Optimal Design and Operation of WHO-EPI Vaccine Distribution Chains

by

Yuwen Yang

Bachelor of Engineering, Beihang University, 2013

Master of Science, University of Pittsburgh, 2016

Submitted to the Graduate Faculty of the

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2020

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Yuwen Yang

It was defended on

July 15, 2020

and approved by

Hoda Bidkhori, Ph.D., Assistant Professor, Department of Industrial Engineering

Bo Zeng, Ph.D., Associate Professor, Department of Industrial Engineering

Jennifer Shang, Ph.D., Professor, Joseph M. Katz Graduate School of Business

Dissertation Director: Jayant Rajgopal, Ph.D., Professor, Department of Industrial Engineering

Optimal Design and Operation of WHO-EPI Vaccine Distribution Chains

Yuwen Yang, PhD

University of Pittsburgh, 2020

Vaccination has been proven to be the most effective method to prevent infectious diseases and in 1974 the World Health Organization (WHO) established the Expanded Programme on Immunization (EPI) to provide universal access to all important vaccines for all children. However, there are still roughly 20 million infants worldwide who lack access to routine immunization services and remain at risk, and millions of additional deaths could be avoided if global vaccination coverage could improve. The broad goal of this research is to optimize the design and operation of the WHO-EPI vaccine distribution chain in underserved low- and middle-income countries. We first formulate a network design problem for a general WHO-EPI vaccine distribution network as a mixed integer program (MIP). We then present three algorithms for typical problems that are too large to be solved using commercial MIP software. We test the algorithms using data derived from four different countries in sub-Saharan Africa and show that with our final algorithm, high-quality solutions are obtained for even the largest problems within a few minutes. Next, we discuss the problem of outreach to remote population centers when direct clinic service is unavailable. We formulate the problem as an MIP that is a combination of a set covering problem and a vehicle routing problem and then incorporate uncertainty to study the robustness of the worst-case solutions and the value of information. Finally, we study a variation of the outreach problem that combines set covering and the traveling salesmen problem and provides an MIP formulation. Motivated by applications where the optimal policy needs to be updated repetitively and where solving this via MIP can be computationally expensive, we propose a machine learning approach

to effectively learn from historical optimal solutions. We also present a case study and show that while the proposed mechanism generates high quality solution repeatedly for problems that resemble instances in the training set, it does not generalize as well on a different set of problems. These mixed results indicate that there are promising research opportunities to use machine learning to achieve tractability and scalability.

Table of Contents

List of Tables

List of Figures

Preface

This dissertation was written and completed during the worldwide pandemic of COVID-19. As of July 2020, the number of people inflected has reached over 11 million in more than 200 countries. While doctors and nurses in the frontlines of the fight have contributed enormous efforts, even including their lives to fight against COVID-19, we will not be able to achieve this mission without the development of a vaccine against it. When researchers and scientists around the world are working around the clock to develop the vaccine, delivering vaccine to end-users remains a major challenge and is an extremely completed task. A few years ago, motivated by the fact that the current hierarchical legacy medical network is not always cost-efficient and reliable, we started this work with fundamentally redesigning the distribution networks to be operated without the need for sophisticated logistics personnel and at minimum cost. In addition, for many geographically dispersed and nomadic populations without direct access to clinics and hospitals, outreach activities are a way to supplement the static immunization facilities. The scope of the work was therefore extended to study outreach operations and mobile clinics. Subsequently, with the breakthroughs in deep learning on a variety of tasks and with many researchers exploring it to solve combinatorial optimization problems, we targeted a similar exploratory approach to combining deep learning with optimization. I hope that these efforts could be a tiny piece of the mission that is related to everyone, and every community on this planet.

With this being said, completing this dissertation was never easy for me. I owe a lot to my parents, Wei Zhou and Xuecheng Yang who have been supportive of every decision I have made in my life. I want to particularly thank my fiancée, Zexu, who has always been there for me during

my times of need, being supportive and caring. I could not finish writing this without her love and our lovely companions, Daidai and Qiuqiu. This dissertation is dedicated to my family.

I would like to extend my deepest gratitude to my advisor, Dr. Jayant Rajgopal, who was the instructor of my first Operations Research course and has encouraged me to challenge myself with the doctoral program. He has given me myriad help, constructive advice, explicit domain knowledge, and has been always very patient and supportive of new research approaches that I proposed. I am very grateful to my committee, Dr. Hoda Bidkhori, Dr. Bo Zeng, and Dr. Jennifer Shang for their comprehensive collaboration, enlightening discussion, patience, and constructive feedback. It has been my pleasure to work with you.

I would also like to acknowledge the National Science Foundation for providing financial support through Award No. CMII-1536430 and thank the World Health Organization for making the information and data available. Thanks to the Department of Industrial Engineering at Pitt for providing the solid training, financial aid, and supportive and inclusive environment. Special thanks to Amazon's Modeling and Optimization team and Microsoft Cloud AI for their training and mentorship, and for allowing me to work on this during internship and full-time employment.

Finally, this dissertation is in memory of the days and nights together with my office mates, Anna Svirsko, Hamdy Salman, Jung Lim, Shan Gong, Yanfei Chen, Zhaohui Geng, and Xin Xu. Many thanks to my friends and fellow graduate students at Pitt, especially, Chaosheng Dong, Colin Gillen, Erfan Mehmanchi, David Abdul-Malak, Jing Yang, Ju Chen, Kai He, Ke Ren, Kezhuo Zhou, Liang Xu, Liuyan Hu, Moataz Abdulhafez, Ruichen Sun, Shadi Sanoubar, Tarik Bilgic, Wei Wang, Yiwei Wu, Yayun Jin, and Xueyu Shi, for making a great community to belong to, helping to prepare for the qualifying exam and defense, and encouragement during the process.

1.0 Introduction

The broad goal of this dissertation is to optimize the design and operation of WHO-EPI vaccine distribution networks. The dissertation has three specific problems that it addresses. First, we develop a general model for designing the network that is applicable across all countries and propose mixed integer programming based algorithms for optimizing the model. Second, we address the issue of optimal outreach policies when resources are limited and direct clinic service is unavailable. Third, we discuss a variation of the outreach problem as a combination of the Set Covering Problem and the Traveling Salesmen Problem, and we propose and study an exploratory learning-based mechanism for solving this combined problem.

1.1 Motivation and Research Objective

Vaccines are biological preparations that can provide active acquired immunity against infections. Typically, a body's immune system is active against a certain disease once the immune system recognizes the disease-causing microorganism as a threat and tries to destroy it. To mimic a specific microorganism, a vaccine usually contains an agent that is made from the microorganism’s toxins, surface protein, or its weakened or killed forms. The agent can thus stimulate the immune system to recognize the agent and then recognize this microorganism once it encounters it at a later time. The body is then equipped with this active acquired immunity to destroy the disease-causing microorganism and prevent the infection. Vaccines can be classified as prophylactic vaccines that can prevent or ameliorate the effect of a natural pathogen, or as therapeutic vaccines such as the potential future cancer vaccines (Bol et al., 2016; Frazer, 2014). According to the World Health Organization (WHO), the current suite of licensed vaccines (e.g., Pentavalent, Oral Polio, Yellow Fever, Tetanus Toxoid) can provide immunity to over twenty-five preventable infections (World Health Organization, 2017).

The term “vaccination” refers to the administration of vaccines. The words *vaccine* and *vaccination* are derived from *Variolae vaccinae* (smallpox of the cow). This term was first devised by Edward Jenner to denote cowpox in his famous *Inquiry* that was published 200 years ago in 1798 (Baxby, 1999). In *Inquiry into the Variolae vaccinae known as the Cow Pox*, he introduced the protective effect of cowpox to prevent against smallpox. In honor of Jenner, the term has since been extended to refer to all protective inoculation.

Vaccination of children includes routine immunization and supplemental immunization. Routine immunization is prescribed according to the national immunization schedule and WHO guidelines. It is administered based on the individual vaccination history of each child. Once a child has received all routine immunizations, he/she is counted as a *fully immunized* individual, and this activity must be recorded on his/her immunization cards and register. General, routine immunization targets people from infants after birth to children five years of age. On the other hand, supplemental immunization works as a supplement to routine immunization by providing additional opportunities to develop the immune system. Supplemental immunization is generally implemented in the form of targeted campaigns when there are outbreaks of diseases. The schedule and policies of supplemental immunization are often determined based on estimates of future disease occurrence (World Health Organization, 2020b).

Vaccination has been widely studied and proven to be the most effective method to prevent illness, disability and death from infections such as HPV (Chang et al., 2009; Fu et al., 2014), chicken pox (Liesegang, 2009), influenza (Fiore, Bridges, & Cox, 2009; Wolff, 2020), cervical cancer, diphtheria, hepatitis B, measles, mumps, pertussis (whooping cough), pneumonia, polio, rabies (Fooks, Banyard, & Ertl, 2019; O'Brien & Nolan, 2019), rotavirus diarrhea, rubella and tetanus. More recently, unprecedented effort has been exerted toward COVID-19, Ebola, Measles, and Polio vaccines (World Health Organization, 2020a). In 2018, the world immunization coverage rate for 3 doses of diphtheria-tetanus-pertussis (DTP3) vaccine was 86% (116.5 million infants), and 129 countries had reached 90% DTP3 vaccine coverage. This has averted an estimated 2 to 3 million deaths in every single year.

However, millions of additional deaths could be avoided if global vaccination coverage could improve, and it is estimated that there are still 19.4 million infants worldwide who lack access to routine immunization services and remain at risk for vaccine-preventable diseases, even in the 21st century (World Health Organization, 2019c). This problem is especially pronounced in low and middle-income countries (LMICs) (Gavi, 2019). Among all these children worldwide, roughly 60% are in only 10 countries. The reasons responsible for the low vaccination rates in these countries are varied and include limited resources, a deficiency of scientific health systems management, competing health priorities, and inadequate monitoring and supervision (de Oliveira, Martinez, & Rocha, 2014; Shen, Fields, & McQuestion, 2014; Yadav, et al., 2014). Even when vaccination services are available, there are cases where patients sometimes refuse vaccines, or delay getting vaccinated based upon complacency, convenience and confidence (Hotez, Nuzhath, & Colwell, 2020; Jarrett et al., 2015; MacDonald et al., 2015).

To address this problem, the WHO established the Expanded Programme on Immunization (EPI) in 1974 with the goal of providing universal access to all important vaccines for all children, with a special focus on underserved developing countries (Bland & Clements, 1998). The program was then expanded with the formation of the Global Alliance for Vaccines and Immunization (Gavi) in 2000 to accelerate access to new vaccines in the poorest countries. EPI and Gavi together have successful contributed to saving millions of lives worldwide by reducing deaths and even eliminating some diseases like measles in high-risk countries (Gavi, 2020; World Health Organization, 2013).

With the help of international organizations and new technological developments, many vaccines can now be produced at low-cost and in mass quantities. However, shipping, storing and delivering vaccines in a cost-efficient fashion while ensuring that vaccines are reliably available to end-users remains a major challenge. The primary characteristic that is responsible for the relatively high cost of vaccine distribution is the fact that most vaccines require narrowly defined temperatures of between 2° and 8° C during storage and transportation. This vaccine supply chain is also referred to as a cold chain. Many of the challenges to get vaccines delivered to children arise from the poor operation and management of the vaccine supply chain; in particular, poor infrastructure, inefficient assignment and use of vaccine storage and transportation devices, a rigid distribution structure and constraints, and mandated replenishment policy are all factors (Acosta, Hendrickx, & McKune, 2019; Yadav et al., 2014; Zaffran, 1996).

In particular, in many LMICs vaccines are usually distributed via a hierarchical legacy medical network, with locations and shipping routes of this network often determined by political boundaries and history. The overarching goal is to ensure that every child has access to vaccines, and along with this, in most LMICs the objective is to design a system that can be operated without the need for sophisticated logistics personnel and at minimum cost.

This fact motivates our study to propose an improved vaccine distribution chain. As an alternative to the current structure, the proposed vaccine distribution chains could be separated from the current legacy health network, while using some appropriate subset of these facilities and with vaccine flows along routes that differ from the current ones. However, the operation of the chain must not deviate from established WHO guidelines and needs to be simple because of the relative lack of sophisticated vaccine management abilities in LMICs.

In particular, this dissertation first focuses on redesigning a complete vaccine network in Chapter 2. The main consideration in redesigning the network includes decisions on the choice of the best set of intermediate hubs from the set of current distribution center locations, obtaining optimal replenish frequencies for each hub, deciding on hub-to-hub connections and the clinic allocations to each hub, determining the actual vaccine flow along all connections, and lastly, selecting the types of storage and transportation devices to use at each location and along each flow path. The re-designed network is not required to follow the four- (or sometimes, three- or five-) tiered hierarchical structure that is currently the norm, nor are they required to follow the replenishment policy currently associated with a particular tier. We present a mixed-integer optimization model and develop methods to get solutions for large scale problems while conducting numerical tests using real data from different LMICs to study the performance of the algorithms. Because of the computational expense and tractability issues of the optimization model, Chapter 3 further present a new algorithm for typical problems that are too large to be solved using commercial MIP software. We test the algorithm using data derived from four different LMICs in sub-Saharan Africa and show that the algorithm is able to obtain high-quality solutions for even the largest problems, within a few minutes.

In addition to the suboptimal structure and operations of a vaccine distribution network, another situation that could cause low vaccination rates is when resources are limited and there are population centers without access to direct clinic services. In this case, an approach known as *outreach* is typically utilized where a team of clinicians and support personnel is sent from an existing clinic to visit one or more of these locations to vaccinate residents there and at other such locations their immediate surrounding area. In Chapter 4 we focus on the problem of outreach. We model the problem of optimally designing outreach efforts as a mixed integer program that is a combination of a set covering problem and a vehicle routing problem. In addition, because elements relevant to outreach (such as populations and road conditions) are often unstable and unpredictable, we incorporate uncertainty to study the robustness of the worst-case solutions and the related issue of the value of information.

Finally in Chapter 5, we looked at the outreach problem defined in Chapter 4 from a different viewpoint and reformulate it as a combination of a Set Covering Problem and a Traveling Salesmen Problem. The Traveling Salesman Problem itself is one of the most intensively studied combinatorial optimization problems due both to its range of real-world applications and its computational complexity. When combined with the Set Covering Problem, it raises even more issues related to tractability and scalability. We provided a mixed integer programming formulation to solve the problem. In many applications where the optimal policy needs to be updated on a regular basis, repetitively solving this via MIP can be computationally expensive. We therefore explore a machine learning approach to effectively deal with this problem by providing an opportunity to learn from historical optimal solutions that are derived from the MIP formulation. We also present a case study using the World Health Organization's vaccine distribution chain, and provided numerical results with data derived from four LMICs in sub-Saharan Africa.

1.1.1 Related Prior Work

There has been some prior research in exploring and improving vaccine distribution via mathematical programming. As early as 1978, Longini et al. developed a deterministic model to describe a single wave of influenza A. The model was then used to generate optimal vaccine distribution patterns among various age groups when vaccine is limited (Longini, Ackerman, & Elveback, 1978). In 1999, Jacobson et al. optimized the procurement of children's immunization vaccines via an integer programming model that minimized the total cost of clinic visits, vaccine purchase and injection costs to fully immunize a child (Jacobson el al., 1999). In 2002, Kaplan et al. used an optimization model to estimate the impact of a smallpox bioterrorist attack in a large U.S. urban area and compared various vaccination policies to alleviate the impact (Kaplan, Craft, & Wein, 2002). Kaplan and Merson proposed a policy that considers both efficiency and equity in allocating federal resources to prevent against HIV (Kaplan & Merson, 2002). Ferguson et al. reviewed the use of mathematical models in smallpox planning within the context of broader epidemiology (Ferguson et al., 2003). Hill and Longini developed a method to generate policies which minimize the quantity of vaccine allocated in order to prevent an epidemic, with heterogeneous subgroups (Hill & Longini, 2003). Earnshaw and Hick utilized a linear programming framework to generate an optimal policy to allocate HIV prevention resources by maximizing the population averted from HIV transmission infections (Earnshaw et al., 2007).

Besides these mathematical modeling approaches, recent research on related topics has used other approaches such as lean, simulation and Markov decision process (MDP) models. Rajgopal et al. developed a spreadsheet model to evaluate the potential impact of several ordering policies at the clinic level in a LMIC. The model identified the optimal policy and correct number of routine vaccines that a health clinic should order (Rajgopal et al., 2011). Norman et al. designed a spreadsheet model to evaluate the impact of different packing schemes and utilized this model to compare the current packing scheme to a proposed modular packing scheme (Norman et al., 2015). To improve and simplify the vaccine inventory system, Lim et al. proposed a set of alternative ordering policies based on lean related concepts that have been used for long in the manufacturing industry (Lim, Norman, & Rajgopal, 2017). The analyses showed that the proposed polices only require a very modest increase in fixed storage and transport requirements. Several comprehensive discrete event simulation models have been presented to describe vaccine networks with consideration of vaccines, storage devices, and transport mechanisms. These experiments have examined various issues such as the impact of switching from 10-dose measles vial size to smaller doses (Assi et al., 2011), changing the current four-tier (central, regional, district, and integrated) to a modified three-tier structure (Assi et al., 2013), and reducing open-vial waste (Heaton et al., 2017; Lee, Assi, Rookkapan, Wateska, et al., 2011; Lee, Assi, Rookkapan, Connor, et al., 2011; Lee et al., 2010). More recently, Mofrad et al. formulated a Markov decision process model that determines the optimal time to conserve vials according to the current vial inventory, time of day, and the remaining clinic-days until the next replenishment. The authors minimize open-vial waste while administering sufficient vaccinations and then present a practical heuristic (Mofrad et al., 2014). Building on this result, Mofrad et al. evaluated several operating strategies to maximize coverage while controlling open vial waste and comparaed optimal and heuristic policies in the presence of random vial yield (Mofrad et al., 2016). A easy-to-implement decision support tool was also generated and made avialible online.

