The BioRECIPE Representation Format

Emilee Holtzapple, Gaoxiang Zhou, Cheryl Telmer, Casey Hansen, Difei Tang, Haomiao Luo,
Natasa Miskov-Zivanov

Mechanisms and Logic of Dynamics (MeLoDy) Lab
Electrical and Computer Engineering Department, Bioengineering Department
University of Pittsburgh

1 Introduction

The BioRECIPE (Biological system Representation for Evaluation, Curation, Interoperability, Preserving, and Execution) representation format was introduced[1] to facilitate seamless human-machine interaction while creating, verifying, evaluating, curating, and expanding executable models of intra- and intercellular signaling. This format allows a human user to easily preview and modify any model component, while it is at the same time readable by machines and can be processed by a suite of model development and analysis tools. The BioRECIPE format is a (directed) graph-based tabular format, that is, it is most suitable for representing influences and executable network models that have a *directed graph* as their underlying structure (example in Figure 1A) although it can also be used for representing undirected network models. Here, we provide details and demonstrate the utility of the BioRECIPE representation format for:

- *event-based interaction lists*
- *element-based executable models*.

When creating interaction lists or models in the BioRECIPE format, information and data can be obtained from different sources, and input into BioRECIPE tables or spreadsheets automatically or manually (Figure 1B). Interaction lists and models written in the BioRECIPE format are convenient for different types of analysis and use (Figure 1C), either with automated tools, or manually, when human input is needed.

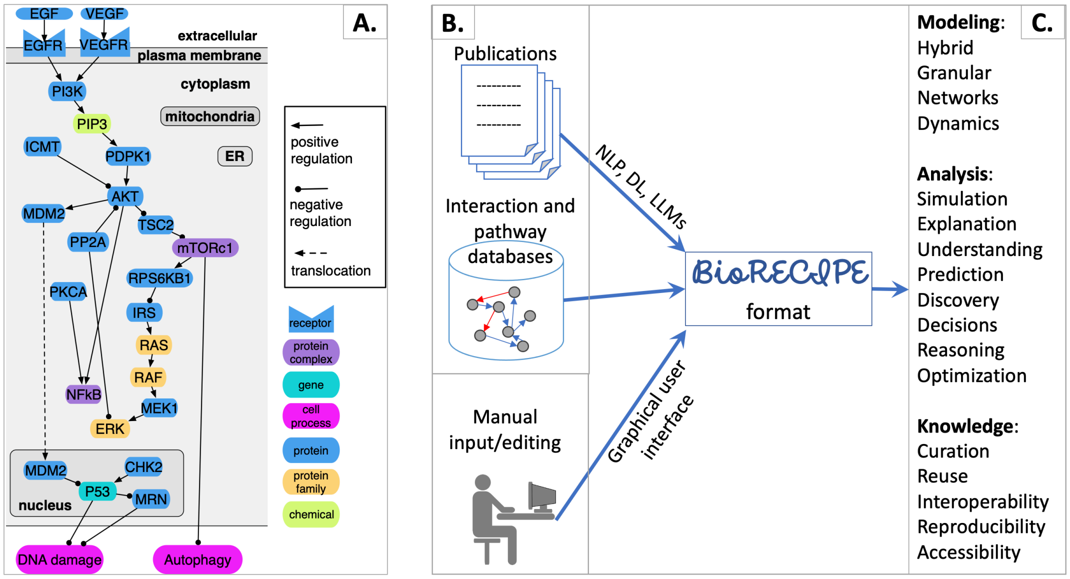


*Figure 1. (A) Signaling network that can be represented in the BioRECIPE format, (B) Multiple types of information are compatible with BioRECIPE format, (C) Models and interaction lists represented in the BioRECIPE format have a wide variety of uses. *ER-endoplasmic reticulum, NLP-natural language processing, DL-deep learning, LLM-large language model.*

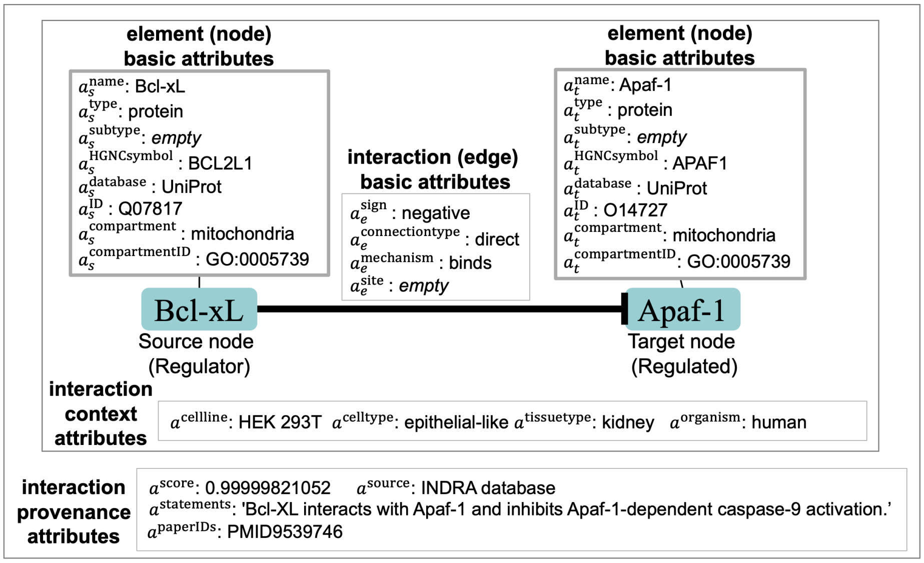
[1] Additional details on how to create files in the BioRECIPE format, examples, and links to related code can be found in BioRECIPE's documentation: https://melody-biorecipe.readthedocs.io

