            - Age and Error in Ma
            - xdist in km, corresponding to location matching the section

Recommended programs

- Text Editors
    - Mac: TextEdit (Default), Xcode, Smultron ($)
    - Windows: Notepad ++ (NOT NOTEPAD), Atom
- SFTP Programs
    - Mac: Fetch
    - Windows: WinSCP
- Command Line Interface
    - Mac: Terminal (default)
    - Windows: Command Line (default), PuTTY
- Plotting/Analysis Software
    - MATLAB

- Image Viewers
    - Mac: Preview (default)
    - Windows: Photos (default), Windows Photo Viewer

5.1 Setup

5.1.1 Creating the Point Cloud (Grid)

Create a 0.5km x 0.5 km grid in MATLAB such that the grid extends from the basal décollement up to the surface. Make sure that the grid exists over 0km in places that form basin fill, (eg if the expected basin is 4.5km deep, make sure in the initial setup your grid extends up to 4.5km elevation in the basin area).

The MATLAB script is titled Create_MOVEPointCloud.m and consists of the following:

```
function [MOVE_PointCloud] =...
    Create_MOVEPointCloud(x_max, x_min, z_max, z_min, resolution)
%   INPUT:
%   x_max, x_min, z_max, z_min: integers of bounding box in [km]
%   resolution: MOVE point cloud resolution in [km]

%   OUTPUT:
%   columns: x,y,z, color ID, unique ID
%   Paul R. Eizenhofer, PhD
%   University of Pittsburgh, peizen@pitt.edu

[X,Z] = meshgrid(x_min:resolution:x_max, z_min:resolution:z_max);
MOVE_PointCloud(:,1) = X(:);
MOVE_PointCloud(:,2) = 0;
MOVE_PointCloud(:,3) = Z(:);
MOVE_PointCloud(:,4) = 0;
for i = 1:length(MOVE_PointCloud(:,3))
    MOVE_PointCloud(i, 5) = i;
end
```

The resolution for our MOVE models is 0.5km.

Then, open the variables section_ptcld and basin_ptcld in MATLAB, copy them into a text editor, and save as a .txt file. (You can paste them into the same text file, or separately, if you prefer to keep them as separate point clouds.)

Note: If you wish to create a non-rectangular grid, you can do so by creating the two clouds separately, but starting the unique IDs (UID) of the second cloud at the max UID of the first plus one. Eg.

```
section_ptcld=Create_MOVEPointCloud(500,0,0,-15,0.5);
basin_ptcld=Create_MOVEPointCloud(650,500.5,5,-15,0.5);
basin_ptcld(:,5)=basin_ptcld(:,5)+max(section_ptcld(:,5));
```

5.1.2 Importing the Point Cloud into MOVE

Open your MOVE Model in cross-section view, use File>Insert (Ctrl+I, Cmd+I) and select the .txt file(s) of your point cloud(s). Change "Select Object Type" to "Points", ensuring the units are in kms. Ensure the columns read: X Y Z Colour ID UID. If you don't have them, right click on the column and select Colour ID and then click "Create New" (for UID). Title it *UID,* Type: *Integer*, and Category: *Dimensionless*.

In the Model Browser, click "Clear Filters", and under "Object Types" find "Point Data" and select your point cloud. Go to Project> To Section, click "Add" or "Collect" and click "Apply". You can do the majority of your work with the visibility toggled off, so that MOVE will run faster.

5.1.3 Deforming the Point Cloud in MOVE

Deform, Load, Erode, Unload, (and sediment load, if required) as normal, but be sure to add the point cloud into the objects being deformed/loaded/unloaded.

Note: Having the point cloud visible, particularly when performing a step that involves deformation or movement, makes MOVE run slower because it has to render the grid. Thus, you can do this with visibility off (the little checkbox) and just select the point cloud, and click "Add" in the Move on Fault or Decompaction Modules. After you have performed the movement, you should toggle the visibility of the point cloud on to make sure that the cloud has actually moved like it should.

5.2 Exporting the Point Cloud

Note on Naming Schemes. The simplest naming scheme is the best. Generally, we prefer to have the total deformation in kms in the name, but this creates problems for MATLAB later on. You can either name with the deformation amounts in the name, and rename them for MATLAB plotting later, or use the scheme suggested below if you don't want to rename it. If you decide to have step identifiers other than step #, be careful to not make the name too long as Pecube/Cascade can cut it off if it's too long.

Model_Step#_(object type) is best for MATLAB plotting later on (Object type = lines, grid, topo)

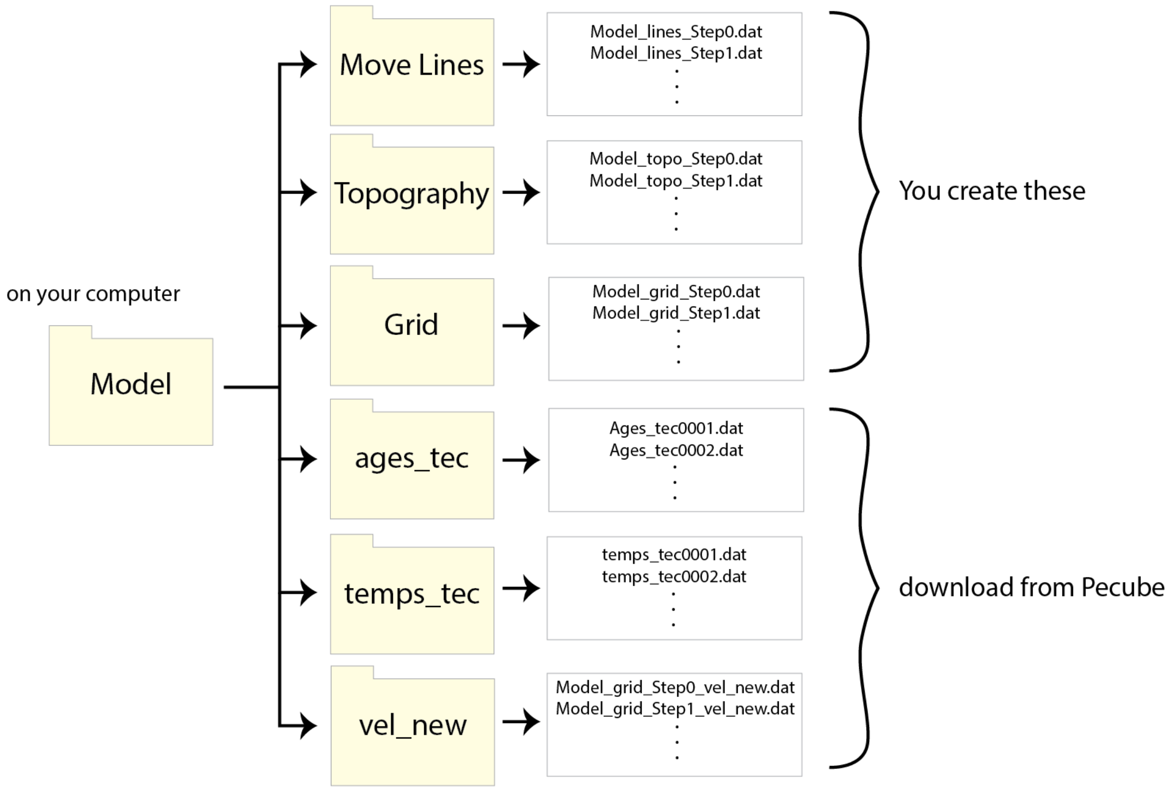


Figure 5.1. Pecube folder structure setup

At the end of each step (DLEU), export the grid, topography, and the Move Lines.

Grid. Make sure (by toggling visibility on) that the grid has actually deformed, loaded, and unloaded. properly. To export the grid (Even with visibility off) select it in the model browser, and go File > Export (Ctrl+Shift+E Cmd+Shift+E). Change File Format > ASCII, and click “Match Model Selection”, then click Next. Select X, Y, Z, Colour ID, and UID as attributes, and change Separator to “Space”. On the Next screen, change the XY and Z units to kilometers, and do not write a header. Select where, and what, to name the file (keeping in mind the *note* above).

Topo. Select the single topography line, and Export. You need X, Y, Z, and ID, space-separated, in kilometers, with no header.

Lines. Select all Move structural and bedding lines (ctrl+A or cmd+A, ctrl or cmd click the section trace and any lines you don't want) and export. You want X, Y, Z, ID, space-separated, in kilometers, WITH A HEADER.

5.3 Pecube Model Setup

You can create all three of the following files using the Matlab script *Pecube_CreateInputFiles.m*. This will allow you to select your grid and topo files, input the ages, thermal paramters, and model names, and will create the velocity.txt file, pecube.in file, and run_pecube.sbatch file.

Note. All files created will have .txt extension. You MUST CHANGE the Pecubein.txt extension to Pecube.in for it to run. If you don't change the sbatch extension, use the command `sbatch run_pecube.txt` *to start the model.*

To change the extension:

- *On Windows, Use Notepad ++ (NOT NOTEPAD); open Pecubein.txt: save as, Pecube.in*
- *On Mac, right click on Pecubein.txt, click "Get Info". In the Name & Extension: blank, change the name to Pecube.in; when a dialog pops up, click "Use .in"*

velocity file: list the ages and corresponding grid for n+1 deformation steps (all def steps+starting); include 10 5 10 1.0 at beginning:

```
10
5
10
1.0
age_start  model_grid_step0.dat
.
.
.
0 model_grid_steplast.dat
```

Note: **DO NOT COPY AND PASTE FROM ANY MICROSOFT PROGRAMS INTO THE TEXT EDITOR.** It makes pecube (AND CASCADE) angry. Pretty much any text editor works, some of use XCode, Smultron, Mac's TextEdit, Atom (Windows).

Note: the first four lines correspond to: (1) the y dimension in km, (2) # nodes in y-direction, (3) interpolation window for averaging over the MOVE grid, and (4) x step-size for interpolation (of topo).

The pecube.in file

This is where you enter the names of the topography files exported from Move, select thermal parameters, select which thermochronometers to plot, and more. n=#def steps.

-*Check the box folder for an example pecube.in file*

(input #2): # of topo files (n+2)

(input #4): Names of topo files (n+2) (step 0 listed twice).

Model_topo_step0.dat
Model_topo_step0.dat
Model_topo_step1.dat
.
.
.
Model_topo_steplast.dat

(input #6): initial, undeformed width, (in node spacing): take the width of your model (eg, it runs from -65 to 600 (so, 665km wide), and you want two nodes per km; so, you type 1330 5). The 5 indicates width in the along strike direction, and we don't change it.

(input #7): spacing of nodes in meters (so, if you did two nodes per km: 500 1000) (we don't change the second 1000 bc we don't really care about the along strike direction in this instance)

(input #9): starting location, in meters, for the pecube grid; (example above: -65000 0.0)

(input #10): # of deformation steps (n+1)

(input #12a): change column (a) to correspond with time steps

(input #14): where you alter thermal parameters: atmospheric lapse rate (i), e-folding depth (k), thermal heat production (j)

(input 17): which Tchron systems you want to be calculated (0=no, 1=yes)

(after input 22): at the end of the pecube.in file, type the name of the velocity file (.txt).

The run_pecube.sbatch file

```
#!/bin/bash -l
## Run script for pecube monte carlo on esd slurm


## General configuration options
#SBATCH -J Pecube_W
#SBATCH -e Pecube_Error%j
#SBATCH -o Pecube_Screen%j
#SBATCH --mail-user=email_id@pitt.edu
#SBATCH --mail-type=ALL


## Machine and CPU configuration
## Number of tasks per job:
#SBATCH -n 1
## Number of nodes:
#SBATCH -N 1
#SBATCH -w u-017-s133

module load pecube
## In some instances, the above may need to be
## module load pecube_new

pecube
```

Note: if you want to run your model on a less busy node, check which nodes are busy using squeue. *(See section 6 for more details on how to implement Command Prompt commands.) Include this in the .sbatch file in file: the highlighted area:*

#SBATCH -w esd_node1 (to run on #38)
#SBATCH -w esd_node2 (to run on #39)
#SBATCH -w argand (to run on #40)
#SBATCH -w u-017-s133
#SBATCH -w u-017-s134
#SBATCH -w u-017-s135
#SBATCH -w u-017-s136

5.4 Setting up SFTP Connections using Fetch (or Another FTP Program; WinSCP for Windows)

Download and install Fetch.
File > New Connection
Hostname: 134.2.5.40
SFTP
Username: whatever Willi assigns you
Password: whatever willi assigns you
Initial folder: /esd/esd01/data/username
Port: 6307
Click the Heart to SAVE
Check add to keychain
In your personal folder, you can do what you want.
You are given the following folders:
data_large
model_runs: for important models; is backed up
scratch: for testing, less important runs; is NOT backed up

5.5 Upload your Model according to the Folder Structure

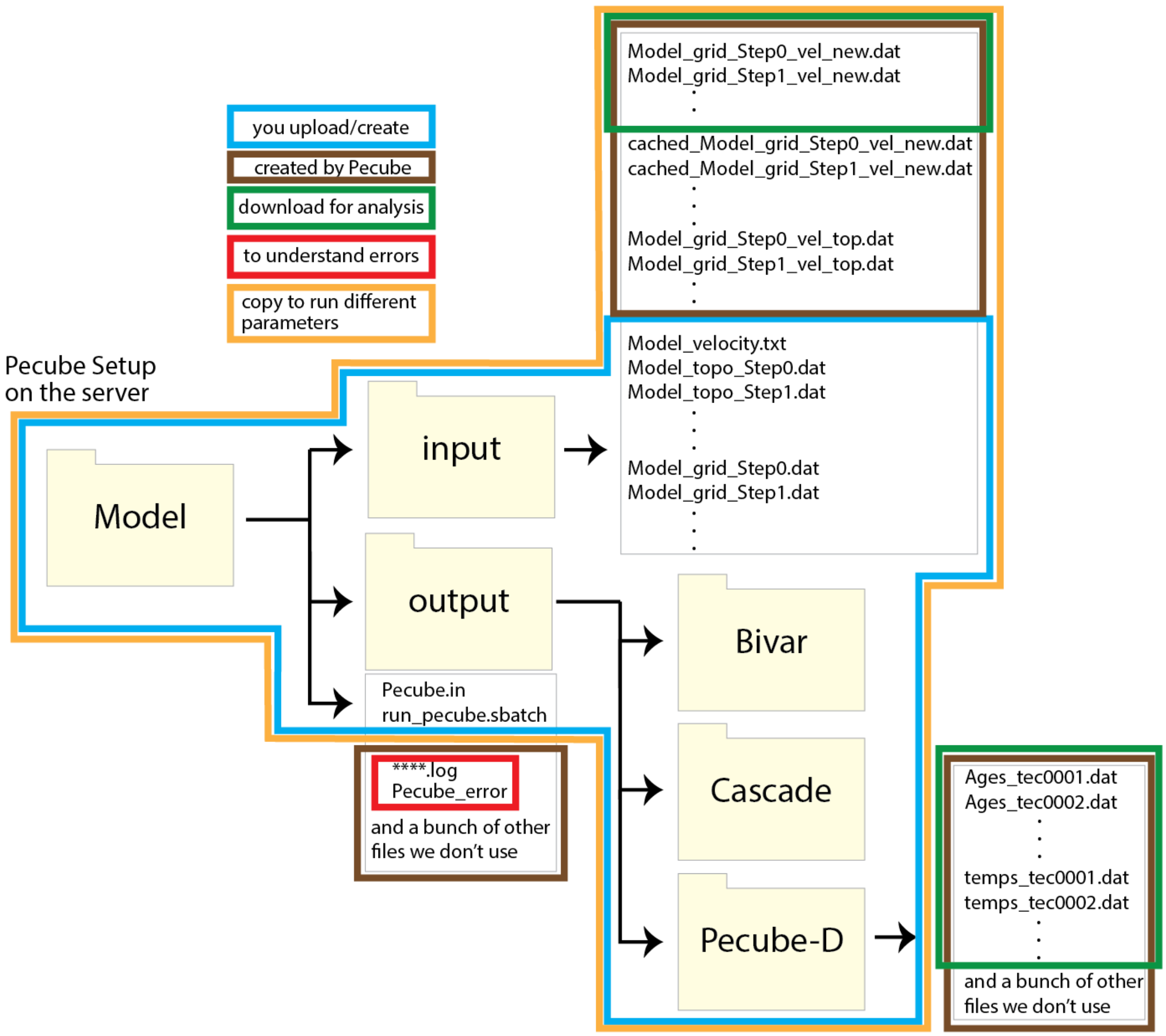


Figure 5.2. Pecube folder structure on the server

5.6 Running your Model

5.6.1 Using Terminal (Mac) or Command Prompt (Windows; or Putty)

a) Use ssh -l username -X -p 6307 134.2.5.40

to log on to the server (replace username with yours). Then type in your password. It won't be visible.

b) Use cd /esd/esd01/data/username/ to switch to your folder, and then navigate to your specific model (eg cd /model_runs/Model).

c) Then, type in sbatch ./run_pecube.sbatch (for pecube). It should confirm the job and give you a number.

d) Check to make sure it's still running with squeue OR squeue -u username. If it computes the velocity files (found in the input file), you're most likely good (this takes 1-4 hrs).

5.6.2 Terminal Commands

ssh -l username -X -p 6307 134.2.5.40
squeue
squeue -u username
cd ..
cd /esd/esd01/data/username/
sbatch run_pecube.sbatch
scancel job#
ls

scp –r source/folder destination/folder
exit

log on to the server
find out what's running
what are you running
go up a directory
change directory to your
run a pecube file
cancel a job
display contents of folder, indicates what other folders are there (Can use dir also)
copy a folder (source) and its contents to destination
log off

Note. If you want to run your model on a less busy node, check which nodes are busy using squeue. *Then, include one of these in the run_pecube.sbatch file:*

#SBATCH –w u-005-s038 (to run on #38)
#SBATCH –w u-005-s039 (to run on #39)
#SBATCH –w u-005-s040 (to run on #40)

5.7 Analyzing your Pecube Model Output

Download the vel_new (from input folder), temps_tec (from output folder), and ages_tec (from output folder). Save them according to the folder structure in in section 2.

Remove temps_tec0000.dat from the other temps_tec folder on your computer (otherwise you’ll have too many files).

Ensure that in addition to the three folders listed above, you also have the move structures /lines exported as listed in section 2.

Place PecubeOutputPlot_batch_DLG_v****.m and inputsdlg.m in your MATLAB folder.

Open PecubeOutputPlot_batch_DLG_v****.m and in Editor, click Run. This will pop up a dialog box where you can select the folders for your lines, velocities, ages, temperatures, and where to save the output.

The **Model Name** is very important, and must be the phrase by which you named your lines and grids (and thus velocities). This is how MATLAB searches for which files to plot.

Model ID is just a catch-all text blank for you to identify which model you are plotting. It gets plotted in the title of the plot, and included in the save name for the figure. It is not used for searching for files.

Deformation front on: right OR left. This controls where the legend is plotted so as to not obscure the data due to the deformation front.

Move Lines Folder: click in this blank and select where your lines/structures are stored

Velocity Folder: click in this blank and select where your (vel_new) velocities are stored

Ages: click in this blank and select where your ages_tec****.dat are stored

Pecube Temperatures Folder: click in this blank; select where your temps_tec**.dat are stored

Save Folder: Select where to save the figures.

XRange: Type in what xrange you want plotted (this is the along cross-section distance; positive to the right, like a normal plot)

ZRange: Type in what depth/elevation range you want to plot

Age Range: Type in what ages you want to plot (in Ma; generally, model start age + time you put in to equilibrate) (0-100Ma is the default)

Figure Format: the extension for the figure to be saved in (eg. .png, .epsc, .fig)

Tchron Smoothing Method: using MATLAB's smoothdata function, this will average the data using the *Method* you select over the *window size* you select. Using a window size of 1 will not average/smooth the data. If you used a 0.5km grid and node spacing (in Pecube), then a window size of 3 will average over the nearest 0.5km on both sides of the point it is computing at. Window size of 5 = nearest 1km. Window size of 7= nearest 1.5km. If you put in an even number, it will do compute the average over a trailing window: eg, with a window size of 4, MATLAB will smooth over two behind and one in front. If you aren't sure which method to use, movmean or movmedian are a good place to start.

Timesteps to plot: Will either plot All OR Range. For Range, you must enter something in the *From X:Y*. If you just want to plot one timestep, just enter the number by itself (eg 12) or a range (23:25).

Tchron data file [txt]: a text file containing a tab-separated list of thermochronometer data. You must have a header row with: Method, Age, Error, and xdist (with that capitalization).

Method options: (as of v 1.4) MAr, ZFT, ZHe, AFT, AHe. Must be tab separated

Thermochronometers to plot: Check which thermochronometers to plot. “Modeled” pulls from the pecube output, while “data T=end” pulls from the Tchron data file.

The function is setup to remember your last inputs as long as you don’t clear MATLAB’s working memory (clear). To rerun subsequent times, either “Run Section” (Cmd+Enter) or comment out (use % at the beginning of the line) the `clear; close all; clc;` line (on or about line 44).

5.7.1 Ensuring No Vertical Exaggeration

Currently, this relies on the MATLAB command `daspect`. However, sometimes this resizes the cross-section view of the subplot so that it is narrower than the Age plot. So, either, turn off the `daspect` command at the end of the plotting, or use the following description to set it up so it works for the ranges you want.

As this depends on the monitor, XRange, and ZRange, the simplest way is to setup your MATLAB runs as follows:

Comment out the `saveas(***)`and `close` lines at the very end.

Make sure the `daspect` command is NOT commented out (at the end of plotting).

Run the script for just one timestep

Adjust the figure window to the size/shape you like.

In the command window, type get(gcf,'Position')

Replace the set(gcf,'Position',[1 5 1280 700]) with the numbers from the command window.

You can now uncomment the saveas(***)and close lines and run the entire script for your entire model (for this specific XRange and ZRange; it may or may not work for other ones) and it will plot with no vertical exaggeration.

5.7.2 Troubleshooting the MATLAB Script

Files appear to be missing!

This means that in the folders you've included, there aren't an equal number of files. The command window should display the number of files it is finding in each folder. Sometimes, this is a problem of leaving temps_tec0000.dat, and sometimes it is a problem with a storage volume in that MATLAB is seeing hidden files. If you know you have the correct number of files in the folders, and you know how many total files that is, you can override this error by typing in the command window: TotalFiles= ##, where ## is the number of files/steps you have.

Warning: Duplicate data points have been detected and removed -corresponding values have been averaged.

You can basically ignore this.

Exceeds matrix dimensions.

This is basically the least helpful error ever, but may be caused by not selecting the exact thermochronometers for pecube to create that this file was written for (AHe, AFT, ZHe, ZFT, and MAr). In the %%% Plot ages %%% section (approx. line 470), you'll have to change which column Age_Avg is pulling the data from. Open an ages_tec file (in excel, or a text editor), and look at the header row. Match up which column goes with which thermochronometer, and alter the y value for each thermochronometer to plot. For example, if you choose for Pecube to create AHe, AFT, ZHe, and MAr, then the script should read as follows (the changes have been highlighted):

```
% MAr
  if MAr==1
  plot(Age_Avg(:,4), Age_Avg(:,10),...
      'o-',...
      'MarkerEdgeColor',[0.6350 0.0780 0.1840],...
      'MarkerFaceColor', [0.6350 0.0780 0.1840],...
      'MarkerSize',4);
  leg(end+1)={['MAr']};
  end
% % ZFT
%     if ZFT==1
%     plot(Age_Avg(:,4), Age_Avg(:,10),...
%         'd-',...
%         'color',[0.4940, 0.1840, 0.5560],...
%         'MarkerFaceColor', [0.4940, 0.1840, 0.5560],...
%         'MarkerSize',4);
%     leg(end+1)={['ZFT']};
%     end
% ZHe
  if ZHe==1
  plot(Age_Avg(:,4), Age_Avg(:,9),...
      'v-',...
      'color',[0.8500, 0.3250, 0.0980],...
      'MarkerFaceColor', [0.8500, 0.3250, 0.0980],...
      'MarkerSize',4);
  leg(end+1)={['ZHe']};
  end
% AFT
  if AFT==1
  plot(Age_Avg(:,4), Age_Avg(:,8),...
      's-',...
      'color',[0, 0.4470, 0.7410],...
      'MarkerFaceColor', [0, 0.4470, 0.7410],...
      'MarkerSize',4);
  leg(end+1)={['AFT']};
  end
% AHe
  if AHe==1
  plot(Age_Avg(:,4), Age_Avg(:,7),...
      'p-',...
      'color',[0.4660, 0.6740, 0.1880],...
      'MarkerFaceColor', [0.4660, 0.6740, 0.1880],...
      'MarkerSize',4);
  leg(end+1)={['AHe']};
  end
```

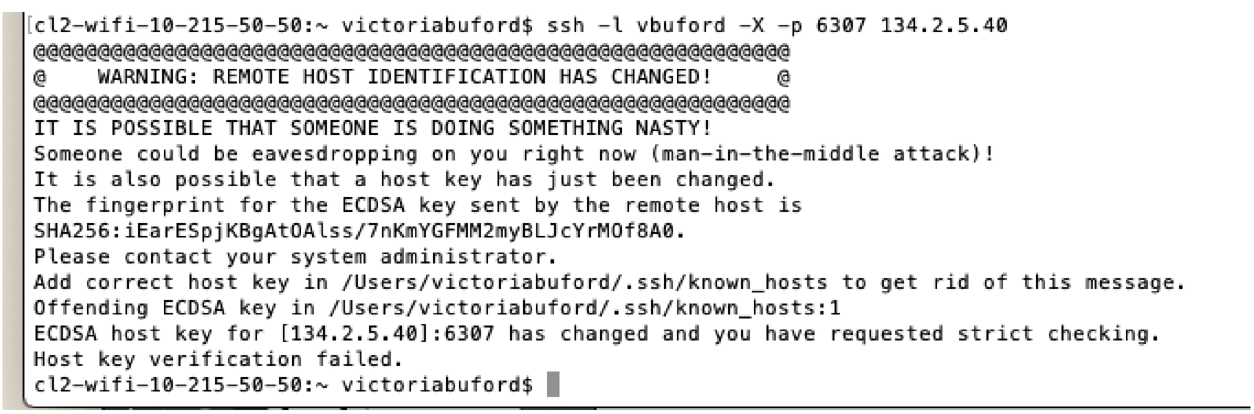
TchronData.Method does not exist.

Oops, either you selected the wrong file to import, or something isn't importing correctly. Sometimes MATLAB is finicky with the importation of tables. Check TchronData by double clicking it in the Workspace. Are the table headers what you typed in, or are they something generic like "Var4"? If they are generic, you can rename them in MATLAB and comment out the line TchronData=readtable(Answer.TchronDataFile,... 'Delimiter','\t'); so that MATLAB will just use the edits you just made rather than reimporting the file.

If they aren't generic, you may have not used the capitalization scheme: Method, Age, Error, xdist. Fix that in your txt file, and it should import fine.

5.7.3 Troubleshooting Login via Command Line

On mac, if they update the server security settings, you may need get a warning like this:



```
cl2-wifi-10-215-50-50:~ victoriabuford$ ssh -l vbuford -X -p 6307 134.2.5.40
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
SHA256:iEarESpjKBgAtOAlss/7nKmYGFMM2myBLJcYrMOf8A0.
Please contact your system administrator.
Add correct host key in /Users/victoriabuford/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /Users/victoriabuford/.ssh/known_hosts:1
ECDSA host key for [134.2.5.40]:6307 has changed and you have requested strict checking.
Host key verification failed.
cl2-wifi-10-215-50-50:~ victoriabuford$
```

Figure 5.3. Command line error

To fix this, ensure you are in /Users/your_user_name/ (by $ pwd), and then

$ rm ~/.ssh/known_hosts

to remove the old authentication keys. Then, log back into the server, and you'll probably have to confirm the new authentification.

```
cl2-wifi-10-215-50-50:/ victoriabuford$ pwd
/
cl2-wifi-10-215-50-50:/ victoriabuford$ cd /Users/victoriabuford/
cl2-wifi-10-215-50-50:~ victoriabuford$ rm ~/.ssh/known_hosts
cl2-wifi-10-215-50-50:~ victoriabuford$ ssh -l vbuford -X -p 6307 134.2.5.40
The authenticity of host '[134.2.5.40]:6307 ([134.2.5.40]:6307)' can't be established.
ECDSA key fingerprint is SHA256:iEarESpjKBgAtOAlss/7nKmYGFMM2myBLJcYrMOf8A0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[134.2.5.40]:6307' (ECDSA) to the list of known hosts.
vbuford@134.2.5.40's password:
Creating directory '/home/vbuford'.
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-38-generic x86_64)
```

Figure 5.4. Resolve the command line error

5.8 Notes on Operating System of the Server

```
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-38-generic x86_64)
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Tue Oct 30 15:44:21 CET 2018
  System load:  2.02              Processes:           342
  Usage of /:   5.8% of 915.40GB   Users logged in:     4
  Memory usage: 9%                IP address for eno1: 134.2.5.40
  Swap usage:   0%

 * Security certifications for Ubuntu!
   We now have FIPS, STIG, CC and a CIS Benchmark.
   - http://bit.ly/Security_Certification

 * Want to make a highly secure kiosk, smart display or touchscreen?
   Here's a step-by-step tutorial for a rainy weekend, or a startup.
   - https://bit.ly/secure-kiosk

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

6.0 Appendix 2: Cascade Setup Tutorial

6.1 Overview

Brief look at how to run Cascade, and export the output.

Updated 8/16/18 for icecascade_v1_80

Summary of files:

- Required input files for Cascade
  - Grids (*.dat)
    - Tab or space separated (space is best)
    - No header
    - In kilometers
    - [X Y Z   Colour ID UID]
  - Topos (*.dat)
    - Tab or space separated (space is best)
    - No header
    - In kilometers
    - [X Y Z   ID]
  - Velocity (*.txt)
    - 4 header lines followed by ages, grid and topo files
  - icecascade.in
    - Edited to reflect your model timing, atmospheric and climate parameters, (and more; see below)
  - Run_icecascade.sbatch
    - Doesn't need to be edited (other than to put your email address on it the first time), unless you want to run on a specific node
    - Can change model name so that on terminal it shows
  - Icecascade (executable)

6.2 Move Setup

See Pecube plotting script tutorial (Section 5.1. Setup).

6.3 Exporting the Point Cloud

See Pecube_Setup-Plotting for how to setup and export grid, topo, and lines.

Note. Cascade doesn't allow selection of where in the cross section to analyze, so the area of interest must start at 0, run in the positive direction, to the width of your model (defined in icecascade.in inputs D1 and D2). If necessary, use MATLAB or some other coding to alter the absolute values of x/y/z to center your model in the desired CASCADE space.

6.4 Model Setup

velocity file: list the ages and corresponding grid for n+1 deformation steps (all def steps+starting); include 10 5 10 1.0 at beginning. (same as pecube)

```
10
5
10
1.0
age_start  model_grid_step0.dat
.
.
.
0 model_grid_steplast.dat
```

You may choose to allow 0.2-0.5My of initialization time for river channels to form. Do this in the velocity.txt file by adding 0.2-0.5 to age_start, and using the same grids twice. EG:

```
10
5
10
1.0
age_start+0.2  model_grid_step0.dat
```

age_start model_grid_step0.dat
age_start model_grid_step1.dat
.
.
.
0 model_grid_steplast.dat

Note. **DO NOT COPY AND PASTE FROM ANY MICROSOFT PROGRAMS INTO THE TEXT EDITOR.** It makes pecube/cascade angry. Pretty much any text editor works, some of use XCode, Smultron, Mac's TextEdit, Atom (Windows).

Note. the first four lines correspond to: (1) the y dimension in km, (2) # nodes in y-direction, (3) interpolation window for averaging over the MOVE grid, and (4) x step-size for interpolation (of topo).

6.5 The 'icecascade.in' File

This is where you enter the select thermal/climate parameters, and more. n=#def steps.

*= change per model; ** Orographic precipitation

y is the distance along the cross-section (our normal x)

6.5.1 Model Processes: (Fluvial Erosion & Deposition, Hillslope Diffusion, Landslide & Glacial Erosion, Orographic Precipitation)

*** **(input A9):** T for orographic precipitation, F for constant/global precipitation (in the new version with imposed precipitation option, this is a numerical input of 1 for global, 2 for original dynamic orography, and 3 for imposed.

6.5.2 Model Output: (Screen Output, Model Output)

(input B2): How often the model writes into the screen output file (how precise do you want your error tracking to be? *Default is 10000*)

(input B4): how frequent to export topo/landscape [yrs] (if model runs for 5Ma, and you export every 50000yrs, then you get 100 files) (for more complete videos, 50k-*100k* screen output is best)

(input B5): name of this model run, also name of output folder (eg *IceCascade* shown in the figure in section 5)

6.5.3 Time Domain: (Dynamic Time, Model Run Time)

(input C1): dynamic time; *best practice == T.*

(input C2): 25-125yrs; *best practice == 25*

***(input C3)**: total model time in years. Should be first age in velocity.txt file

6.5.4 Spatial Domain: (Model Size, # Nodes, etc)

Note: read Paul's paper on legacy topography and see FAQ for info on model size

***(input D1)**: number of nodes along edges of model ➔ (model width [km] *desired Cascade resolution [nodes/km]) + 1 (x – perpendicular to section, y – along section) (start at *1km resolution*, and increase once your model confidence is higher)

***(input D2)**: length of model in D1 in km

***(input D4)**: input an initial topography ➔ *T*

***(input D5)**: name of initial topography file *.dat

(input D6): 3

6.5.5 Fluvial Erosion: (Fluvial/Bedrock Erosion Coefficients & Constants)

(input E1): an (inverse) proxy for rock strength; default is *3.5e-4*. Range of permissible values is approximately 3.5e-3:3.5e-5. For stronger rocks, use smaller numbers (eg 8.5e-5 is half an order of magnitude stronger). For weaker rocks, use larger numbers (eg 8.5e-4 is half an order of magnitude weaker)

6.5.6 Hillslope Erosion: (Hillslope/Landslides Erosion Coefficients & Constants)

Note: [We don't change anything normally]

(input F1): k_{diff} [km^2/yr]: *2e-6*

6.5.7 Ice: (Glaciers)

(inputs G-L)

6.5.8 Tectonics: (Kinematics, Rock Properties, Isostasy, etc)

(input M1): *3*

***(input M1b):** velocity.txt file (which contains ages, move grids, etc)

6.5.9 Climate: (Control [Orographic] Precipitation, Climate, etc)

Used in conjunction with A9 = T or F, or in new version A9= 1 or 2.

(input N1): keep on order with structures, plotting interval, or 25000

(input N2): precipitation grid size in km (might crash if <1km)

****(input N3):** "Precipitation rate for uniform rainfall model [m/yr]" not actually straight rainfall, affected by MAT and climate parameters, only works with orographic precip on

* ****(input N6):** atmospheric moisture; change a0, leave a1 alone

(input N8): wind speed

(input N9a): wind direction (doesn't differentiate between –y and y)[0-180 permissible]

(input N9b): F to do nothing; T to flip the model 180 on the x-axis (to make orographic precipitation flow from the correct side)

(input N10/N11): wind smoothing scale

****(input N13):** atmospheric lapse rate (only works with atmospheric precip on (?))

(input N14): annual variation in daily temperature C

***(input N18)**: min and max temperature at base of the model

6.5.10 Temperature: (Control Climate, For All Runs)

Used in conjunction with A9 = T or F, or in new version A9= 1 or 2.

(input N01): F

***(input N05):** MAT

6.6 The 'run_icecascade.sbatch' File

```
#!/bin/bash -l
## Run script for icecascade on esd slurm


## General configuration options
#SBATCH -J MODEL-NAME
#SBATCH -e IceCascade_Errors%j
#SBATCH -o IceCascade_ScreenOutput%j
#SBATCH --mail-user=EMAIL@pitt.edu
#SBATCH --mail-type=ALL


## Machine and CPU configuration
## Number of tasks per job:
#SBATCH -n 1
## Number of nodes:
#SBATCH -N 1
#SBATCH -w u-005-s040