Mathematical models that address the EPI vaccine distribution chain are relatively few in number. Lee et al. presented a mathematical model in 2012 to optimize the existing vaccine distribution network for Niger when considering new vaccines (Lee et al., 2012). Chen et al.

developed and adapted a planning model for typical vaccine distribution with an expanded discussion of several issues (Chen et al., 2014). Lim et al. presented four quantitative models to determine optimal outreach locations in order to maximize coverage rate and contrasted the models using real world data derived from the state of Bihar in India (Lim et al., 2016). Lim also proposed an initial mixed integer programming model to address the problem of designing an optimal EPI vaccine distribution network (Lim, Norman, & Rajgopal, 2019). Vaccine manufacturers and vaccine supply chain policy makers can also consult these comprehensive studies and review (Chen, 2012; De Boeck, Decouttere, & Vandaele, 2019; Lim, 2016; Mofrad, 2016) to examine optimal vaccine network structures and operational decisions such as order time, order size and production vial size. We will discuss some of these studies in detail and compare them with our approaches in the literature review sections of each chapter.

1.2 Contribution

This dissertation aims to provide additional mathematical models to analyze the issues raised above and develops algorithms to solve these problems. Our major contributions include:

- A general mathematical programming formulation for the design of a WHO-EPI vaccine distribution network in any LMIC, with the goal of minimizing costs while providing the opportunity for universal coverage.
- A column generation based algorithm to solve the MIP model.
- An iterative heuristic that cycles between solving restrictions of the original problem and the associated numerical experiments to show that it can find very good solutions in reasonable time for larger problems that are not directly solvable.

- A novel algorithm for extremely large problems, that solves a sequence of increasingly larger MIP problems and uses insights into the problem structure and principles from cluster analysis to limit the size of each MIP in the sequence, along with related numerical tests using data derived from several different countries in sub-Saharan Africa to illustrate that the algorithm works very well with solution times that scale up in a roughly linear fashion.
- A mathematical programming formulation for the problem of optimally designing outreach efforts as a mixed integer program that is a combination of a set covering problem and a vehicle routing problem.
- Incorporation of uncertainty to study the robustness of the worst-case solutions and the related issue of the value of information.
- The study of a combined Set Covering and Traveling Salesman Problem and a mixed integer programming formulation to solve the problem.
- Design of a machine learning based mechanism to effectively deal with the combined Set Covering and Traveling Salesman Problem by learning from historical optimal solutions that are derived from the MIP formulation. This methodology aims at providing one of the early approaches for an end-to-end learning algorithm for a particular combinatorial optimization problem via deep learning.
- A case study using the World Health Organization's vaccine distribution chain and numerical results with data derived from four countries in sub-Saharan Africa for the machine learning based mechanism to show that it is able to generate high quality results repeatedly for problems that resemble instances in the training set.

- Encouraging proof-of-concept results and definition of new research opportunities to generalize the mechanism to supplement the current exploratory approaches of incorporating machine learning with optimization.

2.0 Optimizing Vaccine Distribution Networks in Low and Middle-income Countries

In this chapter, we formulate a mathematical programming model for the design of a typical WHO-EPI network with the goal of minimizing costs while providing the opportunity for universal coverage. Since it is only possible to solve small versions of the model to optimality, we develop a column generation algorithm to solve the MIP model, and an iterative heuristic that cycles between solving restrictions of the original problem and show that that this heuristic can find very good solutions in reasonable time for larger problems that are not directly solvable. A significant portion of the remainder of Chapter 2 appears in (Yang & Rajgopal, 2020a).

2.1 Problem Development and Literature Review

In most LMICs, vaccines are distributed via a four-tier hierarchical legacy medical network such as the one depicted in Figure 1. Typically, EPI vaccines are purchased in bulk and shipped in by air once or twice a year, then stored in a national distribution center in the capital (or other large city). Required vaccine volumes are transported every three months to *regional* distribution centers using a specialized vehicle such as a large cold truck. Each regional distribution center delivers vaccines to its surrounding *district* centers every month using a smaller cold truck or more commonly, 4×4 trucks with cold storage boxes. Finally, the vaccines are transported from district centers in a vaccine carrier/cooler using locally available means of transportation such as trucks, cars, motorbikes, bicycles, boats, or sometimes even by foot, to local *clinics* where infants, children and pregnant women come to receive vaccinations. This last step is typically, a “pull” operation with monthly pickup by the clinic. A characteristic of EPI vaccines is that they must be stored/transported while maintaining appropriate temperatures (2–8 °C), so that this vaccine distribution chain is often referred to as a cold chain.

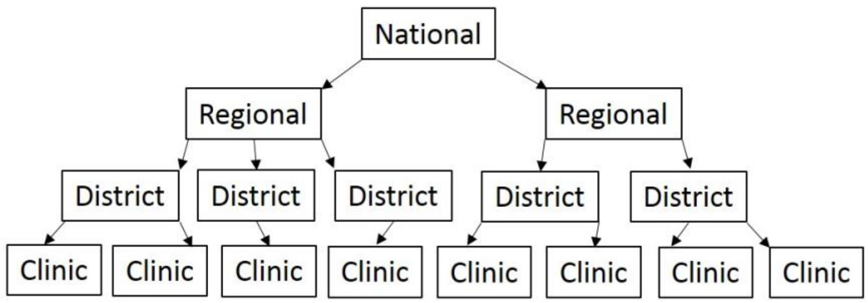

Figure 1 A typical four tier legacy medical network

To develop an optimal vaccine distribution network design along with optimal operational policies for a country that also follow WHO guidelines, we separate this from the existing legacy medical supply chain of which it is typically a component, and model the cold chain independently. Our objective is to minimize the overall cost of transportation, facilities and storage over the whole network, while guaranteeing universal access and following WHO operational guidelines. In the model, vaccines flow from the national center (source node) to clinics (sink nodes), usually via one or more intermediate hubs (transshipment nodes). Although multiple (usually 6 to 8) vaccines are handled in the cold chain, transportation and storage capacities are only affected by the overall space required. Therefore, we only consider the total volume of vaccines shipped or stored. Hub locations are chosen from the current locations of legacy intermediate nodes (regional or district distribution center), and while we retain the choices of monthly or quarterly replenishments as per WHO guidelines, we allow a hub to freely select *either* option. The model determines the clinics assigned to each hub, the national center-to-hub and hub-to-hub connections, the actual vaccine flows on these connections, and the types of storage and transportation devices to be deployed at each location and for each flow.

We make the following assumptions to model the EPI vaccine network:

(1) The network should be capable of meeting all demand that can arise at each clinic, and this demand is determined by the estimated population that the particular clinic serves.

(2) The locations of the clinics and the national distribution center are fixed but we can choose hub distribution centers freely from the current set of regional and district centers.

(3) Each clinic is assigned to a hub for its vaccines (although clinics close to the national center could be directly supplied by it), and replenished once a month. Each hub is supplied by the national center or by another hub.

(4) The national center is the root node of the network and all other nodes (hubs and clinics) have exactly one inbound arc.

(5) As per WHO guidelines, a hub is replenished either quarterly or monthly.

(6) Every open facility has an appropriately sized storage device, to be selected from the WHO's pre-qualified list of devices.

(7) As per WHO guidelines, there is a 25% safety buffer at each clinic location so that the total demand volume is inflated by this factor.

We now discuss these assumptions. First, universal access is a primary goal of the WHO and our model's constraints explicitly capture this. Second, our model is based on using the existing facilities for hubs as opposed to building new ones. Third, in most LMICs, operational simplicity is a driving requirement because resources are very constrained and it can be a challenge to find qualified logisticians and trained personnel who can deal with multiple suppliers and different types of equipment. We therefore retain the current approach of restricting each facility location to having a single supplier, and a single type of storage device that is selected from the WHO's pre-qualified list (World Health Organization, 2009). Finally, for the same reason of operational ease, we do not attempt to determine optimal safety buffers at clinics or reorder points by location; hub locations are restricted to one of two replenishment intervals (monthly or quarterly) and all clinics have the same 25% buffer inventory levels as per WHO guidelines.

Several variations of this class of network design problems have been addressed by the operations research community, including the *p*-median problem, the uncapacitated and capacitated facility location problems (Melo, Nickel, & Saldanha-da-Gama, 2009; Mirchandani, 1990; Şahin & Süral, 2007) and extensions to include transportation cost (Geoffrion & Graves, 1974). The facility location problem is often combined with the vehicle routing problem and typically uses heuristics (Wu, Low, & Bai, 2002). It has also been extended with consideration of risk pooling (Shen, Coullard, & Daskin, 2003) and facility failures (Snyder & Daskin, 2005), with Lagrangian relaxation being a common solution strategy (Daskin, Snyder, & Berger, 2005). Klose and Drexl (Klose & Drexl, 2005) review several facility location models for distribution system design.

As an extension to these facility location problems, the hub selection problem considers the situation where one or more nodes are designated as facilities that serve as consolidation, switching or transshipment points and connect to origin/destination nodes. It has received attention in applications ranging from airlines and emergency services to intermodal logistics and postal delivery services. Interested readers are referred to the surveys on hub location problems presented in (Campbell & O'Kelly, 2012; Farahani et al., 2013). Models for these problems address a variety of objectives and combinations of various problem environments (Farahani et al., 2013). These include the domain of the hub nodes (all network nodes, discrete subset or anywhere along a continuous plane), the number of nodes to be designated as hubs (pre-specified vs. unspecified), hub capacity (limited vs. unlimited), cost of locating hubs (none, fixed or variable), allocation of non-hub nodes to hubs (single vs. multiple) and allocation costs (none, fixed or variable). In particular, our formulation chooses an unspecified number of hubs from a discrete subset of capacitated hub locations that incur fixed costs, and each non-hub location is assigned to a single hub, with allocation costs that are exogenous to the model.

In studying the literature for exact algorithms to find the optimum solution to models similar to ours (capacitated, single allocation, *p*-hubs; albeit with differing objectives), approaches include bi-criteria integer linear programming (Costa, Captivo, & Clímaco, 2008), mixed integer programming (Correia, Nickel, & Saldanha-da-Gama, 2010; Kratica et al., 2011), generalized Benders' decomposition (de Camargo & Miranda, 2012), and fuzzy integer linear programming (Taghipourian et al., 2012). The largest problem size solved is reported as having "up to 10,000 integer variables" (de Camargo & Miranda, 2012). Given the limits to the size of the problems that can be solved optimally, many others have resorted either to metaheuristics (Ernst & Krishnamoorthy, 1999), heuristics based on Lagrangean relaxation or Benders' decomposition (Contreras, Díaz, & Fernández, 2009; de Camargo, de Miranda, & Ferreira, 2011), or heuristics designed for specific formulations (Chen, 2008).

A unique aspect of our model is that replenishment frequencies as well as capacities at hubs and for transportation along arcs are limited to multiple discrete options. This greatly increases the number of binary variables to well over 10,000 even for medium sized problems, and while most of the prior work looks at under 20 hubs, our formulation for an entire country has a number of hubs as well as a total number of binary variables that are an order of magnitude higher. Thus, solving full-country problems using an exact approach is not a viable option. Results from our initial experiments with using Lagrangean relaxation were also not encouraging. Therefore, we develop approaches that are designed for the application domain and the specific class of problems that we study. These and related computational issues are discussed in detail in Sections 2.5 and Section 2.6.

In terms of work specific to vaccine distribution networks, Chen et al. (Chen et al., 2014) were the first to model the network in 2014 as a planning model to maximize the number of children being fully immunized under current network capacity; they then extended it to the case where capacity expansion is allowed. However, this work addresses operations in an existing network as opposed to its design. In 2019, Lim et al. (Lim et al., 2019) proposed a model to design a minimum cost vaccine distribution network, and utilized an evolutionary strategy to solve this problem. Their model assumes that deliveries to hubs are coordinated and done using vehicle loops and fixes the storage devices; thus, multiple trips might have to be made along a route if the volume cannot be handled in a single trip.

In Section 2.2 we present a mixed integer programming model that draws on the initial work by Lim et al. (Lim et al., 2019) but allows for flexibility in replenishment, allows storage devices to be selected in the required size, does not require delivery coordination during replenishment and ensures that all deliveries to a node are made in a single trip as is typically the case in practice. We then provide a column generation algorithm to solve the problem in Section 2.3 and also develop a mathematical programming based iterative heuristic that cycles between solving restrictions of the original problem in Section 2.4; numerical results for this are provided in Section 2.5. As discuss in Section 2.6, we show that it can serve as a simple alternative to solving the formulation directly when a good solution is required quickly.

2.2 Formulation

We now develop our model formulation.

Index sets:

N: National distribution center $=\{0\}$

H: Potential hub distribution centers = $\{1, 2\ldots\ h\}$

C: Local clinics = $\{h+1\ldots\ n\}$, where $n = |H| + |C|$

V: Vertices: $N \cup H \cup C$

A: Arcs:$(i, j)|i \in N \cup H, j \in H \cup C; i \neq j$

T: Transportation vehicles

R: Storage devices

F: Replenishment frequency: {quarterly (=0), monthly (=1)}

Parameters:

c_{ijt}^{T}: Transportation cost per km of vehicle type t between locations i and j ; $(i, j) \in A; t \in T$

c_{jr}^{S}: Annual facility cost when facility j is open and uses storage device r; $r \in R$

p_{t}^{T}: Transportation capacity per trip of vehicle t; $t \in T$

p_{r}^{S}: Storage capacity of device r; $r \in R$

g_f: Annual number of replenishments; $f \in F$ (g_f = 4 if f = 0; g_f = 12 if f = 1)

d_{ij}: Driving distance (km) between location i and location j; $(i, j) \in A$

b_j: Annual demand volume at location j, $j \in C$

Variables:

$U_{ijtf} \in \{0,1\}$: 1 if vaccines flow from location i to location j using vehicle type $t \in T$ and with replenishment frequency $f \in F$, 0 otherwise; $i \in N \cup H, j \in H \cup C$

$W_{irf} \in \{0,1\}$: 1 if hub location $i \in H$ is open and uses storage device of type $r \in R$ and replenishment frequency $f \in F$, 0 otherwise

X_{ij}: Annual flow (volume) of vaccines from location i to location j; $i \in N \cup H, j \in H \cup C$

The mixed integer program for designing the optimal network may then be formulated as follows:

<u>Program MIP-1</u>:

$$Min \sum_{j\in H}\sum_{r\in R}\sum_{f\in F} c_{jr}^{S} W_{jrf} + \sum_{(i,j)\in A}\sum_{t\in T}\sum_{f\in F} 2c_{ijt}^{T} g_f d_{ij} U_{ijtf} \qquad (2\text{-}1)$$

subject to

$$\sum_{r\in R}\sum_{f\in F} W_{jrf} \leq 1 \qquad j \in H \qquad (2\text{-}2)$$

$$\sum_{i\in N\cup H}\sum_{t\in T} U_{ijt1} = 1 \qquad j \in C \qquad (2\text{-}3)$$

$$\sum_{i\in N\cup H}\sum_{t\in T}\sum_{f\in F} U_{ijtf} \leq 1 \qquad j \in H \qquad (2\text{-}4)$$

$$\sum_{r\in R} W_{jrf} - \sum_{i\in N\cup H}\sum_{t\in T} U_{ijtf} = 0 \qquad j \in H, f \in F \qquad (2\text{-}5)$$

$$\sum_{i\in N\cup H} X_{ij} - \sum_{k\in H\cup C} X_{jk} = 0 \qquad j \in H \qquad (2\text{-}6)$$

$$\sum_{i\in N\cup H} X_{ij} = b_j \qquad j \in C \qquad (2\text{-}7)$$

$$\sum_{r \in R} \sum_{f \in F} p_r^S W_{jrf} - (1/g_f) \sum_{i \in N \cup H} X_{ij} \geq 0 \qquad j \in H \qquad (2\text{-}8)$$

$$\sum_{t \in T} \sum_{f \in F} p_t^T U_{ijtf} - (1/g_f) X_{ij} \geq 0 \qquad i \in N \cup H, j \in H \cup C \qquad (2\text{-}9)$$

$$X_{ij} \geq 0 \qquad i \in N \cup H, j \in H \cup C \qquad (2\text{-}10)$$

$$W_{jrf} \in \{0, 1\} \qquad j \in H, r \in R, f \in F \qquad (2\text{-}11)$$

$$U_{ijtf} \in \{0, 1\} \qquad i \in N \cup H, j \in H \cup C, t \in T, f \in F \qquad (2\text{-}12)$$

The objective function (2-1) has two components: annual hub facility costs and total annual round-trip transportation costs. Constraints (2-2) ensure that every open hub j has a single replenishment frequency and a single type of storage device, while Constraints (2-3) ensure that each clinic has exactly one inflow and a monthly replenishment frequency. Constraints (2-4) ensure that each hub has at most one inflow with unique associated replenishment frequency and transport device. Constraints (2-5) ensure there is no flow associated with a hub that is not open and Constraints (2-6) and (2-7) are standard flow balance equations at hubs and clinics. Constraints (2-8) ensure that in each hub there is a sufficiently large storage device to store the vaccines required within each replenishment interval. Finally, Constraints (2-9) ensure that a transportation mode with sufficient capacity is selected to carry the required volume of vaccines for replenishment. Constraints (2-10) – (2-12) are self-explanatory.

2.2.1 Limitations with Solving MIP-1

To explore the solution of the model described by **MIP-1** we tested it with a standard off-the-shelf mixed integer programming solver (IBM ILOG CPLEX 12.6) on a 3.20 GHz processor with 8 GB of memory. Problems of various sizes were generated using data that were derived from information we could access for four different countries in sub-Saharan Africa. While all of these countries currently have a similar four-tiered distribution architecture, they have significantly different demographic characteristics (size, population density, etc.) and also differ in the number of potential hub locations. In general, the effort required for a problem depends largely on the total number of nodes as well as potential hub locations. However, it also depends on the population distribution and the costs, and we were unable to arrive at a systematic way to specify the limits to what is solvable. Our numerical tests are described in more detail in Section 2.5, but as a gross generalization, only problems with about 200-250 nodes or fewer, and a maximum of 15-20 potential hub locations can be solved in reasonable time. Most problems for an entire country are larger than this.