2 Interaction lists

In the BioRECIPE format, interactions can be represented using the *event-based* interaction list format, that is, a format in which each individual interaction (biological event) is represented separately, and interaction participants are represented as arguments of the event. In the tabular form, each interaction is assigned one row and the column headers match interaction attribute names. An example biological interaction, represented as a directed signed edge between two nodes, including node, edge, context, and provenance attributes is illustrated in the figure below (subscripts: *s* – source node, *t* – target node, *e* – edge).

As illustrated in Figure 2, the BioRECIPE representation of an interaction includes four types of attributes:

- *element (node) basic* attributes
- *interaction (edge) basic* attributes
- *interaction context* attributes
- *interaction provenance* attributes



element (node) basic attributes
a_s^{name}: Bcl-xL
a_s^{type}: protein
a_s^{subtype}: *empty*
$a_s^{\text{HGNCsymbol}}$: BCL2L1
a_s^{database} : UniProt
a_s^{ID} : Q07817
$a_s^{\text{compartment}}$: mitochondria
$a_s^{\text{compartmentID}}$: GO:0005739
interaction (edge) basic attributes
a_e^{sign} : negative
$a_e^{\text{connectiontype}}$: direct
$a_e^{\text{mechanism}}$: binds
a_e^{site} : *empty*
element (node) basic attributes
a_t^{name}: Apaf-1
a_t^{type} : protein
a_t^{subtype}: *empty*
$a_t^{\text{HGNCsymbol}}$: APAF1
a_t^{database} : UniProt
a_t^{ID} : O14727
$a_t^{\text{compartment}}$: mitochondria
$a_t^{\text{compartmentID}}$: GO:0005739
Bcl-xL
Source node (Regulator)
Apaf-1
Target node (Regulated)
interaction context attributes
a^{cellline}: HEK 293T a^{celltype}: epithelial-like $a^{\text{tissuetype}}$: kidney a^{organism}: human
interaction provenance attributes
a^{score}: 0.99999821052 a^{source}: INDRA database
$a^{\text{statements}}$: 'Bcl-XL interacts with Apaf-1 and inhibits Apaf-1-dependent caspase-9 activation.'
a^{paperIDs}: PMID9539746

Figure 2. An example interaction in the BioRECIPE interaction list format (subscripts: s - source node, t - target node, e - edge).

Table 1 provides details for each attribute, including attribute name used in the BioRECIPE spreadsheet, a symbol used in detailed definitions in Section 2, a brief description of the attribute, allowed values, and a few examples. In the following, we provide formal definitions of the components of an interaction, and the attributes of these components. We also include additional details about attributes and examples of their values.

2.1 Element (node)

> ***Definition 1***. An element (node), $v = v(\mathbf{a}^v)$, is defined by its name, type, and unique identifier (ID), and these attributes are written as a vector $\mathbf{a}^v = (a^{\text{name}}, a^{\text{type}}, a^{\text{database}}, a^{\text{ID}})$. These are required element attributes in the BioRECIPE format.

The attribute a^{name} is an element name, usually following the standard nomenclature used by biologists and in the literature (e.g., acronym ERK1 is used instead of a longer name “extracellular signal-regulated kinase 1”). The attribute a^{type} represents element type, usually genes, RNAs, proteins, chemicals, or biological processes. Biological entity names often have multiple synonyms (e.g., ERK1 may also be referred to as MAPK3), and therefore, unique identifiers (IDs) are used, which are stored in attribute a^{ID}. These IDs can be obtained from standard databases such as UniProt, PubChem, or the Gene Ontology

Databases (GO). The unique ID attribute is often written as two attributes, the name of the database from which the ID is retrieved, a^{database}, and the ID, a^{ID}. In addition to the required attributes, we include an optional ID attribute, $a^{\text{HGNCsymbol}}$, the gene symbol from the HGNC database, as this is recognized by experts, in contrast to e.g., numbers used by UniProt, and therefore, it can assist in human-driven curation.

The node attribute vector $\mathbf{a}^v$ may also include other attributes that help describe the element. For example, attributes $a^{\text{compartment}}$ and $a^{\text{compartmentID}}$ hold information about the cellular compartment, where the element is found, and the compartment ID, respectively. We use the GO database to obtain these compartment IDs. A subtype attribute, a^{subtype}, may be used to indicate additional type of an element, such as $a^{\text{subtype}} = receptor$ for an element with $a^{\text{type}} = protein$. An element usually represents a biomolecular species, a chemical, or a biological process.