#srun --resv-ports icecascade
srun ./icecascade
```

Note: if you want to run your model on a less busy node, check which nodes are busy using squeue. *(See section 6 for more details on how to implement Command Prompt commands.) Include this in the .sbatch file in file: the highlighted area:*

#SBATCH –w u-005-s038 (to run on #38)
#SBATCH –w u-005-s039 (to run on #39)
#SBATCH –w u-005-s040 (to run on #40)

6.7 Setting up SFTP Connections Using Fetch (or Another FTP Program; WinSCP for Windows)

Follow the instructions in Pecube_Setup-Plotting (Section 5.4 Setting up SFTP Connections using Fetch (or Another FTP Program; WinSCP for Windows)

6.8 Upload your Model According to the Folder Structure

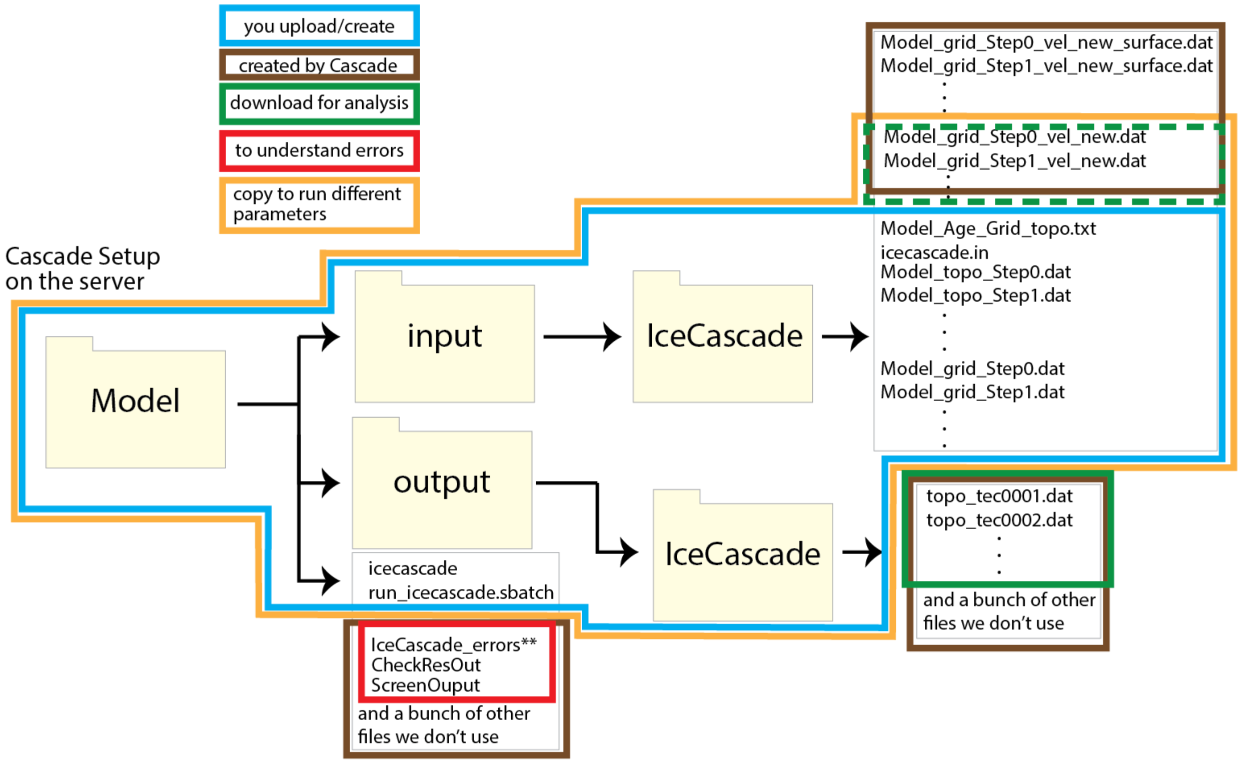


Figure 6.1. Folder structure for CASCADE

Running your model.

a) Using Terminal (Mac) or Command Prompt (Windows; or Putty) use

`ssh -l username -X -p 6307 134.2.5.40`

to logon to the server (replace `username` with yours). Then type in your password. It won't be visible.

b) Use `cd /esd/esd01/data/username/` to switch to your folder, and then navigate to your specific model (eg `cd /model_runs/Model`).

c) Then, type in sbatch run_icecascade.sbatch (for cascade). It should confirm the job and give you a number. Type in chmod 777 * to give yourself permission to run the job. Sometimes you have to run sbatch and chmod twice to give full permission.

d) Check to make sure it's still running with squeue OR squeue -u username. If it computes the velocity files (found in the input file), you're most likely good (this takes 1-4 hrs).

Terminal commands

ssh -l username -X -p 6307 134.2.5.40	log on to the server
squeue	find out what's running
squeue -u username	what are *you* running
cd ..	go up a directory
cd /esd/esd01/data/username/	change directory to yours
sbatch run_icecascade.sbatch	run a cascade job
chmod 777 *	give permission to run a job
scancel job#	cancel a job
ls	display contents of folder
scp –r source/folder destination/folder	copy a folder (source) and its contents to destination
exit	log off

Note. If you want to run your model on a less busy node, check which nodes are busy using squeue. *Include this in the run_pecube.sbatch file:*

#SBATCH –w u-005-s038 (to run on #38)
#SBATCH –w u-005-s039 (to run on #39)
#SBATCH –w u-005-s040 (to run on #40)

6.9 Analyzing Your Cascade Model Output

Cascade exports in topotec format; designed to be plotted with tecplot (an expensive program). You can extract the files using MATLAB since they are in ASCII format, and plot/analyze from there). Use Cascade_plot.m or Cascade_plot_mult(…etc).m to extract and plot general elevation and precipitation over the model space. For Ksn extraction, you'll need

TopoToolbox by Schwanghart and Scherler (there is a an app you can add to matlab, or you can download it as a .m and put it in the home folder).

6.10 FAQ

6.10.1 What Are The "Required" Parameters For Setup?

C3 (model start time)

D1 (model size in km)

D2 (model size in # nodes)

D5 (name of initial topo file)

M1b (name of velocity txt file)

For your first run, do a run with dynamic time off **C1** and time step **C2** 100-200, orographic **A9** off; check to make sure you have uplift, surface velocities, the precipitation is approx what you want, etc.

6.10.2 How Big Should My Model Space Be? How Long Should My Model Run For?

In theory, you need a ~5-10km buffer on each end, the desired model view space at t=now, and (if you're not using dynamic remeshing) the amount of shortening.

However, as we learn more about the effects of legacy topography, it seems you should start (if possible) from a state where the surface is mostly flat (eg while the entire model view is foreland basin. In the initialization time, you may want to create a "false" topography with a slope of 0.1-0.5° (or from 0km to say, 500 or 800m over the model space) to allow CASCADE to create river channels. These will get overprinted by the true landscape (see Paul's paper).

The model space should ideally cover all topography you care about: so, if at t=now you want to see 100km of landscape, and you have 50km of shortening, the model space needs to be *at least* 150km. The model space *definitely* needs to include the "important" locations of uplift: so, a basal ramp, if you have one, the location of imposed uplift, etc.

Note: CASCADE only models from 0 to however big you say the model is. If your deformation front isn't at (or close to 0), you may need to shift the locations.

Andes run time/model space (VBuford, model McQ02N9v1-10): 50km x 200km, node spacing ~1km, ~16My run time

6.10.3 What Should I Change If I Change The Velocity?

C3 (model start time)

M1b (name of velocity txt file)

6.10.4 How Do I Fine Tune The Model?

Increase node spacing **D1** up to ~0.5km; make sure dynamic time **C1** is T and C2=25; turn on adaptive remeshing **D3**; change precipitation node size (caution, may crash) **N2**; increase frequency of model output **B4**.

6.10.5 What Controls Erosion?

Erosion is controlled by: **velocity** (so time for erosion to happen); **uplift** (ramps/loading & imposed); **precipitation** (discussed above); **fluvial and hillslope erosion** (so, E1-5, F1)

6.10.6 What Does "Error Sill" Mean?

This is a super generic error that basically means CASCADE couldn't find a river channel or catchment. This can be due to: (i) no initialization time, (ii) too strong of rocks, (iii) too fast velocity, (iv) not enough precipitation, (v) random error/bad luck/something we haven't figured out yet.

6.10.7 What Parameters Should I Vary?

Constant Precipitation (A9=F): constant precipitation amount is affected by parameters **N6** (atmospheric moisture a0) and **N18** (min and max sea level temperatures) (and to a small degree **E1**). It is not affected by changing N3 (uniform precip rate), N8 (wind speed), N13 (atmospheric lapse rate) or NN05 (mean annual temperature.

The figure(s) below shows the effects of various parameters on the output precipitation. These were run using "default" values (with the exception of N3. N3 default =3.0, oops, but it doesn't affect the results). The following two figures show how N6 and N18 can be varied to affect precipitation; it primarily seems that N6 should be for fine-tuning the amount, while N18 can control larger swings.

Effect of Cascade Parameters on Precipitation
For "Global" Precipitation (A9=F)

N3- Uniform Precip Rate			N8-Wind Speed			N13-Atm. lapse	
1.0	4.0	10	1.00	1.25	1.50	0.0065	5.3
0.6006	0.6006	0.6006	0.6006	0.6006	0.6006	0.6006	0.6006

NN05- Mean Annual Temperature				
7.9	10	15	20	25
0.6006	0.6006	0.6006	0.6006	0.6006

default
has effect

N6-Atmospheric Moisture			
3E-6	.03	.3	3
0.3858	0.4047	0.6006	3.7297

N18-Sea Level Temps					
		Min			
		0	5	10	15
max	6	0.6006			
	10	0.6924	0.8209		
	15	0.8244	0.9741	1.1472	
	20	0.9782	1.1519	1.3524	1.5828
	25	1.1567	1.3578	1.5891	1.8544

Figure 6.2. Variation in output precipitation based on changes in global precip parameters (ca. 2017)

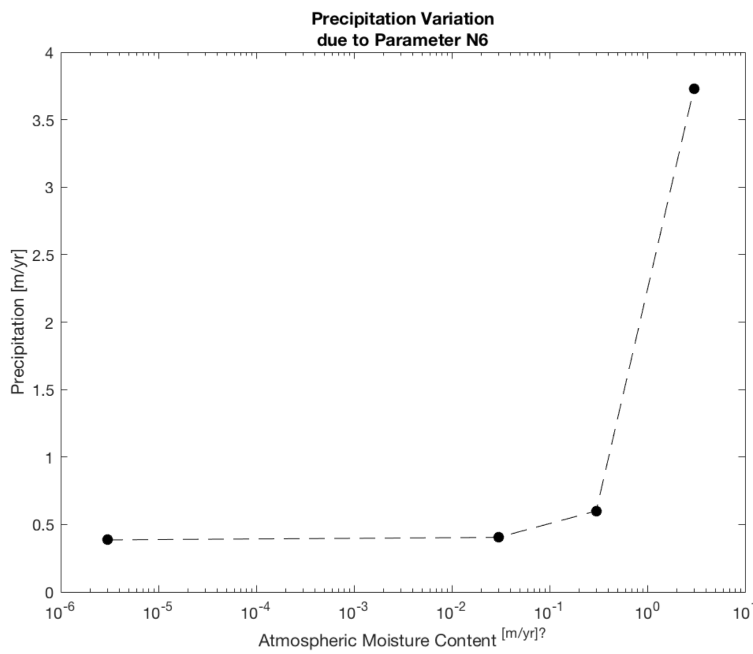


Figure 6.3. Plot of global precipitation variation based on parameter N6

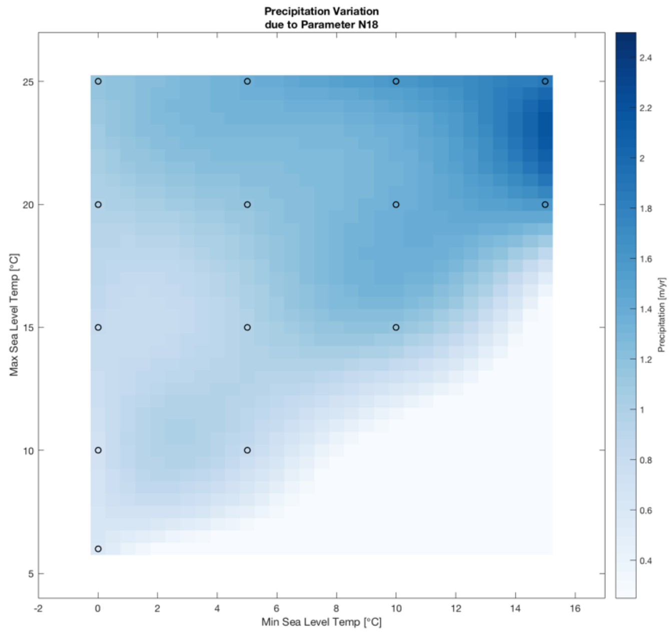


Figure 6.4. Global precipitation variation due to parameters N18

Orographic Precipitation (A9=T): In theory, affected by: **N3** (uniform precip rate), **N6** (atmospheric moisture), **N8** (wind speed), **N13** (atmospheric lapse rate), **N18** (Min and max sea level temperatures), **NN05** (Mean Annual Temperature) (Maybe also N14, N9, N10, N11, N12)

6.11 Note on New Precipitation Module

Though not yet uploaded to the main server, we are currently (as of 12/2020) testing the new 'rainmaker_imposed' module discussed within this dissertation. This new module alters input A9, and adds inputs A9B, and A9C in the icecascade.in file. Additionally, the current version of the executable adds an ability for the user to impose uplift on the rear boundary nodes (input D10) (which should work even if you choose to 'flip' the model with N9b.

Input **A9**: for imposed precipitation, select option '3'

A9B: This should be the filename for the precipitation file, setup like the velocity file. E.g. 'Precip1.txt'

Where Precip1.txt looks like:

```
10
5
10
1.0
9.5 P_95_Plio_Shift136km.dat
7.4 P_74_Plio_Shift36km.dat
6.4 P_64_Plio_Shift36km.dat
5.4 P_54_Plio_Shift16km.dat
4.3 P_43_Plio_Shift13km.dat
3.3 P_33_Plio_Shift6km.dat
2.4 P_24_Plio_MagShifted.dat
1.5 P_15_LGM_MagShifted.dat
0.0 P_NewModernPrecip.dat
```

And then ensure that all of the listed filenames in the right column, as well as Precip1.txt are contained within the model/input/IceCascade/ folder.

A9C: This is the update time (in years) for CASCADE to check whether the precipitation grid should have changed, or to reapply the precip values to the grid based on the lateral translation shift. The rule of thumb I go by is the shortening rate divided by A9C will be the amount of lateral translation allowed before reapplying the precip grid. For the Andes, I've used 100000 without real problem.

D10: This input controls the rear boundary. It is composed of three lines, where the first line is a T/F input, and the next two lines are real numbers in mm/y and km, respectively.

For example, to raise the rear boundary at 2.5 mm/y to a maximum of 4 km:

```
T
2.5
4
```

7.0 Appendix 3: Selected MATLAB Codes

These codes, as well as other selected codes and tutorials, are also available via GitHub here: https://github.com/vmbparks/PhD-Dissertation-Supplement/

7.1 Creation of Pecube Files

7.1.1 Create Pecube Grid for Move Deformation

Saved as ‘Create_MOVEPointCloud.m’.

```
function [MOVE_PointCloud] =...
   Create_MOVEPointCloud(x_max, x_min, z_max, z_min, resolution)
%   INPUT:
%   x_max, x_min, z_max, z_min: integers of bounding box in [km]
%   resolution: MOVE point cloud resolution in [km]

%   OUTPUT:
%   columns: x,y,z, color ID, unique ID
%
%   Paul R. Eizenhöfer, PhD
%   University of Pittsburgh, peizen@pitt.edu

[X,Z] = meshgrid(x_min:resolution:x_max, z_min:resolution:z_max);

MOVE_PointCloud(:,1) = X(:);
MOVE_PointCloud(:,2) = 0;
MOVE_PointCloud(:,3) = Z(:);
MOVE_PointCloud(:,4) = 0;
for i = 1:length(MOVE_PointCloud(:,3))
   MOVE_PointCloud(i, 5) = i;
end
```

7.1.2 Create Pecube Input Files

This file creates the input files for pecube of Pecube.in, velocity.txt, and run_pecube.sbatch. **Note: you must change Pecubein.txt to Pecube.in in your file manager before upload to the server.** Saved as ‘Pecube_CreateInputFiles.m’.

```
%% Creating Pecube.in, velocity.txt, and run_pecube.sbatch files
% Victoria Buford; May 10 2018
% vmb21@pitt.edu

% Have to change Pecubein.txt to Pecube.in outside of MATLAB
% - In Windows, Use Notepad ++ (NOT NOTEPAD); open Pecubein.txt:
% save as, Pecube.in
% - on Mac, right click on Pecubein.txt, click ?Get Info?. In the Name
% & Extension: blank, change the name to Pecube.in; when a dialog
% pops up, click ?Use .in?
%clc;clear;close;
%% Change per Move Model
% In user input now...
% ageinitiation=100; % should not be == ages(1)
% agemodelstart=50;

Xmin=0; % bottom left of start model
Xmax= 600;
%n=21; % number of deformation steps
% Xmin=350; % bottom left of start model
% Xmax= 1130;
% n=38; % number of deformation steps
email='vmb21@pitt.edu';

% don't normally change, but we can:
modelsizem=[500 1000]; % pecube pt grid spacing x and y;

% Which thermochronometers do you want....
% (a) AHe age (Default kinetics)
% (b) AHe age (20 um grain size)
% (c) AHe age (40 um grain size)
% (d) AHe age (70 um grain size)
% (e) AHe age (Low radiation damage)
% (f) AHe age (Moderate radiation damage)
% (g) AHe age (high radiation damage)
% (h) ZHe age
% (i) AFT age
% (j) ZFT age, D0 = 0.001_8, energy = 208.32768_8, grain_size = 3.158_8, geometry_factor = 55.0_8
% (k) Muscovite Ar/Ar age, D0 = 4.0e-8_8,  energy = 180.0_8, grain_size = 750.0_8, geometry_factor =
8.65_8
% (l) Ar in K-feldspar, D0 = 5.6, Ea = 120.0
% (m) Ar in Biotite, D0 = 160.0, Ea = 211.0
% (n) Ar in Muscovite, D0 = 13.2, Ea = 183.0
% (o) Ar in Hornblende, D0 = 14.0, Ea = 176.0
% (p) Apatite U-Th / Pb, D0 = 2.0e-8_8, energy = 230.0_8, grain_size = 50.0_8, geometry_factor = 55.0_8
% (q) Biotite, D0 = 2.0e-13_8, energy = 105.0_8, grain_size = 500.0_8, geometry_factor = 8.65_8
% (r) Ruite U-Pb, D0 = 1.6e-10_8, energy = 243.0_8, grain_size = 250.0_8, geometry_factor = 55.0_8
% (s) Titanite U-Pb, D0 = 1.1e-4_8, energy = 331.0_8, grain_size = 500.0_8, geometry_factor = 55.0_8
% (t) Zircon U-Pb, D0 = 7.8e-3_8, energy = 544.0_8, grain_size = 50.0_8, geometry_factor = 55.0_8
% (u) Titanite U-Th / He, D0 = 5.9e-3, energy = 188, grain_size = 250
% (v) ZHe, low damage D0z = 4.6
% (w) ZHe, med damage D0z = 0.3

AHea=1; %AHe
AHeb=0;
```

```
AHec=0;
AHed=0;
AHee=0;
AHef=0;
AHeg=0;
Zheh=1; % ZHe
AFTi=1; % AFT
ZFTj=1; % ZFT
MArArk=1; % MAr
ArKsparl=0;
ArBiom=1; %BAr
ArMuscn=0;
ArHorno=0;
ApUThp=0;
Bioq=0;
RuiUPbr=0;
TitUPbs=0;
ZirUPbt=0;
TitUThu=0;
ZHelowv=0;
ZHemedw=0;
ZHehighx=0;

ages=[];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Change per Pecube Model
modelname=input('ModelName (for Velocity_file_name): ','s');%'McQ02N9_VelVar5';
abbrevmodelname=input('Abbrev. Model name (8 char display in queue): ','s');%'PN9VV5a3ef12'; % only 8 characters display on server list
NodeAssign=input('Node to assign: 133,134,135,136,\n esd_node1, esd_node2, arduino, \n argand, or 0 for no assignment: ','s');%'0'; % Node to assign: '38', '39','40' or  '0' for no assignment
%%
ef=input('e-folding depth [km]: ');% 12;
ao=input('thermal heat production[muW/m3] (ao): ');% 3.0;
atmlapse=input('atmospheric lapse rate [C/km](4.16): ');% 5.3;
% Navarro-Serrano_etal_2020- S. Peru - 4.16
%%
%%%% Velocities!
VV6=[42.18  40.83   39.45   38.27   37.03   35.90   34.43   33.07 ...
    32.11   31.14   30.18   28.80   27.21   25.83   24.61   23.13   21.75 ...
    20.53   19.41   18.13   16.60   15.09   14.06   12.86   12.01   11.39 ...
    10.76   10.15   9.49    8.74    7.94    6.81    5.75    4.33    3.11 ...
    1.83    0.83    0.00];
ageinitiation=input('Age of thermal initiation: ');
% agemodelstart=input('Age of Step 0: ');
% x=input('Enter ages in brackets [] for Steps 1:end: ');
% sz=size(x);
```

```
% if sz(1)>sz(2)
%     ages=[ageinitiation;agemodelstart;x];
% else
%     ages=[ageinitiation;agemodelstart;x'];
% end

x=input('Enter ages in brackets [] for Steps 0:end: ');
sz=size(x);
if sz(1)>sz(2)
    ages=[ageinitiation;x];
else
    ages=[ageinitiation;x'];
end

%%
% get grid names
disp('Select Grid Files')
[gridfilename,gridfolder]=uigetfile('*.dat','Multiselect','on');
%%
disp('Select Topography Files')
[topofilename,topofolder]=uigetfile('*.dat','Multiselect','on');
%%
disp('Select Save Folder');
SaveFolder=uigetdir;
SaveFolder=strcat(SaveFolder,'/');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%
% modelname=input('ModelName (for Velocity_file_name): ','s');%'McQ02N9_VelVar5';
% abbrevmodelname=input('Abbrev. Model name (8 char display in queue): ','s');
% VV6=[42.18    40.83   39.45   38.27   37.03   35.90   34.43   33.07 ...
%     32.11 31.14   30.18   28.80   27.21   25.83   24.61   23.13   21.75 ...
%     20.53 19.41   18.13   16.60   15.09   14.06   12.86   12.01   11.39 ...
%     10.76 10.15   9.49    8.74    7.94    6.81    5.75    4.33    3.11 ...
%     1.83  0.83    0.00];
% ageinitiation=input('Age of thermal initiation: ');
% agemodelstart=input('Age of Step 0: ');
% x=input('Enter ages in brackets [] for Steps 1:end: ');
% sz=size(x);
% if sz(1)>sz(2)
%     ages=[ageinitiation;agemodelstart;x];
% else
%     ages=[ageinitiation;agemodelstart;x'];
% end
```

```
% disp('Save Folder')
% SaveFolder=uigetdir;
% SaveFolder=strcat(SaveFolder,'/');
%% Model Setup
velname=strcat(modelname,'.txt');
modelsizepts=[(Xmax-Xmin)*(1000/modelsizem(1)) 5];
%%%%%%%%%%% Thermal Parameters %%%%%%%%%%%%%%%%%%%%%%
% % (a) Model thickness (km)
% % (b) Number of z-node planes/layers in the z direction (integer)
% % NOTE: If this value is zero, Pecube will automatically define the z-node plane
% %       distribution such that the elements have a 1:1 (x/y to z) aspect ratio
% %       down to 5 km below the surface, 3:1 down to 15 km below the surface and
% %       ~9:1 down to the model base.
% % (c) Thermal conductivity (W/m K)
% % (d) Specific heat capacity (J/kg K)
% %   *NOTE: diffusivity is now caluclated in Pecube, rather than defined here*
% % (e) Crustal density (kg/m^3)
% % (f) Mantle density (kg/m^3)
% % ** SECOND LINE **
% % (g) Temperature at the base of the model (degrees C)
% % (h) Temperature at z=0 (degrees C)
% %   If lapse=0 this will be the surface temperature everywhere
% % (i) Atmospheric lapse rate (degrees C/km)
% %   NOTE: Positve lapse rate => decreasing T with elevation
% %         Negative lapse rate => increasing T with elevation
% % (j) Crustal volumetric heat production (uW/m^3)
% % (k) e-folding depth of crustal heat production (km)
% %   NOTE: Crustal heat production is constant at the given value for all nodes
% %     above sea level and decreases exponentially below msl. Also, if efold=0,
% %     then crustal heat production will be constant everywhere
% % (l) Mantle volumetric heat production (uW/m^3)
% %   NOTE: mantle HP not yet implemented - does nothing
% %         Also, mantle heat production is assumed to be constant
% % (m) Shear heating
% %  Set brittle shear heating constant below
% %  1 = on
% %  0 = off
% % (n) Shear heating constant (unitless)
% %   Scales shear heating within the brittle realm.
% %   Implemented in same form as used by F. Herman (02/08)
% %   1 = Full (unscaled) brittle shear heating
% %   0 = No brittle shear heating
therml1=[110 220 2.5 800 2500 3300];
therml2=[1300 23.0 atmlapse ao ef 0.01 0 0];

%%%%%%%%%%% Thermochronometers to plot %%%%%%%%%%%%%%%%%%%%%%

% % % (a) AHe age (Default kinetics)
% AHea=1;
% % % (b) AHe age (20 um grain size)
% AHeb=0;
% % % (c) AHe age (40 um grain size)
% AHec=0;
% % % (d) AHe age (70 um grain size)
% AHed=0;
```

```
% % % (e) AHe age (Low radiation damage)
% AHee=0;
% % % (f) AHe age (Moderate radiation damage)
% AHef=0;
% % % (g) AHe age (high radiation damage)
% AHeg=0;
% % % (h) ZHe age
% Zheh=1;
% % % (i) AFT age
% AFTi=1;
% % % (j) ZFT age, D0 = 0.001_8, energy = 208.32768_8, grain_size = 3.158_8, geometry_factor = 55.0_8
% ZFTj=1;
% % % (k) Muscovite Ar/Ar age, D0 = 4.0e-8_8,  energy = 180.0_8, grain_size = 750.0_8, geometry_factor
= 8.65_8
% MArArk=1;
% % % (l) Ar in K-feldspar, D0 = 5.6, Ea = 120.0
% ArKsparl=0;
% % % (m) Ar in Biotite, D0 = 160.0, Ea = 211.0
% ArBiom=0;
% % % (n) Ar in Muscovite, D0 = 13.2, Ea = 183.0
% ArMuscn=0;
% % % (o) Ar in Hornblende, D0 = 14.0, Ea = 176.0
% ArHorno=0;
% % % (p) Apatite U-Th / Pb, D0 = 2.0e-8_8, energy = 230.0_8, grain_size = 50.0_8, geometry_factor =
55.0_8
% ApUThp=0;
% % % (q) Biotite, D0 = 2.0e-13_8, energy = 105.0_8, grain_size = 500.0_8, geometry_factor = 8.65_8
% Bioq=0;
% % % (r) Ruite U-Pb, D0 = 1.6e-10_8, energy = 243.0_8, grain_size = 250.0_8, geometry_factor = 55.0_8
% RuiUPbr=0;
% % % (s) Titanite U-Pb, D0 = 1.1e-4_8, energy = 331.0_8, grain_size = 500.0_8, geometry_factor = 55.0_8
% TitUPbs=0;
% % % (t) Zircon U-Pb, D0 = 7.8e-3_8, energy = 544.0_8, grain_size = 50.0_8, geometry_factor = 55.0_8
% ZirUPbt=0;
% % % (u) Titanite U-Th / He, D0 = 5.9e-3, energy = 188, grain_size = 250
% TitUThu=0;
% % % (v) ZHe, low damage D0z = 4.6
% ZHelowv=0;
% % % (w) ZHe, med damage D0z = 0.3
% ZHemedw=0;
% % % (x) ZHe, high damage D0z = 4.6E+05
% ZHehighx=0;

tchrons=[AHea AHeb AHec AHed AHee AHef AHeg Zheh AFTi ZFTj MArArk ...
   ArKsparl ArBiom ArMuscn ArHorno ApUThp Bioq RuiUPbr TitUPbs ZirUPbt ...
   TitUThu ZHelowv ZHemedw ZHehighx];

%%%%%%%%%%%                                    check                                    size
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear TotalFiles
if length(topofilename)==length(gridfilename) && length(topofilename)==length(ages)-1
   TotalFiles=length(topofilename);
   n=length(topofilename)-1;
   fprintf('Good to go\n')
else
```

```
   errordlg('Topo, Ages, and Grid lengths do not match!')
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%          Create          Pecube          files
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%% create velocity.txt file
% on windows notepad, will need \r\n instead of \n
fidvel=fopen(strcat(SaveFolder,velname),'w');
nrows=TotalFiles;
fprintf(fidvel,'10\n');
fprintf(fidvel,'5\n');
fprintf(fidvel,'10\n');
fprintf(fidvel,'1.0\n');
for row=2:nrows+1
   fprintf(fidvel,'%2.2f %s\n', ages(row),gridfilename{row-1});
end
fclose(fidvel);

%%%%%%%%%%% Create txts to copy into Pecube.in %%%%%%%%%%
%% Pecube.in complete.
% - In Windows, Use Notepad ++ (NOT NOTEPAD); open Pecubein.txt:
% save as, Pecube.in
% - on Mac, right click on Pecubein.txt, click ?Get Info?. In the Name
% & Extension: blank, change the name to Pecube.in; when a dialog
% pops up, click ?Use .in?

% Pecube params we change
% input 1, 2,4,6,7?,9, 10, 12a, 14(thermal), 17 (thermochronometers),
% 22 (name of velocity file.

% Need to change name to pecube.in
%topofilelist=strcat(modelname,'_Pecubein.txt');
fidcomp=fopen(strcat(SaveFolder,'Pecubein.txt'),'w');
nrows=TotalFiles;

%%%% pecube.in writeup
fprintf(fidcomp,'$ ******************************************************************************\n');
fprintf(fidcomp,'$ *** Pecube-D\n');
fprintf(fidcomp,'$ *****\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Input file for running Pecube-D.  University of Tuebingen, Germany (18 May, 2015).\n');
fprintf(fidcomp,'$ Report bugs noticed with this version to Todd Ehlers (todd.ehlers@uni-tuebingen.de).\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ This version of Pecube is based on the distrubtion by Jean Braun.  It has been\n');
fprintf(fidcomp,'$ Modified substatially to account for\n');
fprintf(fidcomp,'$ a.  Calculation of predicted ages for different thermo- geochronometer systems.\n');
```

```
fprintf(fidcomp,'$ b.  Many different options for user defined velocity input fields (e.g. McQuarrie and\n');
fprintf(fidcomp,'$     Ehlers, 2015).\n');
fprintf(fidcomp,'$ c.  Different output format options.\n');
fprintf(fidcomp,'$ d.  calculation of detrital cooling ages for user defined sample points on the topog.\n');
fprintf(fidcomp,'$      (e.g. Whipp et al., 2009)\n');
fprintf(fidcomp,'$ e.  Iterative Inversion of cooling ages for topographic change scenarios (e.g. Olen et al.
2012)\n');
fprintf(fidcomp,'$ f.  Monte Carlo Inversion of cooling ages to identify the denudation histories that\n');
fprintf(fidcomp,'$     that can produce observed ages.  (e.g. Thiede & Ehlers, 2013)\n');
fprintf(fidcomp,'$ g.  Coupling with CASCADE or IceCascade (e.g Yanites and Ehlers, 2013 EPSL)\n');
fprintf(fidcomp,'$ and numerous other significant changes to age prediction, heat production, shear\n');
fprintf(fidcomp,'$ heating, kinematics, and thermal history output\n');
fprintf(fidcomp,'$ are highlighted in the readme folders in  /docs folder. Different options have also been
added\n');
fprintf(fidcomp,'$ for simulation other kinematic fields including an ellipsoidal exhumation field, as well as\n');
fprintf(fidcomp,'$ a coupling to 2D Move restoration files. Note - use of the 2D Move output for velocity
fields\n');
fprintf(fidcomp,'$ also requires the program velocity.py.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Significant program changes have been made to this version with thanks to Willi
Kappler,\n');
fprintf(fidcomp,'$ David Whipp, and Chris Spath.  If you use this program for publications the references
that\n');
fprintf(fidcomp,'$ describe the methods used are:.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Braun , J., 2003. Pecube: A new finite element code to solve the 3D heat\n');
fprintf(fidcomp,'$  transport equation including the effects of a time-varying, finite\n');
fprintf(fidcomp,'$  amplitude surface topography.  Computers and Geosciences, v.29, pp.787-794.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Braun , J., 2002. Quantifying the effect of recent relief changes on age-elevation\n');
fprintf(fidcomp,'$  relationships.  Earth and Planetary Science Letters, v.200, pp.331-343.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Reference to use concerning program changes made by T. Ehlers group:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Olen, S., Ehlers, T.A., Densmore, M.S., 2012, Limits to reconstructing
paleotopography\n');
fprintf(fidcomp,'$ from ther- mochronometer data, J. Geophysical Res ??? Earth Surface, v. 117,\n');
fprintf(fidcomp,'$ doi:10.1029/2011/ JF001985\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Whipp, D.M. Jr., Ehlers, T.A., Braun, J., Spath, C.D., 2009, Effects of exhumation\n');
fprintf(fidcomp,'$ kinematics and topo- graphic evolution on detrital thermochronometer data,\n');
fprintf(fidcomp,'$ J. Geophysical Res. ??? Earth Surface, V. 114, F04021, doi:10.1029/2008JF001195.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Thiede, R.C., Ehlers, T.A., 2013, Large spatial and temporal variations in Himalayan\n');
fprintf(fidcomp,'$ denudation, Earth and Planetary Science Letters, 371-372, pp. 278-293.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ McQuarrie, N., and Ehlers, T.A., 2015 Influence of thrust belt geometry\n');
fprintf(fidcomp,'$ and shortening rate on thermochronometer cooling ages: Insights from the\n');
fprintf(fidcomp,'$ Bhutan Himalaya, Tectonics. 34, doi:10.1002/2014TC003783.\n');
fprintf(fidcomp,'$ Related programs to this distribution:\n');
fprintf(fidcomp,'$      * Bivar - this program takes topography output from cascade and Ice and formats them
for input\n');
fprintf(fidcomp,'$       into Pecube-D\n');
fprintf(fidcomp,'$   * Cascade and Ice (Univ. Tuebingen modified versions, see Yanites and Ehlers, 2012 In
review).\n');
fprintf(fidcomp,'$       These programs can be run prior to\n');
```

```
fprintf(fidcomp,'$      to Pecube-D to provide input topographies.\n');
fprintf(fidcomp,'$      * velocity.py - used to take 2D Move output and create transient velocity fields\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ NOTE - not all of the above files may be in this distribution.  We''re in the process of\n');
fprintf(fidcomp,'$ assembling a more complete distribution.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ You can add as many comment lines as you wish as long as they start with a\n');
fprintf(fidcomp,'$ dollar sign\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ *** HOW TO RUN / EXECUTE THIS PROGRAM ***\n');
fprintf(fidcomp,'$ This version of Pecube is compiled with MPI so that it can run on multicore machines. This is not\n');
fprintf(fidcomp,'$ normally needed if you are doing a single simulation.  However, if you are using the\n');
fprintf(fidcomp,'$ Monte Carlo or Genetic search algorithm then it can run multiple jobs at once.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ The implications of this are that:\n');
fprintf(fidcomp,'$ 1. You have to have a version of MPI installed on your machine.  We are using OpenMPI, but others will likely\n');
fprintf(fidcomp,'$ work.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ 2. Compile the program using scons.  On our system this is done with scons --use-mpi, and then scons -c to clean\n');
fprintf(fidcomp,'$ out the .o files.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ 3. To run the program you need to have the pecube.in file in the same directory as the executable, and then\n');
fprintf(fidcomp,'$ start the job with mpi run.  For example, on our system we do:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$  mpirun -n pecube\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Where -n is the number of cores you want to use.  If you are only running 1 job, then there is likely no speed\n');
fprintf(fidcomp,'$ difference if N=1 or N=8.  So, for 1 job, you write\n');
fprintf(fidcomp,'$ mpirun -n 1 pecube.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ ******************************************************************************\n');

fprintf(fidcomp,'\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ Set mode of pecube operation\n');
fprintf(fidcomp,'$ Valid options:\n');
fprintf(fidcomp,'$ normal_mode: use this option if you just want to run pecube once\n');
fprintf(fidcomp,'$ error_iteration: use this option if you want to run a sequence of pecube simulation\n');
fprintf(fidcomp,'$      one after the other, in order to optimize the erosion rates iteravely\n');
fprintf(fidcomp,'$ monte_carlo: use this option if you want to run thousands of pecube simulation\n');
fprintf(fidcomp,'$      (some of them in parallel) in roder to optimize erosion rates using\n');
fprintf(fidcomp,'$      monte carlo randomisazion\n');
fprintf(fidcomp,'pecube_run_mode: normal_mode\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ If run mode is error iteration:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ Maximum number of error iteration\n');
fprintf(fidcomp,'error_iter_limit: 15\n');
```

```
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ Error tolerance to exit iteration before limit is reached\n');
fprintf(fidcomp,'$ error_iter_tolerance: 1.0\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ Flag whether observables should be created from pecube or not\n');
fprintf(fidcomp,'$ This is currently not working\n');
fprintf(fidcomp,'$ error_iter_create_observables: off\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ If run mode is monte carlo:\n');
fprintf(fidcomp,'$\n');

fprintf(fidcomp,'$ The maximum value for the randomized erosion rates\n');
fprintf(fidcomp,'mc_max_erosion_rate: 4.0\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ The total number of simulations\n');
fprintf(fidcomp,'$mc_num_of_simulations: 10000\n');
fprintf(fidcomp,'mc_num_of_simulations: 20000\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ The tolerance for chi squared\n');
fprintf(fidcomp,'mc_tolerance_chi_squared: 4.0\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ Flag whether to check the minimum threshold and correct it\n');
fprintf(fidcomp,'mc_check_min_threshold: on\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ The actual minimum threshold value\n');
fprintf(fidcomp,'mc_min_threshold_factor: 0.20\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ This factor sets the scale factor for the jitter that is added to\n');
fprintf(fidcomp,'$ each erosion rate values in order to randomize them\n');
fprintf(fidcomp,'$ Lower values means less jitter and higher chance to get stuck in\n');
fprintf(fidcomp,'$ a local minimum\n');
fprintf(fidcomp,'$ Higher values means more jitter and higher chance to miss an optimal value\n');
fprintf(fidcomp,'$ but also avoids to get stuck in a local minimum\n');
fprintf(fidcomp,'$ Useful ranges: 0.01 to 0.1\n');
fprintf(fidcomp,'$ If this value is set to >= 1.0, then evolutionary / genetic algorithm is disabled\n');
fprintf(fidcomp,'$mc_random_jitter_factor: 0.01\n');
fprintf(fidcomp,'mc_random_jitter_factor: 0.05\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ The input file containing the ages and errors\n');
fprintf(fidcomp,'$ This must be a text file with semicolon separated columns\n');
fprintf(fidcomp,'mc_csv_input_file: mhsxls.csv\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ IMPORTANT: This marks the end of the pecube mode configuration\n');
fprintf(fidcomp,'pecube_end_run_mode\n');
```

```
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ ********************************************************************************\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 1) Name of the run (also the name of the folder in which the solution is stored)\n');
fprintf(fidcomp,'$ NOTE: You might need to create this folder manually before running Pecube.\n');
fprintf(fidcomp,'$ AKA: #1:output folder \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'output/Pecube-D\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 2) Number of topography files to be loaded (should be number of time steps+1)\n');
fprintf(fidcomp,'$ 0 = No topography file will be loaded\n');
fprintf(fidcomp,'$ 1 = The same topo file will be used for all time steps. (note relief can still\n');
fprintf(fidcomp,'$     change as specified below). Note the number of time steps, or steps in the\n');
fprintf(fidcomp,'$     tectonomorphic scenario is defined later.\n');
fprintf(fidcomp,'$ >1 = A new topo file will be loaded for each timestep\n');
fprintf(fidcomp,'$ If fewer topo files are listed then the number of time steps, the model will\n');
fprintf(fidcomp,'$   use the last topo file for the subsequent/remaining time steps.\n');
fprintf(fidcomp,'$ When multiple topography files are loaded Pecube will exponentially morph\n');
fprintf(fidcomp,'$   between the two topographies over the given time step.  tau, specified below\n');
fprintf(fidcomp,'$   determines the exponential rate of change in the topography.\n');
fprintf(fidcomp,'$ AKA: #2: # of topo files \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%d\n',n+2);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 3) Flag for topography input\n');
fprintf(fidcomp,'$ 1 = User will list all topography file names below, one file on each new line\n');
fprintf(fidcomp,'$ 2 = User specifies file prefix (see Input 4) and Pecube will load all\n');
fprintf(fidcomp,'$     files with that prefix plus a 4 digit number after it.\n');
fprintf(fidcomp,'$ 3 = All listed topography files are exported from 2d move\n');
fprintf(fidcomp,'$ AKA: #3: Topo files from move? \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'3\n');
fprintf(fidcomp,'\n');


fprintf(fidcomp,'$ (Input 4) Name of the topo file used\n');
fprintf(fidcomp,'$ "Nil" = Topography is assumed to be flat for that time step\n');
fprintf(fidcomp,'$ Otherwise the file should contain nx by ny elevation points (see below)\n');
fprintf(fidcomp,'$    defining the topography in meters\n');
fprintf(fidcomp,'$ Note that the evolution of this topography (in amplitude and elevation offset)\n');
fprintf(fidcomp,'$    can change at each time step, as specified below in Input 12.\n');
fprintf(fidcomp,'$ If multiple topography files are being loaded with a user defined filename\n');
fprintf(fidcomp,'$    prefix filename (e.g., option 2 above) then format is:\n');
fprintf(fidcomp,'$    prefix = "topo_input" (or another user-defined name)\n');
fprintf(fidcomp,'$    which will load files topo_input0000.dat, topo_input0001.dat, etc.\n');
fprintf(fidcomp,'$ Note: If detrital age calculation for a Cascade mesh is specified (value of 1\n');
fprintf(fidcomp,'$       for Input 18) then the topo files must be named topo_pecube_0000.dat,\n');
fprintf(fidcomp,'$       topo_pecube_0001.dat, etc or the prefix topo_pecube_ if automating it\n');
fprintf(fidcomp,'$ AKA: #4: Topo files \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%s\n', topofilename{1});
```