A key fact that makes **MIP-1** hard to solve is that the model has a large number of 0-1 decision variables. For example, if we can choose from three types of transport vehicles and four types of storage devices (i.e., {$|T|$=3, $|R|$=4), Table 1 illustrates the number of decision variables in **MIP-1**. Thus, in order to solve the full problem for one of our instances with 685 nodes and 41 candidate hubs, we end up with 168,838 integer variables. Even for a mid-sized problem with 100 nodes and 15 candidate hubs, the number of binary decision variables is close to 10,000. This clearly calls for heuristics or other approaches.

Table 1 Number of decision variables in MIP-1 with n nodes and h potential hubs

Decision Variables	W_{jrf}	U_{ijtf}	X_{ij}
Type	Integer	Integer	Continuous
Number	$8h$	$6hn$	hn

2.3 A Column Generation Algorithm

Our first approach was to develop a column generation algorithm to solve the **MIP-1** formulation in Section 2.2, which we now describe. Consider the variable W_{jrf}. In any feasible solution we will have some subset of these variables being equal to 1, and for each hub location $j \in H$ that is selected (i.e., $W_{j..} = 1$) we will have an associated value of $r \in R$ and an associated replenishment frequency $f \in F$. We will call any such solution an *enumeration*. Let us define the following in addition:

Index sets:

K: All Possible Enumerations

Parameters:

a_{jrf}^{k}: Indicator parameter that is 1 if enumeration k has hub j being open along with storage device r and replenishment frequency f; 0 otherwise.

Variables:

V_k: 1 if enumeration $k \in K$ is chosen; 0 otherwise

Note that a_{jrf}^{k} satisfies the following conditions for an enumeration k: $\sum_{r \in R} \sum_{f \in F} a_{jrf}^{k} \leq 1$ for $j \in H$. The condition states that if hub j is not open in enumeration k we have $a_{jrf}^{k} = 0$ and if hub j is open then only one a_{jrf}^{k}=1 across all r and f for that j.

The **MIP-1** formulation in Section 2.2 can be then reformulated as:

Program MP:

$$Min \sum_{k\in V}\sum_{j\in H}\sum_{r\in R}\sum_{f\in F} c_{jr}^{S} a_{jrf}^{k} V_k + \sum_{(i,j)\in E}\sum_{t\in T}\sum_{f\in F} 2c_{ijt}^{T} g_f d_{ij} U_{ijtf} \qquad (2\text{-}13)$$

subject to

$$\sum_{k\in K} V_k = 1 \qquad (2\text{-}14)$$

$$\sum_{i\in N\cup H}\sum_{t\in T} U_{ijt1} = 1 \qquad j\in C \qquad (2\text{-}15)$$

$$\sum_{i\in N\cup H}\sum_{t\in T}\sum_{f\in F} U_{ijtf} \le 1 \qquad j\in H \qquad (2\text{-}16)$$

$$\sum_{k\in K}\sum_{r\in R} a_{jrf}^{k} V_k - \sum_{i\in N\cup H}\sum_{t\in T} U_{ijtf} = 0 \qquad j\in H, f\in F \qquad (2\text{-}17)$$

$$\sum_{i\in N\cup H} X_{ij} - \sum_{k\in H\cup C} X_{jk} = 0 \qquad j\in H \qquad (2\text{-}18)$$

$$\sum_{i\in N\cup H} X_{ij} = b_j \qquad j\in C \qquad (2\text{-}19)$$

$$\sum_{k\in K}\sum_{r\in R}\sum_{f\in F} p_r^{S} a_{jrf}^{k} V_k - (1/g_f)\sum_{i\in N\cup H} X_{ij} \ge 0 \qquad j\in H \qquad (2\text{-}20)$$

$$\sum_{t\in T}\sum_{f\in F} p_t^{T} U_{ijtf} - (1/g_f) X_{ij} \ge 0 \qquad i\in N\cup H, j\in H\cup C \qquad (2\text{-}21)$$

$$X_{ij} \ge 0 \qquad i\in N\cup H, j\in H\cup C \qquad (2\text{-}22)$$

$$V_k \in \{0,1\} \qquad k\in K \qquad (2\text{-}23)$$

$$U_{ijtf} \in \{0,1\} \qquad i\in N\cup H, j\in H\cup C, t\in T, f\in F \qquad (2\text{-}24)$$

Here Constraints (2-14) ensure that we pick exactly one enumeration. Constraints (2-17) ensure that the inbound frequency of an open hub in the enumeration is identical to its replenishment frequency. Constraints (2-20) ensure that for the enumeration selected the associated storage device has enough capacity. The remaining constraints are similar to MIP-1 and are self-explanatory.

Our column generation algorithm **(Algorithm 1)** is as follows:

STEP 1: *Initialization of* ***LRMP***

Consider the restricted problem where we only consider a subset of all enumerations in **MP** indexed by $K' \subset K$, and with binary constraints (2-23) and (2-24) replaced by their linear relaxations. Let us refer to this relaxed restricted problem as **LRMP**.

STEP 2: *Solving* ***PP***

Suppose we solve this linear program **LRMP** and obtain the optimal vector of simplex multipliers. Let us denote the $|H|$ simplex multipliers corresponding to (2-20) by the vector π, the $|H|\times|F|$ multipliers corresponding to (2-17) by the vector ρ, and the multiplier corresponding to (2-14) by μ.

In order to find a column (location combination) that could improve the current optimal solution of **LRMP**, the following pricing problem **PP** is solved to find a column with a negative reduced cost:

Variables:

a_{jrf}: Indicator parameter that is 1 if the new column that corresponds to a new enumeration is set to have hub j being open along with storage device r and replenishment frequency f; 0 otherwise.

Program PP:

$$\begin{aligned}
&Min \sum_{j\in H}\sum_{r\in R}\sum_{f\in F} c_{jr}^{S} a_{jrf} - \sum_{j\in H}\pi_j \sum_{r\in R}\sum_{f\in F} p_r^S g_f a_{jrf} - \sum_{j\in H}\sum_{f\in F}\rho_{jf}\sum_{r\in R} a_{jrf} - \mu \\
&= Min \sum_{j\in H}\sum_{r\in R}\sum_{f\in F} c_{jr}^{S} a_{jrf} - \sum_{j\in H}\sum_{r\in R}\sum_{f\in F} \pi_j p_r^S g_f a_{jrf} - \sum_{j\in H}\sum_{f\in F}\sum_{r\in R}\rho_{jf} a_{jrf} - \mu \\
&= Min \sum_{j\in H}\sum_{r\in R}\sum_{f\in F} (c_{jr}^{S} - \pi_j p_r^S g_f - \rho_{jf}) a_{jrf} - \mu
\end{aligned} \tag{2-25}$$

$subject\ to$

$$\sum_{r\in R}\sum_{f\in F} a_{jrf} \le 1 \tag{2-26}$$

$$a_{jrf} \in \{0, 1\} \qquad j \in H, r \in R, f \in F \tag{2-27}$$

Note that **PP** has a closed form solution: for any hub $j \in H$ we pick the device $r \in R$ and replenishment frequency $f \in F$ that has the smallest value for the coefficient of a_{jrf} in (2-25) and if the result of subtracting μ from it is negative, then we set the value of this a_{jrf} = 1, and all other a_{jrf} to zero for that j. For any $j \in H$ where we cannot find a negative value as above we set all a_{jrf} = 0. If the coefficients minus μ are all nonnegative for every j then there are no promising enumerations and we stop. The process may be formalized as shown in Table 2:

Table 2 Algorithm 1.1

Input:

Parameter c_{jr}^S, p_r^S, g_f

Optimal vector of simplex multipliers π_j, ρ_{jf}, μ

---

for $j = 1, 2\ldots, h$:

    Find the smallest $c_{jr}^S - \pi_j p_r^S g_f - \rho_{jf}$ for $\forall r, f$

    if $c_{jr}^S - \pi_j p_r^S g_f - \rho_{jf} - \mu < 0$:

        Set $a_{jrf} = 1$ for this particular r, f

        Set $a_{jrf} = 0$ for all other r, f

    else:

        Set $a_{jrf} = 0$ $\forall r, f$

STEP 3: *Column generation*

Add the column corresponding to the enumeration generated in the previous step and repeat it until no enumeration is generated in Step 2, i.e., all $\sum_{j\in H}\sum_{r\in R}\sum_{f\in F}(c_{jr}^S - \pi_j p_r^S g_f - \rho_{jf})a_{jrf} - \mu \geq 0$

STEP 4: *Obtain integer solution*

We convert the fractional variables into integer variables as follows to obtain a feasible integer solution:

a. If there are multiple V_k that are greater than 0, choose the largest value to set to 1 and the rest to zero.

b. For any node j that is closed in the previous step, set $X_{ij} = 0$ for all i and link each clinic associated with this node to its closest open hub, and choose the smallest feasible device at the hub.

c. For U_{ijtf}, choose the frequencies specified by the final enumeration V_k , if this is infeasible for vehicle type t, choose the cheapest feasible vehicle.

2.3.1 Limitations of the Column Generation Algorithm

We explored the performance of the column generation algorithm using examples derived from the data we had for four different LMICs as discussed in Section 2.2.1; the results are summarized in Table 3. We include results for the 27 test problems that we were able to solve optimally using a standard off-the-shelf mixed integer programming solver (IBM ILOG CPLEX 12.6) on a 3.20 GHz processor with 8 GB of memory. We list the number of potential hub locations, the total number of nodes, the number of binary variables, and a label that identifies the population in the area as being dense, moderate or sparse. The number of nodes and potential hub locations in these problems ranged from 10 to 333, and from 1 to 26, respectively, while the total number of binary variables in the full problem ranged from 68 for the smallest problem to 52,156 for the largest problem we were able to solve optimally. We also list the CPU times for the CPLEX solver to find the optimum solution and for Algorithm 1 to converge, along with the percentage gap between the cost of the solution from the algorithm and the true optimum cost.

Table 3 Computational results for Algorithm 1

No.	Hubs	Nodes	0/1 Variables	Pop. Density	CPU: Optimum	CPU: Algorithm 1	Gap %
1	1	10	68	sparse	<1s	<1s	0%
2	1	18	116	moderate	<1s	<1s	0%
3	2	11	148	sparse	<1s	<1s	0%
4	3	22	420	sparse	2s	<1s	0%
5	2	49	604	sparse	<1s	<1s	0%
6	3	39	726	moderate	2s	<1s	0%
7	4	44	1,088	sparse	3s	<1s	0%
8	4	48	1,184	moderate	<1s	<1s	0%
9	4	55	1,352	moderate	4s	2s	0%
10	4	64	1,568	moderate	7s	2s	0.1%
11	4	65	1,592	dense	8s	3s	0.2%
12	5	77	2,350	moderate	1.6s	<1s	0%
13	8	56	2,752	sparse	16s	21s	0.3%
14	7	99	4,214	dense	4.4s	3s	0.33%
15	11	96	6,424	moderate	10s	37s	0.6%
16	10	117	7,100	moderate	146s	-	-
17	14	101	8,596	sparse	119s	-	-
18	12	128	9,312	dense	116s	-	-
19	8	206	9,952	dense	~10h	76s	0.71%
20	14	148	12,544	moderate	103s	-	-
21	17	141	14,518	moderate	79s	-	-
22	16	162	15,680	dense	1,304s	-	-
23	13	210	16,484	moderate	~1d	-	-
24	14	235	19,852	dense	~2d	-	-
25	19	176	20,216	moderate	4,649s	-	-
26	20	295	35,560	moderate	387s	-	-
27	26	333	52,156	moderate	2,748s	-	-

Unfortunately, it turns out that Algorithm 1 does not address our needs and is actually unable to solve several problems that the off-the-shelf solver is able to. In fact there is only one problem (instance 19 in the table) where it shows any real benefit, as it offers us an excellent solution in about a minute (as opposed to about 10 hours by the solver). More importantly, we found that it could not solve any problem that cannot be directly solved by the solver. One reason that it has no advantage in speed is that as discussed in Section 2.2.1, our difficulty comes from the number of integer solution variables. Although a column generation algorithm addresses this by limiting the enumeration, it is still not sufficiently effective in our case. Second, even with the examples that the algorithm does give solutions to, with **STEP 4** that obtains the integer solution, we cannot guarantee optimality. Given that we have many choices of transportation and storage devices, the solutions of **LRMP** tend to be highly fractional. Choosing the largest V_k and setting it to 1 and the remaining to 0 often fails to give high quality results due to a large integrality gap. In short, better algorithms are needed to solve the problem. We will discuss this in the next section.

2.4 An Iterative Cyclic Algorithm

In Section 2.3.1, we saw that the column generation algorithm has no clear benefit to being able to overcome the limitations of an off-the-shelf solver. Thus, a more powerful algorithm is desirable for larger problems. As discussed in Section 2.2.1, the key fact that makes the **MIP-1** hard to solve, besides the natural difficulty of MIP, is that the model has too many decision variables. In this section, we describe an easy-to-implement MIP-based heuristic that solves a sequence of MIP problems, each of which is a restricted version of Program **MIP-1** that is relatively easy to solve. These restrictions are with respect to either the replenishment frequencies used at hubs or the total number of hubs that are open.

The method is motivated by initial experiments where we tested the **MIP-1** when some variables are fixed, thereby reducing the number of decision variables. First, we formulated a restricted version by fixing all replenishments at hubs to be done either once a month or all replenishments at hubs to be done once a quarter. For smaller problems, these restricted versions yielded solutions in a very short amount of time and with values less than 1% larger than the true optimum. Table 4 demonstrates a sample problem with 385 nodes and 20 candidate hubs when we fixed the replenishment frequency of all hubs to be identical. In the second column, we show results from running model **MIP-1** without any restrictions. In the third column, we show results for the model when we only allow monthly replenishments while in the fourth column, all hubs are restricted to be replenished quarterly. Clearly, the restricted versions yielded solutions in a very short amount of time and with costs only around 0.6% above the complete initial **MIP-1** model. In fact, all examples we tested yielded costs that were within 1% of the optimum value (when it could be obtained).

Table 4 A sample problem when the replenishment frequency is fixed

	MIP	Fix Monthly	Fix Quarterly
Cost	$602,133	$605,646	$605,942
Time	202,158s	102s	1,060s

Next, we formulated an alternative restriction where we fixed the total number of hubs to be open, and the results obtained for a small sample problem are illustrated in Table 5, where the last row corresponds to the original MIP in which we determine how many and which hubs will be open at the optimum. Once again, fixing a portion of the network structure generally yields solutions much more quickly (although as we force more hubs to be open the time does increase).

Table 5 An example when the number of open hubs is fixed

No. Open Hubs	CPU time	Cost
<6	infeasible	-
6	<1	$281,111
7	2s	$259,209
8	3s	$240,800
9	3s	$235,743
10	10s	$229,967
11	12s	$225,300
12	18s	$222,214
13	35s	$224,922
14	440s	$227,541
MIP-1	**103s**	**$222,214**

Based on these observations, our cyclic algorithm starts by solving a restricted version of Program 1 with an initial vector of fixed replenishment frequencies at the hubs, to obtain a locally optimal set of open locations under this frequency vector. The algorithm then fixes these open hub locations and solves another restricted version of Program 1 (with other hubs kept closed) to find the corresponding optimal frequencies. The procedure iterates until we cannot improve the solution. In order to formalize the algorithm, let us denote:

ε : a suitably small constant

$\boldsymbol{f}$: a vector of order $|H|$ indicating the replenishment frequency at hubs; if the ith element is 1 then hub i is set to be replenished quarterly and a frequency-fixing constraint $\sum_{r\in R} W_{jr1} = 0$ is added to the model; alternatively if the ith element is 2 then hub i is set to be replenished monthly, and the frequency-fixing constraint $\sum_{r\in R} W_{jr0} = 0$ is added; if a hub is closed the corresponding element is set to 0.

$\boldsymbol{l}$: a binary vector of order $|H|$ indicating the status of each hub; if the ith element equals 0, hub i is forced to be closed and we add a location-fixing constraint $\sum_{r\in R}\sum_{f\in F} W_{jrf} = 0$ to the model; if the ith element is equal to 1, hub i is set to be open, and we add the location-fixing constraint $\sum_{r\in R}\sum_{f\in F} W_{jrf} = 1$ to the model.

Z_f^k: locally optimum objective value at step k when frequencies are fixed

Z_l^k*:* locally optimum objective value at step k when locations are fixed

Algorithm 2 may then be specified as follows:

STEP 1: *Initialization*

Generate a random vector of length $|H|$ where every entry is one of either 1 or 2 and define it to be $\boldsymbol{f}^1$. Note that initially, every hub is allowed to be open, and if it is open it must use the replenishment frequency specified via $\boldsymbol{f}^1$. Let $k = 1$.

STEP 2: *Local optimum with fixed frequencies*

Set $\boldsymbol{f} \leftarrow \boldsymbol{f}^k$ and solve Program 1 under this fixed frequency vector with the corresponding frequency-fixing constraints. Let Z_f^k be the local optimum value obtained, with corresponding hub locations defined by the vector $\boldsymbol{l}^k$. If $k = 1$ go to **STEP 3** after deleting the frequency-fixing constraints added at this step, else if $Z_l^{k-1} - Z_f^k \leq \varepsilon$, i.e., there is no improvement, stop the algorithm with objective value Z_f^k. Otherwise, it means that the algorithm is still improving the solution, so we delete the frequency-fixing constraints added in this step and continue on to **STEP 3**.

STEP 3: *Local optimum with fixed open locations*

Set $\boldsymbol{l} \leftarrow \boldsymbol{l}^k$ and solve Program 1 under this fixed location vector with the corresponding location-fixing constraints, let Z_l^k be the local optimum value obtained with corresponding replenishment frequencies. If $Z_f^k - Z_l^k \leq \varepsilon$, there is no improvement; stop the algorithm with value Z_l^k. Otherwise, delete the location-fixing constraints added in this step and go to **STEP 4**.

STEP 4: *Update frequency*

Update the frequency vector via $\boldsymbol{f}^{k+1}$ to be the same as the corresponding replenishment frequencies obtained in **STEP 3**. In the case that a hub (say, the *i*th) is not open in the solution obtained at STEP 2, the *i*th element of $\boldsymbol{f}^{k+1}$ is set to 0.

Set $k \leftarrow k + 1$ and then return to **STEP 2**.