2.2 Interaction (edge)

Definition 2. A directed signed interaction (directed edge) $e = e(v_s, v_t, \mathbf{a}^e)$ is defined with its source element v_s, target element v_t, and vector of attributes $\mathbf{a}^e$. The interaction attribute vector always includes at least the sign a^{sign} and connection type $a^{\text{connectiontype}}$ attributes: $\mathbf{a}^e = (a^{\text{sign}}, a^{\text{connectiontype}})$. The direction of an interaction is implicitly defined with source and target nodes, and therefore, not explicitly listed among its attributes.

The a^{sign} attribute indicates the sign (also referred to as polarity) of the influences, and it can take two values, $a^{\text{sign}} = positive$ (e.g., activation) or $a^{\text{sign}} = negative$ (e.g., inhibition). Sometimes, only the information about indirect influences on pathways is known, and therefore, the attribute $a^{\text{connectiontype}}$ is used to indicate whether the interaction e is a direct physical interaction ($a^{\text{connectiontype}} = direct$) or an indirect influence from the source node to the target node ($a^{\text{connectiontype}} = indirect$). Since the interaction definition allows for indirect interactions, it is possible that source and target node are not in the same compartment, and this is the reason we assign the compartment attribute to nodes and not to the interaction.

The list of other attributes is not necessarily fixed; the components in it may vary, dependent on the goals of the analysis. A more specific information about the biological mechanism and the molecular site of an interaction can be included in the $a^{\text{mechanism}}$ and the a^{site} attributes, respectively. We note here that, occasionally, a^{sign} is not explicitly stated in statements about influences that describe mechanisms (e.g., A phosphorylates B). In such cases, it would be up to the user to either fill in this information from other sources or accept a default attribute assignment. For example, the default assignment could be $positive$ for phosphorylation, although this may not always be the case, and would require curation.

The interaction attribute vector can also include the a^{cellline}, a^{celltype}, $a^{\text{tissuetype}}$, a^{organism} attributes, which hold the ***context*** information about the cell line, cell type, tissue type, and organism where the interaction is observed, respectively.

Finally, ***provenance*** attributes can be used. The a^{score} attribute can include either a summary score for confidence, or a quantifier of available evidence for the interaction. The a^{source} attribute indicates whether source of evidence is literature, expert knowledge, databases, or data. The $a^{\text{statements}}$ attribute stores the statements, parts of sentences or sentences where the interaction is mentioned. The a^{paperIDs} attribute holds IDs of the papers (e.g., PMCID) with sentences mentioning the interaction. Whenever the information about the non-essential attributes is not available, these attributes are assigned an $empty$ value.

3 DIRECTED GRAPHS

To describe model representation with the BioRECIPE format, we start with the model topology.

Definition 3. *Model structure (static).* Models that have a directed graph, $G(V, E)$, as their underlying structure, include a set of nodes $V = \{v_1, v_2, \ldots, v_N\}$, where each node $v_i = v(\mathbf{a}_i^v)(i = 1, \ldots, N)$ is one model element, and a set of directed edges $E = \{e_1, e_2, \ldots, e_M\}$, where an edge $e_j = e\left(v_{s_j}, v_{t_j}, \mathbf{a}_j^e\right), \left(v_{s_j}, v_{t_j} \in V, j = 1, \dots, M\right)$ indicates a directed interaction between elements v_{s_j} and v_{t_j}, in which source node v_{s_j} influences target node v_{t_j}. Vectors $\mathbf{a}_i^v$ and $\mathbf{a}_j^e$ include node and edge attributes, respectively.

We note here that the BioRECIPE representation format can also be used for undirected graphs - in that case, the distinction between source and target nodes will not be relevant. Undirected graphs and directed graphs (i.e., static structure of executable models) can be represented in the BioRECIPE format as *lists of interactions* (edges), following the descriptions in Section 2.

*Table 2. The list of interaction attributes, using the formal notation for attributes, brief description of each attribute, examples values for each attribute. For more details on how to form attribute values, see [1]. *-required attribute*

			attribute	description	examples
element	basic		a^{name}*	element name, could be informal, typically used by experts	RAS; ERK1; p53
			a^{type} *	element type	protein; protein family; RNA; gene; chemical; biological process
			$a^{subtype}$	element subtype provides additional details for curation	receptor
			$a^{HGNCsymbol}$	the gene symbol from the HGNC database	BCL2L1; APAF1
			$a^{database}$ *	a database where the element ID is found	UniProt; Pfam; Ensembl; HGNC; PubChem; GO
			a^{ID} *	unique element ID from an open access database	Q07817; O14727
			$a^{compartment}$	cellular compartment name	cytoplasm; plasma membrane; nucleus
			$a^{compartmentID}$	cellular compartment unique identifier from the GO database	0005737; 0005886; 0005634
interaction	basic (influence)		$a^{direction}$	interaction direction	*this is an implicit attribute, determined as a direction from source to target node*
			a^{sign} *	interaction sign (also referred to as "polarity") indicates positive or negative influence	positive; negative
			$a^{connectiontype}$	interaction connection type can be: "direct" ("D"), indicating that the edge between the source and target nodes represents direct physical interaction between elements; "indirect" ("I"), indicating that it is expected or known that there is a path of several connected interactions between the source node and target node	D; I
			$a^{mechanism}$	interaction mechanism indicates the exact physical interaction (biological mechanism); value usually included when $a^{connectiontype}$="D";	binding; phosphorylation; ubiquitination
			a^{site}	molecular site where the interaction occurs	T308; T450; S473 (phosphorylation sites for Akt)
	metadata	context	$a^{cellline}$	cell line where the interaction is observed	GS6-22 (glioblastoma multiforme (GBM) cell lines)
			$a^{celltype}$	cell type where the interaction is observed	T cell; microphage; pancreatic cancer cell; GBM cell
			$a^{tissuetype}$	tissue type where the interaction is observed	pancreas; colon; brain
			$a^{organism}$	organism where the interaction is observed	human; mouse
		provenance	a^{score}	confidence in interaction, e.g., interval 0-1 (INDRA, STRING), present/absent (PCnet)	0.18
			a^{source}	knowledge source(s) where the interaction is found	Literature; expert; data
			$a^{statements}$	statements (sentences) where the interaction is found	"Bcl-XL interacts with Apaf-1 and inhibits Apaf-1-dependent caspase-9 activation"
			$a^{paperIDs}$	if literature, paper IDs where the interaction is found	PMID9539746