```
for row=1:nrows;
   fprintf(fidcomp,'%s\n', topofilename{row});
end
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 5) Coordinate system flag for Pecube input\n');
fprintf(fidcomp,'$ 1 = Degrees\n');
fprintf(fidcomp,'$ 2 = UTM (meters)\n');
fprintf(fidcomp,'$ AKA: #5: Coord syst flag \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'2 \n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 6) Number of points (nx, ny) in the longitude and latitude directions\n');
fprintf(fidcomp,'$   of the topography file being loaded.\n');
fprintf(fidcomp,'$ Note: The shell script make_topo.sh will output this information to the screen\n');
fprintf(fidcomp,'$   if you are using this to create your topo files from ArcGIS grids\n');
fprintf(fidcomp,'$ AKA: #6: model size [points] \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%d %d \n',modelsizepts);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 7) Spacing of longitude and latitude points (in degrees or meters) in\n');
fprintf(fidcomp,'$   the topography input file.\n');
fprintf(fidcomp,'$ Note: The shell script make_topo.sh will also output this information if you\n');
fprintf(fidcomp,'$   use the script to export an ArcGIS DEM grid to Pecube format. Units of the\n');
fprintf(fidcomp,'$   values below should agree with what is specified in Input 5.\n');
fprintf(fidcomp,'$ AKA: #7: model size [m]\n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%d %d \n',modelsizem);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 8) Skipping factor (nskip) for points in the topo input file\n');
fprintf(fidcomp,'$ 1 = All points of the topography are used\n');
fprintf(fidcomp,'$ 2 = Every second point is used, etc.\n');
fprintf(fidcomp,'$ Note: nx, ny AND nskip define the resolution of the finite element mesh in the\n');
fprintf(fidcomp,'$   horizontal directions\n');
fprintf(fidcomp,'$ AKA: #8: skipping factor \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'1\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 9) Geographic location for the origin (bottom left corner) of the\n');
fprintf(fidcomp,'$   Pecube grid.\n');
fprintf(fidcomp,'$ Specify the longitude and latitude (in degrees or meters) of the bottom left\n');
fprintf(fidcomp,'$   corner of the topography file. Units must match above units.\n');
fprintf(fidcomp,'$ NOTE: a) You can set this value to be 0,0 for synthetic topography, or it\n');
fprintf(fidcomp,'$   can be 85670 (utm x), 983443 (utm y) or 109.756 (degrees long), 42.235\n');
fprintf(fidcomp,'$   (degrees lat) if you want Pecube to georeference the grid to your gegraphic\n');
fprintf(fidcomp,'$   area of study.\n');
fprintf(fidcomp,'$ NOTE: b) If you are using a DEM to generate the topography you want to specify\n');
fprintf(fidcomp,'$   an offset below that is 1/2 the topo file spacing specified in Input 7.\n');
fprintf(fidcomp,'$   (e.g., (DEM reolution / 2))\n');
fprintf(fidcomp,'$ AKA: #9: model start loc [m] \n'); % we DO change
fprintf(fidcomp,'\n');
```

```
fprintf(fidcomp,'%d %d \n', [Xmin*1000, 0]);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 10) Number of time steps in the tectonomorphic scenario for your\n');
fprintf(fidcomp,'$   simulation\n');
fprintf(fidcomp,'$ An integer number (>= 1) is required. The value should be 1 less than the\n');
fprintf(fidcomp,'$   number of time step inputs defined in Input 12 below.\n');
fprintf(fidcomp,'$ Examples: a value of 1 will require two input lines for Input 12 below (a line\n');
fprintf(fidcomp,'$   for the starting time condition and one for the final time step condition).\n');
fprintf(fidcomp,'$   A value of 2 below will require 3 lines in Input 12 below. In this case, the\n');
fprintf(fidcomp,'$   first line would be the starting time condition, the second line would be\n');
fprintf(fidcomp,'$   the condition at some intermediate time, and final (third) line would be the\n');
fprintf(fidcomp,'$   final model condition.\n');
fprintf(fidcomp,'$ AKA: #10: # of tect steps (#def+1) \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%d \n', n+1);
fprintf(fidcomp,'\n');
%
fprintf(fidcomp,'$ (Input 11) Erosional time scale (tau, in My) for topographic change\n');
fprintf(fidcomp,'$   This input allows the user to have non-linear morphing of topography with\n');
fprintf(fidcomp,'$   time.  A large value (e.g., 1000) will generate essentially linear changes\n');
fprintf(fidcomp,'$   between the input topography files.  Effectively, this is the e-folding time\n');
fprintf(fidcomp,'$   for the topographic evolution, a.k.a. the exponential decay rate of\n');
fprintf(fidcomp,'$   topography.\n');
fprintf(fidcomp,'$ AKA: #11: Erosional Time Scale \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'1000\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 12a) Definition of the tectonomorphic time steps\n');
fprintf(fidcomp,'$ NOTE: The number of lines should be 1 greater than the value specified in\n');
fprintf(fidcomp,'$   Input 10.\n');
fprintf(fidcomp,'$ Each line formatted as follows:\n');
fprintf(fidcomp,'$ (a) Time (in My in the past)\n');
fprintf(fidcomp,'$   NOTES: (i) The first time step (first line) calculates a steady state\n');
fprintf(fidcomp,'$     thermal solution with the prescribed parameters\n');
fprintf(fidcomp,'$   (ii) Any transient features will occur between the previous listed time step\n');
fprintf(fidcomp,'$     line and the current time step line. For example, for a model with 3 time\n');
fprintf(fidcomp,'$     steps, a 50% decrease in topographic relief and change in the velocity\n');
fprintf(fidcomp,'$     field desired in the final time step would be listed on the last two\n');
fprintf(fidcomp,'$     lines, where the desired final relief and velocity field over that time\n');
fprintf(fidcomp,'$     are listed on the final time step line\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (b) Amplification factor for relief change\n');
fprintf(fidcomp,'$   1 = static topography\n');
fprintf(fidcomp,'$   2 = 200% increase in relief over this time step\n');
fprintf(fidcomp,'$   0.5 = 50% decrease in relief over this time step\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (c) Vertical offset factor (in km) for static topography elevation shifts\n');
fprintf(fidcomp,'$   during simulation.\n');
fprintf(fidcomp,'$   0 = No shift in surface elevations\n');
fprintf(fidcomp,'$   2 = Increase in all surface elevations by 2 km over this time step\n');
fprintf(fidcomp,'$   Why would you use this?  Well, if relief is 2 km, with a mean of 1 km, and\n');
fprintf(fidcomp,'$     relief is decreasing by 50% then if you specify a value of 0.5 (km) here\n');
fprintf(fidcomp,'$     it would shift your mean elevation such that it would remain at 1 km.\n');
```

```
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (d) Flag for output of time-temperature histories\n');
fprintf(fidcomp,'$   Enabling this will output temperature, time, x, y and z positions for all\n');
fprintf(fidcomp,'$     surface points at each step where listed.\n');
fprintf(fidcomp,'$   0 = No output of thermal history at the time step\n');
fprintf(fidcomp,'$   1 = Output of thermal history at the time step\n');
fprintf(fidcomp,'$   Note: Because the first time step is a steady state calculation, there is no\n');
fprintf(fidcomp,'$     thermal history available for the first time step.\n');
fprintf(fidcomp,'$   If the entire thermal history is wanted for surface points at t = 0Ma, then\n');
fprintf(fidcomp,'$     the user should set this flag for thermal output at the last time step\n');
fprintf(fidcomp,'$     specified below\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (e) Kinematic field flag (details of kinematic field specified in subsequent\n');
fprintf(fidcomp,'$   inputs)\n');
fprintf(fidcomp,'$   1 = vertical movement (erosion only)\n');
fprintf(fidcomp,'$   2 = uniform diagonal movement\n');
fprintf(fidcomp,'$   3 = listric fault\n');
fprintf(fidcomp,'$   4 = New Nepal thrust belt model for rotated model (Whipp testing - 10/07)\n');
fprintf(fidcomp,'$   5 = Parabolic uplift field (S. Olen testing - 06/10)\n');
fprintf(fidcomp,'$   6 = Ellipsoid uplift field (M. Schmiddunser - 07/11)\n');
fprintf(fidcomp,'$       An inner and outer ellipse and the uplift rates for the three corresponding\n');
fprintf(fidcomp,'$       areas are specified. The uplift rate between the inner and outer ellipse will\n');
fprintf(fidcomp,'$       be interpolated between the inner and the outer uplift rate\n');
fprintf(fidcomp,'$   7 = As 6, but uses a different function for uplift rate calculation that produces\n');
fprintf(fidcomp,'$       a smoothed uplift profile\n');
fprintf(fidcomp,'$   8 = velocity file names\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (f) Details of kinematics (Peklet)\n');
fprintf(fidcomp,'$   If e=1, value here is the erosion rate (mm/yr)\n');
fprintf(fidcomp,'$   If e=2, value here is the magnitude of the velocity vector at which material\n');
fprintf(fidcomp,'$     is moving laterally (mm/yr) in an Eulerian reference frame\n');
fprintf(fidcomp,'$   If e=3, value here is the maximum slip velocity on fault (mm/yr)\n');
fprintf(fidcomp,'$   If e=4, enter 1, values for velocities are computed within code and scaled\n');
fprintf(fidcomp,'$     by 1 here\n');
fprintf(fidcomp,'$   If e=5, enter 1, values for velocities are based on maximum and minimum\n');
fprintf(fidcomp,'$           velocities defined at (k) and (l)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter value for uplift rate inside the inner ellipse (mm/yr)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ ADDITIONAL OPTIONAL PARAMETERS (depending on kinematic field used)\n');
fprintf(fidcomp,'$ (g)\n');
fprintf(fidcomp,'$   If e=1, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=2, enter fault dip angle theta (degrees). This is the angle from\n');
fprintf(fidcomp,'$     horizontal (positive down) defining the dip of the velocity vector.\n');
fprintf(fidcomp,'$   If e=3, enter the longitude or utm x position of one endpoint of the listric\n');
fprintf(fidcomp,'$     fault trace.\n');
fprintf(fidcomp,'$   Note: If you think of traveling along a line that starts at the first point\n');
fprintf(fidcomp,'$     and ends at the second, the fault would dip off to the left of that line\n');
fprintf(fidcomp,'$   Note:  This value must be entered in km (S. Olen, 06/21/2010)\n');
fprintf(fidcomp,'$   If e=4, enter the horizontal convergence rate (mm/yr) across the Main\n');
fprintf(fidcomp,'$     Frontal Thrust. Note: Fault geometries are hard coded in Pecube\n');
fprintf(fidcomp,'$   If e=5, enter the x-value for the lower right point of the parabola axis (km)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter the x-value for the first focus (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (h)\n');
fprintf(fidcomp,'$   If e=1, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=2, enter angle phi (degrees), the azimuth of the velocity vector in the\n');
```

```
fprintf(fidcomp,'$     x-y plane.\n');
fprintf(fidcomp,'$   If e=3, enter the latitude or utm y position of the first endpoint of the\n');
fprintf(fidcomp,'$     listric fault trace in item (g) above.\n');
fprintf(fidcomp,'$   Note:  This value must be entered in km (S. Olen, 06/21/2010)\n');
fprintf(fidcomp,'$   If e=4, enter the horizontal convergence rate (mm/yr) across the Main\n');
fprintf(fidcomp,'$     Boundary Thrust\n');
fprintf(fidcomp,'$   If e=5, enter the y-value for the lower right point of the parabola axis (km)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter the y-value for the first focus (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (i)\n');
fprintf(fidcomp,'$   If e=1 or e=2, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=3, enter the longitude or utm x of the second end point of the listric\n');
fprintf(fidcomp,'$     fault.\n');
fprintf(fidcomp,'$   Note:  This value must be entered in km (S. Olen, 06/21/2010)\n');
fprintf(fidcomp,'$   If e=4, enter the horizontal convergence rate (mm/yr) across the Main\n');
fprintf(fidcomp,'$     Central Thrust\n');
fprintf(fidcomp,'$   If e=5, enter the x-value for the upper left point of the parabola axis (km)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter the x-value for the second focus (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (j)\n');
fprintf(fidcomp,'$   If e=1 or 2, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=3, enter the latitutde or utm y of the second endpoint of the listric\n');
fprintf(fidcomp,'$     fault.\n');
fprintf(fidcomp,'$   Note:  This value must be entered in km (S. Olen, 06/21/2010)\n');
fprintf(fidcomp,'$   If e=4, enter the horizontal extension rate (mm/yr) across the South Tibetan\n');
fprintf(fidcomp,'$     Detachment\n');
fprintf(fidcomp,'$   If e=5, enter the y-value for the upper left point of the parabola axis (km)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter the y-value for the second focus (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (k)\n');
fprintf(fidcomp,'$   If e=1 or 2, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=3, enter the soling depth (km) of the fault. Note: Fault has an\n');
fprintf(fidcomp,'$     exponential shape.\n');
fprintf(fidcomp,'$   If e=4, enter 0 or 1 for whether or not you want underplating in the Sub-\n');
fprintf(fidcomp,'$     Himalaya during this time step (0=no; 1=yes)\n');
fprintf(fidcomp,'$   If e=5, enter the minimum velocity (mmyr-1)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter semi-major axis of inner elipse (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (l)\n');
fprintf(fidcomp,'$   If e=1 or 2, enter 0, no additional input required\n');
fprintf(fidcomp,'$   If e=3, enter surface dip angle of the fault in degrees\n');
fprintf(fidcomp,'$   If e=4, enter 0 or 1 for whether or not you want underplating in the Lesser\n');
fprintf(fidcomp,'$     Himalaya during this time step (0=no; 1=yes)\n');
fprintf(fidcomp,'$   If e=5, enter the maximum velocity (mmyr-1)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter semi-major axis of outer elipse (in km)\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ (m)\n');
fprintf(fidcomp,'$   If e=1,2,4,5, enter 0., no additonal input required\n');
fprintf(fidcomp,'$   if e=3, enter additional uplift rate (mm/yr, z velocity)\n');
fprintf(fidcomp,'$   If e=6 or e=7, enter uplift rate (mm/yr) outside of outer ellipse.\n');
fprintf(fidcomp,'$       The uplift rate between inner and outer ellipse will be interpolated\n');
fprintf(fidcomp,'$        between the two values$ (Input 12 b) If e=8 in Input 12 above, this defines the min
and max\n');
fprintf(fidcomp,'$ AKA: #12a: Tect Time Steps \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$(a)   (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m)\n');
```

```
params=[1.  0.  1.  8  1.0 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.];
%agelist=strcat(modelname,'_age.txt');
%nrows=TotalFiles;
formatspec='%2.2f \t %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f   %1.0f
%1.0f  %1.0f  %1.0f  %1.0f\n';
for row=1:nrows+1;
   fprintf(fidcomp,formatspec, ages(row),params);
end

fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ (Input 12 b) If e=8 in Input 12 above, this defines the min and max\n');
fprintf(fidcomp,'$ AKA: #12b: allowed velocities \n'); % we don't change
fprintf(fidcomp,'$ allowed velocity values in [mm/years]\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ vx_min vx_max\n');
fprintf(fidcomp,'-100.0 100.0\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ vy_min vy_max\n');
fprintf(fidcomp,'-100.0 100.0\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$ vz_min vz_max\n');
fprintf(fidcomp,'-100.0 100.0\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 13) Isostasy (NOTE: All values listed on one line)\n');
fprintf(fidcomp,'$  Remove input (b) and (c), which are included in input 14... necessary\n');
fprintf(fidcomp,'$  for isostacy to turn on (S. Olen, 1/20/2010)\n');
fprintf(fidcomp,'$ (a) Flag for isostasy\n');
fprintf(fidcomp,'$   1 = isostasy on\n');
fprintf(fidcomp,'$   0 = isostasy off\n');
fprintf(fidcomp,'$ (b) Crustal density (kg/m^3)  ***Remove***\n');
fprintf(fidcomp,'$ (c) Mantle density (kg/m^3)  ***Remove***\n');
fprintf(fidcomp,'$ (d) Young modulus (Pa)\n');
fprintf(fidcomp,'$ (e) Poisson''s ratio\n');
fprintf(fidcomp,'$ (f) Elastic plate thickness (*km*)\n');
fprintf(fidcomp,'$ (g) Size of the FFT grid for elastic rebound calculations (typically 1024 1024\n');
fprintf(fidcomp,'$   but must be a power of 2)\n');
fprintf(fidcomp,'$ AKA: #13: Isostasy \n') ;% we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'0 1.d11 0.25 15. 1024 1024\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 14) Thermal model input parameters\n');
fprintf(fidcomp,'$ NOTE: Values on two lines\n');
fprintf(fidcomp,'$ Note: Pecube currently assumes homogeneous medium.\n');
fprintf(fidcomp,'$ ** FIRST LINE **\n');
fprintf(fidcomp,'$ (a) Model thickness (km)\n');
fprintf(fidcomp,'$ (b) Number of z-node planes/layers in the z direction (integer)\n');
fprintf(fidcomp,'$ NOTE: If this value is zero, Pecube will automatically define the z-node plane\n');
fprintf(fidcomp,'$      distribution such that the elements have a 1:1 (x/y to z) aspect ratio\n');
fprintf(fidcomp,'$      down to 5 km below the surface, 3:1 down to 15 km below the surface and\n');
fprintf(fidcomp,'$      ~9:1 down to the model base.\n');
fprintf(fidcomp,'$ (c) Thermal conductivity (W/m K)\n');
fprintf(fidcomp,'$ (d) Specific heat capacity (J/kg K)\n');
fprintf(fidcomp,'$   *NOTE: diffusivity is now caluclated in Pecube, rather than defined here*\n');
```

```
fprintf(fidcomp,'$ (e) Crustal density (kg/m^3)\n');
fprintf(fidcomp,'$ (f) Mantle density (kg/m^3)\n');
fprintf(fidcomp,'$ ** SECOND LINE **\n');
fprintf(fidcomp,'$ (g) Temperature at the base of the model (degrees C)\n');
fprintf(fidcomp,'$ (h) Temperature at z=0 (degrees C)\n');
fprintf(fidcomp,'$   If lapse=0 this will be the surface temperature everywhere\n');
fprintf(fidcomp,'$ (i) Atmospheric lapse rate (degrees C/km)\n');
fprintf(fidcomp,'$   NOTE: Positve lapse rate => decreasing T with elevation\n');
fprintf(fidcomp,'$         Negative lapse rate => increasing T with elevation\n');
fprintf(fidcomp,'$ (j) Crustal volumetric heat production (uW/m^3)\n');
fprintf(fidcomp,'$ (k) e-folding depth of crustal heat production (km)\n');
fprintf(fidcomp,'$   NOTE: Crustal heat production is constant at the given value for all nodes\n');
fprintf(fidcomp,'$     above sea level and decreases exponentially below msl. Also, if efold=0,\n');
fprintf(fidcomp,'$     then crustal heat production will be constant everywhere\n');
fprintf(fidcomp,'$ (l) Mantle volumetric heat production (uW/m^3)\n');
fprintf(fidcomp,'$   NOTE: mantle HP not yet implemented - does nothing\n');
fprintf(fidcomp,'$         Also, mantle heat production is assumed to be constant\n');
fprintf(fidcomp,'$ (m) Shear heating\n');
fprintf(fidcomp,'$  Set brittle shear heating constant below\n');
fprintf(fidcomp,'$  1 = on\n');
fprintf(fidcomp,'$  0 = off\n');
fprintf(fidcomp,'$ (n) Shear heating constant (unitless)\n');
fprintf(fidcomp,'$   Scales shear heating within the brittle realm.\n');
fprintf(fidcomp,'$   Implemented in same form as used by F. Herman (02/08)\n');
fprintf(fidcomp,'$   1 = Full (unscaled) brittle shear heating\n');
fprintf(fidcomp,'$   0 = No brittle shear heating\n');
fprintf(fidcomp,'$ AKA: #14: Thermal Params \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%d %d %2.1f %d %d %d \n',therml1);
fprintf(fidcomp,'%d %2.2f %2.2f %1.2f %2.1f %.3f %d %d \n',therml2);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 15) Thermal model input parameters for Nepal model geometry\n');
fprintf(fidcomp,'$   **NOTE** This is not used unless the geometry flag above is set to 4.\n');
fprintf(fidcomp,'$   On each line, there are five values. The values are for the Indian Shield,\n');
fprintf(fidcomp,'$   Sub-Himalaya, Lesser Himalaya, Greater Himalaya and Tethyan Himalaya\n');
fprintf(fidcomp,'$   Line 1: Volumetric heat production (uW/m^3)\n');
fprintf(fidcomp,'$   Line 2: Thermal conductivity (W/m K)\n');
fprintf(fidcomp,'$   Line 3: Rock density (kg/m^3)\n');
fprintf(fidcomp,'$   Line 4: Specific heat capacity (J/kg K)\n');
fprintf(fidcomp,'$ AKA: #15:Nepal Thermal \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'0.8 0.8 0.8 1.9 0.8 \n');
fprintf(fidcomp,'2.75 2.75 2.75 2.75 2.75 \n');
fprintf(fidcomp,'2700. 2700. 2700. 2700. 2700. \n');
fprintf(fidcomp,'1000. 1000. 1000. 1000. 1000. \n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 16) Option to read in thermochron data and compare to predicted ages\n');
fprintf(fidcomp,'$ First specify the number of data files for comparison\n');
fprintf(fidcomp,'$ 0 = No data file(s) will be read in\n');
fprintf(fidcomp,'$ For each file name that is specified, the file format should be as follows:\n');
fprintf(fidcomp,'$ First line in file = number of samples (and lines) in rest of file.\n');
fprintf(fidcomp,'$ Each line after that is for an individual sample and should contain (space\n');
fprintf(fidcomp,'$   separated)\n');
```

```
fprintf(fidcomp,'$ (a) Sample longitude or utm x (value in km)\n');
fprintf(fidcomp,'$ (b) Sample latitude or utm y (value in km)\n');
fprintf(fidcomp,'$ (c) Sample elevation\n');
fprintf(fidcomp,'$ (d) Flag for type of AHe age to predict:\n');
fprintf(fidcomp,'$   1=Default diffusion kinetics; 2-4=Use grain size of 20, 40 or 70 um, resp.\n');
fprintf(fidcomp,'$   5-7=Use low, moderate or high eU (radiation damage) values resp.\n');
fprintf(fidcomp,'$   (Schuster et al., 2006)\n');
fprintf(fidcomp,'$   NOTE: These values can be modifed in the Mad_He.f90 subroutine\n');
fprintf(fidcomp,'$   Comments in that subroutine further explain the differences above\n');
fprintf(fidcomp,'$ (e) AHe age (Ma), negative age if non-existant\n');
fprintf(fidcomp,'$ (f) AHe age error, 1s.d. (Ma), use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (g) AFT age (Ma), negative age if non-existant\n');
fprintf(fidcomp,'$ (h) AFT age error, 1s.d. (Ma), use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (i) ZHe age (Ma), negative age if non-existant\n');
fprintf(fidcomp,'$ (j) ZHe age error, 1s.d. (Ma), use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (k) ZFT age (Ma), negative age if non-existant\n');
fprintf(fidcomp,'$ (l) ZFT age error, 1s.d. (Ma), use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (m) MAr age (Ma), negative age if non-existant\n');
fprintf(fidcomp,'$ (n) MAr age error, 1s.d. (Ma), use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (o) Ar in K-feldspar, negative age if non-existant\n');
fprintf(fidcomp,'$ (p) Ar in K-feldspar, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (q) Ar in Biotite, negative age if non-existant\n');
fprintf(fidcomp,'$ (r) Ar in Biotite, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (s) Ar in Muscovite, negative age if non-existant\n');
fprintf(fidcomp,'$ (t) Ar in Muscovite, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (u) Ar in Hornblende, negative age if non-existant\n');
fprintf(fidcomp,'$ (v) Ar in Hornblende, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (w) Apatite U-Th / Pb, negative age if non-existant\n');
fprintf(fidcomp,'$ (x) Apatite U-Th / Pb, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (y) Biotite, negative age if non-existant\n');
fprintf(fidcomp,'$ (z) Biotite, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (a1) Ruite U-Pb, negative age if non-existant\n');
fprintf(fidcomp,'$ (b1) Ruite U-Pb, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (c1) Titanite U-Pb, negative age if non-existant\n');
fprintf(fidcomp,'$ (d1) Titanite U-Pb, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (e1) Zircon U-Pb, negative age if non-existant\n');
fprintf(fidcomp,'$ (f1) Zircon U-Pb, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (g1) Titanite U-Th, negative age if non-existant\n');
fprintf(fidcomp,'$ (h1) Titanite U-Th, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (i1) ZHe, low damage, negative age if non-existant\n');
fprintf(fidcomp,'$ (j1) ZHe, low damage, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (k1) ZHe, med damage, negative age if non-existant\n');
fprintf(fidcomp,'$ (l1) ZHe, med damage, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (m1) ZHe, high damage, negative age if non-existant\n');
fprintf(fidcomp,'$ (n1) ZHe, high damage, use 0 if previous value is negative\n');
fprintf(fidcomp,'$ (o1) Sample ID\n');
fprintf(fidcomp,'$ AKA: #16:Read Tchron? \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'0 \n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ cstmt_ahe_errinc.dat\n');
fprintf(fidcomp,'$ cstmt_aft.dat\n');

fprintf(fidcomp,'$ NOTE: Input 17 and 18 are used for detrital age calculation.  Instructions\n');
```

```
fprintf(fidcomp,'$ on use are in 00README_detrital_ages file if user needs more information.\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 17) Flags for which ages to output\n');
fprintf(fidcomp,'$ 0 = Does not calculate or output predicted ages for this system\n');
fprintf(fidcomp,'$ 1 = Calculates and outputs specified system''s ages\n');
fprintf(fidcomp,'$ NOTE: See Mad_He.f90 subroutine to modify the predicted AHe ages below\n');
fprintf(fidcomp,'$ (a) AHe age (Default kinetics)\n');
fprintf(fidcomp,'$ (b) AHe age (20 um grain size)\n');
fprintf(fidcomp,'$ (c) AHe age (40 um grain size)\n');
fprintf(fidcomp,'$ (d) AHe age (70 um grain size)\n');
fprintf(fidcomp,'$ (e) AHe age (Low radiation damage)\n');
fprintf(fidcomp,'$ (f) AHe age (Moderate radiation damage)\n');
fprintf(fidcomp,'$ (g) AHe age (high radiation damage)\n');
fprintf(fidcomp,'$ (h) ZHe age\n');
fprintf(fidcomp,'$ (i) AFT age\n');
fprintf(fidcomp,'$ (j) ZFT age, D0 = 0.001_8, energy = 208.32768_8, grain_size = 3.158_8, geometry_factor = 55.0_8\n');
fprintf(fidcomp,'$ (k) Muscovite Ar/Ar age, D0 = 4.0e-8_8,  energy = 180.0_8, grain_size = 750.0_8, geometry_factor = 8.65_8\n');
fprintf(fidcomp,'$ (l) Ar in K-feldspar, D0 = 5.6, Ea = 120.0\n');
fprintf(fidcomp,'$ (m) Ar in Biotite, D0 = 160.0, Ea = 211.0\n');
fprintf(fidcomp,'$ (n) Ar in Muscovite, D0 = 13.2, Ea = 183.0\n');
fprintf(fidcomp,'$ (o) Ar in Hornblende, D0 = 14.0, Ea = 176.0\n');
fprintf(fidcomp,'$ (p) Apatite U-Th / Pb, D0 = 2.0e-8_8, energy = 230.0_8, grain_size = 50.0_8, geometry_factor = 55.0_8\n');
fprintf(fidcomp,'$ (q) Biotite, D0 = 2.0e-13_8, energy = 105.0_8, grain_size = 500.0_8, geometry_factor = 8.65_8\n');
fprintf(fidcomp,'$ (r) Ruite U-Pb, D0 = 1.6e-10_8, energy = 243.0_8, grain_size = 250.0_8, geometry_factor = 55.0_8\n');
fprintf(fidcomp,'$ (s) Titanite U-Pb, D0 = 1.1e-4_8, energy = 331.0_8, grain_size = 500.0_8, geometry_factor = 55.0_8\n');
fprintf(fidcomp,'$ (t) Zircon U-Pb, D0 = 7.8e-3_8, energy = 544.0_8, grain_size = 50.0_8, geometry_factor = 55.0_8\n');
fprintf(fidcomp,'$ (u) Titanite U-Th / He, D0 = 5.9e-3, energy = 188, grain_size = 250\n');
fprintf(fidcomp,'$ (v) ZHe, low damage D0z = 4.6\n');
fprintf(fidcomp,'$ (w) ZHe, med damage D0z = 0.3\n');
fprintf(fidcomp,'$ (x) ZHe, high damage D0z = 4.6E+05\n');
fprintf(fidcomp,'$ AKA: #17: which thermochronometers \n'); % we DO change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$(a)(b)(c)(d)(e)(f)(g)(h)(i)(j)(k)(l)(m)(n)(o)(p)(q)(r)(s)(t)(u)(v)(w)(x)\n');
formatspec2='%1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f %1.0f \n';
fprintf(fidcomp,formatspec2,tchrons);
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 18) Flag to calculate detrital age distributions for catchments\n');
fprintf(fidcomp,'$ First line:  0 = no detrital calculation; 1 = detrital calcuation\n');
fprintf(fidcomp,'$ Note:  If a series of CASCADE topographies were loaded in then\n');
fprintf(fidcomp,'$ set Input 18 of the Pecube.in file to be 1, Pecube will output the detrital\n');
fprintf(fidcomp,'$ ages of every cascade catchment at every timestep.  These files will be created\n');
fprintf(fidcomp,'$ in the ''catchments'' folder within your output run directory.  It will create the\n');
fprintf(fidcomp,'$ ''catchments'' folder if it does not exist there.  The files will be named as\n');
fprintf(fidcomp,'$ ''Timestep_0001_Catchment_0001.dat'' and so on for all catchments and timesteps.\n');
fprintf(fidcomp,'$ All the program needs to run properly is to have the tecplot formatted cascade\n');
```

```
fprintf(fidcomp,'$ output files for every timestep (eg. "topo_tec_0001.dat") in\n');
fprintf(fidcomp,'$ the "output/Cascade" directory\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Specifying user defined basins in a file causes Pecube to open the file and read it.\n');
fprintf(fidcomp,'$ The file should contain lines with the following syntax:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ <xpos> <ypos> <age type> <Nil or filename> <yes/Yes or no/No>\n');
fprintf(fidcomp,'$ Ex: 64.8495 128.5548 1 Nil No\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Where xpos is the x value of the basin outlet and ypos is the y value.  The age type is a number 1-11 with the\n');
fprintf(fidcomp,'$ following coding:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ 1 = Apatite Helium Age - Farley, 2000\n');
fprintf(fidcomp,'$ 2 = Apatite Helium Age - Small grain size\n');
fprintf(fidcomp,'$ 3 = Apatite Helium Age - Medium grain size\n');
fprintf(fidcomp,'$ 4 = Apatite Helium Age - Large grain size\n');
fprintf(fidcomp,'$ 5 = Apatite Helium Age - Low radiation damage\n');
fprintf(fidcomp,'$ 6 = Apatite Helium Age - Medium radiation damage\n');
fprintf(fidcomp,'$ 7 = Apatite Helium Age - High radiation damage\n');
fprintf(fidcomp,'$ 8 = Apatite Fission Track Age\n');
fprintf(fidcomp,'$ 9 = Zircon Helium Age\n');
fprintf(fidcomp,'$ 10 = Zircon Fission Track Age\n');
fprintf(fidcomp,'$ 11 = Muscovite Age\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ The next entry for each basin line can be "Nil" or a filename.  If "Nil" is specified, then Pecube uses\n');
fprintf(fidcomp,'$ the TAPES-Grid method of finding upstream points of the basin outlet.  Then, it will write out the x, y,\n');
fprintf(fidcomp,'$ and z positions along with the age data for the upstream points into the main run output directory with\n');
fprintf(fidcomp,'$ the naming convention of "Timestep_0001_Basin_X_64.8495_Y_128.5548.dat".  Also, the PDF for each basin is\n');
fprintf(fidcomp,'$ written to the folder "pdf_data" within the run output directory with the naming convention of\n');
fprintf(fidcomp,'$ "Timestep_0001_Basin_X_64.8495_Y_128.5548_Agetype_01_pdf.dat". If a filename is specified as the entry\n');
fprintf(fidcomp,'$ on the line, then Pecube will open this file and read in each line of data with the format:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ <age> <error>\n');
fprintf(fidcomp,'$ Ex: 35.756469 1.07269407\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Where the age and error are absolute values. The PDF for each of these basins is created in the "pdf_data"\n');
fprintf(fidcomp,'$ folder in the run output directory with the naming convention of "Basin_X_64.8495_Y_128.5548_Agetype_01_pdf.dat".\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ The final entry on each line is a "yes" or "no" on whether the user wants to run the Monte Carlo test for that specified\n');
fprintf(fidcomp,'$ basin.  For the Monte Carlo test to run properly there must be two basins with the same outlet point (x and y positions) and\n');
fprintf(fidcomp,'$ EACH must have a "yes" to run the monte carlo routine. Also, the age types MUST be the same or the Monte Carlo test is not run.\n');
fprintf(fidcomp,'$ An example of correct syntax for the Monte Carlo routine to run properly is as follows:\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ 64.8495 128.5548 7 Nil Yes\n');
```

```
fprintf(fidcomp,'$ 64.8495 128.5548 7 datafile.txt Yes\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Please note that if any or all of these criteria for running the Monte Carlo test are not met, then the program simple does not\n');
fprintf(fidcomp,'$ run the comparison (skips it) and does not output anything for it.\n');
fprintf(fidcomp,'$\n');
fprintf(fidcomp,'$ Note: The subdirectories in the run output directory where Pecube writes most of these files will be\n');
fprintf(fidcomp,'$      automatically created by Pecube if they do not exist already.\n');
fprintf(fidcomp,'\n');
fprintf(fidcomp,'$pdf_tester_for_data.txt\n');
fprintf(fidcomp,'$ AKA: #18: Detrital Age Flag \n'); % we don't change
fprintf(fidcomp,'0\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 19) Minimum number of nodes for a catchment to be output\n');
fprintf(fidcomp,'$ This is a threshold value of nodes that a catchment needs to have in order for an\n');
fprintf(fidcomp,'$   output file to be written for that catchment at that timestep.\n');
fprintf(fidcomp,'$ AKA: #19: # of nodes \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'100\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 20) Name of directory where Cascade tecplot formatted output files are located\n');
fprintf(fidcomp,'$ to be read in by Pecube for PDF calculation\n');
fprintf(fidcomp,'$ Note: This only matters if a "1" is selected above (Input 18) for use of Cascade catchments\n');
fprintf(fidcomp,'$ The program will disregard whatever is here if "0" or a filename is specified\n');
fprintf(fidcomp,'$ AKA: #20: Cascade ouput folder \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'output/Cascade \n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 21) Name of the temperature file if needed. Otherwise Nil\n');
fprintf(fidcomp,'$ AKA: #21: temperature file \n'); % we don't change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'Nil\n');
fprintf(fidcomp,'\n');

fprintf(fidcomp,'$ (Input 22) If e=8 in Input 12 above, filename for velocity file. Otherwise Nil\n');
fprintf(fidcomp,'$ AKA: #22: Velocity.txt file \n');% we change
fprintf(fidcomp,'\n');
fprintf(fidcomp,'%s\n',velname);
fclose(fidcomp);
%% SBatch file

fidsbatch=fopen(strcat(SaveFolder,'run_pecube.txt'),'w');

fprintf(fidsbatch,'#!/bin/bash -l \n');
fprintf(fidsbatch,'## Run script for pecube monte carlo on esd slurm \n');
fprintf(fidsbatch,' \n');
fprintf(fidsbatch,' \n');
fprintf(fidsbatch,'## General configuration options \n');
fprintf(fidsbatch,'#SBATCH -J %s \n',abbrevmodelname);
fprintf(fidsbatch,'#SBATCH -e Pecube_Error%%j \n');
```

```
fprintf(fidsbatch,'#SBATCH -o Pecube_Screen%%j \n');
fprintf(fidsbatch,'#SBATCH --mail-user=%s \n',email);
fprintf(fidsbatch,'#SBATCH --mail-type=ALL \n');
fprintf(fidsbatch,' \n');
fprintf(fidsbatch,' \n');
fprintf(fidsbatch,'## Machine and CPU configuration \n');
fprintf(fidsbatch,'## Number of tasks per job: \n');
fprintf(fidsbatch,'#SBATCH -n 1 \n');
fprintf(fidsbatch,'## Number of nodes: \n');
fprintf(fidsbatch,'#SBATCH -N 1 \n');
% if strcmp(NodeAssign,'0');
%       fprintf(fidsbatch,' \n');
% else
%   fprintf(fidsbatch,'#SBATCH -w u-005-s0%s \n',NodeAssign);
%   fprintf(fidsbatch,' \n');
% end
if strcmp(NodeAssign,'133')||strcmp(NodeAssign,'134') || strcmp(NodeAssign,'135')||
strcmp(NodeAssign,'136')
   fprintf(fidsbatch,'#SBATCH -w u-017-s%s \n',NodeAssign);
   %fprintf(fidsbatch,'#SBATCH -w u-005-s0%s \n',NodeAssign);
   fprintf(fidsbatch,' \n');
elseif strcmp(NodeAssign,'esd_node1')||strcmp(NodeAssign,'esd_node2')||strcmp(NodeAssign,'arduino')
   fprintf(fidsbatch,'#SBATCH -w %s \n',NodeAssign);
   fprintf(fidsbatch,' \n');
else
    fprintf(fidsbatch,' \n');
end

fprintf(fidsbatch,'module load pecube \n');
fprintf(fidsbatch,' \n');

fprintf(fidsbatch,'pecube \n');
%%% end of script
```

7.2 Pecube Plotting and Analysis

7.2.1 Normal Pecube Plotting Script

This requires the supplement inputdlg.m available at the github link above. Saved as ‘PecubeOutputPlot_batch_DLG_v1_65.m’.