The algorithm is summarized in Table 6:

Table 6 Algorithm 2

Input:

MIP-1

H: the set of all candidate hubs

$h = |H|$

STEP 1. *Initialization*

Generate a random binary vector $\boldsymbol{f}^k$ of order h

Set $k = 1$

STEP 2. *Local optimum with fixed frequencies*

Set $\boldsymbol{f} \leftarrow \boldsymbol{f}^k$

Solve the MIP with fixed frequency $\boldsymbol{f}$ and obtain optimal value Z_f^k

if $k > 1$ and $Z_l^{k-1} - Z_f^k \leq \varepsilon$:

Stop

else:

Obtain $\boldsymbol{l}^k$

Delete all frequency-fixing constraints.

Go to **STEP 3**

STEP 3. *Local optimum with fixed open locations*

Set $\boldsymbol{l} \leftarrow \boldsymbol{l}^k$

Solve the MIP with fixed location $\boldsymbol{l}$ and obtain optimal value Z_l^k

if $Z_f^k - Z_l^k \leq \varepsilon$:

Stop

else

Obtain the corresponding replenishment frequencies

Delete all location-fixing constraints.

Go to **STEP 4**

STEP 4. *Update frequency*

Update $\boldsymbol{f}^{k+1}$ and set $k \leftarrow k + 1$

Go to **STEP 2**

Observation: In **STEP 2**, if $k \geq 1$, then $Z_l^{k-1} \geq Z_f^k$ for all k. In **STEP 3**, $Z_f^k \geq Z_l^k$ for all k. In other words, the local optima are always nonincreasing.

At every step before it stops, the method is always improving. From the computational results, we found that the algorithm works best if we start with a frequency where "all hubs are equal," i.e., the initial replenishment frequencies are the same at all hubs. The next section will provide with a detailed computational result.

2.5 Computational Results

We tested **Algorithm 2** using a number of problems; as stated in Sect. 2.2.1 we generated these from the data we had for four different LMICs. Based on experiments we conducted, Table 7 summarizes computational results for the 27 test problems that we were able to solve optimally. We also report on two problems (Nos. 26 and 28) for which optimal solutions are not available.

Table 7 Computational results for Algorithm 2

No.	Hubs	Nodes	0/1 Variables	Pop. Density	CPU: Optimum	CPU: Algorithm	Gap %
1	1	10	68	sparse	<1s	<1s	0%
2	1	18	116	moderate	<1s	<1s	0%
3	2	11	148	sparse	<1s	<1s	0%
4	3	22	420	sparse	2s	<1s	0%
5	2	49	604	sparse	<1s	<1s	0%
6	3	39	726	moderate	2s	<1s	0%
7	4	44	1,088	sparse	3s	<1s	0%
8	4	48	1,184	moderate	<1s	<1s	0%
9	4	55	1,352	moderate	4s	<1s	0%
10	4	64	1,568	moderate	7s	<1s	0%
11	4	65	1,592	dense	8s	<1s	0.27%
12	5	77	2,350	moderate	1.6s	<1s	0%
13	8	56	2,752	sparse	16s	4s	0%
14	7	99	4,214	dense	4.4s	<1s	0%
15	11	96	6,424	moderate	10s	5s	0%
16	10	117	7,100	moderate	146s	29s	0.56%
17	14	101	8,596	sparse	119s	32s	0.28%
18	12	128	9,312	dense	116s	29s	0.17%
19	8	206	9,952	dense	~10h	43s	0.17%
20	14	148	12,544	moderate	103s	49s	0%
21	17	141	14,518	moderate	79s	50s	0%
22	16	162	15,680	dense	1,304s	46s	0.46%
23	13	210	16,484	moderate	~1d	76s	0.52%
24	14	235	19,852	dense	~2d	83s	0.62%
25	19	176	20,216	moderate	4,649s	207s	0.24%
26	11	366	24,244	dense	-	62s	-
27	20	295	35,560	moderate	387s	89s	0.32%
28	28	271	45,752	moderate	-	11s	-
29	26	333	52,156	moderate	2,748s	126s	0.54%

Similar to Table 3, for each problem we list the number of potential hub locations, the total number of nodes in the network, the number of binary variables in the **MIP-1** formulation, and a label that identifies the population in the area as being dense, moderate or sparse. The number of nodes and potential hub locations in these problems ranged from 10 to 333, and from 1 to 28, respectively, while the total number of binary variables in the full problem ranged from 68 for the smallest problem to 52,156 for the largest problem we were able to solve optimally. We also list the CPU times for the CPLEX solver to find the optimum solution and for our algorithm to converge, along with the percentage gap between the cost of the solution from the algorithm and the true optimum cost.

2.6 Discussion

We now discuss the results from the previous section. First, it is clear that there is no obvious, direct relationship with any one specific factor listed in the tables (nodes, hubs, binary variables, and population density); rather, the effort required to solve a problem optimally depends on the combination of these factors. However, as might be expected, the total number of 0/1 variables seems significant. Smaller problems with under (say) 15,000 binary variables are directly solvable in a matter of seconds.

Second, with larger problems, solution times start to increase; there is one problem (no. 18) that took almost 10 h to solve, and in at least two instances (nos. 23 and 24) the solution time was on the order of days. However, there was also a problem (no. 6) that was larger than either of these and that could be solved in a little over 6 min. In general, it is hard to pinpoint what specific characteristics make the problems harder to solve optimally, and if we are presented with a relatively large new problem there does not appear to be any obvious way to say how CPLEX might perform on it.

Third, the iterative cyclic approach of our algorithm appears to be much more stable in its performance when compared to CPLEX. Convergence is achieved in under one minute for 22 of the 27 problems tested, and the longest it took (no. 19) was about 3.5 min. More importantly, the solution that it finds has a cost that is always within 1% (and most often within about 0.5%) of the true minimum cost.

Finally, we also generated much larger problems (including problems that represented the complete network for each country) which could not be solved optimally by CPLEX; our algorithm also failed to converge on most of these because solving even the restricted problems in steps 2 and 3 becomes impossible. , The method did converge to a solution for two instances ((26 and 28 in Table 7) that could not be solved optimally. While a cost comparison is obviously not possible, based on how it does for smaller problems, we can speculate though that these are probably decent solutions.

Overall, **Algorithm 2** generates high quality solutions with a substantially smaller amount of computation time than direct solution. Due to the ease with which it can be implemented, it can serve as a simple alternative to solving **MIP-1** directly when a good solution is required quickly, especially when the number of hub candidate locations is not large.

2.7 Summary

The problem of designing a distribution network for WHO-EPI vaccines is a complex one and one that becomes increasingly harder to solve as the problem size grows. We first formulated a mathematical programming model for the design of a typical WHO-EPI network with the goal of minimizing costs while providing the opportunity for universal coverage and developed a column generation algorithm to solve the formulation. We first explore a column generation algorithm to solve the **MIP-1** formulation, but its performance is not very good and it has no real advantage over directly solving the **MIP-1** via off-the-shelf solver. We then present another algorithm that generates high quality solutions within a fraction of one percent of optimality while using a substantially smaller amount of computation time than direct solution. Due to the ease with which it can be implemented, it can serve as a simple alternative to solving **MIP-1** directly when a good solution is required quickly. However, this method – like direct solution – cannot handle country-level problem formulations. Thus it is clear that we need another heuristic approach that can generate high quality solutions for these larger problems, and we will discuss this in the following chapter.

3.0 A Disaggregation-and-merging Approach to Solve the Network Design Problem

In this chapter we develop a novel algorithm that solves a sequence of increasingly larger MIP problems. To maintain tractability, the approach uses insights into the problem structure and principles from cluster analysis to limit the size of each MIP in the sequence. In order to study the performance of the algorithm, numerical tests using data derived from several different countries in sub-Saharan Africa are conducted. Comparisons with the optimal solution when one is available indicate that the algorithm works very well with solution times that scale up in a roughly linear fashion. The remainder of this chapter appears in (Yang, Bidkhori, & Rajgopal, 2020).

3.1 Problem Development

In Chapter 2 we developed a mixed integer programming (MIP) model **MIP-1** to optimize the design of the distribution network. The model allows for a more appropriate design than current networks, while following the WHO guidelines for operational simplicity. As discussed in Section 2.2.1, to explore the solution of the model described by **MIP-1** we tested it with a standard commercial solver using data derived from the EPI networks in four different countries in sub-Saharan Africa; specifics on the data, as well as the hardware and software used are discussed in Section 3.4, where we describe our numerical experiments in full detail.

As explored in Section 2.2.1, a key fact that makes MIP-1 hard to solve is that the model has a large number of 0-1 decision variables. For example, if we can choose from three types of transport vehicles and four types of storage devices (i.e., {T|=3, |R|=4), recall that in Section 2.2.1,

Table 1 illustrates the number of decision variables in MIP-1. Thus, in order to solve the full problem for one our instances with 685 nodes and 41 candidate hubs, we end up with 168,838 integer variables. Even for a typical 100-node, mid-sized problem with 15 candidate hubs, the number of binary decision variables is close to 10,000.

As the first approach to address this issue, we proposed a heuristic cyclic approach in Section 2.4. Unfortunately, this heuristic can only handle problems with a maximum of roughly 200 nodes. As the problem size increases it starts to slow down dramatically. More importantly, for most large problems with more hub candidates, there are too many combinations of hub locations and hub-to-hub connections for the algorithm to handle. Furthermore, because information on replenishment frequency is available only for open hubs, for some LMICs where there are a large number of candidate hubs but only a small number of these are open in the optimal solution, the heuristic fails to fix the frequency at the next step for a large number of locations, and starts to slow down. In short, the resulting formulation for a national network is too large to solve optimally using standard commercial software or even with the heuristic cyclic approach of the previous chapter. A more powerful algorithm is required to handle large, country-level problem formulations.

To further explore the limits of the problem size that could be solved using a standard solver we also experimented with subsets of the data from each country. That is, we considered successively larger problems: first, with the national center along with a single region (based on how a region is currently defined in the country), then problems with a combination of two regions, three regions, etc. In general, the difficulty associated with a particular problem depends on several factors including the total number of nodes and potential hubs in the problem, as well as the population distribution, transportation cost, and storage cost across the network. Unfortunately, despite extensive computational experimentation it was impossible to pinpoint the limiting characteristics of a tractable problem or establish any clear monotonicity, because of the interrelationships between the problem parameters. Our numerical tests are discussed in more detail in Section 3.4, but as a general rule of thumb, we found that most problems with over 200–250 nodes and over 15–20 potential hubs are impossible to solve directly. Given that in the network for an entire country these limits are almost always exceeded, there is clearly a need for good algorithm if one aims to design an optimal network for the country.

In the next few sections, we propose a sequential MIP-based disaggregation-and-merging algorithm that divides the problem on the entire graph for the distribution network into several subproblems on smaller subgraphs that can be solved with relative ease. The algorithm then intelligently merges the subgraphs together sequentially to obtain a solution to the whole graph. We present numerical comparisons in Section 3.4 and show that the algorithm is able to yield good solutions for even the largest problems.

3.2 Motivation

Our algorithm is motivated by the observation that in a large network, the optimal subnetwork structures in regions that are relatively far apart will tend to be independent of each other. For example, the characteristics of a local clinic are unlikely to have any significant influence on the network structure in locations that are far away, and if a hub is added or removed at some distant location it is unlikely to affect the clinic's supply. The same is also true of a hub that is distant from some other hub whose disposition is changed. Therefore, for larger problems we propose a divide-and-conquer approach where we first divide the whole network into portions that each yield smaller problems that can be solved independently with relative ease. We then systematically merge these smaller problems and solve a sequence of increasingly larger problems. Each of these is formulated using MIP-1, but with the key difference that parts of the structure are fixed based on the optimal solutions to the smaller problems as well as the spatial relationship between the current subnetwork and the new portion being added on. To clarify our approach, we first provide an overview of the method and then provide all of the details.

We start by dividing the entire network into P smaller subnetworks. One could always use existing political boundaries as a natural disaggregation of the network, i.e., each region (or province, or state) of the country is an "independent" network; larger existing regions could be split into smaller ones. Alternatively, we could apply a clustering algorithm to determine the subnetwork nodes. Although the average cluster size will be smaller as we form more clusters, the number of clusters that would give us problems that are small enough to yield a tractable version of MIP-1 is problem dependent, so that it is difficult to prescribe a general value for P in advance. We therefore chose to use *hierarchical clustering* rather than a simpler method such as *K-means clustering*, whereby we can continue to disaggregate the network until each region is small enough for a standard MIP solver to handle; the interested reader is referred to (Aggarwal, 2015) for more details on hierarchical clustering.

Once the independent regions are created we start with the one that contains the national center and optimize its structure via MIP-1 to obtain an initial subnetwork. We now pick a neighboring region to merge with this subnetwork, formulate MIP-1 for the combined set of nodes and solve this (larger) consolidated problem to get a new subnetwork structure with both regions. This process continues until all of the independent regions have been merged into our network. While we will specify details on how each step is executed, the critical thing to note is that at each successive iteration we handle a larger collection of nodes, and therefore have to solve a larger problem. Clearly, the effort required at each stage has to be reasonable; otherwise we are defeating the purpose of the original disaggregation! To ensure that this is the case, we refer to our initial observations on the motivation for this approach, and at each iteration we <u>fix</u> a portion of the current subnetwork, so that we are only using variables associated with a subset of all the nodes corresponding to the current iteration's network. This is done by retaining the locally optimal structure for portions of the subnetwork while allowing for changes in others. In addition, we also use a "shrinking" scheme whereby some of the nodes are aggregated and replaced by a single dummy node so as to further reduce the size of the problem being solved.

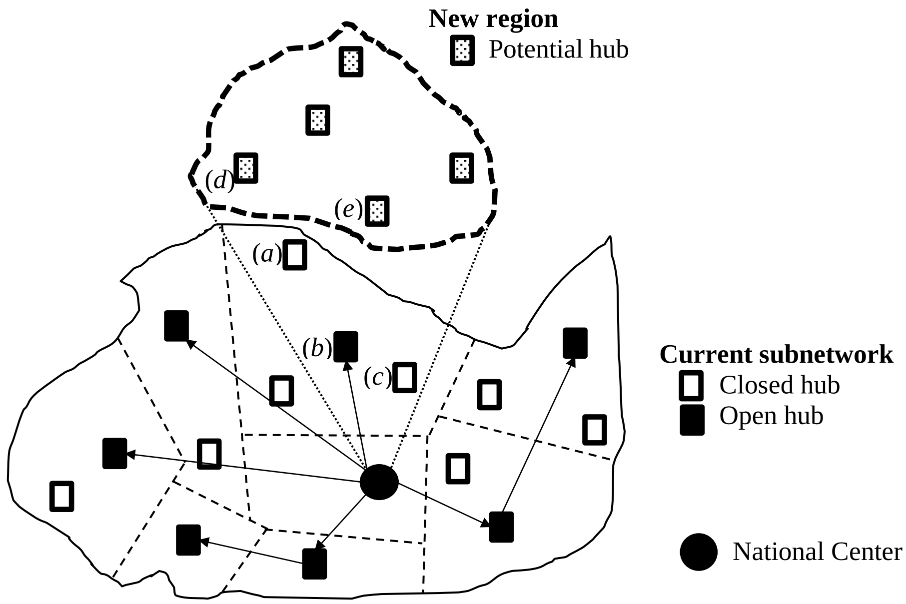

Figure 2 Hub classification during consolidation

To decide upon how to fix parts of the structure at each iteration, we classify all hubs for the current network into three categories. We overview these categories here using the example shown in Figure 2; more mathematically precise definitions are provided when the algorithm is detailed. The lower part of Figure 2 shows the subnetwork we currently have. The seven solid squares represent open hubs while the empty squares denote potential hub locations that were not selected for opening. The dashed lines within the area covered by this subnetwork divide it into eight discrete sections corresponding to the national center and the seven open hubs, and clinics within each section (not shown) are supplied by the corresponding open hub (or national center). Note that some open hubs are supplied directly by the national center while others receive their vaccines from some other open hub. The upper portion of Figure 2 represents the new, neighboring region that we wish to merge with the current subnetwork, along with five potential hub locations within it.

- The first category, which we call *Critical Hubs*, are hub locations "close to" the boundary between the current subnetwork and the new region being merged. The dispositions of these hubs are likelier to change after merger; currently open hubs might close and vice-versa. Clinic assignments to a hub might also change as new hubs might be introduced at geographically proximal locations. In our illustration, nodes (*a*), (*d*) and (*e*) might be critical hubs.
- The second category, which we call *Intermediate Hubs*, are hub locations in the existing subnetwork that are in some sense "in between" the national center and hubs in the new region that is being merged with the existing subnetwork (e.g., hubs labeled (*a*), (*b*) and (*c*) in Figure 2). Since they are *en route* from the center to a possible hub in the new region, such hubs could potentially serve as intermediate transshipment points (while continuing to serve their current clinics if they are currently open). Thus their storage requirements could be larger after merging and/or their replenishment frequencies could possibly change.
- The third category of hub locations, which we call *Non-critical Hubs* can be considered as "independent" during merger. These are locations that are not near the common boundaries or

en route to a potential new hub and we fix their dispositions (open or closed); for the open hubs, their clinic assignments, hub-to-hub connections, storage devices, transportation routes and frequencies are also fixed. In Figure 2 all non-labeled nodes might be non-critical hubs.

We now overview the process to sequence regions for merger. To start with, after the initial set of regions is formed, we solve a sub-problem for each of these regions by applying MIP-1 just to the nodes in that region along with the national center. This yields a locally optimal structure for each region. At each iteration, the region we choose for merger is the one with minimum cluster distance between it and other regions that has already been merged into the consolidated network. Here, we define the cluster distance as the minimum distance between two points that are in different clusters; based on our computational experiments we found that this worked best among the common measures of cluster distance.

3.3 A Disaggregation-and-merging Algorithm

We now formally outline the steps in our algorithm in this section.