Table 1. An illustration of the interaction list in the BioRECIPE format, with all the attributes (current version supports one source node and one target node in each interaction).

	Elements (nodes, v)																Interaction (edge, e)											
	Regulator (source node, v_s)								Regulated (target node, v_t)								Influence				Metadata							
																					Context				Provenance			
	Regulator Name (a_s^{name})	Regulator Type (a_s^{type})	Regulator Subtype ($a_s^{subtype}$)	Regulator HGNC Symbol ($a_s^{HGNCsymbol}$)	Regulator Database ($a_s^{database}$)	Regulator ID (a_s^{ID})	Regulator Compartment ($a_s^{compartment}$)	Regulator Compartment ID ($a_s^{compartmentID}$)	Regulated Name (a_t^{name})	Regulated Type (a_t^{type})	Regulated Subtype ($a_t^{subtype}$)	Regulated HGNC Symbol ($a_t^{HGNCsymbol}$)	Regulated Database ($a_t^{database}$)	Regulated ID (a_t^{ID})	Regulated Compartment ($a_t^{compartment}$)	Regulated Compartment ID ($a_t^{compartmentID}$)	Sign (a_e^{sign})	Connection Type ($a_e^{connectiontype}$)	Mechanism ($a_e^{mechanism}$)	Site (a_e^{site})	Cell Line ($a_e^{cellline}$)	Cell Type ($a_e^{celltype}$)	Tissue Type ($a_e^{tissuetype}$)	Organism ($a_e^{organism}$)	Score (a_e^{score})	Source (a_e^{source})	Statements ($a_e^{statements}$)	Paper IDs ($a_e^{paperIDs}$)
Interaction 1																												
Interaction 2																												
Interaction 3																												
…																												
Interaction M																												

4 EXECUTABLE MODELS

The BioRECIPE format supports representation of the static graph structure of models, as well as attributes necessary to study the dynamics, often through simulations. We refer to the models that can be simulated as *executable models* and represent them in the BioRECIPE format using the *element-based approach*.

In the BioRECIPE format, each element in a model is assigned a row in the model table/spreadsheet. Different from event-based representation of interactions, in the element-based representation all interactions in which a given element participates as a regulated element (i.e., a target node of interaction edges) are combined within a row in the table dedicated to this particular element. In the model representation, the attributes are organized into three groups:

- **Element**:
  - *basic element* attributes
  - *context* attributes
- **Regulation**:
  - *basic regulation* attributes
  - *provenance* attributes
- **Simulation parameters**:
  - *(update) rule* attributes
  - *value* attributes
  - *timing* attributes

The *basic element* attributes, and *context* attributes are inherited from the interactions in which the element participates as a regulated element and are referred to as **Element** attributes in the model representation format. The *basic regulation* attributes and *provenance* attributes are formed by combining corresponding attributes from the individual interactions in which the element participates as a regulated element into an ordered list. These attributes are referred to as **Regulation** attributes, and their value lists are assembled following the same order of original interactions.

We note here that it may occasionally happen (especially if the model assembly process is automated) that the context varies across the interactions where the element is a regulated element. In such cases it is left to the modelers to utilize the context attributes in a manner that best suits their goals. For example, the modeler may either consolidate context attribute values from different interactions and decide to use the one that is most suitable for each attribute, or create a list of values for each attribute, following the same order as for the Regulation attributes.

The BioRECIPE format supports several different model representation *schemes* (Table 3), ranging from less detailed to more detailed, including either previously described "static" attributes only, or some or all of the "dynamic" attributes. These dynamics related attributes are necessary for simulation and the analysis of dynamic behavior; they are referred to as **Simulation parameters**, and are used to determine *element update rules*, and element value and timing parameters for the simulation of executable models. Depending on which attributes are used, we distinguish between the following model representation formats:

- STATIC – only required *basic* **Regulation** attributes
  - simple – only required *basic* **Element** attributes
  - detailed – all **Element** attributes and *provenance* **Regulation** attributes

- DYNAMIC– all *basic* **Regulation** attributes

  scenario-independent – all *rule* attributes and only Variable *value* attribute in **Simulation parameters**
  - simple – only required *basic* **Element** attributes
  - detailed – all **Element** attributes and *provenance* **Regulation** attributes

  scenario-dependent – all **Simulation parameters** attributes
  - simple – only required *basic* **Element** attributes
  - detailed – all **Element** attributes and *provenance* **Regulation** attributes

Table 3. BioRECIPE executable model display and attribute options, where x indicates attributes included in the representation scheme.