```
%% PecubeOutputPlot_batch_DLG v1.65
% Victoria M Buford
% vmb21@pitt.edu
```

```
% updated May 11, 2018

% Now with Goodness of Fit!
% AND accurate checks for # of files when not plotting temp &/or Velocity!

% Batch plot all steps fr   om Pecube functionality.

% Assumptions
    % velocities and structures start at Step0
    % NAMING SCHEME for lines: model_lines_Step***.dat (starting at 0)
                %     vel_new: model_grid_Step***_vel_new.dat (starting at 0)
    % Thermochronometer data must have the following columns:
        % Method, Age, Error, xdist
        % needs a header, with ^^ in it
        % must be tab sepa44
% Required Information:
    % Folder locations where the following are saved:
        % Structures   (eg Move lines: [ x y z ID ] .dat (space separated)
        % Ages         (Ages_tec****.dat from Pecube output)
        % Temperatures (temps_tec***.dat from Pecube output)
        % Velocities   (**_vel_new**.dat from Pecube output)
        % Save         (folder to save output figures)
    % Velocity Scheme
        % constant, late quick, etc. some txt or str ID for you.
    % Smoothing window:
        % If you want the age data averaged over a number of points
    % Ao
        % heat production value (don't actually need this; just labels
        % model more thoroughly)
    % extension
        % what format you want the figure saved in (eg .png, .epsc)
    % Model
        % Model Name (will save in plot names)
    % XRange
        % cross section distance
    % ZRange
        % Range of depth/elevation to plot
    % Age Range
        % Ages to plot (over which the Tchron Data/ model ran)
    % Which thermochronometers you want to plot
        % MAr, ZFT, ZHe, AFT, and AHe are options

%%
clear; close all; clc;

%%
set(0,'DefaultUicontrolFontSize',14)
%%

% Define Default Answers
if exist('Answer','var')==1
    LinesDir    = Answer.MoveLinesFolder;
    VelDir      = Answer.VelocityFolder;
    TempDir     = Answer.TempsFolder;
    SaveDir     = Answer.SaveFolder;
```

```
    AgesDir     = Answer.AgesFolder;
    Modeldef    = Answer.Model;
    IDdef       = Answer.ID;
    XRdef       = Answer.XRange;
    ZRdef       = Answer.ZRange;
    Agedef      = Answer.AgeRange;
    Figdef      = Answer.extension;
    MArdef      = Answer.tchron{1,2};
    ZFTdef      = Answer.tchron{2,2};
    ZHedef      = Answer.tchron{3,2};
    AFTdef      = Answer.tchron{4,2};
    AHedef      = Answer.tchron{5,2};
    BArdatdef   = Answer.tchron{1,3};
    MArdatdef   = Answer.tchron{2,3};
    ZFTdatdef   = Answer.tchron{3,3};
    ZHedatdef   = Answer.tchron{4,3};
    AFTdatdef   = Answer.tchron{5,3};
    AHedatdef   = Answer.tchron{6,3};
    tmethdef    = Answer.methodt{1};
    winddef     = Answer.methodt{2};
    timedef     = Answer.time{1};
    rangedef    = Answer.time{2};
    leglocdef   = Answer.LegLoc;
    TempContDef = Answer.TempContour;
    TchronDataFileDef = Answer.TchronDataFile;
    VelDef      = Answer.tempvel{1,2};
    TempDef     = Answer.tempvel{2,2};
else
    LinesDir    = pwd;
    VelDir      = pwd;
    TempDir     = pwd;
    SaveDir     = pwd;
    AgesDir     = pwd;
    Modeldef    = 'Model';
    IDdef       = 'const';
    XRdef       = [0;600];
    ZRdef       = [-50;7.5];
    Agedef      = [0;100];
    Figdef      = '.png';
    BArdef      = true;
    MArdef      = true;
    ZFTdef      = true;
    ZHedef      = true;
    AFTdef      = true;
    AHedef      = true;
    BArdatdef   = true;
    MArdatdef   = true;
    ZFTdatdef   = true;
    ZHedatdef   = true;
    AFTdatdef   = true;
    AHedatdef   = true;
    tmethdef    = 'movmean';
    winddef     = 1;
    timedef     = 'All';
    rangedef    = '18:23';
    leglocdef   = 'right';
```

```
    TempContDef = true;
    TchronDataFileDef='/Users/victoriabuford/Box
Sync/Research/MoveStuff/SPeru/Tchron/SP1Tchr.txt';%pwd;
    VelDef      = true;
    TempDef     = true;
end

Title = 'Pecube Batch Plotting';

%%%% Setting Up the Dialog Window
% Options.WindowStyle = 'modal';
Options.Resize = 'on';
Options.Interpreter = 'tex';
Options.CancelButton = 'on';
Options.ApplyButton = 'off';
Options.ButtonNames = {'Go','Cancel'};
Option.Dim = 4; % Horizontal dimension in fields

Prompt = {};
Formats = {};
DefAns = struct([]);

%%% Prompts in the Dialog Window

Prompt(1,:) = {['Select Folders and Parameters.'],[],[]};
Formats(1,1).type = 'text';
Formats(1,1).size = [-1 0];
Formats(1,1).span = [1 2]; % item is 1 field x 4 fields

Prompt(end+1,:) = {'Model Name', 'Model',[]};
Formats(2,1).type = 'edit';
Formats(2,1).format = 'text';
Formats(2,1).size = 200; % automatically assign the height
DefAns(1).Model = Modeldef;

Prompt(end+1,:)={'Model ID','ID',[]};
Formats(2,2).type = 'edit';
Formats(2,2).format = 'text';
Formats(2,2).span=[1 1];
DefAns.ID = IDdef;

Prompt(end+1,:)={'Deformation front on', 'LegLoc',[],};
Formats(2,4).type='edit';
Formats(2,4).format='text';
Formats(2,4).size=[35 25];
DefAns.LegLoc=leglocdef;

Prompt(end+1,:) = {'Move Lines Folder','MoveLinesFolder',[]};
Formats(4,1).type = 'edit';
Formats(4,1).format = 'dir';
Formats(4,1).size = [-1 0];
Formats(4,1).span = [1 3];  % item is 1 field x 3 BAfields
DefAns.MoveLinesFolder = LinesDir;
```

```
Prompt(end+1,:)= {'     X Range','XRange',[]};
Formats(4,4).type='edit';
Formats(4,4).format = 'vector';
Formats(4,4).size = [60 50];
DefAns.XRange = XRdef';
%DefAns.XRange = [300 1050];

Prompt(end+1,:) = {'Velocity Folder','VelocityFolder',[]};
Formats(5,1).type = 'edit';
Formats(5,1).format = 'dir';
Formats(5,1).size = [-1 0];
Formats(5,1).span = [1 3];  % item is 1 field x 3 fields
DefAns.VelocityFolder = VelDir;

Prompt(end+1,:)= {'     Z Range','ZRange',[]};
Formats(5,4).type='edit';
Formats(5,4).format = 'vector';
Formats(5,4).size = [60 50];
DefAns.ZRange = ZRdef';
%DefAns.ZRange = [-50 7.5];

Prompt(end+1,:) = {'Ages Folder','AgesFolder',[]};
Formats(6,1).type = 'edit';
Formats(6,1).format = 'dir';
Formats(6,1).size = [-1 0];
Formats(6,1).span = [1 3];  % item is 1 field x 3 fields
DefAns.AgesFolder = AgesDir;

Prompt(end+1,:)= {' Age Range','AgeRange',[]};
Formats(6,4).type='edit';
Formats(6,4).format = 'vector';
Formats(6,4).size = [60 50];
DefAns.AgeRange = Agedef'; %[0 100];

Prompt(end+1,:) = {'Pecube Temperatures Folder','TempsFolder',[]};
Formats(7,1).type = 'edit';
Formats(7,1).format = 'dir';
Formats(7,1).size = [-1 0];
Formats(7,1).span = [1 3];  % item is 1 field x 3 fields
DefAns.TempsFolder = TempDir;

Prompt(end+1,:) = {'Save Folder','SaveFolder',[]};
Formats(8,1).type = 'edit';
Formats(8,1).format = 'dir';
Formats(8,1).size = [-1 0];
Formats(8,1).span = [1 3];  % item is 1 field x 3 fields
DefAns.SaveFolder = SaveDir;

Prompt(end+1,:) = {'Figure Format','extension',[]};
Formats(8,4).format = 'text';
Formats(8,4).size = [45 25];
DefAns.extension = Figdef;

Prompt(end+1,:)={'Tchron Smoothing Method','methodt',[]};
```

```
Formats(9,1).type='table';
Formats(9,1).format={{'movmean','movmedian','gaussian','lowess',...
   'rloess','rloess','sgolay'} 'numeric'}; % see matlab help on function smoothdata
Formats(9,1).items={'Method','window size'};
Formats(9,1).size=[200 45];
%Formats(9,1).span=[1,2];
DefAns.methodt={tmethdef winddef}; %tmethdef;

Prompt(end+1,:) = {'Thermochronometers to plot','tchron',[]};
Formats(9,2).type = 'table';
Formats(9,2).format = {'char', 'logical', 'logical'};
Formats(9,2).items = {'Row' 'Modeled' 'data T=end'};
Formats(9,2).size = [260 160];
Formats(9,2).span = [3 1];  % item is 3 field x 1 fields
DefAns.tchron = {'BAr' BArdef BArdatdef
             'MAr' MArdef MArdatdef
             'ZFT' ZFTdef ZFTdatdef
             'ZHe' ZHedef ZHedatdef
             'AFT' AFTdef AFTdatdef
             'AHe' AHedef AHedatdef};

Prompt(end+1,:)={'Plot?           ','tempvel',[]};
Formats(9,3).type='table'; Formats(9,3).format={'char','logical'};
Formats(9,3).items={'Row' 'Plot'};
Formats(9,3).size=[200 90];
Formats(9,3).span=[1 2];
DefAns.tempvel={'Velocity' VelDef
            'Temperature' TempDef};

Prompt(end+1,:)={'             Timesteps to plot','time',[]};
Formats(10,1).type='table';
Formats(10,1).format={{'All','Range'},'numeric'};
Formats(10,1).items={'Timesteps','From X:Y'};
Formats(10,1).size=[200 45];
DefAns.time={timedef,rangedef};
%
Prompt(end+1,:)={'Contour Temperature?' 'TempContour',[]};
Formats(10,4).type='check';
%Formats(10,4).format={'char','logical'};
Formats(10,4).size=[200 45];
%Formats(10,4).span=[1 2];
DefAns.TempContour=logical(TempContDef);

MaxTemp=600;

Prompt(end+1,:)={'Tchron data file [txt]','TchronDataFile',[]};
Formats(11,1).type = 'edit';
Formats(11,1).format = 'file';
Formats(11,1).size = [250 20];
% Formats(11,1).span = [1 2];  % item is 1 field x 3 fields
DefAns.TchronDataFile = TchronDataFileDef;

[Answer,Cancelled] = inputsdlg(Prompt,Title,Formats,DefAns,Options);
```

```matlab
%% Answers:
% Answer.Model            str
% Answer.ID               str
% Answer.LegLoc           str
% Answer.MoveLinesFolder    str
% Answer.XRange           vector
% Answer.VelocityFolder     str
% Answer.ZRange           vector
% Answer.AgesFolder        str
% Answer.AgeRange          vector
% Answer.TempsFolder       str
% Answer.SaveFolder        str
% Answer.extension         str
% Answer.methodt          1x2 cell
% Answer.tchron           6x2 cell
% Answer.time             1x2 cell
% Answer.tempvel          2x2 cell

%% Interpret Modal Dialog input
Folder_Structures      = Answer.MoveLinesFolder;
Folder_Ages            = Answer.AgesFolder;
Folder_Temperatures    = Answer.TempsFolder;
Folder_Save            = Answer.SaveFolder;
Folder_Velocities      = Answer.VelocityFolder;

window_size = Answer.methodt{2};
method      = Answer.methodt{1};
ID          = Answer.ID;
extension   = Answer.extension;
Model       = Answer.Model;

% Ranges
XMin = Answer.XRange(1);
XMax = Answer.XRange(2);
YMin = Answer.AgeRange(1);
YMax = Answer.AgeRange(2);
ZMin = Answer.ZRange(1);
ZMax = Answer.ZRange(2);

% Thermochronometers
BAr = Answer.tchron{1,2};
MAr = Answer.tchron{2,2};
ZFT = Answer.tchron{3,2};
ZHe = Answer.tchron{4,2};
AFT = Answer.tchron{5,2};
AHe = Answer.tchron{6,2};

% Thermochronometer Data
TchronDatafilename=Answer.TchronDataFile;


% Legend Location
if strcmp(Answer.LegLoc,'right')
   legloc='NorthWest';
```

```matlab
elseif strcmp(Answer.LegLoc,'left')
   legloc='NorthEast';
else
   disp 'Legend location must be left or right.'
end

% Plotting temperatures or velocities
if strcmp(Answer.tempvel{1,2},'true')==1 | Answer.tempvel{1,2}==1
   velplot=1;
else
   velplot=0;
   TotalFiles_Velocities=0;
end

if strcmp(Answer.tempvel{2,2},'true')==1 | Answer.tempvel{2,2}==1
   tempplot=1;
else
   tempplot=0;
end



%% Determine file names and number of files / plots
% Pecube ages
Files_Ages = dir([Folder_Ages '/*.dat']);
FileNames_Ages = {Files_Ages.name};
TotalFiles_Ages = length(FileNames_Ages);

% MOVE structures
Files_Structures = dir([Folder_Structures '/*.dat']);
FileNames_Structures = {Files_Structures.name};
TotalFiles_Structures = length(FileNames_Structures);

% Pecube velocities
if velplot==1
   Files_Velocities = dir([Folder_Velocities '/*.dat']);
   FileNames_Velocities = {Files_Velocities.name};
   TotalFiles_Velocities = length(FileNames_Velocities);
end

% Pecube temperatures
if tempplot==1
   Files_Temperatures = dir([Folder_Temperatures '/*.dat']);
   FileNames_Temperatures = {Files_Temperatures.name};
   TotalFiles_Temperatures = length(FileNames_Temperatures);
end
clear TotalFiles
%%%% Add loop to check if the person wants to plot temps/velocities then
%%%% check total files...
% Check if all folder contain the same number of files
if velplot==1 && tempplot==1 ...
      && TotalFiles_Ages == TotalFiles_Structures ...
      && TotalFiles_Ages == TotalFiles_Velocities...
      && TotalFiles_Ages == TotalFiles_Temperatures
   TotalFiles = TotalFiles_Ages;
```

```
    disp('Good to go')
elseif velplot==0 && tempplot==1 ...
        && TotalFiles_Ages == TotalFiles_Structures ...
        && TotalFiles_Ages == TotalFiles_Temperatures
    TotalFiles = TotalFiles_Ages;
    disp('Good to go')
elseif velplot==1 && tempplot==0 ...
        && TotalFiles_Ages == TotalFiles_Structures...
        && TotalFiles_Ages == TotalFiles_Velocities
    TotalFiles = TotalFiles_Ages;
    disp('Good to go')
elseif velplot==0 && tempplot==0 ...
        && TotalFiles_Ages == TotalFiles_Structures...
    TotalFiles = TotalFiles_Ages;
    disp('Good to go')
else
    errordlg('Files appear to be missing. Check your folders again!')
    fprintf('\n')
    disp('Number of Files')
    fprintf('Structures \t Velocities \t Ages \t Temperatures\n')
    fprintf('%.1f \t\t %.1f \t\t %.1f \t %.1f\n',[TotalFiles_Structures,...
        TotalFiles_Velocities, TotalFiles_Ages, TotalFiles_Temperatures])
    fprintf('\n')
    disp('Files appear to be missing. Check your folders again!')
end


%%
% Time Range to plot
if strcmp(Answer.time{1},'All')==1
    a=1;
    b=TotalFiles;
else
    y=str2num(Answer.time{2});
    if y(end)<=TotalFiles
        if length(y)==1
            a=y(1);
            b=y(1);
        else
            a=y(1);
            b=y(end);
        end
    else
        disp 'Error: Range outside of TotalFiles'
    end
end

%% Plotting Pecube Model and structures
set(0,'defaultAxesFontSize',14)
disp('Processing')
tic
for i = a:b % 1:TotalFiles

    % Import and Clean Up Age data
    if i>=1000
```

```
        File_Age = importdata([Folder_Ages '/Ages_tec' num2str(i) '.dat'], ' ', 4);
    elseif i>=100
        File_Age = importdata([Folder_Ages '/Ages_tec0' num2str(i) '.dat'], ' ', 4);
    elseif i>=10
        File_Age = importdata([Folder_Ages '/Ages_tec00' num2str(i) '.dat'], ' ', 4);
    else
        File_Age = importdata([Folder_Ages '/Ages_tec000' num2str(i) '.dat'], ' ', 4);
    end
    File_Age.data((File_Age.data(:,4) > XMax),:) = [];
    File_Age.data((File_Age.data(:,4) < XMin),:) = [];
    File_Age.data((File_Age.data(:,5) ~= 2),:) = [];

    % Import and Clean Up Temperature Data
    if tempplot==1
%       if i>=1000
%           File_Temperatures = importdata([Folder_Temperatures '/Temps_tec' num2str(i) '.dat'], ' ', 4);
%       elseif i>=100
%           File_Temperatures = importdata([Folder_Temperatures '/Temps_tec0' num2str(i) '.dat'], ' ', 4);
%       elseif i>=10
%           File_Temperatures = importdata([Folder_Temperatures '/Temps_tec00' num2str(i) '.dat'], ' ', 4);
%       else
%           File_Temperatures = importdata([Folder_Temperatures '/Temps_tec000' num2str(i) '.dat'], ' ', 4);
%       end
        File_Temperatures = importdata([Folder_Temperatures '/' Files_Temperatures(i).name], ' ', 4);
        File_Temperatures.data((File_Temperatures.data(:,1) > XMax),:) = [];
        File_Temperatures.data((File_Temperatures.data(:,1) < XMin),:) = [];
        File_Temperatures.data((File_Temperatures.data(:,3) > ZMax),:) = [];
        File_Temperatures.data((File_Temperatures.data(:,3) < ZMin),:) = [];
        File_Temperatures.data((File_Temperatures.data(:,2) ~= 1),:) = [];
    end

    if strcmp(Model,'SP3') ||(strcmp(Model,'SP5') && i==23) ||(strcmp(Model,'SP6') && i>=23)
        % Import Structures
        File_Structures = importdata([Folder_Structures '/' Model '_lines_Step' num2str(i-1) '.dat'], ' ', 1);

%File_Structures=importdata('/Volumes/Files/McQ02N3/Pecube/Lines/McQ02N3_lines_Step23.dat');
        % wrong: %File_Structures = importdata([Folder_Structures '/' Files_Structures(i).name], ' ', 1);
        % Import and Clean Up Velocity Data
        if velplot==1
            File_Velocities = importdata([Folder_Velocities '/' Model '_grid_Step' num2str(i-1) '_vel_new.dat'], '
', 4);
            %Wrong % File_Velocities = importdata([Folder_Velocities '/' Files_Velocities(i).name], ' ', 4);
            File_Velocities.data((File_Velocities.data(:,2) > XMax),:) = [];
            File_Velocities.data((File_Velocities.data(:,2) < XMin),:) = [];
            File_Velocities.data((File_Velocities.data(:,4) > ZMax),:) = [];
            File_Velocities.data((File_Velocities.data(:,4) < ZMin),:) = [];
            File_Velocities.data((File_Velocities.data(:,3) ~= 5),:) = [];
        end
    elseif (strcmp(Model,'SP4') && i>21)
        % Import Structures
        File_Structures = importdata([Folder_Structures '/' Model '_lines_Step' num2str(i-1) '-3.dat'], ' ', 1);

%File_Structures=importdata('/Volumes/Files/McQ02N3/Pecube/Lines/McQ02N3_lines_Step23.dat');
        % wrong: %File_Structures = importdata([Folder_Structures '/' Files_Structures(i).name], ' ', 1);
        % Import and Clean Up Velocity Data
```

```
      if velplot==1
         File_Velocities = importdata([Folder_Velocities '/' Model '_grid_Step' num2str(i-1) '-3_vel_new.dat'],
' ', 4);
         %Wrong % File_Velocities = importdata([Folder_Velocities '/' Files_Velocities(i).name], ' ', 4);
         File_Velocities.data((File_Velocities.data(:,2) > XMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,2) < XMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) > ZMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) < ZMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,3) ~= 5),:) = [];
      end
   elseif (strcmp(Model,'SP4') && i<=21) || (strcmp(Model,'SP5') && i<=21) ||(strcmp(Model,'SP6') &&
i<=21)
      % Import Structures
      File_Structures = importdata([Folder_Structures '/SP3' '_lines_Step' num2str(i-1) '.dat'], ' ', 1);

%File_Structures=importdata('/Volumes/Files/McQ02N3/Pecube/Lines/McQ02N3_lines_Step23.dat');

      % Import and Clean Up Velocity Data
      if velplot==1
         File_Velocities = importdata([Folder_Velocities '/SP3' '_grid_Step' num2str(i-1) '_vel_new.dat'], ' ',
4);
         File_Velocities.data((File_Velocities.data(:,2) > XMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,2) < XMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) > ZMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) < ZMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,3) ~= 5),:) = [];
      end
      elseif (strcmp(Model,'SP5') && i==22) ||(strcmp(Model,'SP5') && i==22) ||(strcmp(Model,'SP5') &&
i==24)||(strcmp(Model,'SP6') && (i==22))
      % Import Structures
      File_Structures = importdata([Folder_Structures '/SP4' '_lines_Step' num2str(i-1) '-3.dat'], ' ', 1);

%File_Structures=importdata('/Volumes/Files/McQ02N3/Pecube/Lines/McQ02N3_lines_Step23.dat');

      % Import and Clean Up Velocity Data
      if velplot==1
         File_Velocities = importdata([Folder_Velocities '/SP4' '_grid_Step' num2str(i-1) '-3_vel_new.dat'], '
', 4);
         File_Velocities.data((File_Velocities.data(:,2) > XMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,2) < XMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) > ZMax),:) = [];
         File_Velocities.data((File_Velocities.data(:,4) < ZMin),:) = [];
         File_Velocities.data((File_Velocities.data(:,3) ~= 5),:) = [];
      end
   end
%%%%%%%%%%%%%%%%
%     else % add else loop for N9 + N10 naming
%        % Import Structures
%        File_Structures = importdata([Folder_Structures '/McQ02N9' '_lines_Step' num2str(i-1) '.dat'], ' ', 1);
%
%File_Structures=importdata('/Volumes/Files/McQ02N3/Pecube/Lines/McQ02N3_lines_Step23.dat');
%
%        % Import and Clean Up Velocity Data
%        if velplot==1
```

```
%                       File_Velocities  =  importdata([Folder_Velocities  '/McQ02N9'  '_grid_Step'  num2str(i-1)
'_vel_new.dat'], ' ', 4);
%           File_Velocities.data((File_Velocities.data(:,2) > XMax),:) = [];
%           File_Velocities.data((File_Velocities.data(:,2) < XMin),:) = [];
%           File_Velocities.data((File_Velocities.data(:,4) > ZMax),:) = [];
%           File_Velocities.data((File_Velocities.data(:,4) < ZMin),:) = [];
%           File_Velocities.data((File_Velocities.data(:,3) ~= 5),:) = [];
%         end
%
%     end
   % remove NaNs in Age Data
   A=size(File_Age.data);
   for j=1:A(2)
      File_Age.data(isnan(File_Age.data(:,j)),:) = [];
   end

   % Sort Ages for line plot
   File_Age.data = sortrows(File_Age.data,4); % x.

   % Compute moving mean
   Age_Avg=[zeros(size(File_Age.data))];
   % id ; x ; y ; real x ; real y ; real z ; AHe ; AFT ; ZHe ; ZFT ; MAr;
   Age_Avg(:,4) = File_Age.data(:,4);
   %window_size=7;
   for k=7:A(2)
      Age_Avg(:,k)=smoothdata(File_Age.data(:,k),method,window_size);
   end
   %Age_Avg=File_Age.data;

%%% Plot ages %%%
   f=figure('visible','off');  % to make loop run faster: comment out for testing.
   subplot(2,1,1); hold on
   leg={};
% BAr
   if BAr==1
   plot(Age_Avg(:,4), Age_Avg(:,12),...
      's-',...
      'MarkerEdgeColor',[0.5 0.5 0.5],...
      'MarkerFaceColor', [0.5 0.5 0.5],...
      'MarkerSize',4,'DisplayName','BAr');
   end

% MAr
   if MAr==1
   plot(Age_Avg(:,4), Age_Avg(:,11),...
      'o-',...
      'MarkerEdgeColor',[0.6350 0.0780 0.1840],...
      'MarkerFaceColor', [0.6350 0.0780 0.1840],...
      'MarkerSize',4,'DisplayName','MAr');
   end
% ZFT
   if ZFT==1
   plot(Age_Avg(:,4), Age_Avg(:,10),...
      'd-',...
      'color',[0.4940, 0.1840, 0.5560],...
```

```
    'MarkerFaceColor', [0.4940, 0.1840, 0.5560],...
    'MarkerSize',4,'DisplayName','ZFT');
  end
% ZHe
  if ZHe==1
  plot(Age_Avg(:,4), Age_Avg(:,9),...
    'v-',...
    'color',[0.8500, 0.3250, 0.0980],...
    'MarkerFaceColor', [0.8500, 0.3250, 0.0980],...
    'MarkerSize',4,'DisplayName','ZHe');
  end
% AFT
  if AFT==1
  plot(Age_Avg(:,4), Age_Avg(:,8),...
    's-',...
    'color',[0, 0.4470, 0.7410],...
    'MarkerFaceColor', [0, 0.4470, 0.7410],...
    'MarkerSize',4,'DisplayName','AFT');
  end
% AHe
  if AHe==1
  plot(Age_Avg(:,4), Age_Avg(:,7),...
    'p-',...
    'color',[0.4660, 0.6740, 0.1880],...
    'MarkerFaceColor', [0.4660, 0.6740, 0.1880],...
    'MarkerSize',4,'DisplayName','AHe');
  end
% Formatting
  axis([XMin, XMax, YMin, YMax]);
  set(gca,'XMinorTick','on','YMinorTick','on');
  grid on;
  xlabel('Distance [km]');
  ylabel('Age [Ma]');
  title(strcat(Model, {' '}, ID,{' '},'Timestep',{' '}, num2str(i-1)), 'Interpreter', 'none');
  box on
% Plot Tchron Data
  if i==TotalFiles
    if max(cell2mat(Answer.tchron(:,3)))==1
      %sum(double(cell2mat(Answer.tchron(:,3))))>=1 %if the user selected to plot the data
      TchronData=readtable(Answer.TchronDataFile,'Delimiter','\t');
      [~,            ~,            raw]            =            xlsread('/Users/victoriabuford/Box
Sync/Research/MoveStuff/SPeru/Tchron/Peru_AZHe_nr_200502.xlsx','AHe_n.r.');
      raw = raw(2:end,:);
      stringVectors = string(raw(:,1));
      stringVectors(ismissing(stringVectors)) = '';
      raw = raw(:,[2,3]);
      data = reshape([raw{:}],size(raw));
      ApGrains200502 = table;
      ApGrains200502.Sample = stringVectors(:,1);
      ApGrains200502.age_Ma = data(:,1);
      ApGrains200502.age_error_Ma = data(:,2);
      clearvars data raw stringVectors;

      [~,            ~,            raw]            =            xlsread('/Users/victoriabuford/Box
Sync/Research/MoveStuff/SPeru/Tchron/Peru_AZHe_nr_200502.xlsx','ZHe_n.r.');
      raw = raw(2:end,:);
```

```
stringVectors = string(raw(:,1));
stringVectors(ismissing(stringVectors)) = '';
raw = raw(:,[2,3]);
data = reshape([raw{:}],size(raw));
ZrGrains200502 = table;
ZrGrains200502.Sample = stringVectors(:,1);
ZrGrains200502.age_Ma = data(:,1);
ZrGrains200502.age_error_Ma = data(:,2);
clearvars data raw stringVectors;
sampledist=table;
sampledist.Dist=TchronData.xdist;
sampledist.Sample_ID=TchronData.Sample_ID;
sampledist.elev=TchronData.Elevation;

dist=[];
elev=[];
count=1;
ApGrList=[];
for jj=1:length(ApGrains200502.Sample)
   for j=1:length(sampledist.Sample_ID)
      if strcmp(sampledist.Sample_ID{j},ApGrains200502.Sample{jj}(1:6))
         dist(jj)=sampledist.Dist(j);
         elev(jj)=sampledist.elev(j);
         ApGrList{count}=ApGrains200502.Sample{jj}(1:6);
         count=count+1;
         break
      else
         dist(jj)=0;
         elev(jj)=0;
      end
   end
end
ApGrains200502.xdist=dist';
ApGrains200502.elev=elev';
ApGrList=unique(ApGrList');
% %%
dist=[];
elev=[];
count=1;
ZrGrList=[];
for jj=1:length(ZrGrains200502.Sample)
   for j=1:length(sampledist.Sample_ID)
      if strcmp(sampledist.Sample_ID{j},ZrGrains200502.Sample{jj}(1:6))
         dist(jj)=sampledist.Dist(j);
         elev(jj)=sampledist.elev(j);
         ZrGrList{count}=ZrGrains200502.Sample{jj}(1:6);
         count=count+1;
         break
      else
         dist(jj)=0;
         elev(jj)=0;
      end
   end
end
ZrGrains200502.xdist=dist';
ZrGrains200502.elev=elev';
```

```
ZrGrList=unique(ZrGrList');
colordata={'ks','k*','ks','kd','ko','kp'};  % define colors for Tchron Data
legdata={'BAr data','MAr data','ZFT data','ZHe data','AFT data','AHe data'}; %legend
markcol=[0.25 0.25 0.25
        0.6350 0.0780 0.1840
        0.4940, 0.1840, 0.5560
        0.8500, 0.3250, 0.0980
        0, 0.4470, 0.7410
        0.4660, 0.6740, 0.1880];
    % Define which data is which in the Tchron file
    %%%%%% MODIFIED FOR Indiv. Grains!
    indZr=~(strcmp(TchronData.Sample_ID,ZrGrList(1))+...
        strcmp(TchronData.Sample_ID,ZrGrList(2))+...
        strcmp(TchronData.Sample_ID,ZrGrList(3))+...
        strcmp(TchronData.Sample_ID,ZrGrList(4))+...
        strcmp(TchronData.Sample_ID,ZrGrList(5))+...
        strcmp(TchronData.Sample_ID,ZrGrList(6)));

    indAp=~(strcmp(TchronData.Sample_ID,ApGrList(1))+...
        strcmp(TchronData.Sample_ID,ApGrList(2))+...
        strcmp(TchronData.Sample_ID,ApGrList(3))+...
        strcmp(TchronData.Sample_ID,ApGrList(4)));

    % Define which data is which in the Tchron file
    MArind = strcmp(TchronData.Method,'MAr');
    BArind = strcmp(TchronData.Method,'BAr');
    ZFTind = strcmp(TchronData.Method,'ZFT');
    ZHeind = strcmp(TchronData.Method,'ZHe')-~indZr;
    ZHeind = ZHeind>0;
    AFTind = strcmp(TchronData.Method,'AFT');
    AHeind = strcmp(TchronData.Method,'AHe')-~indAp;
    AHeind = AHeind>0;
% Matrix of thermochronometers sorted by Method
    index = [BArind MArind ZFTind ZHeind AFTind AHeind];
    [d,e]=size(index);
% Plot the thermochronometer data
for g=1:e % for each thermochronometer
    if Answer.tchron{g,3} % if user wants to plot this Tchron
        if max(index(:,g))==1 % if the tchron data exists
            errorbar(TchronData.xdist(index(:,g)),...
                TchronData.Age((index(:,g))),...
                TchronData.Error((index(:,g))),...
                colordata{g},'MarkerFaceColor',markcol(g,:),...
                'MarkerEdgeColor','k','DisplayName',legdata{g})
            %leg(end+1)={[legdata{g}]};
            if g==4;
                % plot ZHe individual grains
                ZrGrs=find(ZrGrains200502.xdist~=0);
                errorbar(ZrGrains200502.xdist(ZrGrs),ZrGrains200502.age_Ma(ZrGrs),...
                    ZrGrains200502.age_error_Ma(ZrGrs),colordata{g},...
                    'Color',[0.75 0.75 0.75],...
                    'MarkerEdgeColor',markcol(g,:),'MarkerFaceColor',[0.75 0.75 0.75],...
                    'DisplayName','ZrIndiv. Grains')
            elseif g==6;
                % plot AHe individual grains
```

```
                    ApGrs=find(ApGrains200502.xdist~=0);
                    errorbar(ApGrains200502.xdist(ApGrs),ApGrains200502.age_Ma(ApGrs),...
                        ApGrains200502.age_error_Ma(ApGrs),colordata{g},...
                        'Color',[0.75 0.75 0.75],...
                        'MarkerEdgeColor',markcol(g,:),'MarkerFaceColor',[0.75 0.75 0.75],...
                        'DisplayName','ApIndiv. Grains')
                end
            end
        end
    end
  end
end

%    % Legend
legend('Location',legloc)

  hold off
  subplot(2,1,2);
  hold on

  %%% Plot temperature, velocity, and structures%%%
  if tempplot==1
    [x, z] = meshgrid(XMin:XMax, ZMin:ZMax);
    Temperature = griddata(File_Temperatures.data(:,1),...
      File_Temperatures.data(:,3), File_Temperatures.data(:,5), x, z);
    if TempContDef==0
      %%% Plot temperature field %%%
      surf(x, z, Temperature, 'EdgeColor', 'none');
      colormap jet
      c = colorbar('Location', 'southoutside');
      caxis([0 MaxTemp])
      xlabel(c, 'Temperature [∫C]','FontSize',14);
      velcol={'white'};
      hold on
    else
      %%% Plot temperature contours %%%
      [M,c]=contour(x,z,Temperature,[0:100:800],'ShowText','on');
      c.LineWidth=1.5;
      caxis([0 MaxTemp])
      %colormap jet
      colormap(flipud(hot))
      velcol={[0.25 0.5 0.85]};
      hold on
    end
  else
    velcol={[0.25 0.5 0.85]};
  end

  %%% Plot structures %%%
    n = 1;
    while n < length(File_Structures.data(:,4))
      LineID = File_Structures.data(n,4);
      LineID_Indices = find(File_Structures.data(:,4) == LineID);
      plot3(File_Structures.data(LineID_Indices, 1), File_Structures.data(LineID_Indices, 3),...
        (zeros(length(File_Structures.data(LineID_Indices, 3)), 1) + 1e3),...
```

```
         'color', 'black',... %[0.5 0.5 0.5],...
         'LineWidth', 1);
      hold on
      n = LineID_Indices(length(LineID_Indices)) + 1;
   end

%%% Plot velocity field
if velplot==1
   %%% Plot velocity field %%%
      if velplot==1
         QuiverPlotDensity = 10;
         quiver3(File_Velocities.data(1:QuiverPlotDensity:end,2), ...
            File_Velocities.data(1:QuiverPlotDensity:end,4),...
            File_Velocities.data(1:QuiverPlotDensity:end,4)+1000,...
            File_Velocities.data(1:QuiverPlotDensity:end,5),...
            File_Velocities.data(1:QuiverPlotDensity:end,7), ...
            File_Velocities.data(1:QuiverPlotDensity:end,6),...
            0.5,...
            'LineWidth', 0.5,...
            'Color',velcol{1},...
            'MaxHeadSize', 0.010,...
            'AutoScaleFactor', 0.10);
      end
end
if tempplot==0 && velplot==1
   set(gca,'Color',[0.90 0.90 0.90])
end

% Plot modifications
axis([XMin, XMax, ZMin, ZMax]);
set(gcf,'Position',[1 5 1280 700]) % fullscreen for Macbook pro 13" retina
daspect([1 1 1])  % set subplot aspect ratio: no vertical exaggeration
set(gca,'XMinorTick','on','YMinorTick','on');
grid on;
xlabel('Distance [km]');
ylabel('Elevation [km]');
box on
hold off

FileName = char(strcat(Model,'_',ID,...
   '_Timestep_', num2str(i-1),extension));
saveas(gcf,fullfile(Folder_Save, FileName)) %saves figure

%make zoomed in final fig
if i==TotalFiles
   XMin=210;XMax=530;ZMin=-35;legloc='NorthWest';
   subplot(2,1,1)
   legend('Location',legloc)
   axis([XMin, XMax, YMin, YMax]);
   subplot(2,1,2)
   axis([XMin, XMax, ZMin, ZMax]);
   ax=gca;
   ax.Children(1).Visible='off'; %turn off quiver plot
   FileName = char(strcat(Model,'_',ID,...
      '_Timestep_', num2str(i-1),'_data',extension));
```

```
        saveas(gcf,fullfile(Folder_Save, FileName)) %saves figure
    end

    close % closes figure
end
% disp('Finished')
t=toc;
disp('Finished; time elapsed: ')
disp(datestr(datenum(0,0,0,0,0,t),'HH:MM:SS'))
```

7.2.2 Goodness of Fit Computation

Using the normal pecube plotting script, select your files for at least ages, save location, and thermochronometers to match. Saved as ‘Pecube_GOFMaster.m’.