Algorithm 3:

STEP 1: *Disaggregation*

Consider the directed graph G of nodes indexed in V and arcs indexed in A. Divide the set of potential hubs H into P mutually exclusive subsets H_1,..., H_p..., H_P using a clustering algorithm or heuristically. If using a clustering algorithm, to determine whether a cluster of potential hubs indexed in H_p is small enough, define set C_p to be the set of clinics whose closest hub nodes are in H_p and define set $Q_p = H_p \cup C_p \cup \{0\}$ as the complete node set of the region defined by hub nodes in cluster H_p. While there is no obvious way to prescribe the value of P in advance, it is important for the number of nodes in each set Q_p to be small enough that MIP-1 on the subgraph generated by nodes in this set is readily solved. Therefore, define a suitable number M (we suggest a value under 200 based upon our computational experiments) and check whether $|Q_p| \leq M$ for all p; if not, further divide the corresponding cluster H_p, again by using existing political boundaries (such as districts within the region), a clustering algorithm, or heuristically based on the distribution of nodes within the cluster. Continue until the number of nodes in each of H_1, H_2, ..., H_P is no larger than M. Note that depending on the demographic characteristics, the number of nodes in each cluster H_p could be quite different.

Define H' as the set of potential hubs that are currently *not* merged into the consolidated network; thus, at the end of **STEP 1**, $H' = H = \{H_1, H_2, ..., H_P\}$.

STEP 2: *Sub-problem solution and initialization*

Consider the subgraph $G[Q_p]$ that is induced by nodes in Q_p. For each $p \in \{1,2,\dots P\}$ formulate and solve MIP-1 on subgraph $G[Q_p]$ to meet demand optimally at all clinic locations in set Q_p.

Denoting the cluster distance between clusters H_p and H_q (i.e., distance between the nearest pair of nodes i and j, where $i \in H_p$ and $j \in H_q$) as $D(H_p, H_q)$, compute $p^* \in Argmin_{p \in 1,2,\dots,P}\, D(H_p, N)$, where cluster $N = \{0\}$ is an artificial cluster with just the national center in it. Thus cluster H_p has the smallest cluster distance to $\{0\}$.

To start the iterative process set $k = 0$, define $I^0 = Q_{p^*} = H_p \cup C_p \cup \{0\}$ as the index set of all nodes in the initial subnetwork, with corresponding subgraph $G^0 = G[I^0]$. Update $H' \leftarrow H' \backslash \{H_{p^*}\}$.

STEP 3: *Subset Selection*

Set $k = k + 1$ and compute $p^* \in Argmin_{p|H_p \in H'}$ $\{D(H_q, H_p)|H_q \notin H'\}$

Here q and p correspond respectively, to clusters that have and have not yet been merged into the consolidated network, and among the clusters not yet merged p^* has the smallest cluster distance to a cluster that has already been merged. Define $I^k = H_{p^*} \cup C_{p^*} \cup I^{k-1}$ as the complete node set for the consolidated network at iteration k. Define the graph $G^k = G[I^k]$ and update $H' \leftarrow H' \backslash \{H_{p^*}\}$.

STEP 4: *Classification*

Compute $d_{max} = Max_{i,j \in H_{p^*}} d_{ij}$. That is, d_{max} is the largest distance between any pair of hub locations within the new cluster of potential hubs that was just merged. Let $Conv_{p^*}$ be the convex hull of $\{0\}$ and all hub nodes in H_{p^*}: $Conv_{p^*} = Conv(H_{p^*} \cup \{0\})$, and define a positive real number $\alpha \in (0,1)$. Classify the hubs in I^k into three categories as follow:

a. Critical Hubs (H^C): Identify all pairs of nodes (i, j), such that $i \in I^{k-1} \cap H$ and $j \in H_{p^*}$, with $d_{ij} < \alpha d_{max}$, and define i and j as critical hubs. That is, we consider all potential hub pairs with one from the previous consolidated set of regions and one from the new region, and define the two as being critical if they are separated by less than some fraction of the maximum distance between two hub locations in the region being merged. Larger values for α result in more hubs being identified as critical so that the structure of the consolidated network is more flexible, but the associated model formulation is also more difficult to solve. Conversely, when α is smaller, the consolidated problem is easier to solve but a larger portion of the network is fixed. Based upon our computational experiments we suggest a value for α between 0.1 and 0.3.

b. Intermediate Hubs (H^I): Define i as an intermediate hub if $i \in (I^{k-1} \cap H) \ni i \notin H^C, i \in Conv_{p^*}$. That is, these are hub locations (open or closed) in the previous set of consolidated regions that also lie within the convex cone containing the national center and all potential hubs in the new region to be merged.

c. Non-critical Hubs (H^N): Defined as hubs in $I^k \cap H$ that do not belong to H^C or H^I.

STEP 5: *Reduced form of MIP-1*

In this step we add constraints to MIP-1 based upon our classification of hubs:

a. Critical Hubs (H^C): Since the disposition of such a hub is more likely to change during consolidation, we impose no further restrictions on these.

b. Intermediate Hubs (H^I): For every intermediate hub, add constraints that maintain the same clinic assignments that it had in the current solution (if it was open), i.e., for $j \in H^I$ add:

$$\sum_{r \in R} \sum_{f \in F} W_{jrf} = \sum_{r \in R} \sum_{f \in F} w^*_{jrf} \qquad j \in H^I \qquad (2\text{-}28)$$

and for all $i \in C$, $t \in T$

$$X_{ji} = x^*_{ji} \qquad i \in C, j \in H^I \qquad (2\text{-}29)$$

$$U_{jit1} = u^*_{jit1} \qquad i \in C, j \in H^I, t \in T \qquad (2\text{-}30)$$

where u^*_{jitf}, x^*_{ji}, and w^*_{jrf} are from the solution to the MIP defined on $G[I^{k-1}]$. Here (2-28) ensures that open intermediate hubs remain open and closed ones remain closed. Note that since such a hub can potentially supply other hubs, the required capacity of its own storage device and of the inbound transport device might increase, and the replenishment frequency at the hub might also change. Constraints (2-29) and (2-30) ensure the same flow into a clinic with the same device and replenishment frequency.

c. Non-critical Hubs (H^N): Add constraints for each open hub $j \in H^N$ to fix replenishment frequency, storage device, inbound and outbound volumes and vehicle types, and clinic assignment to be the same as they are in the solution to (*i*) the MIP defined on $G[I^{k-1}]$ if $j \in I^{k-1} \cap H$, or (*ii*) MIP-1 defined on $G\left[Q_{j^*}\right]$ if $j \in H_{j^*}$ (in **STEP 2**) and ensure that closed hubs remain closed. That is, for $j \in H^N$, $i \in C$, $l \in N \cup H$, $r \in R$, $t \in T$, $f \in F$, add:

$$X_{ji} = x^*_{ji} \qquad j \in H^N, i \in C \qquad (2\text{-}31)$$

$$U_{jit1} = u^*_{jit1} \qquad j \in H^N, i \in C, t \in T \qquad (2\text{-}32)$$

$$W_{jrf} = w^*_{jrf} \qquad j \in H^N, r \in R, f \in F \qquad (2\text{-}33)$$

$$X_{lj} = x^*_{lj} \qquad l \in N \cup H, j \in H^N \qquad (2\text{-}34)$$

$$U_{ljtf} = u^*_{ljtf} \qquad l \in N \cup H, j \in H^N, t \in T, f \in F \qquad (2\text{-}35)$$

where w^*_{jrf}, x^*_{ji}, u^*_{jitf}, x^*_{lj}, u^*_{ljtf} are values obtained from the solution to the MIP defined on $G[I^{k-1}]$ if $j \in I^{k-1} \cap H$, or the MIP on $G\left[Q_{j^*}\right]$ if $j \in H_{j^*}$.

Note that (2-33) maintains the open/closed status of a hub, (2-31) and (2-32) maintain the same flow into and the same transport device and replenishment frequency for each clinic served by a hub, while (2-34) and (2-35) do the same for the inbound flow into the hub (note that this last feature is different than with intermediate hubs).

STEP 6: *Consolidation*

With the constraints added in **STEP 5**, solve the MIP defined on subgraph $G[I^k]$. If $H' = \phi$, we have merged all hubs; stop and return the solution. Otherwise, delete all new constraints added in **STEP 5,** return to **STEP 3** to add a new region, and repeat the process at the next iteration.

3.3.1 A Refinement to Algorithm 3

In **STEP 5**, there could potentially be hundreds of constraints added at each iteration. We can further manipulate the formulation at this step to obtain the same outcome but with fewer nodes in the graph. Instead of directly formulating the MIP on $G[I^k]$ with the constraints added in **STEP 5**, we could use information previously obtained from the solutions to problems defined on $G[I^{k-1}]$ (in **STEP 6** at the previous iteration) and $G[Q_{p^*}]$ (in **STEP 2**) in order to restrict the problem size. Consider an open intermediate or non-critical hub j that will be restricted to remain open at the next iteration along with the same clinic assignments. To reduce the number of nodes (and hence, the number of binary variables) we could collapse all clinics associated with the hub into a single dummy clinic m with a demand equal to the sum of the demands at these clinics, locate it at the same location as the hub (so that $d_{jm} = 0$) and assign it to hub j. This ensures that the outflows to clinics served by j are the same, so that the solution to the new problem will be the same as the one to the MIP on $G[I^k]$. The only difference is that in the modified problem the total transportation cost to the clinics served by j is zero; however, we can simply add the true cost to the final value obtained by the new MIP.

More formally, consider a hub $j \in H^I \cup H^N$ that is open in the solution to the MIP on $G[I^{k-1}]$ or MIP-1 on $G[Q_{p^*}]$. Define a dummy clinic m with demand D_j^T equal to the total demand across all clinics served by hub j in this solution.

$$D_j^T = \sum_{i \in c} \sum_{t \in T} b_i u_{jit1}^* \qquad (2\text{-}36)$$

Also, define the new index set C^- by removing from set C the indices of all of the clinics serviced by hub j. Then we have the following proposition.

Proposition 1: Given that hub $j \in H^I \cup H^N$ is open in the solution to $G[I^{k-1}]$ or $G[Q_{p^*}]$, **MIP-1** with the additional Constraints (2-29) and (2-30) for all $i \in C$, $t \in T$ is equivalent to **MIP-1** with the following three additional constraints:

$$X_{jm} = D_j^T \qquad (2\text{-}37)$$

$$\sum_{t \in T} U_{jmt1} = 1 \qquad t \in T \qquad (2\text{-}38)$$

$$\sum_{l \in C^-} \sum_{t \in T} U_{jlt1} = 0 \qquad (2\text{-}39)$$

Proof:

Since hub j is open in the solution to $G[I^{k-1}]$ or $G[Q_{j^*}]$, we have :

$$\sum_{r \in R} \sum_{f \in F} W_{jrf} = 1. \tag{2-40}$$

First, suppose that for this j and all $i \in C$, $t \in T$ we add the constraints given by (2-29) and (2-30). This is equivalent to partially fixing the network structure. Specifically, we fix the clinic assignments for hub j. With Constraint (2-7) we have for all $i \in C$ that are served by hub j:

$$X_{ji} = x_{ji}^{*} = b_i \tag{2-41}$$

Therefore, the total volume that goes out of hub j and goes into all of its clinics is also fixed:

$$\sum_{i \in C} X_{ji} = \sum_{i \in C} X_{ji} \sum_{t \in T} U_{jit1} = \sum_{i \in C} \sum_{t \in T} X_{ji} U_{jit1} = \sum_{i \in C} \sum_{t \in T} b_i u_{jit1}^{*} = D_j^T. \tag{2-42}$$

Note that the second and third equalities above hold because of Constraint (2-2).

Now, suppose instead that we delete all clinics assigned to j to obtain the index set C^{-} and add a dummy clinic m with demand given by (2-36), then add Constraints (2-37), (2-38) and (2-39) to **MIP-1**. From these constraints and Constraint (2-6) of **MIP-1**, once again the total volume that goes out of hub j is fixed, i.e., we have:

$$\sum_{l \in C} X_{jl} = X_{jm} = D_j^T \tag{2-43}$$

Since we are not altering any other constraints, this is equivalent to the first case with fixed clinic assignments and the only difference is that the same total outflow is sent to a single clinic. Thus the optimal solutions with either approach are identical.

□

By using **Proposition 1**, for every open intermediate or non-critical hub, we could replace the $\{|C| + |C| \times |T|\}$ constraints in (2-29) and (2-30) with the 3 Constraints (2-37), (2-38) and (2-39). Similarly, we could replace $\{|T| \times |C|\}$ binary variables associated with selecting devices used to send vaccines from the hub to all of its clinics, with $|T|$ binary variable associated with the dummy clinic at the hub and $|T| \times |C^-|$ binary variables associated with each of the clinics not consolidated into the dummy. This results in a reduction of $\{|T| \times (|C| - |C^-| - 1)\}$ in the number of binary variables. As a direct consequence of **Proposition 1**, we have the following:

Proposition 2: For any hub $j \in H^I$ that is open in the solution to $G[I^{k-1}]$, MIP-1 with the constraints added in **STEP 5(b)** is equivalent to MIP-1 with constraints given by (2-37), (2-38) and (2-39) along with (2-28).

Proposition 3: For any hub $j \in H^N$ that is open in the solution to $G[I^{k-1}]$ or $G[Q_{p^*}]$, MIP-1 with the constraints added in **STEP 5(c)** is equivalent to MIP-1 with constraints given by (2-37), (2-38) and (2-39), along with the Constraints (2-33), (2-34) and (2-35) for $l \in N \cup H$, $r \in R$, $t \in T$, $f \in F$.

Note that if we use **Propositions 2** and **Propositions 3** to solve the modified formulation (as opposed to the one in **STEP 5**), we will need to add to the final objective value the following outbound transportation cost (C_j^P) for each hub j that has its clinics consolidated as above:

$$C_j^P = \sum_{i \in C} \sum_{t \in T} 2 c_{jit}^P g_1 d_{ji} u_{jit1}^* \tag{2-44}$$

Based on the preceding discussion, we have the following refined **Algorithm 3***:

Algorithm 3*:

STEPS 1* to **4*:** Identical to **STEPS 1** to **4** in **Algorithm 3**.

STEP 5*: Shrinkage

As in **STEP 5**, we first formulate MIP-1 for $G[I^k]$ but then add constraints and operations based on the category of the hub as follows:

a. Critical Hubs (H^C): No additional action or restrictions.

b. Intermediate Hubs (H^I): If hub $j \in H^I$ is open in the solution to $G[I^{k-1}]$, delete all clinics assigned to that hub, add a dummy clinic m with demand D_j^T computed via (2-36), and add the constraints given by (2-28), (2-37), (2-38) and (2-39) for that hub.

   On the other hand, if hub j is closed in the corresponding solution, add constraints to keep the hub closed:

$$\sum_{r \in R} \sum_{f \in F} W_{jrf} = 0 \tag{2-45}$$

c. Non-critical Hubs (H^N): if a hub $j \in H^N$ is open in the solution to the problem on $G[I^{k-1}]$ or $G[Q_{p^*}]$, delete all clinics assigned to that hub, add a dummy clinic m with demand D_j^T computed via (36), and add the constraints given by (2-33), (2-34), (2-35), (2-37), (2-38) and (2-39) for that hub.

   If hub j is closed, add constraints to keep the hub closed:

$$\sum_{r \in R} \sum_{f \in F} W_{jrf} = 0 \tag{2-46}$$

STEP 6*: Consolidation

Identical to **STEP 6** except that we use (2-25) to add the total cost of additional transportation (C^*_{Total}) to the optimal value to account for the clinics deleted and consolidated in **STEP 5***:

$$C^*_{Total} = \sum_{j \in H^I} C^P_j + \sum_{j \in H^N} C^P_j \tag{2-47}$$

3.4 Numerical Experiments

We tested **Algorithm 3*** as well as a standard commercial solver on a suite of 43 different problems that are derived from the EPI networks in four different countries in sub-Saharan Africa. Due to data confidentiality issues, we denote these as Countries A, B, C and D. Several geographic and demographic characteristics of these four countries are shown in Table 8; for areas and population densities we have normalized the largest values to 1.0 and expressed the other values as respective fractions of these. As one can observe, there are significant differences in these. Countries A and B are relatively large but the population densities are relatively low. Most of the population in Country A is concentrated in a few regions with the remainder being sparsely distributed over the rest of the country in remote desert areas). In contrast, Countries C and D are smaller in area but densely populated and have many more existing vaccination facilities per km^2 of land.

Table 8 Characteristics

Country	A	B	C	D
Area (10^3 km^2)	0.99	1.00	0.09	0.45
Population density (United Nations, 2017) (per km^2)	0.17	0.12	1.00	0.88
No. of potential hubs (h)	41	60	87	141
Total no. of nodes (n+1)	685	933	746	2875

We summarize detailed information on facilities, storage, transportation devices and vaccines in Tables 9 through 12. Note that each country might have different transportation and storage devices to choose from and there can be significant differences in costs as well. For the problem instances that we considered, it happened that $|T|$=3 and $|R|$=4, although there were differences in the specific transport/storage device options in each country as shown in Tables 10 and 11. There are also minor differences in the EPI vaccine regimens within the countries. To obtain the total demand volume at each clinic we first estimate the number of newborns it must serve by multiplying the estimated population in the area served by the clinic and the corresponding national birth rate published by the World Bank. This is then multiplied by the number of required doses and the volume of each dose for each vaccine in the regimen, and adjusted upward to account for anticipated open-vial waste. Finally, the volumes are added across all vaccines.