			Element												Regulation												Simulation parameters														
			Basic						Context						Basic								Provenance				Rule		Value							Timing					
Model representation: attributes and display options			Element Name* (a^{name})	Element Type* (a^{type})	Element Subtype (a^{subtype})	Element HGNC Symbol ($a^{\text{HGNCsymbol}}$)	Element Database* (a^{database})	Element IDs* (a^{ID})	Compartment ($a^{\text{compartment}}$)	Compartment ID ($a^{\text{compartmentID}}$)	Cell Line (a^{cellline})	Cell Type (a^{celltype})	Tissue Type ($a^{\text{tissuetype}}$)	Organism (a^{organism})	Positive Regulator List* ($a^{\text{posreglist}}$)	Positive Connection Type List ($a^{\text{posconnectiontypelist}}$)	Positive Mechanism List ($a^{\text{posmechanismlist}}$)	Positive Site List ($a^{\text{possitelist}}$)	Negative Regulator List* ($a^{\text{negreglist}}$)	Negative Connection Type List ($a^{\text{negconnectiontypelist}}$)	Negative Mechanism List ($a^{\text{negmechanismlist}}$)	Negative Site List ($a^{\text{negsitelist}}$)	Score List ($a^{\text{scorelist}}$)	Source List ($a^{\text{sourcelist}}$)	Statements List ($a^{\text{statementslist}}$)	Paper IDs List ($a^{\text{paperIDslist}}$)	Positive Regulation Rule* ($a^{\text{posregrule}}$)	Negative Regulation Rule* ($a^{\text{negregrule}}$)	Variable* (a^{variable})	Value Type ($a^{\text{valuetype}}$)	Levels* (a^{levels})	State list #* ($a^{\text{statelist}}$)	Const OFF (a^{constOFF})	Const ON (a^{constON})	Increment ($a^{\text{increment}}$)	Spontaneous ($a^{\text{spontaneous}}$)	Balancing ($a^{\text{balancing}}$)	Delay (a^{delay})	Update Group ($a^{\text{updategroup}}$)	Update Rate ($a^{\text{updaterate}}$)	Update Rank ($a^{\text{updaterank}}$)
Static		simple	x	x			x	x							x				x																						
		detailed	x	x	x	x	x	x	x	x	x	x	x	x	x				x				x	x	x	x															
Dynamic	scenario-independent	simple	x	x			x	x							x	x	x	x	x	x	x	x					x	x	x												
		detailed	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x												
	scenario-dependent	simple	x	x			x	x							x	x	x	x	x	x	x	x					x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
		detailed	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

* – required attribute
# – number
multiple 'State list' columns are allowed starting from # = 0
x – attribute included

4.1 Model definitions

In this section, we provide definitions that are relevant for representing executable models, as well as definitions and details for model specific attributes.

> ***Definition 4***. An *input node* is a node that is not a target node of any edge in the model, and an *output node* is a node that is not a source node of any edge in the model. In the graph, input and output nodes are "hanging" from the rest of the model.

> ***Definition 5***. We define a *path* in a model as $n > 1$ connected edges: $p\left(v_{s_p}, v_{t_p}, a^{sign_p}\right) = (e(v_{k_1} = v_{s_p}, v_{k_2}, \mathbf{a}^e_{k_1}),\ e(v_{k_2}, v_{k_3}, \mathbf{a}^e_{k_2}), \ldots, e(v_{k_n}, v_{k_{n+1}} = v_{t_p}, \mathbf{a}^e_{k_n}))$. The direction of the path is implicitly defined with the source node v_{s_p} and target node v_{t_p}. The regulation sign a^{sign_p} is considered positive when the number of negative signs in the set $\{a^{sign}_{k_1}, a^{sign}_{k_2}, \ldots, a^{sign}_{k_n}\}$ is even, and negative when this number is odd. Cycles and feedback loops may be defined in cases where the path source is also the path target, i.e., $p(v_{s_p}, v_{s_p}, a^{sign_p})$.

For example, in Figure 3, on the path from source node v_6 to target node v_{13}, the number of negative regulations is odd, due to only one negative regulation from node v_8 to v_9, and so the sign of this overall path is negative.

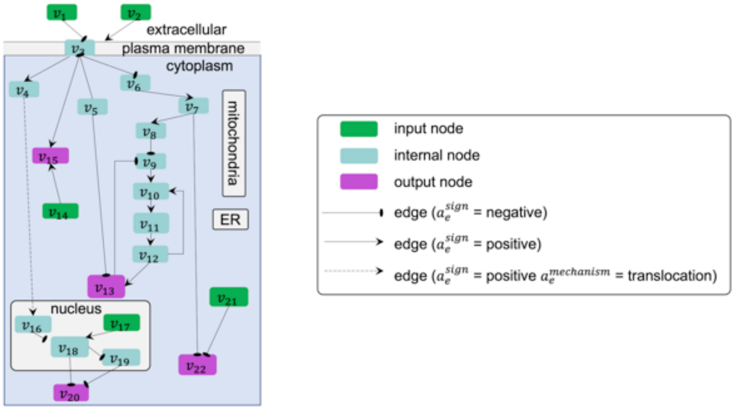


Figure 3. Toy example of a directed signaling network where the sign of the overall path is negative.