```
if i==TotalFiles
    %%% Goodness Of Fit
    GOF_version=1.1;
    samplelocll=zeros(d,e);
    ModelVal=zeros(d,e);
    Matches=zeros(d,e);

    ModelError=1; % in [Ma]
    for mm=2 % to calc mult xsec km errors
        XsecError=mm; % in [km]
        for g=1:e % AFT then ZHe
            if g==6 %AHe
                ii=7;
            elseif g==5 %AFT
                ii=8;
            elseif  g==4 %ZHe
                ii=9; % which index value tchronometer I want
            elseif g==3 % ZFT
                ii=10;
            elseif g==2 %MAr
                ii=11;
            elseif g==1 %BAr
                ii=12;
            end

            % for each sample in table
            for jj=1:d
                % if that sample exists in this thermochronometer
                if index(jj,g)==1
                    Matches(jj,g)=3;
                    %%%%%%% Variables
                    %%%%% Measured Tchron Value
                    %%% TchronData.Age(jj)
```

```
    %%%% Measured Tchron Error
    %%% TchronData.Error(jj)

    %%%% Sample Location (Adjusted to Pecube Grid)
    %%% TchronData.xdist(jj)
    sampleloc(jj,g)=round(TchronData.xdist(jj)*2)/2;

    % Loop over X-error in cross-section
    % Without looping over 1km wide window, N10const had 28.57%
    % fit. With 1km error, 42.86% fit.
    ll=[sampleloc(jj,g)-(XsecError): 0.5: sampleloc(jj,g)+(XsecError)];

    for lll= 1:length(ll)
      samplelocll(jj,g)=ll(lll);

      % Find Sample Location in MODELLED
      IndexSampleLocMod=find(Age_Avg(:,4)==samplelocll(jj,g));
      % Get Modelled Value at that location
      ModelVal(jj,g)=Age_Avg(IndexSampleLocMod,ii);

      % Compare Modelled to Measured
      %%%% To accurately compare, we have 4 cases to test:
      % Case 1. Model + Model Error within measured Age
      % Case 2. Model - Model Error within measured Age
      % Case 3. Measured + Measured error within modelled Age
      % Case 4. Measured - Measured error within modelled Age

      % Case 1. Model Upper Error within measured Age
      if (ModelVal(jj,g) + ModelError)< (TchronData.Age(jj) + TchronData.Error(jj)) ...
          && ...
          (ModelVal(jj,g) + ModelError)> (TchronData.Age(jj) - TchronData.Error(jj))
        Matches(jj,g)=1; % it does match

        % Case 2. Model Lower Error within measured Age
      elseif (ModelVal(jj,g) - ModelError)< (TchronData.Age(jj) + TchronData.Error(jj)) ...
          && ...
          (ModelVal(jj,g) - ModelError)> (TchronData.Age(jj) - TchronData.Error(jj))
        Matches(jj,g)=1; % it does match

        % Case 3. Measured Upper error within modelled Age
      elseif (TchronData.Age(jj) + TchronData.Error(jj)) < (ModelVal(jj,g) + ModelError) ...
          && ...
          (TchronData.Age(jj) + TchronData.Error(jj)) >(ModelVal(jj,g) - ModelError)
        Matches(jj,g)=1; % it does match

        % Case 4. Measured Lower error within modelled Age
      elseif (TchronData.Age(jj) - TchronData.Error(jj)) < (ModelVal(jj,g) + ModelError) ...
          && ...
          (TchronData.Age(jj) - TchronData.Error(jj)) >(ModelVal(jj,g) - ModelError)
        Matches(jj,g)=1; % it does match
        % No else statement needed bc line above defines
        % Matches(jj,g)=3 %Not Matching and this loop only
        % overwrites that if the sample is within the error
      end
    end
  end
```

```
    end
end

% Percent Fit
YesMatches=[];
NoMatches=[];
YesMatches=find(Matches==1);
NoMatches=find(Matches==3);
PercFit=length(YesMatches)/(length(NoMatches)+length(YesMatches))

YesMatchesTchron=[];
nsamples=[];
for kk=1:e
    YesM=length((find(Matches(:,kk)==1)));
    NoM=length((find(Matches(:,kk)==3)));
    nsamples(kk)=YesM+NoM;
    YesMatchesTchron(kk)=YesM/(NoM+YesM);
end

%%% Print goodness of fit to bulk file
%     % Model_Name  Model_ID    Model_Error Sample_Loc_Error    %Fit
%     GOFLoc='/Volumes/Files/VictoriaFiles/Pecube/GoodnessOfFit.txt';
%     fidGOF=fopen(GOFLoc,'at');
%     fprintf(fidGOF,'\n%10s\t%18s\t%0.1f\t%16.1f\t%7.4f',Model,ID,ModelError,XsecError,PercFit);
%     fclose(fidGOF);


%%%% Print goodness of fit + matches + sample info to individual file
GOFLoc_Ind=strcat(Folder_Save(1:end-9),'/GoodnessOfFit_',Model,'_',ID,'_',num2str(mm),'.txt');
fidGOF_ind=fopen(GOFLoc_Ind,'w');
fprintf(fidGOF_ind,'Goodness of Fit Computation; Version %1.1f\n',GOF_version);
fprintf(fidGOF_ind,'Model_Name\tModel_ID\t\t\tModel_Error\tSample_Loc_Error\tPercFit\n');
fprintf(fidGOF_ind,'%s\t\t%s\t\t%0.1f\t\t%f\t%f\n',Model,ID,ModelError,XsecError,PercFit);
fprintf(fidGOF_ind,'\n');
fprintf(fidGOF_ind,'Summary \tBAr\tMAr\tZFT\tZHe\tAFT\tAHe\n');
fprintf(fidGOF_ind,'PercFit \t%0.4f\t%0.4f\t%0.4f\t%0.4f\t%0.4f\t%0.4f\n',YesMatchesTchron');
%fprintf(fidGOF_ind,'\n');
fprintf(fidGOF_ind,'No. samples \t%0.0f\t%0.0f\t%0.0f\t%0.0f\t%0.0f\t%0.0f\n',nsamples');
fprintf(fidGOF_ind,'\nMatches                    [n=%.0f;                    e.g.                    No=3;
Yes=1]\n',(length(NoMatches)+length(YesMatches)));
fprintf(fidGOF_ind,'\n');
fprintf(fidGOF_ind,'Sample \t%s \t%s \t%s \t%s \t%s \t%s \n',legdata{:});
for tt=1:length(Matches)
    fprintf(fidGOF_ind,'%10s  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \n',TchronData.Sample_ID{tt},
Matches(tt,:));
    %fprintf(fidGOF_ind,'%s \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \n',num2str(tt), Matches(tt,:));
end
fclose(fidGOF_ind);


MatchSum=sum(Matches');

GOFMaster='/Volumes/Files/VictoriaFiles/Pecube/SP5/GOFMaster2.txt';
fidGOF_Master=fopen(GOFMaster,'a');
%             fidGOF_Master=fopen(GOFMaster,'w');
%
```

```
    %          fprintf(fidGOF_Master,'Goodness of Fit Computation; Version %1.1f\n',GOF_version);
    %
    %          fmt=['%s ' repmat(' %s',1,numel(TchronData.Sample_ID)) '\n'];
    %          fprintf(fidGOF_Master,fmt,'Model',TchronData.Sample_ID{:})
    %

    fmt=['%s =' repmat(' %1.0f',1,numel(MatchSum)) '\n'];
    fprintf(fidGOF_Master,fmt,strcat(Model,'_',ID),MatchSum)
    fclose(fidGOF_Master)

    %fprintf(fidGOF_Master,'\t%.0f')

    %
    %          for tt=1:length(MatchSum)
    %             fprintf(fidGOF_Master,'%10s \t%.0f \n',TchronData.Sample_ID{tt}, MatchSum(tt));
    %                    %fprintf(fidGOF_ind,'%s \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \n',num2str(tt),
Matches(tt,:));
    %          end
    %
  end
  %%%%% End GOF

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%
end
```

7.2.3 Chi-Square Test

Prepare the matlab workspace in the same method as goodness of fit, via the normal pecube plotting script. Saved as ‘Pecube_ChiTest.m’.

```
if i==TotalFiles
  %%% Goodness Of Fit
  GOF_version=1.1;
  samplelocll=zeros(d,e);
  ModelVal=zeros(d,e);
  Matches=zeros(d,e);


  %         %%
  ModelError=1; % in [Ma]
  for mm=2 % to calc mult xsec km errors
    XsecError=mm; % in [km]
    ChiSample=zeros(d,e,(XsecError/0.5*2+1));%length(ll)
    for g=1:e % AFT then ZHe
      if g==6 %AHe
```

```
    ii=7;
elseif g==5 %AFT
    ii=8;
elseif  g==4 %ZHe
    ii=9; % which index value tchronometer I want
elseif g==3 % ZFT
    ii=10;
elseif g==2 %MAr
    ii=11;
elseif g==1 %BAr
    ii=12;
end

% for each sample in table
for jj=1:d
    % if that sample exists in this thermochronometer
    if index(jj,g)==1
        Matches(jj,g)=3;
        %%%%%%% Variables
        %%%% Measured Tchron Value
        %%% TchronData.Age(jj)

        %%%% Measured Tchron Error
        %%% TchronData.Error(jj)

        %%%% Sample Location (Adjusted to Pecube Grid)
        %%% TchronData.xdist(jj)
        sampleloc(jj,g)=round(TchronData.xdist(jj)*2)/2;

        % Loop over X-error in cross-section
        % Without looping over 1km wide window, N10const had 28.57%
        % fit. With 1km error, 42.86% fit.
        ll=[sampleloc(jj,g)-(XsecError): 0.5: sampleloc(jj,g)+(XsecError)];

        for lll= 1:length(ll)
            samplelocll(jj,g)=ll(lll);

            % Find Sample Location in MODELLED
            IndexSampleLocMod=find(Age_Avg(:,4)==samplelocll(jj,g));
            % Get Modelled Value at that location
            ModelVal(jj,g)=Age_Avg(IndexSampleLocMod,ii);
            ChiSample(jj,g,lll)=((TchronData.Age(jj)-ModelVal(jj,g))^2)/(TchronData.Error(jj)^2);

            % Compare Modelled to Measured
            %%%% To accurately compare, we have 4 cases to test:
            % Case 1. Model + Model Error within measured Age
            % Case 2. Model - Model Error within measured Age
            % Case 3. Measured + Measured error within modelled Age
            % Case 4. Measured - Measured error within modelled Age

            % Case 1. Model Upper Error within measured Age
            if (ModelVal(jj,g) + ModelError)< (TchronData.Age(jj) + TchronData.Error(jj)) ...
                    && ...
                    (ModelVal(jj,g) + ModelError)> (TchronData.Age(jj) - TchronData.Error(jj))
                Matches(jj,g)=1; % it does match
```

```
          % Case 2. Model Lower Error within measured Age
        elseif (ModelVal(jj,g) - ModelError)< (TchronData.Age(jj) + TchronData.Error(jj)) ...
            && ...
            (ModelVal(jj,g) - ModelError)> (TchronData.Age(jj) - TchronData.Error(jj))
          Matches(jj,g)=1; % it does match

          % Case 3. Measured Upper error within modelled Age
        elseif (TchronData.Age(jj) + TchronData.Error(jj)) < (ModelVal(jj,g) + ModelError) ...
            && ...
            (TchronData.Age(jj) + TchronData.Error(jj)) >(ModelVal(jj,g) - ModelError)
          Matches(jj,g)=1; % it does match

          % Case 4. Measured Lower error within modelled Age
        elseif (TchronData.Age(jj) - TchronData.Error(jj)) < (ModelVal(jj,g) + ModelError) ...
            && ...
            (TchronData.Age(jj) - TchronData.Error(jj)) >(ModelVal(jj,g) - ModelError)
          Matches(jj,g)=1; % it does match
          % No else statement needed bc line above defines
          % Matches(jj,g)=3 %Not Matching and this loop only
          % overwrites that if the sample is within the error
        end
      end
    end
  end
end

ChiMin=min(ChiSample,[],3);
Chi=sum(sum(ChiMin))
%          %%
% Percent Fit
YesMatches=[];
NoMatches=[];
YesMatches=find(Matches==1);
NoMatches=find(Matches==3);
PercFit=length(YesMatches)/(length(NoMatches)+length(YesMatches))

YesMatchesTchron=[];
nsamples=[];
for kk=1:e
  YesM=length((find(Matches(:,kk)==1)));
  NoM=length((find(Matches(:,kk)==3)));
  nsamples(kk)=YesM+NoM;
  YesMatchesTchron(kk)=YesM/(NoM+YesM);
end

%%% Print goodness of fit to bulk file
%    % Model_Name  Model_ID    Model_Error Sample_Loc_Error    %Fit
%    GOFLoc='/Volumes/Files/VictoriaFiles/Pecube/GoodnessOfFit.txt';
%    fidGOF=fopen(GOFLoc,'at');
%    fprintf(fidGOF,'\n%10s\t%18s\t%0.1f\t%16.1f\t%7.4f',Model,ID,ModelError,XsecError,PercFit);
%    fclose(fidGOF);


%%%% Print goodness of fit + matches + sample info to individual file
GOFLoc_Ind=strcat(Folder_Save(1:end-9),'/GoodnessOfFit_',Model,'_',ID,'_',num2str(mm),'.txt');
fidGOF_ind=fopen(GOFLoc_Ind,'w');
```

```
        fprintf(fidGOF_ind,'Goodness of Fit Computation; Version %1.1f\n',GOF_version);
        fprintf(fidGOF_ind,'Model_Name\tModel_ID\t\t\tModel_Error\tSample_Loc_Error\tPercFit\n');
        fprintf(fidGOF_ind,'%s\t\t%s\t\t%0.1f\t\t%f\t%f\n',Model,ID,ModelError,XsecError,PercFit);
        fprintf(fidGOF_ind,'Chi^2: %0.4f\n',Chi);
        fprintf(fidGOF_ind,'\n');
        fprintf(fidGOF_ind,'Summary \tBAr\tMAr\tZFT\tZHe\tAFT\tAHe\n');
        fprintf(fidGOF_ind,'PercFit \t%0.4f\t%0.4f\t%0.4f\t%0.4f\t%0.4f\t%0.4f\n',YesMatchesTchron');
        %fprintf(fidGOF_ind,'\n');
        fprintf(fidGOF_ind,'No. samples \t%0.0f\t%0.0f\t%0.0f\t%0.0f\t%0.0f\t%0.0f\n',nsamples');
        fprintf(fidGOF_ind,'\nMatches                    [n=%.0f;                    e.g.                    No=3;
Yes=1]\n',(length(NoMatches)+length(YesMatches)));
        fprintf(fidGOF_ind,'\n');
        fprintf(fidGOF_ind,'Sample \t%s \t%s \t%s \t%s \t%s \t%s \n',legdata{:});
        for tt=1:length(Matches)
            fprintf(fidGOF_ind,'%10s  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \t%.0f  \n',TchronData.Sample_ID{tt},
Matches(tt,:));
            %fprintf(fidGOF_ind,'%s \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \n',num2str(tt), Matches(tt,:));
        end
        fclose(fidGOF_ind);

        MatchSum=sum(Matches');

        GOFMaster='/Volumes/Files/VictoriaFiles/Pecube/SP5/GOFMaster2.txt';
        fidGOF_Master=fopen(GOFMaster,'a');
        %           fidGOF_Master=fopen(GOFMaster,'w');
        %
        %           fprintf(fidGOF_Master,'Goodness of Fit Computation; Version %1.1f\n',GOF_version);
        %
        %           fmt=['%s ' repmat(' %s',1,numel(TchronData.Sample_ID)) '\n'];
        %           fprintf(fidGOF_Master,fmt,'Model',TchronData.Sample_ID{:})
        %

        fmt=['%s =  Chi^2=%0.6f' repmat(' %1.0f',1,numel(MatchSum)) '\n'];
        fprintf(fidGOF_Master,fmt,strcat(Model,'_',ID),Chi,MatchSum)
        fclose(fidGOF_Master)

        %fprintf(fidGOF_Master,'\t%.0f')

        %
        %           for tt=1:length(MatchSum)
        %               fprintf(fidGOF_Master,'%10s \t%.0f \n',TchronData.Sample_ID{tt}, MatchSum(tt));
        %                   %fprintf(fidGOF_ind,'%s \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \t\t%.0f \n',num2str(tt),
Matches(tt,:));
        %           end
        %
    end
    %%%%% End GOF

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
```

```
end
```

7.3 Cascade & Geomorph Codes

7.3.1 Cascade Video Plotting

Saved as ‘Cascade_PlotBatchPrecipDefL.m

```
%% The following script plots and saves a complete CASCADE model run and optionally creates a video
from the topography files
% Victoria Buford
%   edited from Paul Eizenhoefer's version
%% Instructions
% 1. Place script in folder where you want to have the script output (e.g. video and MATLAB figure files) to
be saved
% 2. Run script in MATLAB
% 3. Answer command window questions
% 4. Wait (coffee break?)
% 13, 14, 20:30, 32
%% CASCADE output folder and files
close;clc;clear;
disp('Locate the folder containing CASCADE output files topo_tec_****.dat.');
OutputFolder = uigetdir;
%%

prompt={'X-range [xmin xmax] [km]','Y-range [ymin ymax] [km]',...
    'Z-range [zmin zmax] [km]', 'time per file [ka]',...
    'video (0=no; 1=yes)','Model Name'};

defaultans = {'[0 50]','[0 150]','[0 10]','100','1','CASCADE_Video'};
inp= inputdlg(prompt,'Plot Ranges',1,defaultans);
XRange=str2num(inp{1});
YRange=str2num(inp{2});
ZRange=str2num(inp{3});
TimeStep=str2num(inp{4});
Video=str2num(inp{5});
Model_name=(inp{6});

ZRangePlotColor=[0 7.5];

 PrecMax=2;%max(File.precipitation_mpy);


%% Plot Figures and get videoframes
tic
```

```
CASCADE_Files = dir([OutputFolder '/*.dat']);
FileNames = {CASCADE_Files.name};
TotalFiles = length(FileNames);

if Video == 1
   Frames = VideoWriter(strcat(OutputFolder(1:end-8), Model_name), 'MPEG-4');
   Frames.Quality = 100;
   Frames.FrameRate =  1000/TimeStep ;% should equate to 1Ma/s
   open (Frames);
end
f=figure('visible','off');
set(f,'Position',[1 1 1300 700]); %1280 704

%XRange=[5 45];YRange=[5 190];


for i = 1:TotalFiles

%     for j=1:TotalFiles % Eg.... whoops time out = 5ky
%     i=(j-1)*20+1;
   %%% Load CASCADE output files %%%
   if (i-1)>=1000
      File = readtable([OutputFolder '/topo_tec_' num2str(i-1) '.dat'], 'HeaderLines', 4);
   elseif (i-1)>=100
      File = readtable([OutputFolder '/topo_tec_0' num2str(i-1) '.dat'], 'HeaderLines', 4);
   elseif (i-1)>=10
      File = readtable([OutputFolder '/topo_tec_00' num2str(i-1) '.dat'], 'HeaderLines', 4);
   else
      File = readtable([OutputFolder '/topo_tec_000' num2str(i-1) '.dat'], 'HeaderLines', 4);
   end

   OutputVariables={'x_km','y_km','z_km','node','precipitation_mpy',...
      'fluvial_erosion_rate_mpy','diffusion_erosion_rate_mpy',...
      'landslide_erosion_rate_mpy','total_erosion_rate_mpy',...
      'catchment_color','catchment_number',...
      'glacial_erosion_rate_mpy','ice_thickness_m','mass_balance_1py',...
      'total_topography_m','sliding_velocity_mpy','gerode_term_mpy',...
      'rock_contact_km','isostatic_deflection_mpy','slope_mpkm',...
      'totalflexiso_m','constriction','cumulative_erosion_m',...
      'surface_area_km2'}';
   File.Properties.VariableNames=OutputVariables';

   % Clean up output - Remove NaNs
   Nanss = ismissing(File,{NaN}); % creates table size of A, with 1 or 0 for NaN
   NanNodes = Nanss(:,4); clear Nanss % pull only the nodes column
   File = File(~NanNodes,:); clear NanNodes % remove any row that the node is NaN
   % Compute Delaunay triangulation
   Triangulation = delaunay(File.x_km, File.y_km);

    % Plot
%     f=figure;
%     set(f,'Position',[1 1 1300 700])
   % 3D Figure.
   subplot(4,3,[1,5])
```

```
trimesh(Triangulation, File.x_km, File.y_km, File.z_km, 'FaceColor', 'flat')
daspect([1 1 1])
xlabel('[km]')
ylabel('[km]')
zlabel('[km]')
ax1=gca;
c1 = colorbar('Location', 'westoutside');
xlabel(c1,'Elevation [km]')

%%% Timestep
% use this one for topo_tec0000 = 0Ma
% Time = (i-1) * TimeStep / 1e3; % paul's time: model start = 0Ma

%%% Timestep
% use this one for topo_tec**last = 0Ma
 sz = size(CASCADE_Files);
 TimeOrig = TimeStep * sz(1) - TimeStep;
 if (TimeOrig - (TotalFiles-1) * TimeStep) / 1e3 ==0
     Time = (TimeOrig - (i-1) * TimeStep) / 1e3;
 else
     TimeOrig = TimeStep * sz(1);
     Time = (TimeOrig - (i-1) * TimeStep) / 1e3;
 end
 clear TimeOrig
%%%%

title(strcat(num2str(Time), ' Ma'), 'FontSize',20);
demcmap([ZRangePlotColor])
axis([XRange YRange ZRange])
light('Position',[0 180 50],'Style','local')

% Precipitation Plot
subplot(4,3,[3,6])
trimesh(Triangulation, File.x_km, File.y_km, File.z_km, File.precipitation_mpy,  'FaceColor', 'flat')
daspect([1 1 1])
c2 = colorbar('Location', 'westoutside');
xlabel(c2,'Precipitation [m/yr]')
xlabel('[km]')
ylabel('[km]')
zlabel('[km]')
ax2=gca;
pcmap=othercolor('Blues4');
colormap(ax2,pcmap)
caxis([0 PrecMax])
axis([XRange YRange ZRange])
light('Position',[0 180 50],'Style','local')
view(0,90)

%%%% X-Sec Plot
% Precipitation
subplot(4,3,[7,9])
xx=min(File.x_km):0.5:max(File.x_km);
yy=min(File.y_km):0.5:max(File.y_km);
[x,y]=meshgrid(xx,yy);%clear xx yy
z=griddata(File.x_km,File.y_km,File.z_km,x,y);
```

```
precip_mpy=griddata(File.x_km,File.y_km,File.z_km,File.precipitation_mpy,...
    x,y,z);
zprec=0:0.5:ZRange(2);
[zzprec,yyprec]=meshgrid(zprec,yy);
caxis([0 PrecMax])
xprec=max(File.x_km)*ones(size(zzprec));
for k=1:length(y)
    avgprecip(k) = mean(precip_mpy(k,:)) ;
end
avgprecip=smoothdata(avgprecip,'movmean',11); % over 5km, per convo w/ Paul
avgprecipgrid=ones(size(yyprec)).*avgprecip';

s=surf(xprec,yyprec,zzprec,avgprecipgrid,'EdgeColor','none');
hold on
ax3=gca;

%pcmap=othercolor('Blues3');
colormap(ax3,pcmap)
caxis([0 PrecMax])
view([-90 0])
ax3.Color='none';
daspect([1 1 1])
axis([0 50 YRange ZRange])
ax3.YTick=[];
ax3.ZTick=[];

% Topography
subplot(4,3,[10,12])
trimesh(Triangulation, File.x_km, File.y_km, File.z_km, 'FaceColor', 'flat')
    xlabel('[km]')
ylabel('[km]')
zlabel('[km] or [m/y]')
hold on

plot3(zeros(size(avgprecip)),yy,avgprecip,'b','LineWidth',1)
daspect([1 1 1])
ax4=gca;
demcmap([ZRangePlotColor])
axis([0 50 YRange ZRange])
light('Position',[0 180 50],'Style','local')
view(-90,0)

set(ax4,'Color','None')
box on
set(ax4,'Position',[0.13 0.11 0.775 0.1577])
set(ax3,'Position',[0.13 0.11 0.775 0.1577])




% Positions of various axes...
 set(ax1,'Position',[0.1164    0.3141    0.6850    0.5782])
 set(c1,'Position',[0.1297    0.3002    0.0144    0.5941])
 set(ax2,'Position',[0.7781    0.2757    0.1745    0.6157])
 ax1.Color='None';
```

```
 ax1.YTick=[0:10:YRange(2)];
 ax2.YTick=[0:10:YRange(2)];
 ax4.YTick=[0:10:YRange(2)];


%savefig([OutputFolder strcat(num2str(Time), ' Ma.fig')])
%save([OutputFolder strcat(num2str(Time), ' Ma.png')])
if Video == 1
   set(gcf, 'Position', [0,0,1300,700])
   Frame = getframe(gcf);
   writeVideo(Frames,Frame);
end

   % for last time step, plot swath topography
if i==TotalFiles
   % import modern topo (McQ02N)
   filename = '/Volumes/Files/VictoriaFiles/Cascade/McQ02NModernTopo.dat';
   delimiter = ' ';
   formatSpec = '%f%f%f%f%[^\n\r]';
   fileID = fopen(filename,'r');
   dataArray = textscan(fileID, formatSpec, 'Delimiter', delimiter, 'MultipleDelimsAsOne', true, 'TextType',
'string', 'EmptyValue', NaN,  'ReturnOnError', false);
   fclose(fileID);
   ModernTopo = table(dataArray{1:end-1}, 'VariableNames', {'x','z_min','z_mean','z_max'});
   clearvars filename delimiter formatSpec fileID dataArray ans;
   plot3(zeros(size(ModernTopo.x)),ModernTopo.x-100,ModernTopo.z_min/1e3,'r','LineWidth',1)
   plot3(zeros(size(ModernTopo.x)),ModernTopo.x-100,ModernTopo.z_mean/1e3,'r','LineWidth',1)
   plot3(zeros(size(ModernTopo.x)),ModernTopo.x-100,ModernTopo.z_max/1e3,'r','LineWidth',1)
   if Video == 1
      set(gcf, 'Position', [0,0,1300,700])
      Frame = getframe(gcf);
      writeVideo(Frames,Frame);
   end
end

%close(f);
end
if Video == 1
   close (Frames);
end
close
toc
```

7.3.2 Ksn Computation

Saved as 'KsnThings.m'

```
%% code to import, process DEM to fill sinks/ create streams etc, then create KSN and topo swaths
% Victoria M Buford Parks
% April 1, 2019
```

```
% Requires topo toolbox(, and maybe topographic analysis kit?)
clc;close;clear;

% dem='/Volumes/Files/VictoriaFiles/ArcGIS/DEM/GDEMTan-WGSUTM.tif';
DEM8 = GRIDobj(dem);
DEM8.Z(DEM8.Z>10000)=NaN;
DEM8.Z(DEM8.Z==0)=NaN;

DEMfill8=fillsinks(DEM8);
%you'll have to get the coordinates to crop with...
%DEMfill8=crop(DEMfill,x5,y5);
DEMfill8=crop(DEMfill);

% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%
% If you don't want to crop the DEM yourself, just
% load('DEM8_All.mat')
% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%--------------------%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp([datestr(clock) ' -- DEM 8:FlowObj'])
FDm8=FLOWobj(DEMfill8);%,'dinf');
disp([datestr(clock) ' -- DEM 8:FlowAcc'])
A8=flowacc(FDm8);
disp([datestr(clock) ' -- DEM 8:Streams'])
S8=STREAMobj(FDm8,'minarea',4e3);
disp([datestr(clock) ' -- DEM 8:Hydro Condition'])

% Hydrologically condition
theta_ref=0.5;
segment_length=2000;
zc=mincosthydrocon(S8,DEMfill8,'interp',0.2);
DEMc=GRIDobj(DEMfill8);
DEMc.Z(DEMc.Z==0)=NaN;

DEMc.Z(S8.IXgrid)=zc;
% %%
disp([datestr(clock) ' -- DEM 8:KsnComputation'])
[ksn,ksn_ms]=KSN_Quick(DEMfill8,DEMc,A8,S8,theta_ref,segment_length);

%
disp([datestr(clock) ' -- DEM 8:Export to Shapefile for use in Arc'])
shapewrite(ksn_ms,'ksn_S8-2km.shp')

%%
[xx,yy]=getcoordinates(DEMfill8);
[X,Y]=meshgrid(xx,yy);
disp([datestr(clock) ' -- DEM 8:Gridding Ksn'])
ksn_cell=cell(numel(ksn_ms),1);
for ii=1:numel(ksn_ms)
    ksn_cell{ii}=ones(numel(ksn_ms(ii).X),1)*ksn_ms(ii).ksn;
```

```
end
ksn_x=vertcat(ksn_ms.X); ksn_y=vertcat(ksn_ms.Y); ksn_ksn=vertcat(ksn_cell{:});

Fk=scatteredInterpolant(ksn_x,ksn_y,ksn_ksn);
ksn_int=Fk(X,Y);
KSNGrid8=GRIDobj(xx,yy,ksn_int);
%
IDX=GRIDobj(DEM,'logical');
KSNGrid8.Z(IDX.Z)=NaN;

M=IDX;
M.Z(~isnan(DEMfill8.Z))=true;
KSNGrid8=crop(KSNGrid8,M,NaN);


%% % The above Data is stored in DEM8_All.mat
% Rak17/McQ08Swath Then SwathOBJ
width=50;% in km
% 1.0e+06 *
%
%    0.5666    8.2086
%    0.8113    8.3613
%    0.8113    8.3613
 x=[0.5666;0.8113]*1e6;
 y=[8.2086;8.3613]*1e6;
 disp([datestr(clock) ' -- DEM 8:Swath Topo'])
 swathtopo=SWATHobj(DEMfill8,x,y,'width',width*1e3);
%swathtopo=SWATHobj(DEMfill8,'width',width*1e3);
disp([datestr(clock) ' -- DEM 8:Swath Ksn'])
swathKsn=SWATHobj(KSNGrid8,swathtopo.xy0(1:2,1),swathtopo.xy0(1:2,2),'width',width*1e3);

% %%
figure(1)
imagesc(DEMfill8)
demcmap([0 6500])
colorbar
hold on
plot(swathtopo.X,swathtopo.Y,'r') %shows where swath data is from
plot(swathtopo.xy0(1:2,1),swathtopo.xy0(1:2,2),'k')%Center/sectionline

hold off



% %%
figure(2)
subplot(2,1,1)
plotdz(swathtopo)
axis([0 3e5 0 6500])
title('Swath Topo')

%axis([0 3e5 0 6500])
hold on
%yyaxis right
subplot(2,1,2)
```

```
plotdz(swathKsn)
ylabel('Ksn []')
axis([0 3e5 0 1000])
% %%

%% Get Coords of Box
MinX=min(min(swathtopo.X));
ID=find(swathtopo.X==MinX);
MinY=swathtopo.Y(ID);

MinYY=min(min(swathtopo.Y));
ID=find(swathtopo.Y==MinYY);
MinXX=swathtopo.X(ID);

MaxX=max(max(swathtopo.X));
ID=find(swathtopo.X==MaxX);
MaxY=swathtopo.Y(ID);

MaxYY=max(max(swathtopo.Y));
ID=find(swathtopo.Y==MaxYY);
MaxXX=swathtopo.X(ID);
figure(3)
imagesc(DEMfill8)
demcmap([0 6500])
hold on
plot([MinX,MinXX,MaxX,MaxXX,MinX],[MinY,MinYY,MaxY,MaxYY,MinY],'-')%,'MarkerSize',10)
hold off

%%


sz=size(swathtopo.Z);
mean_Sw=zeros(sz(2),1);
median_Sw=zeros(sz(2),1);
mode_Sw=zeros(sz(2),1);
std_Sw=zeros(sz(2),1);
min_Sw=zeros(sz(2),1);
max_Sw=zeros(sz(2),1);

for i=1:sz(2)
   mean_Sw(i)=mean(swathtopo.Z(:,i));
   median_Sw(i)=median(swathtopo.Z(:,i));
   mode_Sw(i)=mode(swathtopo.Z(:,i));
   std_Sw(i)=std(swathtopo.Z(:,i));
   min_Sw(i)=min(swathtopo.Z(:,i));
   max_Sw(i)=max(swathtopo.Z(:,i));
end
% %%
mode_SwAvg=zeros(sz(2),1);
%mean over 3 widths
mode_width=8;
for i=1+mode_width:sz(2)-(1+mode_width)
   %mean_Sw(i)=mean(Perez16.Z(:,i));
   %median_Sw(i)=median(Perez16.Z(:,i));
```

```
    datta=swathtopo.Z(:,i-mode_width:i+mode_width);
    mode_SwAvg(i)=mode(datta(:));
    %std_Sw(i)=std(Perez16.Z(:,i));
end
%%
sz=size(swathKsn.Z);
mean_SwKsn=zeros(sz(2),1);
median_SwKsn=zeros(sz(2),1);
mode_SwKsn=zeros(sz(2),1);
std_SwKsn=zeros(sz(2),1);
min_SwKsn=zeros(sz(2),1);
max_SwKsn=zeros(sz(2),1);

for i=1:sz(2)
    mean_SwKsn(i)=mean(swathKsn.Z(:,i));
    median_SwKsn(i)=median(swathKsn.Z(:,i));
    mode_SwKsn(i)=mode(swathKsn.Z(:,i));
    std_SwKsn(i)=std(swathKsn.Z(:,i));
    min_SwKsn(i)=min(swathKsn.Z(:,i));
    max_SwKsn(i)=max(swathKsn.Z(:,i));
end

% %%
mode_SwAvgKsn=zeros(sz(2),1);
%mean over 3 widths
mode_width=8;
for i=1+mode_width:sz(2)-(1+mode_width)
    %mean_Sw(i)=mean(Perez16.Z(:,i));
    %median_Sw(i)=median(Perez16.Z(:,i));
    datta=swathKsn.Z(:,i-mode_width:i+mode_width);
    mode_SwAvgKsn(i)=mode(datta(:));
    %std_Sw(i)=std(Perez16.Z(:,i));
end

%%
f3.Renderer='painters'; % to ensure that the file is editable in AI.
saveas(f3,strcat('/Volumes/Files/AreaMap'),'epsc')

f2.Renderer='painters'; % to ensure that the file is editable in AI.
saveas(f2,strcat('/Volumes/Files/Swaths'),'epsc')

%%
function [ksn,ksn_ms]=KSN_Quick(DEM,DEMc,A,S,theta_ref,segment_length)
    g=gradient(S,DEMc);
    G=GRIDobj(DEM);
    G.Z(S.IXgrid)=g;

    Z_RES=DEMc-DEM;

    ksn=G./(A.*(A.cellsize^2)).^(-theta_ref);

    ksn_ms=STREAMobj2mapstruct(S,'seglength',segment_length,'attributes',...
        {'ksn' ksn @mean 'uparea' (A.*(A.cellsize^2)) @mean 'gradient' G @mean 'cut_fill' Z_RES @mean});
end
```

7.3.3 Ksn Extraction from CASCADE Output Topo Tec Files

Saved as ‘Cascade_GeomorphologyAnalysis.m’.

```
%% Instructions and notes
% Script that extracts user-defined topography swaths and normalised river
% steepness values.
% Simply follow the prompts in the command window.
% Note: Imported DEMs need to be in WGS 1984 (for now...)
%
% Modified by Victoria M. Buford Parks (Dec 2020) from
% Paul R. Eizenhˆfer, Eberhard-Karls-Universit‰t T¸bingen
% paul-reinhold.eizenhoefer@uni-tuebingen.de
% January 2020

% October 2020:
%   - added local relief output
%   - added precipitation output (requires TRMM precipitation file)
%   - cleaned up plots

%% Initiate script
tic
clear

% Analyses options
UserInput = 'DEM Type? (GeoTiff = 0, CASCADE = 1) ';
DEMInput = input(UserInput);

if DEMInput == 1
   disp('Locate CASCADE output file topo_tec_****.dat.');
   [File_CASCADE, Path_CASCADE] = uigetfile('topo_tec_****.dat', 'Select CASCADE output file.');

   CASCADE_File = readtable([Path_CASCADE File_CASCADE], 'HeaderLines', 4);

   OutputVariables = {'x_km','y_km','z_km','node','precipitation_mpy',...
      'fluvial_erosion_rate_mpy','diffusion_erosion_rate_mpy',...
      'landslide_erosion_rate_mpy','total_erosion_rate_mpy',...
      'catchment_color','catchment_number',...
      'glacial_erosion_rate_mpy','ice_thickness_m','mass_balance_1py',...
      'total_topography_m','sliding_velocity_mpy','gerode_term_mpy',...
      'rock_contact_km','isostatic_deflection_mpy','slope_mpkm',...
      'totalflexiso_m','constriction','cumulative_erosion_m',...
      'surface_area_km2'}';
   CASCADE_File.Properties.VariableNames = OutputVariables';

   % Clean up output - Remove NaNs
   Nanss = ismissing(CASCADE_File,{NaN}); % creates table size of A, with 1 or 0 for NaN
   NanNodes = Nanss(:,4); clear Nanss % pull only the nodes column
   CASCADE_File = CASCADE_File(~NanNodes,:); clear NanNodes % remove any row that the node is
NaN

   % Interpolation to regular grid
```

```
    UserInput = ('Choose DEM resolution in [m]: ');
    DEMResolution = input(UserInput);

    UserInput = ('Choose DEM width in [m] (should correspond to CASCADE model width): ');
    DEMWidth = input(UserInput);

    UserInput = ('Choose DEM length in [m] (should correspond to CASCADE model length): ');
    DEMLength = input(UserInput);

    [x_DEM,y_DEM] = meshgrid(0:DEMResolution:DEMWidth, 0:DEMResolution:DEMLength);
    z_DEM = griddata(CASCADE_File.x_km * 1e3, CASCADE_File.y_km * 1e3, CASCADE_File.z_km *
1e3, x_DEM, y_DEM, 'natural');
    Erosion_DEM = ...
        griddata(CASCADE_File.x_km * 1e3, CASCADE_File.y_km * 1e3,
CASCADE_File.total_erosion_rate_mpy, x_DEM, y_DEM, 'natural');
    Precipitation_DEM = ...
        griddata(CASCADE_File.x_km * 1e3, CASCADE_File.y_km * 1e3,
CASCADE_File.precipitation_mpy, x_DEM, y_DEM, 'natural');

    DEM = GRIDobj(x_DEM, y_DEM, z_DEM);
    DEM_Erosion = GRIDobj(x_DEM, y_DEM, Erosion_DEM);
    DEM_Precipitation = GRIDobj(x_DEM, y_DEM, Precipitation_DEM);

    % Define Swath Geometry
    UserInput = 'Enter [longitude latitude] of profile start in [m]: ';
    ProfileStartInput = input(UserInput);

    UserInput = 'Enter [longitude latitude] of profile end in [m]: ';
    ProfileEndInput = input(UserInput);

else
    disp('Select DEM for geomorphic analysis...')
    [File_DEM, Path_DEM] = uigetfile('****.tif', 'Locate DEM');

    UserInput = 'DEM resolution in [m]? ';
    DEM_Resolution = input(UserInput);

    disp('DEM is loading...')