Table 9 Facility cost ($/year)

Facility type	Country A	Country B	Country C	Country D
National	40,000	14,870	52,500	158,191
Hub	4,500	450	2,389	20,992
Clinic	800	150	140	1,825

Table 10 Storage details

Country	Device Type	Capacity (L)	Cost ($/year)
A	Cold Room 1	18,000	8,116
	Cold Room 2	1,200	1,200
	Refrigerator 1	700	900
	Refrigerator 2	340	610
B	Cold Room 1	5,000	17,534
	Cold Room 2	1,500	1,800
	Refrigerator 1	700	900
	Refrigerator 2	504	624
C	Cold Room	1,500	1,500
	Refrigerator 1	340	650
	Refrigerator 2	200	550
	Refrigerator 3	53	462
D	Cold Room	5,000	17,534
	Refrigerator 1	504	624
	Refrigerator 2	340	510
	Refrigerator 3	84	394

Table 11 Transportation details

Country	Vehicle Type	Capacity (L)	Cost ($/km)
A	Cold truck	9,293	0.97
	4×4 truck	172	0.54
	Motorbike	5	0.23
B	Cold truck	9,500	0.78
	4×4 truck	308.44	0.51
	Motorbike	3.4	0.1
C	Truck 1	331.2	1.4
	Truck 2	110.4	0.4667
	Motorbike	3	0.13
D	Cold truck	15,000	1.12
	4×4 truck	82.8	0.38
	Motorbike	3	0.12

Table 12 Vaccine details

Country	Name	Pharmaceutical Form	Open-vial waste	Dose Volume (cc)	Required Dosage
A	Tuberculosis	Lyophilized	0.5	1.2	1
	Tetanus Toxoid	Liquid	0	3	3
	Measles	Lyophilized	0.4	2.1	1
	Oral Polio	Liquid	0	1	4
	Yellow Fever	Lyophilized	0.4	2.5	1
	DTC-HepB-Hib	Liquid	0	16.8	3
	Rotavirus	Liquid	0	45.9	3
	PCV13	Liquid	0	12	3
B	Tuberculosis	Lyophilized	0.5	1.2	1
	Tetanus Toxoid	Liquid	0.15	3	2
	Measles	Lyophilized	0.45	3.5	1
	Oral Polio	Liquid	0.17	1	4
	Yellow Fever	Lyophilized	0.45	2.5	1
	DTC-HepB-Hib	Liquid	0.1	11	3
	Rotavirus	Liquid	0	17.1	2
	PCV13	Liquid	0.05	12	3
C	Tetanus Toxoid	Liquid	0.05	3	2
	Measles	Lyophilized	0.45	3.5	1
	Oral Polio	Liquid	0.17	1	4
	Yellow Fever	Lyophilized	0.1	2.5	1
	BCG	Lyophilized	0.5	1.2	1
	Pentavalent	Liquid	0.15	5.3	3
	Pentavalent	Liquid	0.45	12.9	3
D	Tetanus Toxoid	Liquid	0.1	2.5	2
	Measles	Lyophilized	0.45	3.5	1
	Oral Polio	Liquid	0.17	2	4
	Yellow Fever	Lyophilized	0.05	2.46	1
	PCV10	Liquid	0.45	4.8	3
	BCG	Lyophilized	0.5	1.2	1
	Pentavalent	Liquid	0.15	9.7	3

We tested the algorithm using a computer with an Intel Core i5-6500 CPU and 3.20 GHz processor with 8.0 GB memory. For solving **MIP-1** directly we used IBM ILOG CPLEX 12.6. Since none of the complete problems for any of the four countries could be directly solved, we started with smaller subproblems and worked on successively larger ones based on how a region is currently defined in the country. Detailed results for each of the four countries studied may be found in Tables 13 through 16. Each entry in a table corresponds to a problem over a region, part of a region, or a set of regions in the corresponding country. The last row in each table denotes the problem with the full set of nodes across all regions of the country. We summarize the number of potential hubs, the total number of nodes, total number of binary variables and a qualitative characterization of the population density associated with each instance, in order to illustrate the diversity of the problems that we formulated. To further interpret the results we also subjectively label our test problems as “large” or “small” using a cutoff of 20,000 binary variables. For the problems whose optimal solutions could be obtained via CPLEX, we report the percentage gap between the optimum cost and the objective value for the solution returned by **Algorithm 3***. For a design problem such as this, computational times are obviously less relevant than being able to solve the problem; nevertheless, as a matter of record we also list the solution times for **MIP-1** using the commercial software (when an optimum solution is available) and for the solution found by **Algorithm 3***.

Table 13 Computational results for Country A

No.	Hubs	Nodes	Binary Variables	Population Density	Size Label	Gap	CPU Times	
							MIP-1	**Algorithm 3***
1	1	10	68	sparse	small	0%	<1s	<1s
2	2	49	604	sparse	small	0%	<1s	<1s
3	4	48	1,184	moderate	small	0%	<1s	<1s
4	5	77	2,350	moderate	small	0%	1.6s	<1s
5	7	99	4,214	dense	small	0%	4.4s	2s
6	8	206	9,952	dense	small	0%	~10h	15s
7	14	148	12,544	moderate	small	0%	103s	14s
8	13	210	16,484	moderate	small	0%	~1d	24s
9	14	235	19,852	dense	small	0%	~2d	25s
10	19	176	20,216	moderate	*large*	0%	4,649s	16s
11	20	384	46,240	dense	*large*	0.29%	~1.1w	51s
12	33	599	118,866	moderate	*large*	-	-	81s
Full	41	685	168,838	moderate	*large*	-	-	127s

Table 14 Computational results for Country B

No.	Hubs	Nodes	Binary Variables	Size Label	Population Density	Gap	CPU Times	
							MIP-1	**Algorithm 3***
1	8	56	2,752	small	sparse	0%	16s	1s
2	14	101	8,596	small	sparse	0%	119s	12s
3	12	128	9,312	small	dense	0%	116s	13s
4	16	162	15,680	small	dense	0%	1,304s	15s
5	28	271	45,752	*large*	dense+moderate	-	-	41s
6	41	372	91,840	*large*	moderate	-	-	64s
7	57	510	174,876	*large*	moderate	-	-	79s
8	65	591	231,010	*large*	moderate+sparse	-	-	104s
Full	87	746	390,108	*large*	moderate+sparse	-	-	191s

Table 15 Computational results for Country C

No.	Hubs	Nodes	Binary Variables	Size Label	Population Density	Gap	CPU Times	
							MIP-1	Algorithm 3*
1	1	18	116	small	moderate	0%	<1s	<1s
2	2	11	148	small	sparse	0%	<1s	<1s
3	3	22	420	small	sparse	0%	2s	<1s
4	3	39	726	small	moderate	0%	2s	<1s
5	4	44	1,088	small	sparse	0%	3s	<1s
6	4	55	1,352	small	moderate	0%	4s	<1s
7	4	64	1,568	small	moderate	0%	7s	<1s
8	4	65	1,592	small	dense	0%	8s	<1s
9	11	96	6,424	small	moderate	0.07%	10s	<1s
10	17	141	14,518	small	moderate	0.15%	79s	2s
11	20	295	35,560	*large*	moderate	0.42%	387s	3s
12	26	333	52,156	*large*	sparse+moderate	0.69%	2,748s	37s
13	34	601	122,876	*large*	moderate+dense	-	-	218s
Full	60	933	336,360	*large*	sparse+dense	-	-	476s

Table 16 Computational results for Country D

No.	Hubs	Nodes	Binary Variables	Size Label	Population Density	Gap	CPU Times	
							MIP-1	Algorithm 3*
1	10	117	7,100	small	moderate	0.25%	146s	6s
2	14	270	22,792	*large*	dense	-	-	19s
3	11	366	24,244	*large*	dense	-	-	23s
4	27	540	87,696	*large*	dense	-	-	72s
5	38	906	206,872	*large*	dense	-	-	168s
6	83	1,718	856,228	*large*	dense+moderate	-	-	453s
Full	141	2,875	2,433,378	*large*	moderate	-	-	713s

As the tables show, the number of nodes, potential hubs, and binary variables in the largest problem that CPLEX could solve directly vary with each country. While problem size is certainly a factor, the geographical and population characteristics of the underlying network for a problem also play a role in determining whether it can be solved optimally. Based on extensive testing, it is our conclusion that (with a few exceptions) direct solution of MIP-1 using commercial software is feasible only in problems with fewer than approximately 200 nodes and 15 potential hubs, which is much smaller than the full network for almost all countries.

While the results indicate that the ability (and the effort) required to solve a problem optimally depends on the combination of factors listed in the tables, there is no systematic relationship with any one specific factor that could be established. However, as might be expected, the total number of binary variables seems significant. Based upon our characterization of the test problems as “large” or “small” (using a cutoff of 20,000 binary variables), we have a total of 24 small and 19 large instances (including the full problem for each country) in our test set; the distribution of these labels for each country was different and depended on the specific characteristics of that country. The results show that while all the small instances could be solved optimally, **MIP-1** corresponding to 15 of the 19 large instances could not be solved to optimality. In particular, countries C and D which have denser populations with more nodes per unit area proved harder to handle. Even for the instances that could be solved to optimality, the required effort can be inconsistent. For example, there are a couple of small instances (instances 8, 9 for Country A) that took a long time to solve, and while three of the four large instances that could be solved yielded solutions in reasonable times, one (instance 11, Country A) took over a week to solve (which incidentally, was also the largest problem that we were able to solve to optimality). Also, in the case of the problems that could not be solved, there was no pattern to the integrality gap when the solver failed.

On the other hand, the disaggregation-and-merging approach of **Algorithm 3*** was robust and able to generate solutions for every problem that we formulated (and in well under about 5 minutes in almost all cases; even the largest problem that we tested, with over 2 million variables, took only approximately 12 minutes). In the 28 instances where the optimal solution was available for comparison, Tables 13-16 show that **Algorithm 3*** also converged to the optimal solution in 22 instances while finding a solution with a cost within 0.5% of the optimum value for five of the six remaining instances; our cost for the largest problem that could be solved optimally was 0.69% higher than the optimal cost. Moreover, even though the demographic characteristics also have an effect, Figure 3 shows that the computational effort appears to be approximately linear in the number of binary variables (the data point for the full Country D problem with over 2 million binary variables is omitted in the graph to maintain a better visual scale; the effort for this problem is actually proportionately smaller).

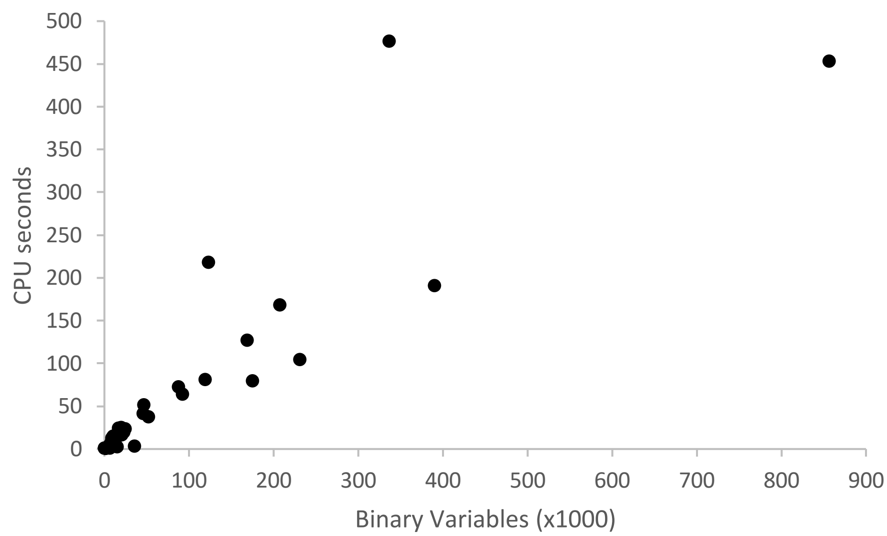

Figure 3 Computational times with Algorithm 3*

With the 15 problems for which the optimal solutions to **MIP-1** are not available there was no way to compute the cost difference between the solution from **Algorithm 3*** and the optimum value. We could visually verify that the network structures generated were reasonable in all instances, and while there is no guarantee that they are optimal, they are certainly better than anything that could be derived by inspection or *ad hoc* methods. One comparison that we can do is to compare the cost of the solution generated by our algorithm (the last row in each of Tables 13-16) against the current cost for the entire country, since this is an existing network. To compute the country-wide costs for the current structure, we used the same unit costs for facilities, transport and storage as those used in our numerical tests. The results are shown in Table 17. Note that the total costs in both cases include identical operational costs for the national center as well as the clinics. The difference is that for the existing structure we have operational costs for regional and district centers and transportation costs from the national center to regions, regions to districts, and districts to clinics; while in our case, we have operational costs for the selected hubs and transport costs into the hubs and from the hubs to clinics (as captured by cost expression (2-1) in the formulation). Also note that we do not consider all the costs associated with the existing network in the comparison; rather we compare only the storage and transport costs currently incurred for EPI vaccines, with the corresponding costs in the redesigned network. The results indicate that even though we do not have a guarantee of optimality our distribution network produces overall savings ranging from approximately 6% to 27% for the four countries studied.

Table 17 Network cost for Country A, B, C, D

Country	A	B	C	D
Original Cost ($)	2,453,690	791,164	5,239,822	8,674,722
Optimized Cost ($)	1,844,129	743,903	3,819,622	7,869,399
Savings	24.84%	5.97%	27.10%	9.28%

3.5 Summary and Directions for Future Work

The WHO-EPI vaccine distribution network is of critical importance in low and middle-income countries, and designing it optimally can be of significant economic and social benefit to these countries. In this paper we present a general MIP formulation of the design problem that is applicable to any country. While this problem can be solved optimally when the network is small, it rapidly becomes intractable as the problem size grows and a different solution approach is needed to address the problem for an entire country. We present a novel MIP-based disaggregation-and-merging algorithm that is based on the simple observation that changes to the structure in a part of the network are unlikely to have a significant effect on the structure in other parts that are far away. The algorithm thus uses a divide-and-conquer approach to intelligently generate and solve a sequence of MIPs. Extensive tests based on real-world data derived from four different countries in sub-Saharan Africa show that it yields solutions that are optimal or within 0.5% of the best cost where optimality can be verified, and for large instances that are impossible to solve optimally, it is uniformly robust and yields good solutions in a few minutes.

There are several directions for future work. First, a possible limitation of our approach is the fact that it is designed for problems where it is reasonable to assume that structures for subnetworks that are physically distant will tend to be independent of each other. While this is generally true if costs are uniform, it might not be true in other networks. For example, it might be appropriate to locate the hub serving a portion of the network at a location that is far away if the facility costs there are substantially lower, and transportation costs are also low. It would be interesting to see what degradation is obtained in the results when our approach is applied to such networks. Second, one could consider and model uncertainty, which could be associated with both demand and supply. Finally, a related direction would be to develop more sophisticated vaccine inventory management policies than the current practice of using an across-the-board buffer of 25% with a fixed monthly/quarterly replenishment interval. From an implementation standpoint though, the challenge here is twofold. First, to quantify and evaluate stochasticity we would need much more data than is currently available. Second, from a personnel standpoint, it would require far more sophistication in vaccine inventory management than is currently available in most LMICs.

4.0 Outreach and Mobile Clinic Strategies for Vaccine Distribution

In Chapter 2 and 3, we addressed the network design problem of vaccine distribution chains and once the network is fixed, targeted populations are assumed to travel to the clinics in the network. However, in many low and middle-income countries with geographically dispersed and nomadic populations, last-mile vaccine delivery can be extremely complex. Because newborns in remote locations within these countries often do not have direct access to clinics and hospitals, they face significant risk from diseases and infections. An approach known as outreach is typically utilized to raise immunization rates in these situations. A set of these remote population centers is chosen, and over an appropriate time horizon, teams of clinicians and support personnel are sent from a depot (a district center or hub, or an existing health clinic) to set up mobile clinics at these locations to vaccinate people there and in the immediate surrounding area. In this paper, we model the problem of optimally designing outreach efforts as a mixed integer program that is a combination of a set covering problem and a vehicle routing problem. In addition, because elements relevant to outreach (such as populations and road conditions) are often unstable and unpredictable, we incorporate uncertainty to study the robustness of the worst-case solutions and the related issue of the value of information. The remainder of Chapter 4 may also be found in (Yang & Rajgopal, 2019).

4.1 Introduction

As a biological preparation against infectious disease, vaccines have averted 2 to 3 million deaths annually (World Health Organization, 2019b), and coverage rates have improved significantly over the years under the guidance of the World Health Organization's Expanded Programme on Immunization (WHO-EPI) and the Global Alliance for Vaccines and Immunization (Gavi) (Gavi, 2020; World Health Organization, 2013). However, in many of the poorest countries, getting childhood vaccines delivered to their final destinations can be an extremely complex process. Although many low and middle-income countries (LMICs) can often obtain vaccines at low cost, operating a vaccine distribution system can be a challenge. Many vaccines require a narrow temperature range of between 2 and 8°C during storage and transportation, which in turn brings with it high distribution and storage costs. In addition to the challenge of planning for storage devices and transportation capabilities to distribute vaccines throughout the country, geographically dispersed or nomadic populations also present a major challenge. As a result, in many countries significant portions of the population have no direct access to health clinics.

Inadequate infrastructure and geographic barriers such as poor road conditions or limited access to transportation can further compound this problem. For example, in Niger, around 90% of the roads are not paved (Blanford et al., 2012). A recent study published in The Lancet Global Health estimated that across 48 sub-Saharan countries, 28.2% of women of child-bearing age are more than 2 hours travel time (combined walking and motorized) away from the nearest hospital (Ouma et al., 2018). The study also found wide variations with the percentage ranging from under 25% in South Sudan to over 90% in several countries including Nigeria, Kenya, Swaziland and Burundi. Another recent study in Uganda (Malande et al., 2019) concluded that difficulty in access to immunization centers due to poor road terrain has a significant effect and results in low immunization coverage. Thus, people living in remote locations in LMICs often face significant difficulty in obtaining routine vaccinations, and the WHO estimates that almost 20 million infants worldwide are at high risk from vaccine-preventable diseases such as polio, measles, yellow fever and tuberculosis (World Health Organization, 2019c).

To supplement the fixed vaccine distribution network, an approach known as *outreach* is typically utilized to raise immunization rates, especially in remote areas where direct access to clinic services is limited or unavailable. Clinicians and support personnel are sent from an existing clinic location to render services at one or more of these remote population locations. While the exact terminology varies from country to country we will refer to the existing location as a *depot* and the remote location as a *mobile clinic*. People at the location of the mobile clinic and other locations that are within a reasonable distance from it come there to get vaccinated. Note that this service is distinct from a *campaign* (a one-time attempt), in that outreach is periodic and repeated at regular time intervals. This interval might range from 1 month to 6 months in different countries.