Definition 6. An *element-based executable model* is a triple $\mathcal{M}(G, \mathcal{X}, \mathcal{F})$, where $G(V, E)$ is a static network structure of the model (see ***Definition 3***), $\mathcal{X} = \{x_1, x_2, \dots, x_N\}$ is a set of N state variables corresponding to nodes in $V = \{v_1, v_2, \dots, v_N\}$, and $\mathcal{F} = \{f_1, f_2, \dots, f_N\}$ is a set of N regulatory (update) functions such that each element $v_i \in V$ has a corresponding function $f_i \in \mathcal{F}$.

In element-based modeling, the function and element types are usually decided based on the knowledge or the information available about the modeled system and its components. In other words, the element-based modeling approach can represent indirect influences between elements, and it can model systems where the knowledge about element interaction mechanisms is incomplete. An example of element-based model is a discrete model, where each element state variable is assigned a discrete set of values. Boolean models [2-4] are a subset of discrete models, where elements can have only two values, 0 (also referred to as OFF or False) and 1 (also referred to as ON or True).

Definition 7. When element update functions $f_i \in \mathcal{F}$ have different mathematical form across elements $v_i \in V$ within the same model, for example, logical, discrete, or continuous functions, we refer to these models as *hybrid element-based executable models*.

In hybrid element-based models, individual elements within the same model can have very different update functions. The set or interval of possible values assigned to each model element can also vary. Using such hybrid collection of element update rules within a single model enables model simulation and studies of cell dynamics, state transitions, and feedback loops, while utilizing the available information, in the absence of complete knowledge of interaction mechanisms. These hybrid element-based models enable integration of both prior knowledge and data, and the analysis of hybrid networks (systems involving protein-protein interactions, gene regulations, and/or metabolic pathways).

Definition 8. A source node v_j of an edge in graph $G(V, E)$ that has v_i as a target node is called a *regulator* of v_i. In other words, for each element v_i, any element v_j that influences the state of v_i such that the function f_i is sensitive to the value of x_j is called a regulator of v_i.

Definition 9. For each element v_i, an *influence set*, denoted as $V_i^{influence} \subset V$, consists of all regulators of v_i. The state variables that correspond to the elements in $V_i^{influence}$ form set $\mathcal{X}_i^{influence}$.

Definition 10. Any element $v_j \in V_i^{influence}$, for which the edge $e(v_j, v_i, \mathbf{a}^e)$ has a positive sign, $a_e^{\text{sign}} = positive$, belongs to the *positive regulator list* for element v_i, denoted as $v_j \in V_i^{influence,+} \subset V_i^{influence}$, represented with attribute $a^{\text{posreglist}}$. Any element $v_j \in V_i^{influence}$, for which the edge $e(v_j, v_i, \mathbf{a}^e)$ has a negative sign, $a_e^{\text{sign}} = negative$, belongs to the *negative regulator list* for element v_i, denoted as $v_j \in V_i^{influence,-} \subset V_i^{influence}$, represented with attribute $a^{\text{negreglist}}$.

Definition 11. For each element $v_i \in V$, its *state variable* $x_i \in \mathcal{X}$ can take any value from a set or an interval of values X_i. The state variable x_i is represented with attribute a^{variable}, and is assigned either the amount or activity *value type* of v_i, represented with attribute $a^{\text{valuetype}}$.

Definition 12. The state variables x_j that correspond to elements in $V_i^{influence,+}$ form set $\mathcal{X}_i^{influence,+} \subset \mathcal{X}_i^{influence}$, and are used for creating a *positive regulation rule* for v_i, represented with attribute $a^{\text{posregrule}}$. The state variables x_j that correspond to elements in $V_i^{influence,-}$ form set $\mathcal{X}_i^{influence,-} \subset \mathcal{X}_i^{influence}$, and are used for creating a *negative regulation rule* for v_i, represented with attribute $a^{\text{negregrule}}$.

Definition 13. When X_i is a set of discrete values, $|X_i|$ is referred to as the *number of levels* of v_i, represented with attribute a^{levels}.

Definition 14. An array of k state values $X_i^0, X_i^{t_1}, X_i^{t_2}, \dots, X_i^{t_{k-1}}$ that are assigned to v_i at $t_0, t_1, t_2, \dots, t_{k-1}$ time steps during simulation, where t_0 is the initial time step, and $t_0 < t_1 < t_2 < \cdots < t_{k-1}$, is called **state list** and is represented with attribute $a^{\text{statelist}}$. Multiple state lists are allowed within the BioRECIPE table, in consecutive columns, named "State list #" where #=0,1,2,...