    DEM = GRIDobj([Path_DEM File_DEM]);
    DEM = reproject2utm(DEM, DEM_Resolution);

    disp('Done.')

    disp('Select TRMM file for precipitation swath...')
    [File_Precipitation, Path_Precipitation] = uigetfile('****.tif', 'Locate TRMM file');

    UserInput = 'Precipitation data resolution in [m]? ';
    Precipitation_Resolution = input(UserInput);

    UserInput = 'Enter UTM Zone [integer]: ';
    UTM_Zone = input(UserInput);
```

```
    UserInput = 'Northern or southern hemisphere (N = 0; S = 1)? ';
    HemisphereCheck = input(UserInput);

    disp('Precipitation data are loading...')

    PrecipitationData = GRIDobj([Path_Precipitation File_Precipitation]);

    disp('Done.')

    % Define Swath Geometry
    UserInput = 'Enter [longitude latitude] of profile start in [decimal degrees]: ';
    ProfileStartInput = input(UserInput);

    UserInput = 'Enter [longitude latitude] of profile end in [decimal degrees]: ';
    ProfileEndInput = input(UserInput);
end

UserInput = 'Swath profile resolution? (in [m]) ';
ProfileResolution = input(UserInput);

UserInput = 'Width of analysis swath? (in [m]) ';
ProfileWidth = input(UserInput);

%% TOPOGRAPHY SWATH

disp('Extracting topography and precipitation swaths...')

if DEMInput == 1
    ProfileStart = ProfileStartInput;
    ProfileEnd = ProfileEndInput;
else
    [ProfileStart(1), ProfileStart(2)] = ll2utm(ProfileStartInput(2), ProfileStartInput(1));
    [ProfileEnd(1), ProfileEnd(2)] = ll2utm(ProfileEndInput(2), ProfileEndInput(1));
end

ProfileLength = sqrt((ProfileStart(1) - ProfileEnd(1))^2 + (ProfileStart(2) - ProfileEnd(2))^2);
ProfileAngle = asin(abs(ProfileStart(2) - ProfileEnd(2)) / ProfileLength); % Note: angle in radians

% Extract Swath Topography

% Define temporary profile end points
% Swath to the left
for i = 1:round(ProfileWidth / ProfileResolution / 2)
    ProfileStart(i+1,1) = ProfileStart(1,1) - i * cos(pi/2 - ProfileAngle) * ProfileResolution;
    ProfileStart(i+1,2) = ProfileStart(1,2) + i * sin(pi/2 - ProfileAngle) * ProfileResolution;
    ProfileEnd(i+1,1) = ProfileEnd(1,1) - i * cos(pi/2 - ProfileAngle) * ProfileResolution;
    ProfileEnd(i+1,2) = ProfileEnd(1,2) + i * sin(pi/2 - ProfileAngle) * ProfileResolution;
end

% Swath to the right
for j = 1:round(ProfileWidth / ProfileResolution / 2)
    ProfileStart(i+j+1,1) = ProfileStart(1,1) + j * sin(ProfileAngle) * ProfileResolution;
```

```
    ProfileStart(i+j+1,2) = ProfileStart(1,2) - j * cos(ProfileAngle) * ProfileResolution;
    ProfileEnd(i+j+1,1) = ProfileEnd(1,1) + j * sin(ProfileAngle) * ProfileResolution;
    ProfileEnd(i+j+1,2) = ProfileEnd(1,2) - j * cos(ProfileAngle) * ProfileResolution;
end

% Extract profiles
% Extract topographic profiles
if DEMInput == 0
    if (UTM_Zone > 30 && HemisphereCheck == 0)
        PrecipitationData = crop(PrecipitationData, [0 0 180.02 180.02], [0 36.02 0 36.02]);
    elseif (UTM_Zone > 30 && HemisphereCheck == 1)
        PrecipitationData = crop(PrecipitationData, [0 0 180.02 180.02], [0 -36.02 0 -36.02]);
    elseif (UTM_Zone < 30 && HemisphereCheck == 0)
        PrecipitationData = crop(PrecipitationData, [0 0 -180.02 -180.02], [0 36.02 0 36.02]);
    elseif (UTM_Zone < 30 && HemisphereCheck == 1)
        PrecipitationData = crop(PrecipitationData, [0 0 -180.02 -180.02], [0 -36.02 0 -36.02]);
    end
    PrecipitationData = reproject2utm(PrecipitationData, Precipitation_Resolution);
end

for k = 1:(j+i+1)
    [temp_d,temp_z,~,~] = ...
        demprofile(DEM,round(ProfileLength/ProfileResolution),...
        [ProfileStart(k,1) ProfileEnd(k,1)], [ProfileStart(k,2) ProfileEnd(k,2)]);
    if DEMInput == 1
        [~,temp_Erosion,~,~] = ...
            demprofile(DEM_Erosion, round(ProfileLength/ProfileResolution),...
            [ProfileStart(k,1) ProfileEnd(k,1)], [ProfileStart(k,2) ProfileEnd(k,2)]);
        [~,temp_Precipitation,~,~] = ...
            demprofile(DEM_Precipitation, round(ProfileLength/ProfileResolution),...
            [ProfileStart(k,1) ProfileEnd(k,1)], [ProfileStart(k,2) ProfileEnd(k,2)]);
        ProfileErosion(k,:) = temp_Erosion;
        ProfilePrecipitation(k,:) = temp_Precipitation;
    else
        [~,temp_Precipitation,~,~] = ...
            demprofile(PrecipitationData, round(ProfileLength/ProfileResolution),...
            [ProfileStart(k,1) ProfileEnd(k,1)], [ProfileStart(k,2) ProfileEnd(k,2)]);
        ProfilePrecipitation(k,:) = temp_Precipitation;
    end
    ProfileDistance = temp_d;
    ProfileElevation(k,:) = temp_z;

end
% Calculate minimum, maximum and mean topographies and precipitation
for k = 1:length(ProfileDistance)
    MaximumTopography(k) = max(ProfileElevation(:,k));
    MeanTopography(k) = nanmean(ProfileElevation(:,k));
    MinimumTopography(k) = min(ProfileElevation(:,k));

    if DEMInput == 1
        MeanErosion(k) = nanmean(ProfileErosion(:,k));
        MeanPrecipitation(k) = nanmean(ProfilePrecipitation(:,k));
    else
        MaximumPrecipitation(k) = max(ProfilePrecipitation(:,k)/1e3);
        MeanPrecipitation(k) = nanmean(ProfilePrecipitation(:,k)/1e3);
```

```
        MinimumPrecipitation(k) = min(ProfilePrecipitation(:,k)/1e3);
    end
end

Topography = [ProfileDistance MeanTopography' MaximumTopography' MinimumTopography'];

if DEMInput == 1
    Erosion = [ProfileDistance MeanErosion'];
    Precipitation = [ProfileDistance MeanPrecipitation'];
else
    Precipitation = [ProfileDistance MeanPrecipitation' MaximumPrecipitation' MinimumPrecipitation'];
end

disp('Done.')

%% KSN SWATH
disp('Performing local gradient and Ksn analysis...')

% DEM flow routing and sink filling
FD = FLOWobj(DEM,'preprocess','carve');
DEM = imposemin(FD,DEM);
DEM = fillsinks(DEM);
FDm = FLOWobj(DEM,'dinf');
% FDm = Flowobj(DEM, 'multi');
A = flowacc(FDm);

% Calculate stream network
S = STREAMobj(FDm, 'unit', 'mapunits', 'minarea', 1e7); % Define minimum drainage area here

% Calculate Ksn values and drainage area for all streams
G = gradient8(DEM);
DRAINAGE_AREA = A.*(A.cellsize^2);
KSN = G./DRAINAGE_AREA.^-0.45; % Define stream concavity index here

% Extract streams
[x,y,ksn,localgradient,drainage_area] = STREAMobj2XY(S,KSN,G,DRAINAGE_AREA);
Streams = [x, y, ksn, localgradient, drainage_area];

% Filter data (NaN's and ksn = 0 will be removed)
Streams(any(isnan(Streams),2),:) = [];
Streams(Streams(:,3) == 0,:) = [];

% Rotation
RotationAngle = -rad2deg(ProfileAngle);
Rotation = rotz(RotationAngle); % insert rotation angle here (counterclockwise)
RotationCentre = [ProfileStart(1,1); ProfileStart(1,2); 0]; % insert rotation centre here, ideally beginning of section line

% All streams
Streams_rotated = (Rotation * ([Streams(:,1)'; Streams(:,2)'; Streams(:,3)'] - RotationCentre)) + RotationCentre;
Streams_rotated = [Streams_rotated; Streams(:,4:5)'];
disp('Done.')
```

```
% Extract ksn values
% Area of interest
x_range = [RotationCentre(1); RotationCentre(1) + ProfileLength]; % define length of section starting at the rotation centre
y_range = [RotationCentre(2) - ProfileWidth;  RotationCentre(2) + ProfileWidth]; % define area across section

% All streams
Streams_filtered = Streams_rotated(:,...
    Streams_rotated(1,:) > x_range(1) & Streams_rotated(1,:) < x_range(2) &...
    Streams_rotated(2,:) > y_range(1) & Streams_rotated(2,:) < y_range(2));
KsnValues = bin(Streams_filtered(1,:), Streams_filtered(3,:), round(ProfileLength/ProfileResolution));
LocalGradient = bin(Streams_filtered(1,:), Streams_filtered(4,:), round(ProfileLength/ProfileResolution));% Define number of bins
% Streams_binned_SmoothMedian = smoothdata(Streams_binned(:,2), 'movmedian'); % Define width of moving average window

%% Plots
% Topography and Ksn
fig = figure;
set(fig,'defaultAxesColorOrder',[[0 0 0]; [0 0 0]]);

if DEMInput == 1
    subplot(3,4,1)
else
    subplot(3,3,1)
end
imageschs(DEM,[],'colormap',[1 1 1],'colorbar',false,'ticklabel','nice');

hold on

plot([ProfileStart(1,1); ProfileEnd(1,1)], [ProfileStart(1,2); ProfileEnd(1,2)], 'r:', 'LineWidth', 2)
plot([ProfileStart(i+1,1); ProfileEnd(i+1,1)], [ProfileStart(i+1,2); ProfileEnd(i+1,2)], 'r', 'LineWidth', 2)
plot([ProfileStart(i+j+1,1); ProfileEnd(i+j+1,1)], [ProfileStart(i+j+1,2); ProfileEnd(i+j+1,2)], 'r', 'LineWidth', 2)
plot(ProfileStart(:,1), ProfileStart(:,2), 'r', 'LineWidth', 2)
plot(ProfileEnd(:,1), ProfileEnd(:,2), 'r', 'LineWidth', 2)

xlabel('Longitude [m]')
ylabel('Latitude [m]')
title({'River Steepness';...
    ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
    '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});

plotc(S,KSN);
h1 = colorbar; hcb_title1 = get(h1, 'Title'); set(hcb_title1, 'String', 'K_{sn}');
colormap(h1, jet);
caxis([0 1000])

% Topography and Local Relief
if DEMInput == 1
    subplot(3,4,2)
else
    subplot(3,3,2)
```

```
end
imageschs(DEM,G,'colormap','parula','ticklabel','nice','colorbarylabel','Slope [ ]','caxis',[0 1])

hold on

plot([ProfileStart(1,1); ProfileEnd(1,1)], [ProfileStart(1,2); ProfileEnd(1,2)], 'r:', 'LineWidth', 2)
plot([ProfileStart(i+1,1); ProfileEnd(i+1,1)], [ProfileStart(i+1,2); ProfileEnd(i+1,2)], 'r', 'LineWidth', 2)
plot([ProfileStart(i+j+1,1); ProfileEnd(i+j+1,1)], [ProfileStart(i+j+1,2); ProfileEnd(i+j+1,2)], 'r', 'LineWidth', 2)
plot(ProfileStart(:,1), ProfileStart(:,2), 'r', 'LineWidth', 2)
plot(ProfileEnd(:,1), ProfileEnd(:,2), 'r', 'LineWidth', 2)

xlabel('Longitude [m]')
ylabel('Latitude [m]')
title({'Local Relief';...
    ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
    '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});


h2 = colorbar; hcb_title2 = get(h2, 'Title'); set(hcb_title2, 'String', 'Slope [ ]');
colormap(parula);

% Precipitation
if DEMInput == 1
    subplot(3,4,3)
    imageschs(DEM,DEM_Precipitation,'colormap','cool','ticklabel','nice','colorbarylabel','Precipitation
[m/yr]')

    hold on

    plot([ProfileStart(1,1); ProfileEnd(1,1)], [ProfileStart(1,2); ProfileEnd(1,2)], 'r:', 'LineWidth', 2)
    plot([ProfileStart(i+1,1); ProfileEnd(i+1,1)], [ProfileStart(i+1,2); ProfileEnd(i+1,2)], 'r', 'LineWidth', 2)
    plot([ProfileStart(i+j+1,1); ProfileEnd(i+j+1,1)], [ProfileStart(i+j+1,2); ProfileEnd(i+j+1,2)], 'r', 'LineWidth',
2)
    plot(ProfileStart(:,1), ProfileStart(:,2), 'r', 'LineWidth', 2)
    plot(ProfileEnd(:,1), ProfileEnd(:,2), 'r', 'LineWidth', 2)

    xlabel('Long. [m]')
    ylabel('Lat. [m]')
    title({'Precipitation';...
        ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
        '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
else
    subplot(3,3,3)

    imagesc(PrecipitationData)

    hold on

    plot([ProfileStart(1,1); ProfileEnd(1,1)], [ProfileStart(1,2); ProfileEnd(1,2)], 'r:', 'LineWidth', 2)
    plot([ProfileStart(i+1,1); ProfileEnd(i+1,1)], [ProfileStart(i+1,2); ProfileEnd(i+1,2)], 'r', 'LineWidth', 2)
    plot([ProfileStart(i+j+1,1); ProfileEnd(i+j+1,1)], [ProfileStart(i+j+1,2); ProfileEnd(i+j+1,2)], 'r', 'LineWidth',
2)
    plot(ProfileStart(:,1), ProfileStart(:,2), 'r', 'LineWidth', 2)
```

```
    plot(ProfileEnd(:,1), ProfileEnd(:,2), 'r', 'LineWidth', 2)

    xlabel('Longitude [m]')
    ylabel('Latitude [m]')
    title({'TRMM Precipitation';...
        ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
        '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
    xlim(xlim(h1))
    ylim(ylim(h1))
end

caxis([0 5])
h3 = colorbar; hcb_title3 = get(h3, 'Title'); set(hcb_title3, 'String', 'Precipitation [m/yr]');
colormap(h3, cool);



% Erosion (LEMs only, unless erosion maps exist at some point...)
if DEMInput == 1
    subplot(3,4,4)
    imageschs(DEM,DEM_Erosion,'ticklabel','nice','colorbarylabel','Erosion [m/yr]')

    hold on

    plot([ProfileStart(1,1); ProfileEnd(1,1)], [ProfileStart(1,2); ProfileEnd(1,2)], 'r:', 'LineWidth', 2)
    plot([ProfileStart(i+1,1); ProfileEnd(i+1,1)], [ProfileStart(i+1,2); ProfileEnd(i+1,2)], 'r', 'LineWidth', 2)
    plot([ProfileStart(i+j+1,1); ProfileEnd(i+j+1,1)], [ProfileStart(i+j+1,2); ProfileEnd(i+j+1,2)], 'r', 'LineWidth',
2)
    plot(ProfileStart(:,1), ProfileStart(:,2), 'r', 'LineWidth', 2)
    plot(ProfileEnd(:,1), ProfileEnd(:,2), 'r', 'LineWidth', 2)

    xlabel('Long. [m]')
    ylabel('Lat. [m]')
    title({'Erosion';...
        ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
        '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
    h4 = colorbar; hcb_title4 = get(h4, 'Title'); set(hcb_title4, 'String', 'Erosion [m/yr]');
    %colormap(h4, turbo);
    caxis([-5e6 0])
end

% Ksn and Topography Swaths
if DEMInput == 1
    subplot(3,4,[5,8])
else
    subplot(3,3,[4,6])
end
x_temp = [Topography(:,1)', fliplr(Topography(:,1)')];
TopographySwath = [Topography(:,3)', fliplr(Topography(:,4)')];
fill(x_temp, TopographySwath, [0.86, 0.86, 0.86]);
hold on;
plot(Topography(:,1)', Topography(:,3)','r');
plot(Topography(:,1)', Topography(:,2)', 'k');
plot(Topography(:,1)', Topography(:,4)','b');
```

```
xlabel('Distance along Swath [m]')
ylabel('Elevation [m]')
title({'Topography and River Steepness';...
   ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
   '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
box on
grid on

yyaxis right
plot(KsnValues(:,1) - RotationCentre(1), KsnValues(:,2), 'm-');

ylabel('Median K_{sn}')
legend([num2str(ProfileWidth) ' m Swath'], 'Max. Topography', 'Mean Topography', 'Min. Topography',...
   'Median K_{sn}')
ax=gca;
ax.XDir='reverse'

% Precipitation and Local Relief
if DEMInput == 1
   subplot(3,4,[9,12])

   plot(Precipitation(:,1)', Precipitation(:,2)','b');
   hold on
   plot(Erosion(:,1)', -Erosion(:,2)', 'r', 'LineWidth', 2)
   ylabel('Precipitation/Erosion [m/yr]')
   box on
   grid on

   yyaxis right

   plot(LocalGradient(:,1) - RotationCentre(1), LocalGradient(:,2), 'g-');
   xlabel('Distance along Swath [m]')
   ylabel('Slope [ ]')
   title({'Local Gradient and Precipitation';...
      ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
      '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
   legend('Precipitation', 'Erosion', 'Local Relief')
   ax=gca;
   ax.XDir='reverse';
else
   subplot(3,3,[7,9])

   x_temp = [Precipitation(:,1)', fliplr(Precipitation(:,1)')];
   PrecipitationSwath = [Precipitation(:,3)', fliplr(Precipitation(:,4)')];
   fill(x_temp, PrecipitationSwath, [0.86, 0.86, 0.86]);
   hold on;
   plot(Precipitation(:,1)', Precipitation(:,3)','r');
   plot(Precipitation(:,1)', Precipitation(:,2)', 'k');
   plot(Precipitation(:,1)', Precipitation(:,4)','b');
   ylabel('Precipitation [m/yr]')
   box on
   grid on

   yyaxis right
```

```
    plot(LocalGradient(:,1) - RotationCentre(1), LocalGradient(:,2), 'g-');
    xlabel('Distance along Swath [m]')
    ylabel('Slope [ ]')
    title({'Local Gradient and Precipitation';...
        ['[' num2str(round(ProfileStartInput(1), 2)) ' ' num2str(round(ProfileStartInput(2), 2))...
        '] to [' num2str(round(ProfileEndInput(1), 2)) ' ' num2str(round(ProfileEndInput(2), 2)) ']']});
    legend([num2str(ProfileWidth) ' m Swath'], 'Max. Precipitation', 'Mean Precipitation', 'Min. Precipitation',...
        'Local Relief')
end

%% Clean up
% CASCADE file input
clear File_CASCADE File_DEM OutputVariables Path_CASCADE Path_DEM UserInput x_DEM y_DEM
z_DEM
clear Erosion_DEM Precipitation_DEM localgradient

% Topography section
clear temp_d temp_x temp_y temp_z UserInout ProfileLatitude ProfileLongitude ProfileAngle Path_DEM i
j k KsnOutput
clear HemisphereCheck File_DEM ProfileDistance ProfileElevation x_temp ProfileStart ProfileEnd
TopographyOutput
clear MaximumTopography MinimumTopography MeanTopography TopographySwath UserInput
clear ProfileErosion ProfilePrecipitation temp_Erosion temp_Precipitation MeanErosion MeanPrecipitation

% Ksn section
clear Rotation A drainage_area FD h h1 h2 h3 h4 hcb_title1 hcb_title2 hcb_title3 hcb_title4
clear ksn KsnSwath RotationAngle Streams
clear Streams_rotated Streams_filtered x y x_range y_range fig

toc
```

8.0 Appendix 4: CASCADE Fortran90 Files

These are the files edited for the version of the executable used to produce the imposed precipitation models discussed in Chapter 4.

8.1 CASCADE Input Files

This is an example CASCADE input file 'icecascade.in' used with the executable that allows rear boundary uplift and imposed uplift.

```
# Input file for coupled Ice and Cascade Program.  All input parameters
# are specified in this file, no need to change source code and recompile
# for different simulations.
# version W. Kappler, B. Yanites, T. Ehlers, 15 Nov. 2010
#
# *****************************
# ****MODEL PROCESSES
# INPUT A
# *****************************

# INPUT A1 *** Enable fluvial erosion
# fluvial_erosion
T

# INPUT A2 *** Enable fluvial deposition [T or F]
# ideposition
T

# INPUT A3 *** Enable hillslope diffusion [T or F]
# idiffusion
T

# INPUT A4 *** Enable landslide erosion parameters [T or F]
# ilandslide
T
```

```
# INPUT A5 *** Flag to permanently turn on/off Ice. T = (true) on, F = (false) off
# If set to F it will never call ice/grow a glacier.
# iceflag
F

# INPUT A6 *** Glacial erosion
# glacial_erosion
F

# INPUT A7 *** flag to permit flexural isostasy (F = no flexure)
# iflexure
F

# INPUT A8a *** Allow horizontal advection of nodes and topography. Only required if
# vy and vx are non-zero.
# ihorizontal
T

# INPUT A8b *** Select mode for horizontal movement (added by PRE Jan2017)
# 1 = Global value applied throughout the model domain as defined in input M2a and M2b
# 2 = Variable horizontal movement as given by a MOVE/Pecube velocity file
# ihorizontal_mode
2

# INPUT A9 *** Enable orographic precipitation [T or F]
# If iflag_oro = F, the iflag_uni is set to T (uniform precipitation
# iflag_oro
# Edited by vmbp, Jan 13 2020
# 1 = No, Global precipitation
# 2 = Yes, Dynamic Orographic Precipitation from Rainmaker
# 3 = Imposed Precipitation file

3

# INPUT A9B *** Filename for input precipitation
# Added by vmbp, Jan 13 2020
NewPrecip19.txt

# INPUT A9C *** Update time for imposed precipitation in years
# Added by vmbp, Jan 13 2020
100000

# INPUT A10 *** Higher-Order Ice Physics Switch (using COMSOL)
# .FALSE. = just use update_height from BY (modified SIA)
# .TRUE. = use update_height_comsol
# use_comsol
```

```
F

# ******************************
# ****MODEL OUTPUT
# INPUT B
# ******************************

# INPUT B1 *** Define type of output wanted
# F (=false): All output files written
# T (=true): Only the tecplot formatted output files and the
#    geometry and topography files are written
# tecflag
T

# INPUT B2 *** Frequency of screen output [time steps]
# nshortwrite
50000

# INPUT B3 *** Frequency of flux outputs [time steps]
# (=0 no flux save otherwise frequency in time steps)
# this output is of total flux through the fixed nodes on the boundaries (if any)
# iflux
0

# INPUT B4 *** Frequency of writing model output [y]
# When only a cascade simulation is run the writetime varialbe is what used.
# When a cascade and ice simulation are run then the writeice variable is used.
# If it is a mixed simulation (e.g. 2 Myr Cascade only, then ice), the writetime
# variable will be used until ice is called.
# writetime
# writeice
#20000
#20000
100000
100000


# INPUT B5 *** name of this run; must also be the name
# of an existing folder where all output files will be stored
# If the folder does not exits, it will be created
# run_name
IceCascade

# ******************************
# ****MODEL SET UP: TIME DOMAIN
# INPUT C
```

```
# ******************************

# INPUT C1 *** Enable dynamic time step calculation
# iadjust
T

# INPUT C2 *** Initial time step length [y]
# dt0
25

# INPUT C3 *** Total model run time [y]
# endtime
# 1e7
2.41e6


#******************************
#****MODEL SET UP: SPATIAL DOMAIN
# INPUT D
#******************************

# INPUT D1 *** Number of nodes along x and y edges of model, note these will be changed if
# addrim=1 after addrim subroutine.  If readmesh is on to read in an initial topography,
# make sure these values equal number x-y values in the tecplot file.
# BY: DELETE THIS LINE?  If creating SS fluvial, be sure to turn addrim off.
# nx
# ny
51
151

# INPUT D2 *** Length/size of x and y sides of the model [km]
# sidex
# sidey
50
150

# INPUT D3 *** Enable adaptive remeshing
# iadapt
F

# INPUT D4 *** Allow Cascade to read in an initial topography/mesh with dimensions the same
as above
# meshread
T

# INPUT D5 *** Name of the initial topography to read in [file_name].
```

This is only used if meshread = T.
meshname
N9v163_topo_tec_0017.dat

INPUT D6 *** Specify the format of the meshfile above.
1 = tecplot formatted file:
File needs to include number of nodes, and then columns for x,y,h,h0 for each node.
h and h0 are the surface and bedrock elevations (?)
the difference between the two is the sediment thickness
2 = ascii file:
The file should contain nx by ny elevation points
with the elevations of each point in [m].
File should be 3 columns with row=nx*ny
Column 1: x coordinate (m); column 2: y coordinate (m); column 3: elevation (m)
3 = MOVE ascii topography export
The 2DMOVE topography should be exported as [x y z PointID] ascii file
meshformat
1

INPUT D7 *** Add 'shelf' to boundary of model domain to allow glaciers to flow out of valleys
and not disappear out of model domain, 0,0 is on lower left so Exl is extended rim left,
Eyd is extended rim down [km]
The 'shelf' in reality could be a below modern sea level continental shelf
or an above sea-level coastal plain or a piedmont.
Exr = distance shelf will be created to the RIGHT of the domain y-axis specified above. (optional)
Exl = distance shelf will be created to the LEFT of the domain y-axis specified above. (optional)
Exd = distance shelf will be created to the DOWN/BELOW of the domain x-axis specified above. (optional)
Exu = distance shelf will be created to the UP/ABOVE of the domain x-axis specified above. (optional)
addshelf
Exr
Exl
Eyd
Eyu
T
0
0
0
0

INPUT D8 *** shelf/piedmont slopes for the above shelf added to the model:
positivite values slope away from original topography
slopexr

```
# slopexl
# slopeyd
# slopeyu
0.0
0.0
0.0
0.0

# INPUT D9 *** number of extra nodes to add in x and y direction for the shelf area added (if any).
# These nodes will only be added to the 'shelf' area, and can be set to have a coarser mesh on the edges of the model.
# Suggestion is to add both distances (e.g. Exl+Exr) from above and use an average
# node spacing of your liking.  This method is best used when reading a mesh.
# In that scenario, your shelf can have different nodal spacings.
# If you are generating new topograpy, make sure nxe, nye will generate
# the spacing you prefer (i.e. what nx and ny give you in INPUT D1).
# Essentially if a new mesh is generated, spacing in the model domain as well as on the
# shelf will be the same, so make sure nxe and nye (in relation to Exl,Exr,Eyd,Eyl) will
# give the proper nodal spacing.  Initialize_nodal_geometry.f90 does not differentiate between
# shelf and regular nodes.  Read_nodal_geometry.f90 does differentiate, so a different
# mesh spacing is possible if reading in a mesh.
# Note: you can run IceCascade for one time step and use that topography as an input mesh in order to
# use a different node spacing
# nxe
# nye
5
20

# INPUT D10: impose uplift at rear boundary
# If you want to T/F
# Uplift rate [mm/yr]
# Max elevation to obtain [km]
F
1
2


#*****************************
#****EROSION: FLUVIAL
# INPUT E
#*****************************

# INPUT E1 *** Fluvial erosion constant [unitless]
# xkf
```

```
1.0e-4

# INPUT E2 *** Bedrock erosion length scale [m]
# xlf_BR
1000.0

# INPUT E3 *** Coefficient controlling channel width as function of discharge [sqrt(yr/m)]
# width_c
0.1

# INPUT E4 *** Discharge threshold for channel formation [m km2/yr]
# thresh
4.0

# INPUT E5 *** Fluvial erosion length scale for alluvial material [m]
# xlf_AL
100.0

# INPUT E6 *** Define sea level elevation; no fluvial erosion below sea level [km]
# sea_level
0.0


#******************************
#****EROSION: HILLSLOPES
# INPUT F
#******************************

# INPUT F1 *** Hillslope diffusion coefficient [km2/yr]
# xkdiff
2.e-6

# INPUT F2 *** Landslide method to be used
# 1: probabilistic method (might need testing/debugging)
# 2: simple slope threshold
#    pmax is the critical slope [degrees]
#    distmax is the maximum distance of transport [???]
# 3: simpler slope threshold in landslide_simple.f
#    pmax is the critical slope [degrees]
# Landslide method 1 and 2 use subroutine landslide.f
# Landslide method 3 uses subroutine landslide_simple.f
# lsmeth
3

# INPUT F3 *** Threshold hillslope angle for landsliding (used for lsmeth=1-3) [degrees]
# Used by landslide method: 1,2 and 3
```

For method 2 and 3, this value dictates if a landslide occurs or not (if slope is greater
than this value, then a landslide occurs, if it's less, no landslide)
For method 1, this value is the maximum possible slope, but landslides can occur on slopes
less than this value. As slope steepens, the probability increases that a landslide will occur.
If the slope equals pmax, then the probability is 1. Probability also increases as a function of time
since last landslide (see xk0).
pmax
30.0

INPUT F4 *** Maximum transport distance (used for lsmeth=1,2) [m]
Used by landslide method: 1 and 2
distmax
0.0

INPUT F5 *** Parameters for lsmeth=1
Hillslope effective cohesion
I (Brian) think that a large value will give you hillslope angles that cluster near pmax
whereas a small value will give you hillslope angles of a greater distribution.
Used by landslide method: 1
cohes
0.0

INPUT F6 *** Density of hillslope material [kg/m3]
Used by landslide method: 1
rho
0.0

INPUT F7 *** Acceleration due to gravity [m/s2]
Used by landslide method: 1
grav
9.81

INPUT F8 *** Landslide frequency scaling parameters for lsmethod 1, suggested (?) values
xk0, a coefficient used along with time since last landslide for a given node to calculate
probability of a landslide occurring at that node. Can be thought of as a 'preparation' timescale
i.e. if you have a landslide, it will take some time (related to xk0) before another landslide
will occur. I (Brian) guess this means that if you set to 0, landslide probability will not depend on time since last landslide.
xk1, is the number of landslides per timescale set by dtc (below) for the average deluany area per node.
So if xk1=1 and dtc=100, then on average 1 landslide will occur every 100 yrs at a given node.
dtc is a measure of landslide frequency used in the probabilistic calculations. Used to divide
both the current timestep length as well as the time since last landslide to calculate a
landslide probability for a given node.
xk0=0.01, xk1=1., dtc = 100. yrs

```
# Used by landslide method: 1
# xk0
# xk1
# dtc
0.0
0.0
0.0


#******************************
#****ICE TIME SET UP
# INPUT G
#******************************

# INPUT G1 *** After ice is turned on, this parameter controls how often [yr] rainmaker is called.
# The one (calc_rain) set in initialize_general_parameters.f90 is used before ice is turned on.
# calc_rain_ice
0

# INPUT G2 *** Shallow Ice Approximation time step size (yr)
# shallow_ice
0

# INPUT G3 *** Maximum dh/dt allowed for each timestep in the ice thickness.  This prevents
# run-away glacier sizes.
# dh_allowed
0

# INPUT G4 *** Erosion time step (in yrs)
# dt_icetime
0

# INPUT G5 *** End of time stepping
# This must be zero if Cascade is calling ICE.
# If ICE is running alone this is the end time for the ICE simulation
# ice_tfinal
0.0

#******************************
#****ICE FLAGS
# INPUT H
#******************************

# INPUT H1 *** Flag to determin if you have polythermal or isothermal ice.
# T = polythermal, calculate glacier geotherms.
# F = isothermal, glacier temperature = surface temperature.
```

itemp
T

#******************************
#****ICE MECHANICS
INPUT I
#******************************

INPUT I1 *** Ice-flow constant, B (1/(sPa**3)), it is good to use Paterson's value but not necessary...
ice_flow
0

INPUT I2 *** Sliding law constant, Bs [units depend on value of the exponent below, e.g. 1/shearstress**n]
sliding_law
0

INPUT I3 *** Power law exponent, n
power_law_exp
0

INPUT I4 *** Exponent sliding law, ns (usually set between 2 and 3)
exp_sliding_law
0

INPUT I5 *** Density of ice (kg/m**3), rho_ice
was 'rho', now rho_ice
rho_ice
0

INPUT I6 *** Constriction constant - used to approximate higher order effects not implicitly covered in SIA approach. [1/m ?]
was 'gamma' now 'constriction'
constriction
0

#******************************
#****ICE EROSION**************
INPUT J
#******************************

INPUT J1 *** Erosion rate constant for glacial erosion
Sliding velocity is multiplied by this value. [unitless]
was 'alpha' now 'erosion_rate'

```
# erosion_rate
0

# INPUT J2 *** the power of the relationship between ice velocity and erosion.
# ice_ero_pow
0


#*****************************
#****ICE THERMAL FIELD
# INPUT K
#*****************************

# INPUT K1 *** Basal heat flux [Wm-2]
# basal_heat_flux
0

# INPUT K2 *** Ice conductivity [Wm-1k-1] (?)
# conductivity
0

#*****************************
#****MISCELLANEOUS ICE
# INPUT L
#*****************************

# INPUT L1 *** Threshold snow angle [degrees]
# Any snowcovered hillslope steeper than this angle will experience snow avalanching
# critangle
0

# INPUT L2 *** Calving rate coefficeint [1/yr]
# Coefficient to calculate the rate of calving velocity.
# This value times the depth of ice below water gives the rate.
# calvecoef
0


#*****************************
#**** TECTONICS (cascade code)
# INPUT M
#*****************************
# INPUT M1 *** uplift_mode
# 1 = Use global uplift_rate
# 2 = Use MOVE_velocity_file
# 3 = Use text file with list of MOVE grids and their corresponding time
```

3

INPUT M1a *** uplift_rate
5.0

INPUT M1b *** MOVE_velocity_file (M1 = 2) OR text file listing all MOVE grids (M1 = 3)
McQ02N9_VV16-end5.txt

INPUT M1c *** Timesteps corresponding to changing velocity field (for M1 = 3)
5

INPUT M2a *** Maximum advection velocity in x direction [mm/y ????]
advec_velx
5.0

INPUT M2b *** Maximum advection velocity in y direction [mm/y ????]
advec_vely
0.0

INPUT M3 *** size (in km) of the square mesh on which the thin elastic plate calculations are done
hflex
1000.0

INPUT M4 *** flag to permit flexure in the x-direction
ixflex
F

INPUT M5 *** flag to permit flexure in the y-direction
iyflex
F

INPUT M6 *** elastic thickness in km
thickflex
15.0

INPUT M7 *** young modulus (in Pa)
young_mod
1.0e11

INPUT M8 *** poisson's ratio
pratio
0.25

INPUT M9 *** density of crustal rocks for isostatic calculation (in kg/m3)
rhocflex

```
2750.0

# INPUT M10 *** asthenospheric density for isostatic calculation (in kg/m3)
# rhoaflex
3300.0

# INPUT M11 *** Time [yr] when the flexure calculation begins.
# Set to a non-zero number [e.g. 1.0e6] to avoid numerical issues early on in the simulation.
# If you want flexure for the entire simulation duration, just set to 0
# flexon
4.5e9


#******************************
#****CLIMATE
# INPUT N
#******************************

# *** Precipitation parameters

# INPUT N1 *** Time between updates of precipitation field in orographic precip model [yr]
# calc_rain
25000

# INPUT N2 *** Grid size of square grid cells used in orographic precipitation model to calculate
# precipitation on cascade topography [km]
# del_gr
1

# INPUT N3 *** Precipitation rate for a uniform rainfall model [m/yr]
# rain_vel
4.0

# INPUT N4 *** how often do you want to update Ice if topography hasn't changed significantly.
# Make this a multiple of the rainmaker calls (calc_rain) defined above.
# calc_ice
100

# INPUT N5 *** Model time at which ICE can start running an glaciers might start to grow.
# If you want to not call Ice for a certain time period (i.e. climate will
# be too warm anyway, or you want a step change into a glacial period, etc).
# For Ice to be turned on at begining of simulation, set to 0
# iceon
#3.0e6
0.0
```

INPUT N6 *** Background atmospheric moisture content parameters
a0 [m/yr] & a1 [m/yr per (m/s)]
0.01
110.0

INPUT N7 *** Reciprocal vertical velocity variance [1/(m/s)]
alf
110.0

INPUT N8 *** Mean annual wind speed [m/s]. Typical values are around 1.0 to 1.5
wnd
0.1

INPUT N9 *** Wind direction = counter-clockwise angle rel. to y=0 line or x-axis [degrees]
90 = wind blows in direction of y-axis, 0 = wind blows in direction of x-axis.
angle
70

INPUT N9b *** Flip model along y-axis (y_min = y_max) to make orographic precipitation work properly
T = flip, F = do nothing
F

INPUT N10 *** Cross wind smoothing scale [km]. Scale that atmospheric model smooths
precipitation over topography within that window. Most orogens have values between 20-75 km.
xwind_s
15

INPUT N11 *** Upwind smoothing scale. Scale that atmospheric model smooths
precipitation over topography within that window. Most orogens have values between 20-75 km.
upwnd_s
15

INPUT N12 *** Topographic subsampling to speed up orographic precip calculation. Use integer values,
rainmaker will skip nodes based on this value. Typically a value of 2 is enough.
subsam
1

INPUT N13 *** Atmospheric lapse rate (C/m)
xlapse_rate
0.00001

*** Snow fall parameters below.

INPUT N14 *** Annual variation in daily temperature [C]

```
# Half amplitude of temperature shifts throughout a year
# This value is used to calculate a positive degree day melting algorithm in glacial calculations.
# At
0

# INPUT N15 *** Maximum snow accummulution rate (m/yr)
# xpmax
0

# INPUT N16 *** Use max out accumulation
# use_max_accu
T

# INPUT N17 *** Positive degree day melting constant (kmelt) (m/C)
# USed for mass balance calculation in glaciers
# Typical value 0.008 from Braithwaite, 1995
# kmelt
0

# INPUT N18 ***  Min and max sea level temperatures (C)
# These values are used in rainmaker for oscillating (sine) climate over glacial/interglacial cycles.
# These temperature variations are also put into rainmaker, and used for precipitation rates during
# glacial and non-glacial conditions in the model.
# If you want a constant climate during the simulation - set the following two values equal to
eachother.
# temp0min
# temp0max
#10.0
#16.0
#2.0
#8.0
#0.0
22.0
30.0

# INPUT N19 *** Period of surface temperature oscillation (in yrs)
# These values are used in rainmaker for oscillating (sine) climate over glacial/interglacial cycles.
# These temperature variations are also put into rainmaker, and used for precipitation rates during
# glacial and non-glacial conditions in the model.
# xp
1.0e5

#******************************
#****Temperature Input
# INPUT N
#******************************
```

INPUT O1 *** Use temperature data from file (T = yes , F = No)
temp_from_file
F
INPUT O2 *** Filename for temperature data
temp_file_name
test.txt

INPUT O3 *** Amplitude which is used for calculating the temperature swing
amp_temp
8

Input O4 *** Normalised O18 Values for Present and LGM
IG_O18
LGM_O18
0.2129
0.9609

Input O5 *** MAT at target latitude (Zuerich)
mean_t
7.9

8.2 CASCADE FORTRAN90 Executable Files

8.2.1 Main File from Cascade

```
      program cascade
! ---------------------------------------------------
! |                                                 |
! | CCCCC    A    SSSSS CCCCC    A    DDDD  EEEEE |
! | C       A A   S     C       A A   D   D E     |
! | C      AAAAA  SSSSS C      AAAAA  D   D EEEE  |
! | C     A     A     S C     A     A D   D E     |
! | CCCCC A     A SSSSS CCCCC A     A DDDD  EEEEE |
! |                                                 |
! ---------------------------------------------------

! The program cascade was developed by:

!      Jean Braun
!      Research School of Earth Sciences
!      Australian National University
```

```
!       Canberra, ACT, 0200
!       Australia
!       Tel: +61-2-6249-5512
!       Fax: +61-2-6249-5443
!       email: Jean.Braun@anu.edu.au

! (Canberra, June 1st, 1995)
! (Present version August 28, 1998)

! Please report any problem and certainly any improvement you bring to
! cascade.