Compared to a fixed clinic, mobile clinics can offer more flexibility and viability when treating vulnerable and isolated populations (World Health Organization, 2018; 2019a) and avoid unnecessary fixed facility, inventory, and labor cost (Daskin & Dean, 2004). Furthermore, outreach is proven to dramatically raise the overall immunization rates in resource-deprived countries that suffer from extremely low coverage rates. An early study in Kenya estimated that outreach increased the coverage rate in the lowest density zone in Kenya from 25% to 57% and from 54% to 82% in the area with greatest population density (World Health Organization, 1977). With the support of the WHO, outreach activities encompassing 1,982 mobile clinics and 5,964 personnel were able to cover 80% of targeted infants in September 2015 in Yemen; 290,498 children were vaccinated by these actions; and in 2018, 44 mobile clinics were set up to serve populations to hear-to-reach areas in Syrian (World Health Organization, 2019a)

While Chapters 2 and 3 and other relatively recent work (Chen et al., 2014; Lim et al., 2019; Yang et al., 2020; Yang & Rajgopal, 2020a) have focused on the network design phase of the WHO vaccine distribution chains, it is somewhat surprising that outreach has not received more attention in the academic literature and that there are almost no quantitative models available to help decision makers create an optimal outreach strategy. Lim et al. were the first to present quantitative models to determine optimal outreach locations and policies to maximize coverage rate (Lim et al., 2016). The authors contrasted various coverage models using data derived from the state of Bihar in India. In more recent work, Mofrad has proposed a mixed integer programming model to obtain an optimal mix of fixed and outreach vaccination services under demand uncertainty (Mofrad, 2016).

In the remainder of this chapter we propose a general model for LMICs to build on these early studies. Section 4.2 provides some background and our assumptions. We then provide an MIP model formulation in Section 4.3. An extension of the model with uncertainty considerations is presented in Section 4.4. We introduce the notion of value of information in Section 4.5 and present numerical results in Section 4.6. Finally, we discuss the results and summarize our work in Section 4.7.

4.2 Problem Development and Literature Review

While outreach has been proven to be effective at increasing vaccination rates in resource-deprived regions of the world, there is no standard structure or process that every country follows.

A typical process might be one where a medical team departs from an existing district center or clinic in a van or truck, carrying supplies and vaccines in cold boxes. The team then sets up at one or more mobile clinic location(s) and vaccinates the area's residents as well as residents from nearby areas. If multiple locations are visited, the team might go to each location sequentially and return to the original depot at the end of the day. However, each country has its own outreach policy and criteria to conduct outreach. For example, in some countries an outreach team might consist of clinicians and workers who come from multiple locations. In some countries, the vaccines might be delivered via a truck by a separate logistical team and stored in refrigerators at mobile clinics before the clinical team arrives there. In some countries, it might be possible for the team to stay in a mobile location overnight. In general, unlike with the operation of fixed clinics and the associated distribution system, there are no clear guidelines or standards on how outreach should be conducted.

Despite significant variations in economy, geography, demography, etc., and thus in how outreach is done across all these countries, this chapter aims to provide a relatively rigorous process for outreach trips across all countries to meet the WHO's goal of providing the entire targeted population with the opportunity to be vaccinated. We present a mixed integer programming (MIP) formulation to optimize the outreach strategies.

To retain a tractable MIP model while accounting for the various associated complexities and diversity as best we can, we include three sets of decisions into our consideration. The first is choosing the locations of mobile clinics for outreach as a subset of the existing targeted population centers. The second set of considerations is on how to assign population centers to mobile clinics. Note that a population center can be assigned to a mobile clinic only if it is closer than the maximum coverage distance (MCD) to that mobile clinic. The MCD can be defined as the maximum distance that people must travel to get vaccinated (e.g., 5 km) and is determined by the planners in the country. A mobile clinic could have the capability to serve multiple population centers, and each population center is assigned to a specific mobile clinic. Third, we determine an optimal set of vehicle trips that ensure that all mobile clinic locations are visited once within some suitable planning horizon (typically, 3 months or 6 months). Each trip would carry the required clinical and support personnel along with the required amount of vaccine for the location(s) served by the trip. Note that the vaccine regimens are not identical across countries, and based on the demand that is expected at population centers and the vaccine vial volumes, we can estimate the total volume associated with expected demand at each population center at each outreach session. Within the planning horizon, multiple vehicle trips can be undertaken but each vehicle trip must depart from a fixed depot and return to that depot after it visits one or multiple mobile clinic locations. The vehicles utilized in outreach trips are typically trucks or vans with several coolers or cold boxes and are thus capacitated in terms of how much vaccine can be carried.

To model a realistic process for outreach, we consider time windows on vehicle trips. There is a maximum trip duration (MTD) for each vehicle trip, e.g., 8 to 12 hours if all personnel need to return to the depot on the same day. In the case that they could stay overnight at a mobile clinic location, the MTD could possibly be longer. We also consider service time. This includes time required to set up the mobile clinic and time allocated to vaccinate targeted population members who come to the clinic. Different clinics do not always have identical service times; a clinic at a location that serves a larger population is likely to have longer service times. Depending on the situation, the service time at the originating depot can be set to zero or to the actual time required to load vaccines and prepare the team on the day of the trip. The travel times between the depot and clinics are obtained by dividing the corresponding distances by the average vehicle speed (e.g., 25 km/h).

We consider two components of cost in our objective function. The first is the direct cost associated with running a mobile clinic at a remote location. This cost includes the setup at the outreach site, the cost of renting or obtaining space, any labor costs for vaccination operations onsite, potential storage and energy consumptions cost, and any other local cost. The second cost component is the trip-related cost that is assumed to be proportional to the duration of the trip. This might include fuel costs, vehicle depreciation, hourly wages/allowances paid to the team and driver, vehicle rental costs, etc. We assume this results in an average cost per hour that is used to compute the cost of the trip based on its planned duration. The total cost is thus determined by the locations of the mobile clinics and the routes taken by the vehicles on their trips. In summary:

1) Our objective is to minimize the sum of direct mobile clinic costs and trip costs.
2) Mobile clinics for outreach are chosen from a set of existing targeted population centers.
3) A population center is said to be *covered* by a mobile clinic if it is within the specified MCD of the clinic.
4) A population center can be *assigned* to a mobile clinic only if it is covered by that mobile clinic, and each population center is assigned to one mobile clinic where the entire population assigned to the depot has the opportunity to be vaccinated.
5) Multiple outreach trips are made within the planning horizon, and every mobile clinic must be visited once within the planning horizon by an outreach trip.
6) We consider time windows on outreach trips and assume that a trip cannot be longer than some given maximum duration. There is a service time at each mobile clinic and a travel time between locations.

7) In each outreach trip, the vehicle departs from the depot, visits one or more mobile clinic locations, and returns to the depot within the MTD.
8) The vehicle is capacitated, and we assume the capacity is more than that what is required at any single population center.

The MCD in assumption 3 is assumed to be set to a value that is acceptable in the country being considered. Assumption 4 captures the WHO policy of ensuring that every child has the opportunity to be vaccinated. Assumptions 5, 6 and 7 are based on the most common practice, and assumption 8 is required to ensure feasibility.

The proposed problem under these assumptions can be viewed as a combination of a set covering problem (SCP) and a vehicle routing problem with time windows (VRPTW): the process of choosing mobile clinics and assigning population centers to each can be viewed as an SCP while the routes to visit these mobile clinics can be viewed as a VRPTW. A typical SCP in this context would choose the optimal facility locations with the objective of minimizing cost or maximizing the total demand covered (Berman & Krass, 2002; Church & ReVelle, 1974). This is a well-studied problem in the operations research community and has been widely applied in the heath care area (Daskin & Dean, 2004). The VRPTW is a variation of the Vehicle Routing Problem (VRP), which due to its wide application and importance in distribution networks, has also been widely studied by researchers. The goal of VRP is to obtain an optimal vehicle trip strategy to serve a set of customers. However, due to its complexity, exact algorithms such as branch-and-cut and branch-and-price usually have a size limit of 50 to 100 nodes; the problem is thus often solved by approximation algorithms and heuristics to find high quality solutions (Kumar & Panneerselvam, 2012; Prins, 2004; Toth & Vigo, 2014; Vidal et al., 2012).

VRPTW has the additional complication that customers need to be served within predefined time windows and within total trip time durations at minimum total cost, and belongs to the class of NP-hard problems (Lenstra & Kan, 1981). These problems are often solved by heuristics such as genetic algorithms (Potvin & Bengio, 1996; Thangiah, Nygard, & Juell, 1990), Tabu search (Cordeau, Laporte, & Mercier, 2001; Taillard et al., 1997), evolutionary algorithms (Bräysy, Dullaert, & Gendreau, 2004; Homberger & Gehring, 1999; Koç et al., 2015), large neighborhood search (Braaten et al., 2017; Bräysy, 2003; De Backer et al., 2000; Shaw, 1998), guided local search (Kilby, Prosser, & Shaw, 1999), ant colony algorithm (Balseiro, Loiseau, & Ramonet, 2011), and hybrid metaheuristics (Bent & Hentenryck, 2006; Homberger & Gehring, 2005; Reil, Bortfeldt, & Mönch, 2018; Yu, Yang, & Yao, 2011). Exact solution approaches (Qureshi, Taniguchi, & Yamada, 2009) and iterative route construction and improvement algorithms (Figliozzi, 2010; 2012) have also been suggested for solving VRPTW with "soft" time windows (a relaxation in the length of the time windows).

4.3 Model Formulation

In this section we develop our model formulation.

Parameters:

n: Total number of targeted population centers

i: Index of locations. $1 \leq i \leq n$ if i is a population center; $i = 0, n{+}1$ if i is depot

k: Index of outreach trips

b_i: Volume of vaccine demanded at population center i over the planning horizon

f_i: Fixed cost of running a mobile clinic at population center i

c: Average transportation cost per hour

d_{ij}: Distance between location i and location j (with $d_{ii} = 0$)

D: Maximal coverage distance (MCD)

$a_{ij} \in \{0,1\}$: 1 if location i is within a distance D from location j, 0 otherwise

K: Maximum number of outreach trips that can be made within the planning horizon

t_{ij}: Travel time from location i to location j

s_i: Service time at location i

r: Maximum trip duration (MTD)

p: Capacity of vehicle

Variables:

$X_{ij} \in \{0,1\}$: 1 if population center j is assigned to mobile clinic at location i, 0 otherwise

$Y_i \in \{0,1\}$: 1 if there is a mobile clinic at location i, 0 otherwise

$Z_{ijk} \in \{0,1\}$: 1 if location i is followed by location j in outreach trip k, $k \leq K$

U_{ik}: Cumulative vaccine volume already distributed by outreach trip k when arriving at mobile clinic location i, $k \leq K$

W_i: Total volume of vaccine sent to mobile clinic at location i

Program MIP-2:

$$Min \sum_{1 \leq i \leq n} f_i Y_i + \sum_k \sum_i \sum_j c t_{ij} Z_{ijk} \tag{4-1}$$

subject to

$$X_{ij} \leq a_{ij} \qquad \forall i,j \qquad (4\text{-}2)$$

$$X_{ij} \leq Y_i \qquad 0 \leq i \leq n, 1 \leq j \leq n \qquad (4\text{-}3)$$

$$\sum_{i \leq n} X_{ij} = 1 \qquad 1 \leq j \leq n \qquad (4\text{-}4)$$

$$W_i = \sum_j b_j X_{ij} \qquad i \leq n \qquad (4\text{-}5)$$

$$\sum_j Z_{0jk} = 1 \qquad \forall k \qquad (4\text{-}6)$$

$$\sum_j Z_{j0k} = 0 \qquad \forall k \qquad (4\text{-}7)$$

$$\sum_i Z_{i(n+1)k} = 1 \qquad \forall k \qquad (4\text{-}8)$$

$$\sum_i Z_{(n+1)ik} = 0 \qquad \forall k \qquad (4\text{-}9)$$

$$\sum_j Z_{ijk} = \sum_j Z_{jik} \qquad \forall k, 1 \leq i \leq n \qquad (4\text{-}10)$$

$$\sum_j \sum_k Z_{ijk} = Y_i \qquad 1 \leq i \leq n \qquad (4\text{-}11)$$

$$U_{ik} - U_{jk} + pZ_{ijk} \leq p - W_j \qquad \forall i,j,k \qquad (4\text{-}12)$$

$$W_i \leq U_{ik} \leq p \qquad \forall i,k \qquad (4\text{-}13)$$

$$\sum_i \sum_j (t_{ij} + s_i) Z_{ijk} \leq r \qquad \forall k \qquad (4\text{-}14)$$

$$\sum_i \sum_{j \leq n} Z_{ijk-1} \geq \sum_i \sum_{j \leq n} Z_{ijk} \qquad k \geq 2 \qquad (4\text{-}15)$$

$$Z_{iik} = 0 \qquad \forall i,k \qquad (4\text{-}16)$$

$$X_{ij} \in \{0,1\} \qquad \forall i,j \qquad (4\text{-}17)$$

$$Y_i \in \{0,1\} \qquad \forall i \qquad (4\text{-}18)$$

$$Z_{ijk} \in \{0,1\} \qquad \forall i,j,k \qquad (4\text{-}19)$$

$$U_{ik} \geq 0 \qquad \forall i,k \qquad (4\text{-}20)$$

$$W_i \geq 0 \qquad \forall i \qquad (4\text{-}21)$$

The objective function (4-1) minimizes the overall cost, which has two components: clinic operation costs and outreach trip transportation costs. Constraints (4-2) ensure that a mobile clinic can only serve population centers within the MCD of the clinic. Constraints (4-3) ensure that a location can serve other locations only if a mobile clinic is scheduled there. Constraints (4-4) ensure that each population center is assigned to a mobile clinic. Constraints (4-5) compute the total vaccine volume handled at a mobile clinic based on the population that the clinic serves. These four sets of constraints define a typical facility location problem.

The next set of constraints relate to the vehicle routing problem. Note that node 0 denotes the origin and node (n+1) is the final node at the end of a trip; both represent the depot. Constraints (4-6) and (4-7) imply that each vehicle trip departs from the depot (i=0) exactly once, while Constraints (4-8) and (4-9) imply that each vehicle trip enters back into the depot (i=n+1) exactly once. Constraints (4-10) ensure that the flow that enters and departs any population center i is balanced in each outreach trip k. Constraints (4-6) – (4-10) thus ensure that every vehicle trip is indeed a (0)-(n+1) path.

Constraints (4-11) state that exactly one vehicle enters and departs each population center during a planning period if there is a mobile clinic at this location (i.e., $Y_i = 1$). Constraints (4-12) are the vehicle-specific version of MTZ subtour elimination constraints introduced by Miller, Tucker, and Zemlin (Miller, Zemlin, & Tucker, 1960). Note that for a particular vehicle route k in which j follows i, $Z_{ijk} = 1$ implies $U_{jk} \geq U_{ik} + W_j > U_{ik}$. Suppose there exists a subtour ($i, j, \ldots$ i), with $i \neq 0, n$. Then, $U_{jk} > U_{ik} > U_{jk}$ will lead to a contradiction. Constraints (4-13) ensure that a vehicle carries enough vaccine for each mobile clinic but does not exceed its capacity. Note that if location i is not a part of trip k the values of U_{ik} are irrelevant to the problem as long as they satisfy (4-13).

Constraints (4-14) state that the total travel time and service time of a route cannot be larger than the MTD. Constraints (4-15) are added to avoid degeneracy by ensuring that route k is never utilized if route k-1 is not utilized; with Constraints (4-15), we reduce the search space by making sure that vehicle routes are chosen in a sequence of 1, 2, 3, etc. In addition, it ensures that vehicle trips with more stops will have a lower index value. Constraints (4-16) – (4-21) are self-explanatory.

4.4 A Two-period Stochastic Model

In Sections 4.2 and 4.3 we introduced a model that assumed all parameters are constant and deterministic, and the model is solved once. However, conditions in many targeted outreach locations are not always stable and predictable. It can often be difficult to obtain accurate estimates of all problem parameters ahead of time, and these might change as we get closer to the implementation of the outreach trips. For example, because demand is a function of population and birth rate, it can be more accurate to think of it as being stochastic, as both the population and the birth rate within a location could vary from year to year or even within a year. Similarly, in Assumption (7) we estimate the travel time from i to j as a constant based on the distance and the average vehicle speed. However, traffic and road conditions in the targeted zones can be unstable, so that this assumption might also need to be reconsidered. With an extreme event such as a flood or a landslide, a road might even be blocked. Conversely, improvements to infrastructure might actually reduce travel times. Therefore, it would in general be suboptimal to determine a fully fixed strategy ahead of time and simply repeat it in every successive planning period.

On the other hand, it can also be problematic if we update all parameters and obtain completely revised plans for each planning period. Recall that our problem is to minimize costs while providing the *opportunity* for 100% coverage; however, for a variety of reasons, in practice not every patient will show up at a clinic. A major goal of the WHO is to make access to vaccinations as easy as possible so as to minimize the number of these lost opportunities, and from this viewpoint, it is desirable to have a stable set of mobile clinic locations and for the populations assigned to each to be aware of when and where clinics will be conducted on a regular basis (e.g., the second Tuesday of every month; the first Monday in January, April, July and October; March 15 and September 15; etc.). We draw a compromise here by fixing locations but allowing for flexibility in timings. It is undesirable to move mobile clinic locations because it is disruptive and confusing for the populace to be directed to a different location each time for vaccination services. In contrast, it is relatively easy to inform people of a change in the timing of a clinic (because of a change in how we do the vehicle routing), especially if it is only for some clinics and the new times are not too different from those in the previous session.

Suppose we consider our problem using a two-period stochastic model. The two main uncertainties we consider during each planning period are (a) with respect to the population (and hence the volume of vaccines required) at each location, and (b) with respect to the travel times between locations i and j. Suppose that the volume of vaccines demanded at location i within each planning period is stochastic and represented by the random variable $\tilde{b}_i$, but we can constrain it to lie within some range $(\underline{b_i}, \overline{b_i})$. Similarly, the travel time between i and j is assumed to be stochastic and given by $\tilde{t}_{ij} \in (\underline{t_{ij}}, \overline{t_{ij}})$. This range might in general, be as wide as desired in order to account for inherent uncertainties and the upper bounds would reflect the worst-case scenarios. We can then utilize **MIP-2** but incorporate this information to now minimize either the *expected* cost or the *maximum* cost possible. We choose the latter option as it is more desirable in LMICs if the goal is to follow the WHO guidelines of reaching every child. As we will see, this also has the advantage of not requiring a characterization of the distributions associated with the stochastic variables. The solution to this yields the optimal mobile clinic locations, the assignment of population centers to these locations, and the associated routes for outreach trips for use within the first planning period.