Definition 15. When the state variable x_i has a constant 0 value throughout the entire simulation, this is referred to as a *constant OFF state*, and represented with attribute a^{constOFF}.

Definition 16. When the state variable x_i has a constant nonzero value (e.g., the highest value from X_i) throughout the entire simulation, this is referred to as a *constant ON state*, and represented with attribute a^{constON}.

Definition 17. The *next state* of element v_i, denoted as x_i^*, is computed using the element update rule f_i and current states of all elements in its influence set, that is, current values of all variables in $\mathcal{X}_i^{influence}$: $x_i^* = f_i(\mathcal{X}_i^{influence})$.

5 TRANSLATORS AND COMPATIBILITY

The BioRECIPE documentation is also available as ReadtheDocs pages [1], where further details are provided on how to create interaction lists and executable models in this format, as well as links to examples and available translators to and from other representation formats and various tool inputs and outputs.

The BioRECIPE format is compatible with several common representation formats (e.g., SBML) used by the systems and computational biology community that are created mainly for machine use.

The BioRECIPE format is also compatible with the output of several machine readers (TRIPS [5] and REACH [6]), as well as with the INDRA output [7]. Through the model translation, BioRECIPE is compatible with a number of tools and databases that are used to analyze or store models (e.g., CellCollective [8], NDEx [9], BioModels [10], many of the simulators available in the BioSimulators [11] and CoLoMoTo [12] repositories).

Finally, BioRECIPE is also compatible with a suite of tools within the Dynamic System Explanation (DySE) framework [13]. For model simulation and analysis, the DiSH simulator [14] and PIANO [15] accept files in BioRECIPE format. Several tools that curate, verify, and extend models, such as FLUTE [16], VIOLIN [17], CLARINET [18], ACCORDION [19], FIDDLE [20], and MINUET [21], all use BioRECIPE format and output their results in a BioRECIPE-compatible format.

6 CONCLUSION

The BioRECIPE representation format is a valuable tool for the field of systems and computational biology by promoting model curation using both human and machine curators. The complexity of cellular signaling pathways and their components necessitate modeling methods that can account for a multitude of details. To compensate, the BioRECIPE representation format allows for many element and interaction attributes to be included, as well as attributes for simulation. The BioRECIPE format is compatible with multiple tools for model curation, information extraction, and existing formats for systems biology. This interoperability ensures that researchers can seamlessly integrate BioRECIPE into their existing workflows. Future directions include additional functionality to translate between existing formats (such as SBOL [22]), or integration of the BioRECIPE representation format as input to model curation and storage platforms (such as CellCollective [8] or NDEx [9]). These platforms play a crucial role in managing and disseminating computational models, and closer integration with BioRECIPE could streamline the process of model sharing and collaboration within the scientific community. In conclusion, the BioRECIPE representation format offers a leap forward in the field of systems biology. Its ability to support comprehensive model curation makes it a valuable resource for researchers and practitioners in the biological sciences.

7 Acknowledgments

Many previous and current members of MeLoDy Lab have contributed to the brainstorming and development of the BioRECIPE format. The format has gone through several versions and revisions. The first version was developed by Dr. Khaled Sayed, Dr. Cheryl Telmer, Adam Butchy, and Dr. Natasa Miskov-Zivanov. It was inspired by the evaluation format used as part of DARPA's Big Mechanism program [23-25], and suggested by the MITRE team led by Dr. Lynette Hirschman. The original BioRECIPE format was presented at the Machine Learning, Optimization, and Big Data (MOD) conference [26], where it received the Best Paper Award. We would also like to thank Dr. Kara Bocan, Dr. Yasmine Ahmed, Dr. Brent Cochran, and Stefan Andjelkovic for their constructive feedback and suggestions while developing the current version of the format.

8 References

[1] The MeLoDy Lab. "BioRECIPE documentation." https://melody-biorecipe.readthedocs.io/en/latest/ (accessed 2023).

[2] I. Albert, J. Thakar, S. Li, R. Zhang, and R. Albert, "Boolean network simulations for life scientists," *Source Code for Biology and Medicine,* journal article vol. 3, no. 1, p. 16, November 14 2008, doi: 10.1186/1751-0473-3-16.

[3] J. D. Schwab, S. D. Kühlwein, N. Ikonomi, M. Kühl, and H. A. Kestler, "Concepts in Boolean network modeling: What do they all mean?," *Computational and Structural Biotechnology Journal,* vol. 18, pp. 571-582, 2020/01/01/ 2020, doi: https://doi.org/10.1016/j.csbj.2020.03.001.

[4] R. S. Wang, A. Saadatpour, and R. Albert, "Boolean modeling in systems biology: an overview of methodology and applications," (in eng), *Phys Biol,* vol. 9, no. 5, p. 055001, Oct 2012, doi: 10.1088/1478-3975/9/5/055001.

[5] N. UzZaman and J. Allen, "TRIPS and TRIOS system for TempEval-2: Extracting temporal information from text," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, 2010, pp. 276-283.

[6] M. A. Valenzuela-Escarcega, G. Hahn-Powell, M. Surdeanu, and T. Hicks, "A Domain-independent Rule-based Framework for Event Extraction," in *ACL*, 2015.