! cascade is a geomorphic program to compute the evolution of landscapes
! by erosion/deposition and tectonic uplift/subsidence. Two types of processes
! are included: 1) short-range (hillslope) processes, modeled by a simple linear
!                  diffusion equation
!               2) long-range (river) processes, modeled by an equation of
!                  reaction between water flowing in a river network and the
!                  substratum

! The main advantage/difference between cascade and other geomorphic models
! is its ability to handle arbitrary, non-rectangular, grids at the corners
! of which the computations are done. This opens opportunities to vary spatial
! discretization in various parts of the model or to adjust spatial discretiza-
! tion to the results of the model (evolving discretization).

! cascade uses the theory of Delaunay triangulation and Voronoi diagrams
! to compute the possible "connections" between neighbouring nodes on the
! arbitrary grid and the surface area of the part of the landscape the height
! of which is slave to any node of the grid.

! The river network is calculated from the "bucket passing" algorithm (BPA)
! that does not require a complete ordering of the nodes according to their
! heights but rather a local-only ordering that depends on the position of the
! nodes with respect to local maxima in heights. In BPA, each node is given a
! bucket full of water and asked to pass it to its lowest neighbour. After that
! operation, all nodes that have not received anything are local maxima and
! are put at the top of a stack. After the next bucket passing step, those that
! have not received anything are put on the stack, and so on until all nodes
! have been put on the stack. The stack contains an ordering of the nodes that
! is appropriate to calculate the effect of river flow on the landscape.

! note that because this version of the bucket algorithm can handle local
! minima, runs tend to be slower at the beginning when many local minima
! exist. As the minima disappear due to river erosion and deposition, one
! should notice an increase efficiency per time step
```

```
! In this version, the diffusion equation is solved by using a "classical"
! 3 node linear finite element code with an iterative solver (Gauss-Seidel).
! The diffusion equation solver is the slow part of the code and future versions
! of cascade will be improved in this regard.

! cascade has the possibility of including a stratified crust with
! regards to erosional parameters
! Developed by Peter van der Beek, April '96

! IMPORTANT NOTE:
!****************
! Please, note that this software CANNOT be freely distributed. You must
! obtain Jean Braun's permission to use it (or part of it) or to give to other
! potential users. Please, respect this condition of use. I am trying to
! protect parts of the Delaunay/Voronoi algorithms that we are using in a
! commercial venture with Malcolm Sambridge. This means that some of our
! "clients" had to pay to use this software commercially.

! Good luck.

! NOTE from check_mesh.f on requirements for remeshing ...
! It is important that, if this routine is used, that is if dynamic
! remeshing is turned on, all nodal parameters that you have added (such
! as a new nodal property) be passed here for what is called permutation.
! During permutation, nodes are renumbered (see near bottom of the
! subroutine) and nodal properties have to be updated accordingly.
!  In this new version all the properties and parameters that have to
! be permuted are stored in memory and param

!------------------------------------------------------------------------

! subroutines called:
! - debug
! - initialize_general_parameters
! - initialize_nodal_geometry
! - erosional_properties
! - update_time_step
! - find_neighbours
! - write_output
! - change_sea_level
! - erosional_properties
! - tectonic_uplift
! - tectonic_movement
! - orography
! - find_donors
```

```
! - find_catchment
! - fluvial_erosion
! - diffusion_erosion
! - flexure
! - show
! - check_mesh
! - update_bedrock
! - write_output
! - mdian2
! - update_time_step
! - landslide or
! - landslide_simple

! the module rt_param contains the data structure for the run time parameters
! and the functions to read those parameters from file
      use rt_param
      use cascade_globals

! for each file there is a module:
      use m_change_sea_level
      use m_update_bedrock
      use m_terosion
      use m_debug
      use m_find_dslope
      use m_update_flags
      use m_surface_temperature
      use m_erosional_properties
      use m_interp_ice
      use m_find_donors
      use m_tectonic_uplift
      use m_gerode_node
      use m_diffusion_erosion
      use m_mass_balance
      use m_update_time_step
      use m_UpdateBoundaryNodes
      use m_find_catchment
      use m_find_neighbours
      use m_tectonic_movement
      use m_write_output
      use m_initialize_nodal_geometry
      use m_landslide_simple
      use m_fluvial_erosion
      use m_read_nodal_geometry2
      use m_flexure
      use m_landslide
      use m_ice
```

```
      use m_check_mesh
      use m_rainmaker
      use m_rainmaker_imposed
      use m_MOVEtoCascade
      use m_checkMeshResolution
      implicit none

! define global data structure for configuration items
      type(config) :: ConfigData

! only used in this context / scope. (= not global)
      integer(4) :: i, k, system

      iseed = 1



! initialize variables, WK
      nt = ntmax

! system time variables
      print *,'**********************************'
      print *,'Start time:'
      tsys0 = system('date')
      print *,'**********************************'

! use dynamic arrays, WK

! one dimension
      allocate(x(nnodemax))
      allocate(y(nnodemax))
      allocate(hi(nnodemax))
      allocate(h(nnodemax))
      allocate(h0(nnodemax))
      allocate(isodh(nnodemax))
      allocate(xd(nnodemax))
      allocate(yd(nnodemax))
      allocate(hd(nnodemax))
      allocate(xl(nnodemax))
      allocate(yl(nnodemax))
      allocate(hl(nnodemax))
      allocate(xu(nnodemax))
      allocate(yu(nnodemax))
      allocate(hu(nnodemax))
      allocate(xr(nnodemax))
      allocate(yr(nnodemax))
```

```
      allocate(hr(nnodemax))
      allocate(dhg(nnodemax))
      allocate(hicerem(nnodemax))
      allocate(lastice_h(nnodemax))
      allocate(ldh(nnodemax))
      allocate(gerode_term(nnodemax))
      allocate(work(nnodemax))
      allocate(water(nnodemax))
      allocate(sediment(nnodemax))
      allocate(orwater(nnodemax))
      allocate(slope(nnodemax))
      allocate(length(nnodemax))
      allocate(ndon(nnodemax))
      allocate(nb(nnodemax))
      allocate(nb2(nnodemax))
      allocate(ibucket(nnodemax))
      allocate(iorder(nnodemax))
      allocate(itype_node(nnodemax))
      allocate(nwork(nnodemax))
      allocate(ncat(nnodemax))
      allocate(nsill(nnodemax))
      allocate(nempty(nnodemax))
      allocate(nlake(nnodemax))
      allocate(prec(nnodemax))
      allocate(y_gr(nnodemax))
      allocate(x_gr(nnodemax))
      allocate(nodes(nnodemax))
      allocate(vis_tlist(nnodemax))
      allocate(vis_elist(nnodemax))
      allocate(add_tlist(nnodemax))
      allocate(nodelist(nbmax))
      allocate(tlist(nbmax))
      allocate(c_list(2*nbmax))
      allocate(mask(3*nnodemax))
      allocate(inactive(nnodemax))
      allocate(anthi(nnodemax))

! two dimensions
      allocate(memory(nnodemax,nmemory))
      allocate(param(nnodemax,nparam))
      allocate(nn2(nbmax,nnodemax))
      allocate(nn(nbmax,nnodemax))
      allocate(kcon(ntmax,nnodemax))
      allocate(jcon(ntmax,nnodemax))
      allocate(flex(nflex,nflex))
      allocate(work_flex(nflex,nflex))
```

```
        allocate(zpt(nxi,nyi))
        allocate(bipt(nxi,nyi))
        allocate(bface(nxi,nyi))
        allocate(tbipt(nxi,nyi))
        allocate(antitoc(nxi,nyi))
        allocate(points(2,nnodemax))
        allocate(vertices(3,nnodemax*3))
        allocate(neighbour(3,nnodemax*3))
        allocate(v_local(3,nbmax*2))
        allocate(n_local(3,nbmax*2))
        allocate(mask_e(3,nnodemax*3))

! three dimensions
        allocate(cell(nnodemax,nbmax,2))

! initializes memory
        do k=1,nmemory
           do i=1,nnodemax
              memory(i,k)=0.0_8
           enddo
        enddo

        do k=1,nnodemax
           do i=1,nparam
              param(k,i)=0.0_8
           enddo
        enddo

        do k=1,nbmax
           do i=1,nnodemax
              nn2(k,i)=0
              nn(k,i)=0
           enddo
        enddo

        do k=1,ntmax
           do i=1,nnodemax
              kcon(k,i)=0
              jcon(k,i)=0
           enddo
        enddo

        do k=1,nflex
           do i=1,nflex
              flex(k,i)=0.0_8
              work_flex(k,i)=0.0_8
```

```
   enddo
enddo

do k=1,nxi
   do i=1,nyi
      zpt(k,i)=0.0_8
      bipt(k,i)=0.0_8
      bface(k,i)=0.0_8
      tbipt(k,i)=0.0_8
      antitoc(k,i)=0
   enddo
enddo

do k=1,2
   do i=1,nnodemax
      points(k,i)=0.0_8
   enddo
enddo

do k=1,3
   do i=1,nnodemax
      mask_e(k,i)=.FALSE.
   enddo
enddo

do k=1,3
   do i=1,nbmax*2
      v_local(k,i)=0
      n_local(k,i)=0
   enddo
enddo

do k=1,nnodemax
   x(k) = 0.0_8
   y(k) = 0.0_8
   hi(k) = 0.0_8
   h(k) = 0.0_8
   h0(k) = 0.0_8
   isodh(k) = 0.0_8
   xd(k) = 0.0_8
   yd(k) = 0.0_8
   hd(k) = 0.0_8
   xl(k) = 0.0_8
   yl(k) = 0.0_8
   hl(k) = 0.0_8
   xu(k) = 0.0_8
```

```
    yu(k) = 0.0_8
    hu(k) = 0.0_8
    xr(k) = 0.0_8
    yr(k) = 0.0_8
    hr(k) = 0.0_8
    dhg(k) = 0.0_8
    hicerem(k) = 0.0_8
    lastice_h(k) = 0.0_8
    ldh(k) = 0.0_8
    gerode_term(k) = 0.0_8
    work(k) = 0.0_8
    water(k) = 0.0_8
    sediment(k) = 0.0_8
    orwater(k) = 0.0_8
    slope(k) = 0.0_8
    length(k) = 0.0_8
    ndon(k) = 0
    nb(k) = 0
    nb2(k) = 0
    ibucket(k) = 0
    iorder(k) = 0
    itype_node(k) = 0
    nwork(k) = 0
    ncat(k) = 0
    nsill(k) = 0
    nempty(k) = 0
    nlake(k) = 0
    prec(k) = 0.0_8
    y_gr(k) = 0.0_8
    x_gr(k) = 0.0_8
    nodes(k) = 0
    vis_tlist(k) = 0
    vis_elist(k) = 0
    add_tlist(k) = 0
    inactive(k) = .FALSE.
    anthi(k) = 0
  enddo

  do k=1,3
    do i=1,nnodemax*3
      vertices(k,i) = 0
      neighbour(k,i) = 0
    enddo
  enddo

! ivocal = .TRUE. means CASCADE will save debugging information in debug.out
```

```
! should be set to ivocal = .FALSE. in production runs

      ivocal = .TRUE.

      if (.not.ivocal) then
         open (89,file='debug.out',status='unknown')
         write (89,'(a)') 'Debugger not activated'
         write (89,'(a)') 'To activate set ivocal to 1 in cascade'
         close (89)
      else
         open (89,file='debug.out',status='unknown')
         rewind (89)
      endif

! if ivocal=1 opens a log file to record which nodes are being
! added/removed from the mesh

      if (ivocal) open (22,file='checkMeshResolution.out',status='unknown')

! read configuration from file "input/IceCascade/icecascade.in"
! fortran uses call by reference, so the data structure is not copied and no
! pointer is needed here!
      call readConfig(ConfigData)

!       print *, "configData:"
!       print *, configData

!       print *, "end of test readConfig"

      call flush()

! this is no longer needed and can be removed later:
! initialize general parameters
      if (ivocal) call debug ('cascade$')

! prep intialize geometry if adding a shelf.
! Takes variables defined in initialize nodal geometry and resets nnode.
! Done in cascade main becasue I was getting seg faults otherwise.  Should move this at some point
      if (configData%addshelf) then

        print *, "sidex: ", configData%sidex, ", sidey: ", configData%sidey

        ! new width and height of domain
        configData%sidex = configData%sidex + configData%Exl + configData%Exr
        configData%sidey = configData%sidey + configData%Eyu + configData%Eyd
```

```
print *, "new sidex: ", configData%sidex, ", new sidey: ", configData%sidey

! new grid step size
dye = int(configData%Eyd + configData%Eyu) / configData%nye
dxe = int(configData%Exl + configData%Exr) / configData%nxe

if (dxe == 0) then
  dxe=dye
end if

if (dye == 0) then
  dye=dxe
end if

if ((dxe > 0) .and. (dye > 0)) then
  print *, "dxe: ", dxe, ", dye: ", dye

    nexd=int(configData%sidex)/dxe
    neyd=int(configData%Eyd)/dye - 1

    if (neyd < 0) then
      neyd = 0
    end if

    ned=nexd*neyd

    print *, "nexd: ", nexd, ", neyd: ", neyd, ", ned: ", ned

    nexu=int(configData%sidex)/dxe
    neyu=int(configData%Eyu)/dye - 1

    if (neyu < 0) then
      neyu = 0
    end if

    neu=nexu*neyu

    print *, "nexu: ", nexu, ", neyu: ", neyu, ", neu: ", neu

    neyl=int(configData%sidey-configData%Eyd-configData%Eyu)/dye
    nexl=int(configData%Exl)/dxe
    nel=neyl*nexl

    print *, "nexl: ", nexl, ", neyl: ", neyl, ", nel: ", nel

    neyr=int(configData%sidey-configData%Eyd-configData%Eyu)/dye
```

```
        nexr=int(configData%Exr)/dxe
        ner=neyr*nexr

        print *, "nexr: ", nexr, ", neyr: ", neyr, ", ner: ", ner

        ! nextra: number of additional shell nodes
        nextra=ned+neu+nel+ner
        print *, "cascade, add shelf:"
        print *, "nextra: ", nextra
        print *, "nnode: ", configData%nnode
        print *, "nx, ny: ", configData%nx, configData%ny
        configData%nnode = configData%nnode + nextra
        print *, "nnode: ", configData%nnode

      end if
    end if

!  Prep rainmaker
    nxs = int(configData%sidex)/configData%del_gr + 1
    nys = int(configData%sidey)/configData%del_gr + 1
    nxice=nxs
    nyice=nys
    delx=configData%sidex/dble(nxice-1)
    dely=configData%sidey/dble(nyice-1)

! allocate memory once
! we meed to do it here, because we need the values nxs and nys

    allocate(prec_gr(nxs,nys))
    allocate(z(nxs,nys))
    allocate(hforice(nxs,nys))
    allocate(slidetoc(nxs,nys))
    allocate(htoc(nxs,nys))
    allocate(httoc(nxs,nys))
    allocate(iceftoc(nxs,nys))
    allocate(constoc(nxs,nys))

    do k=1,nys
      do i=1,nxs
        z(i,k) = 0.0_8
        hforice(i,k) = 0.0_8
        prec_gr(i,k) = 0.0_8
        slidetoc(i,k) = 0.0_8
        htoc(i,k) = 0.0_8
        httoc(i,k) = 0.0_8
        iceftoc(i,k) = 0.0_8
```

```
      constoc(i,k) = 0.0_8
    enddo
  enddo

global_cascade_dt = 1.0

! initialize the node geometry
  if (configData%meshread) then
    ! Mesh format described in rt_param.f90
    select case (configData%meshformat)
      case (1)
        if (ivocal) then
          call debug ('read_nodal_geometry$')
        end if
        call read_nodal_geometry2(configData)
      case (2)
        ! WK: TODO
        print *, "Mesh format not implemented yet: ", configData%meshformat
        stop
      case (3)
        ! added by PRE Jan2017
        call MOVEtoCascadeTopo(configData)
      case (4)
        ! WK: TODO
        print *, "Mesh format not implemented yet: ", configData%meshformat
        stop
      case default
        print *, "Unknown mesh format: ", configData%meshformat
        stop
    end select
  else
    if (ivocal) call debug ('initialize_nodal_geometry$')
      call check_h(1)
      call initialize_nodal_geometry(configData)
      call check_h(2)
  end if
  tott=h
  if (ivocal) call debug ('cascade$')
  nnode0 = configData%nnode

  do i=1,configData%nnode
    if (.not.configData%meshread) then
      h0(i) = h(i)
    end if
    hi(i) = h(i)
  enddo
```

```
        temperature = configData%temp0min


! opens various output files

        if (configData%nrun_name == 0) then
          print *,'No run name available'
          stop
        endif

        inquire (file=configData%run_name(1:configData%nrun_name), &
             exist=cascade_FileExists, iostat=cascade_ioStatus)

        if (.not.cascade_FileExists) then
          print *, 'creating folder: ', configData%run_name(1:configData%nrun_name)
          system_Result = system('mkdir ' // configData%run_name(1:configData%nrun_name))
        end if

        open (7, file=configData%run_name(1:configData%nrun_name) // '/topography', &
            status='unknown', iostat=cascade_ioStatus)

        if (cascade_ioStatus /= 0) then
          print *,'error, could not open file: ', (configData%run_name(1:configData%nrun_name) //
'topography')
          stop
        end if

        open                    (10,file=configData%run_name(1:configData%nrun_name)//'/geometry',
status='unknown')

        if (.not.configData%tecflag) then
          open
(8,file=configData%run_name(1:configData%nrun_name)//'/connectivity',status='unknown')
          open
(9,file=configData%run_name(1:configData%nrun_name)//'/donors',status='unknown')
          open
(11,file=configData%run_name(1:configData%nrun_name)//'/properties',status='unknown')
          open
(12,file=configData%run_name(1:configData%nrun_name)//'/discharge',status='unknown')
          open
(13,file=configData%run_name(1:configData%nrun_name)//'/erosion_rate',status='unknown')
          open
(14,file=configData%run_name(1:configData%nrun_name)//'/catchments',status='unknown')
          open
(15,file=configData%run_name(1:configData%nrun_name)//'/lakes',status='unknown')
```

```
          open
(16,file=configData%run_name(1:configData%nrun_name)//'/slides',status='unknown')
          open
(17,file=configData%run_name(1:configData%nrun_name)//'/slopearea',status='unknown')
          open
(77,file=configData%run_name(1:configData%nrun_name)//'/precip',status='unknown')
      endif

! initialize erosional nodal properties
      if (ivocal) call debug ('erosional_properties$')
      call check_h(3)
      call erosional_properties (configData)
      call check_h(4)
      if (ivocal) call debug ('cascade$')

! initialize time step
      if (ivocal) call debug ('update_time_step$')
      call check_h(5)
      call update_time_step (configData, .FALSE.)
      call check_h(6)
      if (ivocal) call debug ('cascade$')


! writes the initial conditions

      if (ivocal) call debug ('write_output$')
      timeint=-1
      call check_h(7)
      call write_output(configData, .TRUE.)
      call check_h(8)
      if (ivocal) call debug ('cascade$')

! WK: DEBUG
!       stop

! finds Delaunay triangulation and voronoi cell surface areas for initial
! set of nodes
      if (ivocal) call debug ('find_neighbours$')

!       print *, "x: ", x(1:10)
!       print *, " y: ", y(1:10)
!       print *, "nn: ", nn(1:10, 1)
!       print *, "nb: ", nb(1:10)
!       print *, "nnode: ", configData%nnode
!       print *, "nbmax: ", nbmax
!       print *, "nn2: ", nn2(1:10, 1)
```

```
!       print *, "nb2: ", nb2(1:10)
!       print *, "points: ", points(1:10, 1)
!       print *, "vertices: ", vertices(1:10, 1)
!       print *, "neighbour: ", neighbour(1:10, 1)
!       print *, "nodes: ", nodes(1:10)
!       print *, "vis_tlist: ", vis_tlist(1:10)
!       print *, "vis_elist: ", vis_elist(1:10)
!       print *, "add_tlist: ", add_tlist(1:10)
!       print *, "nt: ", nt
!       print *, "memory(1, 7): ", memory(1, 7)
!       print *, "memory(1, 6): ", memory(1, 6)
!       print *, "eps: ", eps
!       print *, "xy: ", xy
!       print *, "pp: ", pp
!       print *, "aa: ", aa(1:10, 1)
!       print *, "bb: ", bb(1:10)
!       print *, "surfscale: ", configData%surfscale
!       print *, "cell: ", cell(1:10, 1, 1)
!       print *, "finish"

!       print *, "nnodemax: ", nnodemax
!       print *, "nmemory: ", nmemory

      call check_h(9)
      call find_neighbours (configData)
      call check_h(10)

      if (ivocal) call debug ('cascade$')

! writes the initial conditions
      if (ivocal) call debug ('write_output$')
      timeint=-1
      call check_h(11)
      call write_output(configData, .TRUE.)
      call check_h(12)
      if (ivocal) call debug ('cascade$')

! start of time stepping
      cascade_istep=0

! initalize orographic time counter (via TE code 10/06)
      oro_time=configData%calc_rain + 1.0_8

! intialize ice time counter BJY 101109
      ice_time=dble(configData%calc_ice + 1)
      global_iceIsRunning = .FALSE.
```

```
! initialize time since last addition of nodes in tectonic_movement.f
      tcheck = 0.0_8

      do while (time < configData%endtime)
         cascade_istep=cascade_istep+1

! update flags for ice and flexure
         call check_h(13)
         call update_flags(configData)
         call check_h(14)

! change in sea level
         if (ivocal) call debug ('change_sea_level$')
         call check_h(15)
         call change_sea_level (configData)
         call check_h(16)
         if (ivocal) call debug ('cascade$')

! initializes the flux of material entering and leaving the grid
! for this time step
         influx=0.0_8
         outflux=0.0_8

! distributes erosional properties to rocks according to the amount eroded
! i.e. makes layered crust (Peter, April '96)
         if (ivocal) call debug ('erosional_properties$')
         call check_h(17)
         call erosional_properties (configData)
         call check_h(18)
         if (ivocal) call debug ('cascade$')


! add tectonic uplift/subsidence component to landscape height (modified by PRE Jan2017)
! (note that heights are in km)

         if (configData%uplift_mode == 1) then
            if (ivocal) call debug ('tectonic_uplift$')
               call check_h(19)
               call tectonic_uplift (configData)
               call check_h(20)
            if (ivocal) call debug ('cascade$')
               call check_h(20)
            if (ivocal) call debug ('cascade$')
         end if
         if (configData%uplift_mode == 2) then
```

```
      call MOVEtoCascadeUplift(configData) ! Added by PRE Jan2017
   end if
   if (configData%uplift_mode == 3) then ! Added by PREFeb2017
      call MOVEtoCascadeVelocityField(configData)
      call MOVEtoCascadeUplift(configData)
   end if

! Update Boundary Nodes for Uplift
   if (configData%imposeRearBoundaryUplift) then
   call UpdateBoundaryNodes(configData) ! Added by VMBP Jan2020
   end if

! add tectonic horizontal movement (modified by PRE Jan2017)
   if (configData%ihorizontal) then
      tcheck = tcheck + global_cascade_dt
      if (configData%ihorizontal_mode == 1) then
         if (ivocal) call debug ('tectonic_movement$')
         call check_h(21)
         call tectonic_movement (configData)
         call check_h(22)
         if (ivocal) call debug ('cascade$')
      end if
      if (configData%ihorizontal_mode == 2) then ! added by PRE Jan2017
         if (configData%uplift_mode == 2) then
            call MOVEtoCascadeHmove(configData)
         elseif (configData%uplift_mode == 3) then
            call MOVEtoCascadeHmove(configData)
         end if
      end if
   end if

!  Following subroutine implements a variable climate that is fed into both rainmaker and
mass_balance
   call check_h(23)
   call surface_temperature (configData)
   call check_h(24)

! Calculate Precipitation - added by DW 10/06 (using TE code)
   if (ivocal) call debug ('rainmaker$')

! call new routine.
! call every time step.
! if update time exceeded then set oro_flag to one
! which cases reinterpolation of the nodal topography
! on to the regular grid in routine orography_new ghr 07/01
```

```
        ! added third option for precipitation Jan 2020
        ! Victoria M Buford Parks
! DEBUG MODE 10/11/01 on next line
        water=orwater
        oro_time = oro_time+global_cascade_dt
        if ((oro_time > configData%calc_rain).or.(cascade_istep == 1)) then
          if (configData%iflag_precip.lt.3) then ! for options 1 and 2
            print*,'rainmaker: total topography (surface +ice)',oro_time, configData%calc_rain
            oro_time = 0.0_8

!  NOTE:  CHanged h from 'tott' total topography because weird things were happening with ice
now that rainmaker builds ice topography
            call check_h(25)
            call rainmaker(configData)
            call check_h(26)
            orwater=water
          endif
        end if

        if (configData%iflag_precip.eq.3) then
          if ((oro_time > configData%imposedPrecipUpdateTime).or.(cascade_istep == 1)) then
            print*, "time: ",time,"oro_time: ", oro_time, "Calling rainmaker_imposed"
            call rainmaker_imposed(configData)
            ! Restart orographic time counter
            oro_time = 0.0_8
            orwater=water
          endif

        endif
        if (ivocal) call debug ('cascade$')


! find the donor for each node (its lowest neighbour)

        if (ivocal) call debug ('find_donors$')
!       write(6,*) 'finding donors... '
        call check_h(27)
        call find_donors (configData)
        call check_h(28)
        if (ivocal) call debug ('cascade$')

! find rain function (water) - commented out by DW 10/06

!     if (ivocal.eq.1) call debug ('orography$')
!     call orography (x,y,h,water,memory(1,7),length,work,
!    &              ndon,nn,nb,nbmax,
```

```
!    &              nwork,
!    &              oro_length,oro_height,oro_scale,
!    &              nnode,wind_direction,rain_vel)
!     if (ivocal.eq.1) call debug ('cascade$')

! find catchment for each node

        if (ivocal) call debug ('find_catchment$')
        call check_h(29)
        call find_catchment (configData)
        call check_h(30)
        if (ivocal) call debug ('cascade$')


!  CALL ICE HERE
        if (global_iceIsRunning) then
           ice_time=ice_time+global_cascade_dt
           ice_topotime=ice_topotime+int(global_cascade_dt)

! checking maximum change in topography for a node since the last time Ice was called
           ldh=(h-lastice_h)
           ldh=abs(ldh)
           lastice_dh=maxval(ldh)


!  Add more requirements such as (maybe): max(gbalance)>0  or max(iceth>0) or  topography
changes by 25 m or more

           if ((ice_time > dble(configData%calc_ice)).or.(cascade_istep == 1)) then
              ice_time=0.0_8

!     print*,'--converting Cascade to Ice--',lastice_dh


              lastice_h=h

              print *,'*******Updating Ice, Ice, baby*******'
              ice_topotime=0

!   call ice, returns ice
              hicerem=iceth

              if (ivocal) call debug ('ice, interp_ice$')
              call check_h(31)
              call ice(configData)
              call check_h(32)
```

```
            call interp_ice(configData)
            call check_h(33)

            print *, "ice finished!"
         end if ! ice_time
      end if ! end global_iceIsRunning

!   Find where to erodoe by glacial erosion and erode there

      if (configData%glacial_erosion) then
         if (ivocal) call debug ('gerode_node$')
         call check_h(34)
         call gerode_node(configData)
         call check_h(35)
      endif
! river erosion

!       print*,dt,time,cascade_istep

      if (configData%fluvial_erosion) then
         if (ivocal) call debug ('fluvial_erosion$')
         call check_h(36)
         call fluvial_erosion (configData)
         call check_h(37)
      end if

      if (ivocal) call debug ('cascade$')

! diffusion erosion
      if (configData%idiffusion) then
         if (ivocal) call debug ('diffusion_erosion$')
         call check_h(38)
         call diffusion_erosion (configData)
         call check_h(39)
      endif
      if (ivocal) call debug ('cascade$')

! landsliding
      if (configData%ilandslide) then
         if ((configData%lsmeth == 1).or.(configData%lsmeth == 2)) then
            if (ivocal) call debug ('landslide$')
            do i=1,configData%nnode
               smax(i)=0.0_8
               tt(i)=tt(i)+global_cascade_dt
            enddo
            call check_h(40)
```

```
            call landslide(configData)
            call check_h(41)
        elseif (configData%lsmeth == 3) then
            if (ivocal) call debug ('landslide_simple$')
            call check_h(42)
            call landslide_simple(configData)
            call check_h(43)
        endif
    endif

    if (ivocal) call debug ('cascade$')


! calculate isostatic rebound
    if (configData%iflexure) then
!     print *,'Calculating Flexure',configData%iflexure
        if (ivocal) call debug ('flexure$')
        call check_h(44)
        call flexure(configData)
        call check_h(45)
        if (ivocal) call debug ('cascade$')
    endif


! check the grid (in case adaptive grid is allowed)

    if (configData%iadapt) then
        if (ivocal) call debug ('check_mesh$')
        call check_h(46)
        call check_mesh (configData)
        call check_h(47)
        if (ivocal) call debug ('cascade$')
    endif

! If horizontal tectonic movement is turned on, mesh needs to be remeshed regularly PRE
NOV2017
    if (configData%ihorizontal) then
        if (meshtime >= dble((delta / 2e-4_8))) then
            call check_h(150)
            print *, 'Updating mesh... (checkMeshResolution)'
            call checkMeshResolution(configData)
            call check_h(151)
            meshtime = 0.0_8
!   if (dtold /= 0.0_8) then
!                   global_cascade_dt=dtold
!                   dtold=0.0_8
```

```
!                 end if
              remeshflag = 1
           end if
end if

! check for bedrock incision

        if (ivocal) call debug ('update_bedrock$')
        call check_h(48)
        call update_bedrock (configData)
        call check_h(49)
        if (ivocal) call debug ('cascade$')

! update time
        time=time+global_cascade_dt
        shorttime=shorttime+global_cascade_dt
        meshtime=meshtime+global_cascade_dt ! Added by PRE Nov2017

        call check_h(50)
        call terosion()
        call check_h(51)

! writing output

        if (shorttime == dble(configData%writetime)) then
           if (ivocal) call debug ('write_output$')

! find downstream slope
           call check_h(52)
           call find_dslope(configData)
           call check_h(53)
           call mass_balance(configData)
           call check_h(54)

! added h0 to the parameter list
           call flush(6)

           call check_h(55)
           call write_output(configData, .FALSE.)
           call check_h(56)

           if (ivocal) call debug ('cascade$')

           shorttime=0.0_8
           if (dtold /= 0.0_8) then
              global_cascade_dt=dtold
```

```
            dtold=0.0_8
          endif
        endif

! write a short line to the screen
        if (mod(itime, configData%nshortwrite) == 0) then
!         if ((itime/configData%nshortwrite*configData%nshortwrite) == itime) then
          hmin=h(1)
          hmax=h(1)
          do i=1,configData%nnode
            hmin=min(hmin,h(i))
            hmax=max(hmax,h(i))
          enddo
          hmedian=(hmax-hmin)/2.0_8

! commenting out mdian2 as it seems to give an error, some
! h(i) values are being set to infinity...

          print *, 'TIME ',cascade_istep,' (',time,'), ', 100.0_8 * time / configData%endtime
          tsys0 = system('date')
          print *, 'hrange=[',hmin,',',hmedian,',',hmax,']'
          print *, 'h', maxval(h),minval(h),global_cascade_dt,dhmaxglac,maxval(slide)
          print *, "nnode: ", configData%nnode
          print *, "nx, ny: ", configData%nx, configData%ny
          print *, "sidex, sidey: ", configData%sidex, configData%sidey
!    &    maxval(h-h0)
        endif

! update time step
        if (ivocal) call debug ('update_time_step$')
        call check_h(57)
        call update_time_step (configData, .TRUE.)
        call check_h(58)
        if (ivocal) call debug ('cascade$')

! end of time stepping

! WK: for gprof:
!       exit

      enddo

! get stop time
      print *,'**********************************'
      print *,'End time:'
      tsys0 = system('date')
```

```
    print *,'*********************************'

  contains
    subroutine check_h(num)
      use cascade_globals
      use m_check_mesh
      implicit none

      integer(4) :: i, num

      do i=1,nnodemax
        if (h(i) /= h(i)) then
          print *, "num: ", num
          print *, "h(i) is NaN: ", i, h(i)
          print *, "memory(i,5): ", memory(i,5)
          stop
        end if
      end do

      if (global_cascade_dt == 0.0) then
        print *, "num: ", num
        print *, "global_cascade_dt is zero"
        stop
      endif
    end subroutine check_h
  end
```

8.2.2 Cascade Globals

This is the new/adjusted 'Cascade_globals.f90'.

! This version of CASCADE is a fork from the original version published by Braun and Sambridge, 1997 Basin Research. The modifications made to the original version of CASCADE are described in Yanites and Ehlers, 2012, 2016 EPSL, and Eizenhoefer et al., 2019 JGR-ES.
! If you published results from this version of the program we would appreciate that you reference:
! 1. Publications listed by J. Braun in the cascade.f90 file.
! 2. Yanites, B.J., and Ehlers, T.A., 2012, Global climate and tectonic controls on the denudation of glaciated mountains, Earth and Planetary Science Letters, v. 325-326, pp. 63-75. doi.org/10.1016/j.epsl.2012.01.030.

! 3. Yanites, B.J., and Ehlers, T.A., 2016, Intermittent glacial sliding velocities explain variations in long-timescale denudation, SW British Columbia. Earth and Planetary Science Letters, 450, pp. 52-61.
! 3. Eizenhoefer, P.R., McQuarrie, N., Shelef, E., and Ehlers, T.A., 2019 (in press), Fluvial responses to horizontal displacements in convergent orogens over geologic time scales, Journal of Geophysical Research - Earth Surface.

! Please direct all inquiries concerning this fork of the source code to Todd Ehlers (Univ. Tuebingen, Germany) tehlers@icloud.com

```
! cascade_globals.f90
! This module contains all global variables and constants used in icecascade

      module cascade_globals
        implicit none
        save

!
! nx and ny are the linear dimensions of the initial rectangular grid
! note that there is no reason for the grid to be rectangular
! this is done here because it is still the most popular way of
! designing numerical meshes

! nnodemax is the maximum number of nodes that the grid is allowed to grow
! into. If you decide not to use the idynamic=1 option you should chose
! nnodemax as close as possible to nnode

! nbmax is the maximum number of "natural neighbours" that any point can have
! for a general set of points nbmax=20 seems appropriate. However it is
! possible that for arbitrary sets of points, nbmax could be > 20. If it is
! the case, a warning message will be produced by cascade

! ntmax is the maximum nuber of triangles the circumcircles of which contain
! a common point

! Note that it will be assumed that the maximum number of triangles is
! three times the maximum number of nodes (this is very safe as the number
! of triangles is usually of the order of twice the number of nodes).

! nparam is the number of geomorphic parameters to be carried
! by the nodes

! nmemory is the number of working arrays carried by the nodes

! nflex is the discretization used to solve the flexural problem
```

```
! it must be a power of 2

!  nxi is the number of gridded x nodes for ice
!  nyi is the number of gridded y nodes for ice be sure they match the .in file

!      parameter (nnodemax=205*79,nparam=3,nmemory=8,
!     &           nbmax=70,ntmax=30,nflex=256)

! nodemax boosted up to run higher resolution models on petrarch cluster
! dwhipp 04/07

        integer(4), parameter :: nnodemax = 501*301
        integer(4), parameter :: nparam = 3
        integer(4), parameter :: nmemory = 9
        integer(4), parameter :: nbmax = 600
        integer(4), parameter :: ntmax = 30
        integer(4), parameter :: nflex = 256
        integer(4), parameter :: nxi = 200
        integer(4), parameter :: nyi = 120

! gridsize is no longer used and will be deleted
!          integer(4), parameter :: gridsize = 1000

! global constants, TODO: add more constants, replace
! magic numbers in code with constants

        real(8), parameter :: GLOBAL_PI = 3.1415926535897932 39_8
        ! SEC_IN_YR was "secinyr"
        real(8), parameter :: SEC_IN_YR = 365.25_8 * 24.0_8 * 3600.0_8


! global variables:

        real(8) :: global_cascade_dt
        logical :: global_iceIsRunning

! x and y are the x- and y-coordinates of the nodes in km
! h is the current topography
! hi is the initial topography
! h0 is the location of the bedrock interface
! all h's are in m,iceth(nnodemax)
        real(8) :: ice_time,lastice_dh

        real(8), dimension (:), allocatable :: x
        real(8), dimension (:), allocatable :: y
        real(8), dimension (:), allocatable :: hi
```

```
        real(8), dimension (:), allocatable :: h
        real(8), dimension (:), allocatable :: h0
        real(8), dimension (:), allocatable :: isodh
        real(8), dimension (:), allocatable :: xd
        real(8), dimension (:), allocatable :: yd
        real(8), dimension (:), allocatable :: hd
        real(8), dimension (:), allocatable :: xl
        real(8), dimension (:), allocatable :: yl
        real(8), dimension (:), allocatable :: hl
        real(8), dimension (:), allocatable :: xu
        real(8), dimension (:), allocatable :: yu
        real(8), dimension (:), allocatable :: hu
        real(8), dimension (:), allocatable :: xr
        real(8), dimension (:), allocatable :: yr
        real(8), dimension (:), allocatable :: hr
        real(8), dimension (:), allocatable :: dhg
        real(8), dimension (:), allocatable :: hicerem
        real(8), dimension (:), allocatable :: lastice_h
        real(8), dimension (:), allocatable :: ldh
        real(8), dimension (:), allocatable :: gerode_term

!
! param are geomorphic parameters attached to each nodes
! there are nparam of them
!

! param(*,1)=fluvial erosion constant
! param(*,2)=bedrock erosion length scale
! param(*,3)=diffusion erosion constant

        real(8), dimension (:,:), allocatable :: param


! memory are variables that have to be stored from one step to the next
! for each node
! there are nmemory of them
!

! memory(*,1)=dhcrit
! memory(*,2)=dhfluvial
! memory(*,3)=dhdiff
! memory(*,4)=hiso
! memory(*,5)=fix
! memory(*,6)=newsurface
! memory(*,7)=surface
! memory(*,8)=dhlandslide (added by Ehlers 6/01)
```

```
! memory(*,9)=dhglacier (added by Yanites 10/09)

        real(8), dimension (:,:), allocatable :: memory


! work is a working array
!
        real(8), dimension (:), allocatable :: work

! water is the amount of water that drains down the landscape
! it is equivalent to the discharge
! sediment is the sediment load in the rivers

        real(8), dimension (:), allocatable :: water
        real(8), dimension (:), allocatable :: sediment
        real(8), dimension (:), allocatable :: orwater

! slope is the slope between a node and its donor neighbour
! note that slopes are in meter per kilometer as our horizontal
! length unit is a kilometer while the horizontal unit is the meter

        real(8), dimension (:), allocatable :: slope
        real(8), dimension (:), allocatable :: length

! ndon is the name of the donor neighbour node
! nn is the list of neighbours
!  nb is the number of neighbours for each node

        integer(4), dimension (:), allocatable :: ndon
        integer(4), dimension (:), allocatable :: nb
        integer(4), dimension (:), allocatable :: nb2
        integer(4), dimension (:,:), allocatable :: nn2
        integer(4), dimension (:,:), allocatable :: nn

! ibucket is a working array that is used in the
! "pass the bucket" algorithm on which the cascade method is based
! to define the river network (the ndon array)

        integer(4), dimension (:), allocatable :: ibucket

! the following arrays are also used in the cascade algorithm

        integer(4), dimension (:), allocatable :: iorder
        integer(4), dimension (:), allocatable :: itype_node

! nwork is a working array used in determing the catchment to which each
```

```
! node belongs; that is stored in the ncat array which has the name
! of the exiting node of the catchment (that the way catchments are named)

        integer(4), dimension (:), allocatable :: nwork
        integer(4), dimension (:), allocatable :: ncat

! nsill and nempty are used in the algorithm that looks for
! sill nodes in case of local minima
! nlake is a flag that determined whether a node belong to a
! lake or not

        integer(4), dimension (:), allocatable :: nsill
        integer(4), dimension (:), allocatable :: nempty
        integer(4), dimension (:), allocatable :: nlake

! influx is the flux of material into the landscape
! brought in the system by the tectonic uplift
! outflux is the flux out of the system, ie through the nodes
! where fix=0.

        real(8) :: influx,outflux

!
! initialize precip. array - added by DW 10/06
! TAE 7/01
! NOTE: if you unccomment the following line the program will execute, but
! if start to pass prec to other subroutines you will seg fault.  I'm not
! sure why at this point? TAE
        real(8), dimension (:), allocatable :: prec
        real(8), dimension (:), allocatable :: y_gr
        real(8), dimension (:), allocatable :: x_gr
        integer(4) :: nxs,nys
        real(8), dimension (:,:), allocatable :: prec_gr
        real(8), dimension (:,:), allocatable :: z

! the following arrays are needed in the natural neighbour routines
! have a look inside the library routienes to figure out what
! their purpose is

        real(8), dimension (:,:), allocatable :: points
        real(8) :: eps
        integer(4), dimension (:,:), allocatable :: vertices
        integer(4), dimension (:,:), allocatable :: neighbour
        integer(4), dimension (:), allocatable :: nodes
        integer(4), dimension (:), allocatable :: vis_tlist
        integer(4), dimension (:), allocatable :: vis_elist
```

```
        integer(4), dimension (:), allocatable :: add_tlist
        integer(4), dimension (:), allocatable :: nodelist
        integer(4), dimension (:), allocatable :: tlist
        logical, dimension (:), allocatable :: c_list
        integer(4), dimension (:,:), allocatable :: v_local
        integer(4), dimension (:,:), allocatable :: n_local
        logical, dimension (:), allocatable :: mask
        logical, dimension (:,:), allocatable :: mask_e
        logical, dimension (:), allocatable :: inactive

! the following arrays are used to calculate the surface of
! voronoi cells

! WK: needs to be changed to 8 bytes later
! WK: c code currently also uses float instead of double
        real(8) :: xy(2),pp(2,nbmax),aa(nbmax,2),bb(nbmax)

! the following arrays are used to solve the diffusion equation iteratively

        real(8) :: hp(nnodemax)
        integer(4) :: nkcon(nnodemax)
        real(8) :: ael1(6,nnodemax*3),ael2(6,nnodemax*3)
        real(8) :: bel(nnodemax),diag(nnodemax)
        integer(4), dimension (:,:), allocatable :: kcon
        integer(4), dimension (:,:), allocatable :: jcon

! itadd and jtadd are used when dynamic remeshing is turned
! on they are used to determine where resolution has to be increased

        integer(4) :: itadd(nnodemax),jtadd(nnodemax*3)

! the following arrays are used in the flexural isostasy
! calculations
! nflex is the resolution at which the FFT are done to calculate the
! flexural response; nflex has to be a power of 2

        real(8), dimension (:,:), allocatable :: flex
        real(8), dimension (:,:), allocatable :: work_flex


! Variables needed for landsliding routine landslide.f
! The following lines were added by Ehlers 6/01
        real(8) :: smax(nnodemax),tt(nnodemax)
        integer(4), dimension (:,:,:), allocatable :: cell
```

```
! landslide time series variables DS 6/15/1
!  these store info between write_output calls
!  might be problems if there are more than nnodemax slides
!  during a period

! dslope:  downstream slope found after all erosion DS 11/18/1
        real(8) :: dslope(nnodemax)

        integer(4) :: bdry(nnodemax)

! timeint: needed for naming tecplot output files
        integer(4) :: timeint

! ice grids and arrays
        real(8), dimension (:,:), allocatable :: zpt
        real(8), dimension (:,:), allocatable :: bipt
        real(8), dimension (:,:), allocatable :: bface
        real(8), dimension (:,:), allocatable :: hforice
        real(8), dimension (:,:), allocatable :: tbipt
        real(8), dimension (:,:), allocatable :: htoc
        real(8), dimension (:,:), allocatable :: httoc
        real(8), dimension (:,:), allocatable :: constoc
        real(8), dimension (:,:), allocatable :: slidetoc
        real(8), dimension (:,:), allocatable :: iceftoc

        integer(4), dimension(:,:), allocatable :: antitoc
        integer(4), dimension(:), allocatable :: anthi
        integer(4), dimension(nnodemax) :: glacier

        real(8) :: iceth(nnodemax),tott(nnodemax),slide(nnodemax)
        real(8) :: gbalance(nnodemax),strict(nnodemax)
        real(8) :: delx, dely
        real(8) :: shelf(nnodemax),sort(nnodemax),th(nnodemax)

        real(8)  totalerosion(nnodemax)

        integer(4) :: nt, dye, dxe, nexd, neyd, ned, nel, ner, neu, nexl, nexr, nextra, nexu
        integer(4) :: neyl, neyr, neyu, remeshflag = 0

        integer(4) :: tsys0, tsysT, cascade_ioStatus, system_Result

        integer(4) :: iseed, ice_topotime, cascade_istep, itime, nnode0, nxice, nyice
        logical :: ivocal, cascade_FileExists

        real(8) :: dhmaxglac, dtold, hmax, hmedian, hmin, oro_time, shorttime, meshtime, tcheck,
temperature
```

```
    real(8) :: time, delta, dhmaxfluv, dhminfluv, dhminglac, GLOBAL_C, GLOBAL_CS
    real(8) :: imposed_oro_time ! Edited vmbp
    real(8) :: diffusivity, dtdzb, dhmaxdiff, dhmindiff, dtmin, dtmax

  end module cascade_globals
```

8.2.3 New Rainmaker – Imposed Precipitation

This is the code of the new 'Rainmaker_imposed.f90' file.

```
!-----------------------------------------------------------------
module m_rainmaker_imposed
  use rt_param
  use cascade_globals

  implicit none
  contains
!#######################################################################!
!-----------------------------------------------------------------
! Option for Imposed Precipitation
! This module was added by Victoria M Buford Parks January 2020
!-----------------------------------------------------------------
  subroutine rainmaker_imposed(configData)

! code to import external precipitation data apply to  CASCADE...
      ! only executes if using imposed precipitation (icecascade.in : A9=3)
      ! only called on initialization of cascade and when
      !   imposed_oro_time counter is greater than configData%imposedPrecipUpdateTime  (value
in cascade.f90)
      ! Defined in cascade.f90: oro_time = oro_time+global_cascade_dt

!     inputs: nnode            no. of nodes
!             x,y,h            x,y, and h of nodes
!             surface          surface area assoc. with each node
! outputs:
!             prec             precipitation rate at each node
!             water            precip *area at each node
!             x_gr,y_gr,prec_gr x,y,precip on reg. grid
      ! Notes on: Input precipitation grids
      ! The .txt referenced in icecascade.in / A10 must be
      ! in this format:
        ! first line: depth in y dimension in km
        ! second line: number of nodes in y dimension
```

```
    ! third line: number of samples to take for interpolation
    ! fourth line: step size in x direction for interpolation
    ! fifth line to end of file: time in Myrs and filename
    !
    ! example:
    ! 10.0
    ! 5
    ! 10
    ! 1.0
    ! 8.10 file1.dat
    ! 3.2 file2.dat
    ! 0 file3.dat
    !
    ! The first four lines will be ignored herein
    ! The ages are the end age at which the precipitaiton grid will be applied:
    ! e.g. file1.dat will be applied from the model start time (input C3 in icecascade.in) until
8.1Ma

  ! each grid (e.g. filei.dat) must have a header and
  ! three input columns: x y precip (units: km, km, and m/yr)
    ! example:
    ! x[km]     y[km]   P[m/yr]
    ! 0.000     0.000   1.200
    ! 0.076     0.001   0.250
    ! 15.000    22.250  4.600

  use rt_param
  use cascade_globals
  use m_check_var

  implicit none
! begin routine.
  type(config) :: configdata
  real(8), dimension(:), allocatable :: x_precip, y_precip, z_precip, &
    InputPrecipTimeStep, PTimeStep,x_grid,y_grid
  real(8) :: resolution
  character(255), dimension(:), allocatable :: PrecipGridFiles
  character(255) :: PrecipFileName_List, PFileName_Grid1,FileName_Precip, &
    FileName3,FileName4
  integer(4) ::  FileUnit1 = 92, FileUnit3 = 94, &
    FileUnit4 = 95, Filelines, NumberOfPrecipFiles
  integer(4) :: i, j, k, jj, count, iflag_precip
  integer :: ios
  logical :: Test
```

```
print*,'Module: rainmaker_imposed'

PrecipFileName_List = trim('input/IceCascade/'//configData%precipname)
! determine the number of precip files
call LengthOfFile(PrecipFileName_List, FileLines)

NumberofPrecipFiles = FileLines - 4
FileLines = 0

! open list of precip files
open(FileUnit1, file=PrecipFileName_List, status='old',action='read')
! Read first four lines & do nothing
   do i = 1,4
      read (FileUnit1, *)
   end do

   ! Read said input .txt file (list of precip files)
   allocate (InputPrecipTimeStep(NumberofPrecipFiles), &
     PrecipGridFiles(NumberofPrecipFiles),PTimeStep(NumberofPrecipFiles+1))
   ! Store the names of the precip files
     ! & compute intended time (in cascade reference frame) of each file
     PTimeStep(i)=0.0_8
     print*,'Model Run Time [yrs]:',configData%endtime
     print*,"i=",1," PTime(i)= ",int(PTimeStep(1))
   do i = 1, NumberofPrecipFiles
      ! Read time steps and input file names;
      read(FileUnit1, *, iostat=ios), InputPrecipTimeStep(i), PrecipGridFiles(i)
      if (ios/=0) exit

      ! reverse time line
      ! This only works if precip files start at the same time as the model start time
      !PTimeStep(i) = (InputPrecipTimeStep(1) - InputPrecipTimeStep(i)) * 1e6_8
      PTimeStep(i+1) = configData%endtime - (InputPrecipTimeStep(i) * 1e6_8) ! in years
      print*,"i=",i+1,' PTime(i) [y]: ',int(PTimeStep(i+1))
      print*,'input time (i) [Ma]: ',int(InputPrecipTimeStep(i)),'Grid:',PrecipGridFiles(i)
      call ShortenString(PrecipGridFiles(i), 4)
      !print *, "PrecipGridFiles(i): ", PrecipGridFiles(i)
   end do
close(FileUnit1)

resolution=4 ! points per km

! Create Precip Files in initialization
if (cascade_istep.eq.1) then
   print*,'Initializing rainmaker_imposed'
   call Write_CascadePrecipFiles(PrecipGridFiles, NumberofPrecipFiles, &
```

```
        int(configData%sidex), int(configData%sidey), &
        configData%y_flip, PTimeStep,resolution)
    ! then read in first file and interpolate onto precip nodes
    FileName_Precip = 'input/IceCascade/'//trim(PrecipGridFiles(1))//'_interpolated.dat'
    ! go to 100
endif
! ! Determine which precip file to use, and interpolate it onto nodes.
 do i=2,NumberofPrecipFiles+1
    if    ( (time.ge.0) .and. (time.le.PTimeStep(2)) ) then
        FileName_Precip = 'input/IceCascade/'//trim(PrecipGridFiles(1))//'_interpolated.dat'
        jj=i-1
    elseif ( (time.gt.PTimeStep(i-1)) .and. (time.le.PTimeStep(i)) ) then
        FileName_Precip = 'input/IceCascade/'//trim(PrecipGridFiles(i-1))//'_interpolated.dat'
        jj=i-1
    endif
enddo
print *, "FileName_Precip: ", FileName_Precip

call LengthOfFile(FileName_Precip,FileLines)

            ! ! Trying to call and create precip files in case they get deleted mid-run;
            ! ! error rank mismatch somewhere.
            ! inquire(file=FileName_Precip, exist=Test)
            !     if (Test) then
            !         !FileName(1:(FileName_StringLength - ReduceBy))
            !         call ShortenString(FileName_Precip,17)
            !         FileName_Precip=FileName_Precip//'.dat'
            !         call Write_CascadePrecipFiles(FileName_Precip, int(1), &
            !             int(configData%sidex), int(configData%sidey), &
            !             configData%y_flip, PTimeStep,resolution)
            !         call ShortenString(FileName_Precip,4)
            !         FileName_Precip=FileName_Precip//'_interpolated.dat'
            !     end if

allocate(x_grid(FileLines-1), y_grid(FileLines-1), z_precip(FileLines-1))
call Read_CascadePrecipFile(FileName_Precip,FileLines-1,x_grid,y_grid,z_precip)
            ! Interpolate nice precip data onto CASCADE nodes
            !----------------------------------------

            ! !----------------------------------------
            ! !----------------------------------------
            ! !----------------------------------------
            ! ! write a test precip file !! For testing only!
            ! FileName3='input/IceCascade/test_input_interpolated.dat'
            ! do j  = 1,configdata%sidey*resolution
            !     y_precip(j) = y_grid(j)
```

```
        !     !count=count+configdata%sidex*resolution
        ! enddo! fix this above to read in y_precip data correctly

        ! count=1
        ! do j  = 1,configdata%sidex*resolution
        !     x_precip(j) = x_grid(j+count)
        !     count=count+configdata%sidey*resolution
        !      enddo!      fix      this      above      to      read      in      y_precip      data
correctlyx_precip=x_precip(1,configdata%sidex*resolution)
        ! open(FileUnit3, file=FileName3, status='new')
        ! do j=1,size(y_precip)
        ! do kk=1,size(x_precip)
        !     prec_gr(kk,j)=z_precip( (j-1)*(configdata%sidex*resolution) + kk)
        !     write(FileUnit3, *) x_precip(kk), y_precip(j), prec_gr(kk,j)
        ! end do
        ! !count=count+configdata%sidex*resolution
        ! end do
        ! close(FileUnit3)
        ! !! For testing only!
        ! !----------------------------------------
        ! !----------------------------------------
        ! !----------------------------------------
   call interp2(configData%nnode,x_grid,y_grid,z_precip,resolution)
        ! !----------------------------------------
        ! !----------------------------------------
        ! !----------------------------------------
        ! ! write a test output precip file !! For testing only!
        ! FileName4='input/IceCascade/test_output_interpolated.dat'
        ! open(FileUnit4, file=FileName4, status='new')
        ! ! count=0
        !     do j=1,size(prec)
        !        write(FileUnit4, *)  x(j), y(j), prec(j)
        !     end do
        ! close(FileUnit4)
        ! !! For testing only!
        ! !----------------------------------------
        ! !----------------------------------------
        ! !----------------------------------------

  deallocate(InputPrecipTimeStep, PrecipGridFiles, PTimeStep, z_precip, &
  x_grid, y_grid)

  ! multiply precip rate by nodal area to get rate of water input.
  !       surface_area     = surface associated with each node, [km2] = memory(1,7)

  do i =1,configData%nnode
```

```
        water(i) = memory(i, 7)*prec(i)
    enddo
    !z=zint

return
end subroutine rainmaker_imposed

! memory(*,1)=dhcrit
! memory(*,2)=dhfluvial     (amount of material eroded/deposited over the time step)
! memory(*,3)=dhdiff        (total height removed/added)
! memory(*,4)=hiso
! memory(*,5)=fix           (Boundary condition array)
! memory(*,6)=newsurface    (which nodal surface area has to be updated)
! memory(*,7)=surface_area  [km2]
! memory(*,8)=dhlandslide   (added by Ehlers 6/01)
! memory(*,9)=dhglacier     (added by Yanites 10/09)

!       x,y         = x- and y-nodal positions [km]
!       h           = present topography [m]
!       dt          = time step length [yr]
!       water       = water discharge at each point
!       prec_gr [m/yr]
!       bckgr [m/yr]

!####################################################################!

!----------------------------------------
subroutine
Write_CascadePrecipFiles(PrecipGridFiles,NumberofPrecipFiles,size_x,size_y,flip,PTimeStep,r
esolution)
!----------------------------------------
! added by vmbp Jan 13 2020
    implicit none

    real(8), dimension(:), allocatable :: x_precip, y_precip, z_precip, &
        x_grid,y_grid,PTimeStep
    real(8), dimension(:,:), allocatable :: z
    real(8) :: resolution !, dimension(:)
    !real(8) :: prec_gr
    character(255), dimension(:) :: PrecipGridFiles
    character(255) ::  PFileName_Grid1, FileName_Precip
    integer(4) :: FileUnit5 = 74, Filelines, NumberOfPrecipFiles,size_x,size_y,i,j,k,kk
    integer :: ios
    logical :: Test,flip

    ! Create Precipitation Files
```

```
  do i = 1, NumberOfPrecipFiles
     ! create new Precip file
     FileName_Precip = trim('input/IceCascade/'//PrecipGridFiles(i))//'_interpolated.dat'

     ! Test whether file already exists
     inquire(file=FileName_Precip, exist=Test)
     if (Test) then
        cycle
     end if

     ! Initialize Precip
     ! Read in Precipitation Grid
     PFileName_Grid1 = 'input/IceCascade/'//trim(PrecipGridFiles(i))//'.dat'
     call LengthOfFile(PFileName_Grid1, FileLines)
     print*, "PFileName_Grid1:", PFileName_Grid1, 'FileLines= ', FileLines
     ! Each input precip data grid has A HEADER!
     allocate (x_precip(FileLines-1), y_precip(FileLines-1), z_precip(FileLines-1))

     call Read_InputPrecipGrid(PFileName_Grid1, FileLines, y_precip, x_precip, z_precip)

     ! Performing y-flip switch if enabled
     if (flip) then
        do j=1,FileLines-1
           x_precip(j) = size_y - x_precip(j)
        end do
     end if

     print*, 'Writing Precipitation file for time step: '
     print *, "i: ", i, "PtimeStep(i): ",nint(PTimeStep(i)),"years PrecipGridFiles(i): ",
PrecipGridFiles(i)

     ! Create Regular Precipitation grid interpolated from import
     open(FileUnit5, file = FileName_Precip, status='new')
     print*, FileName_Precip
     write(FileUnit5, *), 'REGULAR GRID # x,y,precip,; units: km, mm/year'
     ! define regular grid.
     ! with 1/4km grid spacing
     ! configdata%sidex = length of model side
     allocate (x_grid(size_x*resolution),y_grid(size_y*resolution))
     do k  = 1,size_x*resolution
           x_grid(k) = dble((k-1)/resolution)
     enddo
     do j  = 1,size_y*resolution
           y_grid(j) = dble((j-1))/resolution
     enddo
     print*, 'size(x_grid):', size(x_grid), 'size(y_grid):', size(y_grid)
```

```
        ! modify x_gr and y_gr to appropriate size
        print *, "x_precip size: ", size(x_precip), ", y_precip size: ", size(y_precip), &
        ", z_precip size: ", size(z_precip), ", size_x: ",size_x, ", size_y: ",size_y
        ! calculate/interpolate my precipitation on regular grid
           ! subroutine grid2(nnode,x,y,h,x_gr,y_gr,nxs,nys,z)
        allocate(z(size(x_grid), size(y_grid)))
        call grid2_notglobalvar(size(z_precip(:)), x_precip, y_precip,  z_precip, &
           x_grid,y_grid,size(x_grid), size(y_grid),z)
        do k=1,size(x_grid)
           do kk=1, size(y_grid)
           write(FileUnit5,*), x_grid(k), y_grid(kk), z(k,kk)
           enddo
        enddo
        close(FileUnit5)

        deallocate (x_precip, y_precip, z_precip,x_grid,y_grid, z)

    end do
    print*, 'DONE WRITING IMPOSED PRECIP FILES!'

end subroutine Write_CascadePrecipFiles
!----------------------------------------
! Subroutine that reads in Input Precip grids
!----------------------------------------
! added by vmbp Jan 13 2020
subroutine Read_InputPrecipGrid(FileName, Lines, x, y, z)
    implicit none

    character(255), intent(in) :: FileName
    integer(4) :: FileUnit11 = 12, ios, i
    integer(4), intent(in) :: Lines
    real(8), dimension(:), intent(out) :: x, y, z

    open(FileUnit11, file=FileName, status='old')
        read(FileUnit11,*) ! Skips 1st line: header
        do i = 1,Lines
            read(FileUnit11, *, iostat=ios), x(i), y(i), z(i)
        end do
    close(FileUnit11)
end subroutine Read_InputPrecipGrid

!--------------------------------------------------------------------
! Subroutine to interpolate regularly gridded data onto Cascade mesh
!--------------------------------------------------------------------
! added by vmbp Jan 13 2020
subroutine interp2(nnode2,x_grid,y_grid,prec_grid,resolution)!,resolution)
```

```
! written by Victoria M. Buford Parks (vmbparks@gmail.com)
! for use with imposed orography, Jan 2020
   use cascade_globals
   use m_check_var

   implicit none
   ! subroutine to do linear interpolation from regular grid back onto nodes
   !    inputs:
   ! (global)   nnode           number of nodes
   ! (global)   x,y             coords at nodes
   !          x_grid,y_grid        coordinates of regular grid
   ! (global)   prec_grid          precipitation on regular grid
   !          resolution       spacing of x_grid, y_grid values
   !
   !    outputs:
   !          prec             precipitation on nodes
   real(8), dimension(:), intent(in) ::  x_grid, y_grid !resolution
   real(8), dimension(:), intent(in) :: prec_grid(size(x_grid))
   !real(8), intent (out) :: prec(nnode)
   integer(4)  :: i,k, nnode2,&
      idx_xMax_yMax,idx_xMin_yMax,idx_xMax_yMin,idx_xMin_yMin
   real(8)     :: xqmax, xqmin, yqmax, yqmin, &
                  xmx_ymx_wt, xmx_ymn_wt, xmn_ymx_wt, &
                  xmn_ymn_wt, resolution
   !resolution=abs(x_grid(2)-x_grid(1))
   ! begin loop over nodes
   do i=1, nnode2
      ! determine all nearest neighbors
      xqmax    = ceiling(x(i)*resolution)/resolution
      xqmin    = floor(x(i)*resolution)/resolution
      yqmax    = ceiling(y(i)*resolution)/resolution
      yqmin    = floor(y(i)*resolution)/resolution
      ! remove issues with nodes near the boundary of modle
      if (xqmax.lt.0) then
         xqmax=0
      elseif (xqmin.lt.0) then
         xqmin=0
      elseif (yqmax.lt.0 )then
         yqmax=0
      elseif (yqmin.lt.0 )then
         yqmin=0
      elseif (xqmax.gt.maxval(x_grid)) then
         xqmax=maxval(x_grid)
      elseif (xqmin.gt.maxval(x_grid)) then
         xqmin=maxval(x_grid)
      elseif (yqmin.gt.maxval(y_grid)) then
```

```
   yqmin=maxval(y_grid)
elseif (yqmax.gt.maxval(y_grid)) then
   yqmax=maxval(y_grid)
end if
! Find nearest 4 grid locations to sample point
do k=1,size(x_grid)
   if ((x_grid(k).eq.xqmax) .and. (y_grid(k).eq.yqmax)) then
      idx_xMax_yMax = k
      !cycle
   elseif ((x_grid(k).eq.xqmin) .and. (y_grid(k).eq.yqmax)) then
      idx_xMin_yMax = k
      !cycle
   elseif ((x_grid(k).eq.xqmax) .and. (y_grid(k).eq.yqmin)) then
      idx_xMax_yMin = k
      !cycle
   elseif ((x_grid(k).eq.xqmin) .and. (y_grid(k).eq.yqmin)) then
      idx_xMin_yMin = k
      !cycle
   endif
 end do
 ! intend to average the above points, but there is some error in weighting. thus, use only one.
! do k=1,size(x_grid)
!    ! Find MOVE velocities within resolution range of regular grid point
!    allocate(LocArray(size(x_grid)))
!    do l=1,FileLines
!       if ((Grid_y1(l) <= (RegGrid_X(k) + RegGrid_Resolution)) .and. &
!          (Grid_y1(l) > (RegGrid_X(k) - RegGrid_Resolution)) .and. &
!          (Grid_z1(l) <= (RegGrid_Z(k) + RegGrid_Resolution)) .and. &
!          (Grid_z1(l) > (RegGrid_Z(k) - RegGrid_Resolution))) then
!          LocArray(l) = l
!       else
!          LocArray(l) = 0.0_8
!       end if
!    end do
!    LocArray = pack(LocArray, LocArray /= 0.0_8 .and. LocArray < FileLines)

!    ! Calculate average vertical and horizontal velocities of found record.
!    ! If record is empty, set velocities to zero.
!    if (size(LocArray) == 0) then
!       RegGrid_Xvel(k) = 0.0_8
!       RegGrid_Zvel(k) = 0.0_8
!       deallocate(LocArray)
!       allocate(LocArray(1))
!    else
!       RegGrid_Xvel(k) = sum(Vel_y(LocArray)) / size(LocArray)
!       RegGrid_Zvel(k) = sum(Vel_z(LocArray)) / size(LocArray)
```

```
       !     end if

       !     ! Write output to file
       !      write(FileUnit2,*), k,  0.0_8, RegGrid_X(k), RegGrid_Z(k), 0.0_8, RegGrid_Xvel(k),
RegGrid_Zvel(k), 7
       !     deallocate(LocArray)
       ! end do

       ! compute linearly interpolated, weighted precipitaiton
       ! prec(i)  =  (prec_grid(idx_xMax_yMax) * xmx_ymx_wt + &
       !            prec_grid(idx_xMax_yMin)  * xmx_ymn_wt + &
       !            prec_grid(idx_xMin_yMax)  * xmn_ymx_wt + &
       !            prec_grid(idx_xMin_yMin)  * xmn_ymn_wt) /&
       !            (xmx_ymx_wt+xmx_ymn_wt+xmn_ymx_wt+xmn_ymn_wt)
        prec(i)  =  (prec_grid(idx_xMax_yMax))! + prec_grid(idx_xMax_yMin) + &
                   ! prec_grid(idx_xMin_yMax) + prec_grid(idx_xMin_yMin))/4

        if (i.eq.2) then
           print*,"prec(i=2)", prec(i)
           !exit
       end if
   enddo
end subroutine interp2

!----------------------------------------
! Subroutine Read_CascadePrecipFile
!----------------------------------------
! added by vmbp Jan 13 2020
subroutine Read_CascadePrecipFile(FileName, Lines, loc_x, loc_y, loc_z)
! Reads an imposed precipitation file file
   implicit none

   character(255), intent(in) :: FileName
   integer(4) :: FileUnit6 = 13, ios, j
   integer, intent(in) :: Lines
   real(8), dimension(:), intent(out) :: loc_x, loc_y, loc_z

   open(FileUnit6, file=FileName, status='old')
   read(FileUnit6,*) ! read header and do nothing
      do j = 1,Lines-1
         read(FileUnit6, *, iostat=ios), loc_x(j),  loc_y(j), loc_z(j)

         if (ios/=0) then
            exit
         end if
      end do
```

```
   close(FileUnit6)
end subroutine Read_CascadePrecipFile
!----------------------------------------
! Subroutine grid2_notglobalvar
!----------------------------------------
! borrowed from rainmaker.f90
subroutine grid2_notglobalvar(nnode2,x_data,y_data,h_data,x_gr,y_gr,nxs2,nys2,z)
   implicit none
!    inputs:
!          nnode2     number of nodes
!          x_data,y_data        co-ords of nodes
!          h_data          elevation of nodes
!          x_gr, y_gr co-ords of reg grid
!          nxs, nys   number of values in x_data and y_data direction
!          subsam    subsampling of input nodes.
!
!     outputs:
!          z       elevation on regular grid
   ! FIX: initialization /create precip files vs after precip files exist + time comparison

   integer(4), intent(in) :: nnode2, nxs2, nys2
   real(8), intent(in) :: x_data(nnode2), y_data(nnode2), h_data(nnode2), x_gr(nxs2), y_gr(nys2)
   real(8), intent(out) :: z(nxs2, nys2)

   integer(4) :: i, j, k, n1, n2, n3, n4
   real(8) :: radius, d, d1, d2, d3, d4
   real(8) :: factor1, factor2, factor3, factor4, factor_sum

   ! consider a radius of 10 km
   radius = 10.0_8
   n1 = 1
   n2 = 1
   n3 = 1
   n4 = 1

   print *, "x_data min: ", minval(x_data), &
   ", x_data max_data: ", maxval(x_data), &
   ", y_data min: ", minval(y_data), &
   ", y_data max_data: ", maxval(y_data)
   print *, "x_gr min: ", minval(x_gr), ", x_gr max_data: ", maxval(x_gr)
   print *, "y_gr min: ", minval(y_gr), ", y_gr max_data: ", maxval(y_gr)

   do i=1,nxs2
      do j=1,nys2
```

```
! now look for th_datae 4 nearest neigh_databors
d1 = huge(d1)
d2 = huge(d2)
d3 = huge(d3)
d4 = huge(d4)

do k=1,nnode2
   d = sqrt((x_gr(i) - x_data(k))**2 + (y_gr(j) - y_data(k))**2)

   if (d < d1) then
      d2 = d1
      d3 = d2
      d4 = d3
      d1 = d

      n2 = n1
      n3 = n2
      n4 = n3
      n1 = k
   else if (d < d2) then
      d3 = d2
      d4 = d3
      d2 = d

      n3 = n2
      n4 = n3
      n2 = k
   else if (d < d3) then
      d4 = d3
      d3 = d

      n4 = n3
      n3 = k
   else if (d < d4) then
      d4 = d

      n4 = k
   endif
enddo

factor1 = exp(-d1/radius)
factor2 = exp(-d2/radius)
factor3 = exp(-d3/radius)
factor4 = exp(-d4/radius)

factor_sum = factor1 + factor2 + factor3 + factor4
```

```
        z(i, j) = ((h_data(n1)*factor1) + (h_data(n2)*factor2) + (h_data(n3)*factor3) + (h_data(n4)*factor4)) / factor_sum

      enddo
   enddo
end subroutine grid2_notglobalvar

!----------------------------------------
! Subroutine that shortens a given string by an integer number of
!----------------------------------------
! added by vmbp Jan 13 2020
subroutine ShortenString(FileName, ReduceBy)
   implicit none

   character(255), intent(out) :: FileName
   integer(4), intent(in) :: ReduceBy
   integer(4) :: FileName_StringLength

   FileName_StringLength = len_trim(FileName)

   if (FileName_StringLength < ReduceBy) then
      print*, 'Subroutine ShortenString: String is too short to be shortened by ', ReduceBy, ' characters.'
   else
      FileName = FileName(1:(FileName_StringLength - ReduceBy))
   end if
end subroutine ShortenString
!----------------------------------------
! Subroutine that determines the length of a file
!----------------------------------------
! added by vmbp Jan 13 2020
subroutine LengthOfFile(FileName, LineCounter)
   implicit none

   character(255), intent(in) :: FileName
   integer(4) :: FileUnit2 = 11, ios
   integer(4), intent(out) :: LineCounter

   open(FileUnit2, file=FileName, status='old')
      LineCounter = 0
      do
         read(FileUnit2, *, iostat=ios)
         if (ios/=0) then
            exit
         end if
         LineCounter = LineCounter + 1
```

```
        end do
    close(FileUnit2)
end subroutine LengthOfFile
end module m_rainmaker_imposed
```

8.2.4 Rear Boundary Nodes

This is the new ‘UpdateBoundaryNodes.f90’that allows the user to control the uplift of the rear boundary, regardless of user use of N9b (to flip the model space).

```
! tectonic_uplift
        module m_UpdateBoundaryNodes
          contains
          subroutine UpdateBoundaryNodes(configData)
                use rt_param
                use cascade_globals
                implicit none

!       subroutine tectonic_uplift(x,y,h,h0,hi,nnode,fix,dt,time,influx,surface,uplift_rate,shelf)

! this routine defines the tectonic uplift function
! it may vary in space and time

! INPUT: x,y        = x- and y-nodal coordinates (in km's)
!        h          = present topography (in m's)
!        h0         = bedrock-alluvials interface (in m's)
!        hi         = original topograpy (in m's)
!        nnode      = number of nodes
!        fix        = boundary conditions
!        dt         = time step length
!        time       = current time
!        surface    = surface associated with each node
!        uplift_rate = rate in m/yr - defined w/ general params
!       shelf       =array controlling where uplift occurs
! OUTPUT:  h        = update topography
!          hi       = updated original topograpy
!          h0       = updated bedrock-alluvials interface
!          influx   = updated influx of material by tectonic uplift

! subroutines called:
! NONE
```

```
!      real     x(nnode),y(nnode),h(nnode),h0(nnode),hi(nnode)
!      real     fix(nnode),shelf(nnode)
!      real     surface(nnode),influx
!      real     uplift_rate

!      real     dij

  type(config) :: configData
  integer(4) :: inode,count
  real(8) :: dh




! the example is a square topography that stops growing after a while
! note that you can use the array fix to prevent the base level to move
! with tectonic uplift

! commented out 6/15/1, now in initialize_general_parameters.f
!      uplift_rate=2.E-3
  !print*,"Rear Boundary Uplifting"
  !print*,"Max Elevation [m]:",configData%imposeRearBoundaryUpliftMaxElevation*1e3_8
  !print*,"Uplift Rate [m/yr]:",configData%imposeRearBoundaryUpliftRate / 1e3_8
  !print*,"dt:",global_cascade_dt
  count=0

  do inode=1,configData%nnode

! uplift the topography keeping the boundary pinned
!     if (memory(inode, 5) > 0.5_8) then

!       dh=configData%uplift_rate*global_cascade_dt*shelf(inode)

! ! uplift interior
!       h(inode) = h(inode) + (dh*shelf(inode))
!        if (shelf(inode) > 0.0 ) then
!            print *, "h(inode), dh, uplift_rate, dt, shelf: ", &
!            h(inode), dh, configData%uplift_rate, global_cascade_dt, shelf(inode)
!        endif

!       print*,dh,shelf(inode),h(inode),dt,uplift_rate
! added 6/15/1
!  rectangular uplift pattern
!        if (x(inode).gt.3.0.and.x(inode).lt.30.-3.0.and.
!     &      y(inode).gt.3.0.and.y(inode).lt.30.-3.0) then
!         h(inode)=h(inode)+dh
```

```
!       endif
!

!  rectangular uplift pattern
!      if (x(inode).gt.2.0.and.x(inode).lt.8.0) then
!       h(inode) = h(inode) + dh
!      endif

! changes added 6/14/1
!     circular uplift pattern
!     dij = sqrt((x(inode)-25.)**2+(y(inode)-25.)**2)
!      if (dij.lt.10.) then
!       h(inode)=h(inode)+dh
!      endif

    ! x and y are the x- and y-coordinates of the nodes in km
! h is the current topography
! hi is the initial topography
! h0 is the location of the bedrock interface
! all h's are in m, iceth(nnodemax)

    !configData%imposeRearBoundaryUpliftMaxElevation is in km (CONVERT TO m's)
    !configData%imposeRearBoundaryUpliftRate is in mm/yr (or km/My) (CONVERT TO m's)

    ! Elseif condition Added by Victoria M Buford Parks Jan 2020
    ! For nodes that are at the rear boundary (eg y=200)
        if (y(inode).eq.configData%sidey) then
        !    print*, "inode:",inode,"y:",int(y(inode)),"h:",h(inode)
            if (h(inode).lt.configData%imposeRearBoundaryUpliftMaxElevation*1e3_8) then
                dh = configData%imposeRearBoundaryUpliftRate / 1e3_8 * global_cascade_dt
                h(inode)=h(inode)+dh

                if (h(inode).gt.configData%imposeRearBoundaryUpliftMaxElevation*1e3_8) then
                    h(inode)=configData%imposeRearBoundaryUpliftMaxElevation*1e3_8
                endif

              !  print*, "inode:",inode,"y:",y(inode),"new h:",h(inode),"dh",dh

            endif
        count=count+1
        ! Return to original code VMBP Jan 2020
        ! else
        !     dh = 0.0_8

    ! do not touch the following lines
    ! they update h0, hi and calculate the influx of material into the
```

```
   ! landscape
      h0(inode)=h0(inode)+(dh)
      hi(inode)=hi(inode)+(dh)
      influx=influx+dh*memory(inode,7)
      endif

   enddo
   !print*,"# nodes at rear boundary", count
  return
  end subroutine UpdateBoundaryNodes
end module m_UpdateBoundaryNodes
```