[7] B. M. Gyori, J. A. Bachman, K. Subramanian, J. L. Muhlich, L. Galescu, and P. K. Sorger, "From word models to executable models of signaling networks using automated assembly," *Molecular systems biology,* vol. 13, no. 11, pp. 954-954, 2017, doi: 10.15252/msb.20177651.

[8] T. Helikar *et al.*, "The Cell Collective: Toward an open and collaborative approach to systems biology," *BMC Systems Biology,* vol. 6, no. 1, p. 96, 2012/08/07 2012, doi: 10.1186/1752-0509-6-96.

[9] D. Pratt *et al.*, "NDEx, the Network Data Exchange," (in eng), *Cell Syst,* vol. 1, no. 4, pp. 302-305, Oct 28 2015, doi: 10.1016/j.cels.2015.10.001.

[10] V. Chelliah *et al.*, "BioModels: ten-year anniversary," *Nucleic Acids Research,* vol. 43, no. D1, pp. D542-D548, 2014, doi: 10.1093/nar/gku1181.

[11] B. Shaikh *et al.*, "BioSimulators: a central registry of simulation engines and services for recommending specific tools," *Nucleic Acids Research,* vol. 50, no. W1, pp. W108-W114, 2022, doi: 10.1093/nar/gkac331.

[12] A. Naldi *et al.*, "Cooperative development of logical modelling standards and tools with CoLoMoTo," (in eng), *Bioinformatics,* vol. 31, no. 7, pp. 1154-9, Apr 1 2015, doi: 10.1093/bioinformatics/btv013.

[13] C. A. Telmer *et al.*, "Dynamic system explanation: DySE, a framework that evolves to reason about complex systems - lessons learned," presented at the Proceedings of the Conference on Artificial Intelligence for Data Discovery and Reuse, Pittsburgh, Pennsylvania, 2019.

[14] K. Sayed, Y. Kuo, A. Kulkarni, and N. Miskov-Zivanov, "DiSH simulator: Capturing dynamics of cellular signaling with heterogeneous knowledge," in *2017 Winter Simulation Conference (WSC)*, 3-6 Dec. 2017 2017, pp. 896-907, doi: 10.1109/WSC.2017.8247841.

[15] G. Zhou, K.-W. Liang, and N. Miskov-Zivanov, *Intervention Pathway Discovery via Context-Dependent Dynamic Sensitivity Analysis*. 2019.

[16] E. a. T. C. A. a. M.-Z. N. Holtzapple, "FLUTE: Fast and reliable knowledge retrieval from biomedical literature," *Database,* vol. 2020 , ISSN = 1758-0463 , DOI = 10.1093/database/baaa056, 2020. [Online]. Available: https://doi.org/10.1093/database/baaa056.

[17] C. Hansen, J. Kisslinger, N. Krishna, E. Holtzapple, Y. Ahmed, and N. Miskov-Zivanov, "Classifying Literature Extracted Events for Automated Model Extension," 2021, doi: 10.1101/2021.09.30.462421.

[18] Y. Ahmed, C. Telmer, and N. Miskov-Zivanov, "CLARINET: Efficient learning of dynamic network models from literature," *Bioinformatics Advances,* 2021, doi: 10.1093/bioadv/vbab006.

[19] Y. Ahmed, C. Telmer, and N. Miskov-Zivanov, "ACCORDION: Clustering and Selecting Relevant Data for Guided Network Extension and Query Answering," *arXiv preprint,* p. arXiv:2002.05748, 2020.

[20] A. A. Butchy, C. A. Telmer, and N. Miskov-Zivanov, "Automating Knowledge-Driven Model Recommendation: Methodology, Evaluation, and Key Challenges," *arXiv preprint arXiv:2301.11397,* 2023.

[21] E. Holtzapple, B. Cochran, and N. Miskov-Zivanov, "Automated verification, assembly, and extension of GBM stem cell network model with knowledge from literature and data," *bioRxiv,* p. 2021.07.04.451062, 2021, doi: 10.1101/2021.07.04.451062.

[22] J. A. McLaughlin *et al.*, "The Synthetic Biology Open Language (SBOL) Version 3: Simplified Data Exchange for Bioengineering," (in eng), *Front Bioeng Biotechnol,* vol. 8, p. 1009, 2020, doi: 10.3389/fbioe.2020.01009.

[23] M. Peterson, T. Korves, C. Garay, R. Kozierok, and L. Hirschman, "Final Report on MITRE Evaluations for the DARPA Big Mechanism Program," *arXiv preprint arXiv:2211.03943,* 2022.

[24] P. R. Cohen, "DARPA's Big Mechanism program," (in eng), *Phys Biol,* vol. 12, no. 4, p. 045008, Jul 16 2015, doi: 10.1088/1478-3975/12/4/045008.

[25] J. Elliott. "Big Mechanism (Archived)." Defense Advanced Research Projects Agency. https://www.darpa.mil/program/big-mechanism (accessed 2023).

[26] K. Sayed, C. A. Telmer, A. A. Butchy, and N. Miskov-Zivanov, "Recipes for Translating Big Data Machine Reading to Executable Cellular Signaling Models," Cham, 2018: Springer International Publishing, in Machine Learning, Optimization, and Big Data, pp. 1-15.