At the end of the first planning period we review our estimates of the demand and travel time parameters and update these based on the most current information. For example, estimates of the population in some locations might have changed because of seasonal migrations or because of updated information from public health - or other - sources. Similarly, we might perhaps know that because of some natural catastrophe certain roads will be unavailable over the next planning period, or that driving times along certain routes will be longer or shorter because of changes in the season or changes in road conditions.

In the second period problem, we assume that the locations of mobile clinics and their population center assignments are fixed at the values obtained earlier, but we use updated parameters $(\underline{b_i'}, \overline{b_i'})$ and $(\underline{t_{ij}'}, \overline{t_{ij}'})$ to obtain revised routes for outreach trips in the second planning period. Typically, the range of parameter values resulting from these updated estimates will be tighter than the ones with the period 1 problem previously solved because of additional information that might now be available.

We illustrate this via a simple example shown in Figure 4. Suppose that we are developing the outreach strategy for an area containing a fixed clinic (which serves as the depot) and 15 population centers that must be served by it via outreach, with a visit every three months. We use our initial estimates of the demand and travel time to obtain the mobile clinic locations, along with the population assignments to each. We also obtain a set of outreach trips with routes as shown in Figure 4, where arrows represent vehicle routes and dotted lines represent assignment of outside population centers to a clinic: we have 8 outreach sessions with mobile clinics at locations 2, 5, 6, 9, 10, 12, 14, 15, spread across three separate outreach trips: trip 1 visits and holds outreach clinics at locations 2, 5 and 6 before returning to the depot; trip 2 does the same with locations 10 and 9, and trip 3 with locations 12, 14 and 15.. While each mobile clinic serves the population at its location, the clinic at location 2 also serves population centers 3 and 4, which are within the MCD of location 2. Similarly location 6 also serves 7; 9 also serves 8; 12 also serves 11 and 13; and people at location 1 are served by the fixed clinic (location 0).

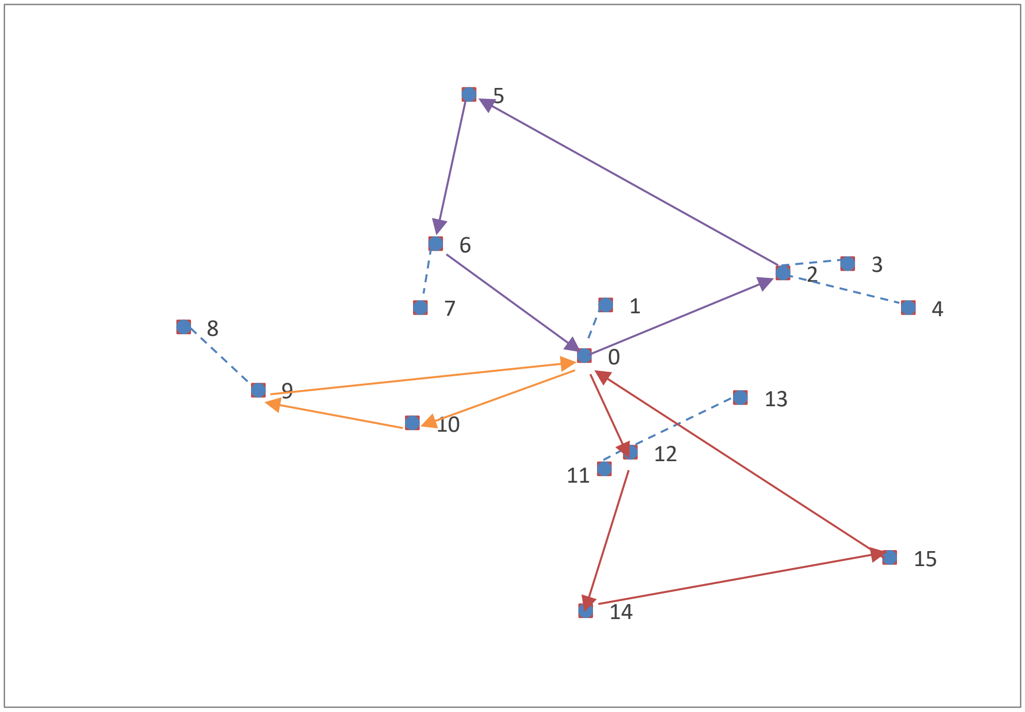

Figure 4 Initial solution

This plan is implemented in the first planning period (e.g., quarter). At the end of the period we get updated information and learn that the travel time along each edge will be a lot shorter because we have a new vehicle now, but that the roads connecting locations 2 and 5, as well as 0 and 6 will be closed because of major repairs. Without changing the locations of our mobile clinics, if possible we would like to obtain a better set of outreach trips to cover these same locations during the next planning period based on the updated information. This results in the strategy displayed in Figure 5. We still have our mobile clinics at the same eight locations but now we only have two outreach trips (0-12-14-15-2-0 and 0-5-6-9-10-0).

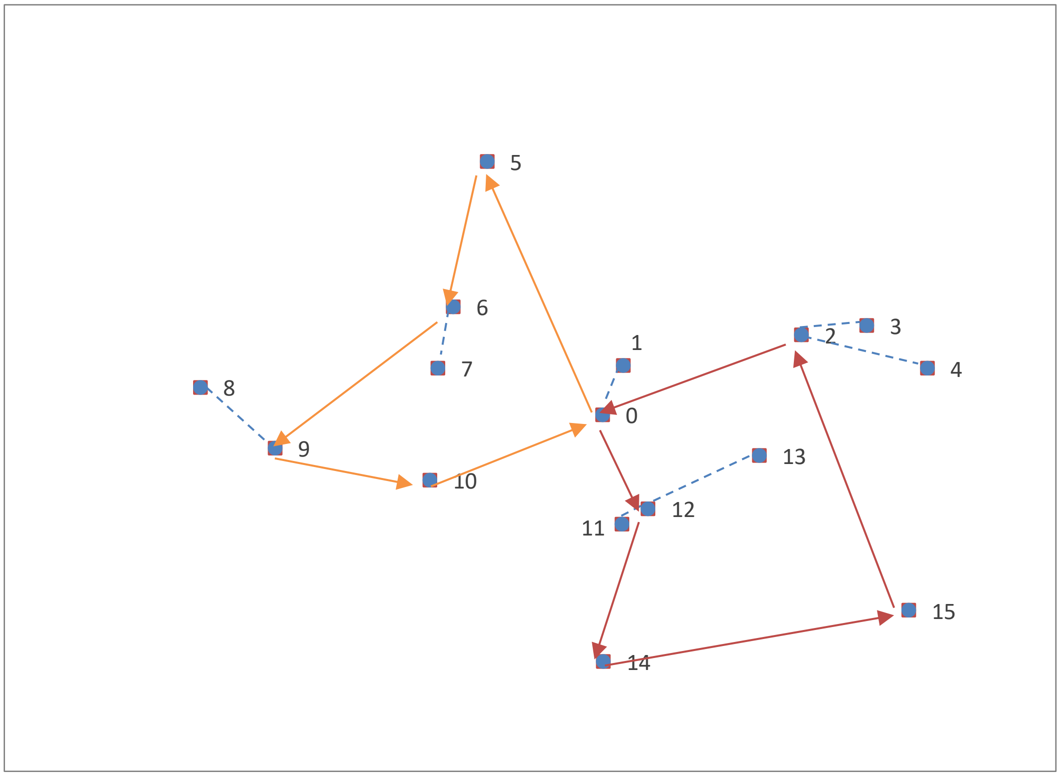

Figure 5 Updated solution

This process can then be repeated for subsequent planning periods, and as we obtain new information we can obtain updated solutions for the outreach trips each time. In summary, we have the following two-period stochastic mixed integer programming models (TS-MIP):

<u>TS-MIP-1</u>:

$$Z_1 = Min_{X,Y,Z,U,V,W}\ Max \sum_{1 \le i \le n} f_i Y_i + \sum_i \sum_j \sum_k c\tilde{t}_{ij} Z_{ijk} \qquad (4\text{-}22)$$

$subject\ to$

Constraints (4-2) – (4-4)

$$\sum_i W_i = \sum_j \tilde{b}_j X_{ij} \qquad i \le n \qquad (4\text{-}23)$$

Constraints (4-6) – (4-13)

$$\sum_i \sum_j (\tilde{t}_{ij} + s_i) Z_{ijk} \leq r \qquad \forall k \qquad (4\text{-}24)$$

Constraints (4-15) – (4-21)

$$\tilde{t}_{ij} \in (\underline{t_{ij}}, \overline{t_{ij}}) \qquad \forall i, j \qquad (4\text{-}25)$$

$$\tilde{b}_i \in (\underline{b_i}, \overline{b_i}) \qquad \forall i \qquad (4\text{-}26)$$

Let Y_i^* be the optimal value of Y_i in the solution to **TS-MIP-1**. Then we have

TS-MIP-2:

$$Z_2 = \left(\sum_{1 \leq i \leq n} f_i Y_i^* + \right) Min_{Z,U,V,W}\ Max \sum_i \sum_j \sum_k c\tilde{t}_{ij} Z_{ijk} \qquad (4\text{-}27)$$

subject to

Constraints (4-2) – (4-4), with $Y_i = Y_i^*$in (4-3)

$$\sum_i W_i = \sum_j \tilde{b}_j X_{ij} \qquad i \leq n \qquad (4\text{-}28)$$

Constraints (4-6) – (4-13), with $Y_i = Y_i^*$in (4-11)

$$\sum_i \sum_j (\tilde{t}_{ij} + s_i) Z_{ijk} \leq r \qquad \forall k \qquad (4\text{-}29)$$

Constraints (4-15) – (4-17), (4-19) – (4-21)

$$\tilde{t}_{ij} \in (\underline{t'_{ij}}, \overline{t'_{ij}}) \qquad \forall i, j \qquad (4\text{-}30)$$

$$\tilde{b}_i \in (\underline{b'_i}, \overline{b'_i}) \qquad \forall i \qquad (4\text{-}31)$$

Note that **TS-MIP-2** is solved with the clinic locations fixed, and optimizes deliveries over an updated range of values for the demands $\tilde{b}_i$ and travel times $\tilde{t}_{ij}$.

Proposition 4: Assuming feasibility, solving **TS-MIP-1** followed by **TS-MIP-2** is equivalent respectively to (a) solving **TS-MIP-1** with $\tilde{t}_{ij} = \overline{t_{ij}}$ in (4-22), (4-24) and $\tilde{b}_j = \overline{b_j}$ in (4-23); and (b) solving **TS-MIP-2** with $\tilde{t}_{ij} = \overline{t'_{ij}}$ in (4-27), (4-29) and $\tilde{b}_j = \overline{b'_j}$ in (28).

Proof: First, note that from (4-23) or (4-26), as the value of $\tilde{b}_j$ increases, so does the value of W_i. This in turn reduces the size of the feasible regions for **TS-MIP-1** and **TS-MIP-2** by tightening the constraints defined by (4-12) and (4-13). Similarly, an increase in $\tilde{t}_{ij}$ tightens the constraints defined by (4-24) or (4-29) while also increasing the cost coefficient for Z_{ijk} in the objective. So with these changes, assuming feasibility, the objective function can only increase from its current value (or at best, stay the same). Its maximum value is thus obtained when each $\tilde{b}_j$ and $\tilde{t}_{ij}$ is at its largest possible value.

□

The above result is intuitive: when the population (demand) increases, it is possible that limitations arising from the vehicle capacity might increase the number of trips required to cover all locations, and when travel times along an arc i-j increase, the total travel costs rise; it is also possible that the length of a trip might exceed the trip MTD (r), again causing an increase in the number of trips. **Proposition 4** states that if we are conservative and plan for the worst-case scenario with respect to the period 2, then this corresponds to when travel times and populations are as large as they could get. We refer to the solutions for these worst-case scenarios as *robust* solutions.

4.5 Robustness and the Value of Information

In Section 4.4 we introduced a two-period procedure to address the unstable outreach environment that is typical in practice. In this section we compare, discuss and interpret the costs associated with the robust solutions to the period 1 and period 2 problems.

We can interpret Z_1, the optimal value of **TS-MIP-1** as the optimal cost associated with the conservative strategy at the beginning of the first period that addresses the worst-case scenario. The optimal value of **TS-MIP-2** given by Z_2 is also for a conservative strategy but with an updated worst case scenario and with clinic locations fixed based upon the optimal solution to **TS-MIP-1** for the first period. Any difference between Z_1 and Z_2 is a result of possibly updated outreach trips with better vehicle routes. While Z_2 could in general be larger or smaller than Z_1, if the updated upper bounds are the same or smaller than before, then as the following corollary states, Z_2 will be smaller.

Corollary 1: If $\overline{b_i'} \leq \overline{b_i}$ and $\overline{t_{ij}'} \leq \overline{t_{ij}}$, then $Z_2 \leq Z_1$.

Proof: In proving **Proposition 4** we saw that as the values of $\tilde{b}_i$ and $\tilde{t}_{ij}$ increase, the feasible regions for both problems shrink, and when they decrease the region expands. Therefore, Z_1 and Z_2 are monotone non-decreasing in both $\tilde{b}_{ij}$ and $\tilde{t}_{ij}$. Further, **TS-MIP-2** has the same locations as the optimal locations in **TS-MIP-1** (at i corresponding to $Y_i^* = 1$), and if $\overline{b_i'} \leq \overline{b_i}$ and $\overline{t_{ij}'} \leq \overline{t_{ij}}$ it has an expanded feasible region for choosing the delivery routes; so $Z_2 \leq Z_1$.

□

Definition 1: The percentage improvement in the robust cost that arises from tighter upper bounds is defined as $\Delta Z = 100 * (Z_1 - Z_2)/Z_1$.

Note that the mobile clinic locations used in **TS-MIP-2** were obtained by solving **TS-MIP-1**, and in general, these need not be optimal with the updated problem parameter estimates. If we had the ability to also relocate mobile clinics in each time window, we could design a network and an associated outreach strategy with a possibly lower cost than Z_2 for the new planning period. To see this, we define the following One-Stage Stochastic Mixed Integer Programming model (**OS-MIP**):

<u>OS-MIP</u>:

$$Z_0 = Min_{X,Y,Z,U,V,W}\, Max \sum_{1\le i\le n} f_i Y_i + \sum_i \sum_j \sum_k c\tilde{t}_{ij} Z_{ijk} \tag{4-32}$$

$subject\ to$

Constraints (4-2) – (4-4)

$$\sum_i W_i = \sum_j \tilde{b}_j X_{ij} \qquad i \le n \tag{4-33}$$

Constraints (4-6) – (4-13)

$$\sum_i \sum_j (\tilde{t}_{ij} + s_i) Z_{ijk} \le r \qquad \forall k \tag{4-34}$$

Constraints (4-15) – (4-21)

$$\tilde{t}_{ij} \in (\underline{t'_{ij}}, \overline{t'_{ij}}) \qquad \forall i,j \tag{4-35}$$

$$\tilde{b}_i \in (\underline{b'_i}, \overline{b'_i}) \qquad \forall i \tag{4-36}$$

Along the lines of **Proposition 4**, we have the following Proposition that is related to **OS-MIP**:

Proposition 5: Program **OS-MIP** is equivalent to solving it with $\tilde{t}_{ij} = \overline{t'_{ij}}$ in (4-32), (4-34) and $\tilde{b}_j = \overline{b'_j}$ in (4-33).

The following proposition relates **OS-MIP** to **TS-MIP-2**:

Proposition 6: $Z_0 \leq Z_2$.

<u>Proof</u>: It is clear that **OS-MIP** is a relaxation of **TS-MIP-2**, with the option of picking locations other than those given by $Y_i = Y_i^*$. Therefore, $Z_0 \leq Z_2$.

□

Note that the optimal value of **OS-MIP** (Z_0) is yet another conservative cost, and corresponds to the theoretical best robust solution to the outreach problem for the second period. Any difference between Z_2 and Z_0 is due to the fact that in OS-MIP we have the freedom to update mobile clinic locations. We may also interpret this reduction as the value of having better information on the parameter bounds at the beginning of the first period, as opposed to having to wait for it until the beginning of the second period (because we would then have obtained this solution for the second period in our initial design for the first period).

Definition 2: The value of information is defined as $V = 100 * (Z_2 - Z_0)/Z_2$.

Thus V is the percentage savings possible (in the worst-case scenario), from obtaining information in the form of correct bounds on $\tilde{t}_{ij}$ and $\tilde{b}_i$ at the beginning of the first period.

4.6 Numerical Experiments

We tested the procedure introduced in the previous sections on data that we adapted from four countries in sub-Saharan Africa. Due to issues with data confidentiality we label these countries A through D. Country B is smaller and has a high population density. The other three are larger in area and have some pockets of dense population with others (such as areas in the Sahara desert) where it is much more sparsely populated. To explain our numerical experiments and demonstrate some of the insights to be gained, we will first describe in detail an illustrative example with data derived from Country D. Following this, we analyze and summarize results from a larger set of instances.

Our illustrative example has 9 population centers on a 20 km by 20 km graph, with the depot is in the middle of the graph. Suppose each population center has an average of 100 newborns in a year. For implementation in the first period a robust solution (with value Z_1) is obtained for problem **TS-MIP-1** using initial estimates of upper bounds on demand and travel times. The mobile clinic locations and the population centers assigned to each location in this solution are then fixed. Next, using the most current information on demands and travel times vehicle routes are updated for the second period by obtaining a robust solution (with value Z_2) to problem **TS-MIP-2**. We also solve **OS-MIP** to obtain the theoretical best robust solution to the period 2 problem (with value Z_0). Under the assumption that the bounds on the updated estimates are always tighter than the initial ones, we study the impact of upper bound changes (i) only in b (demand), (ii) only in t (travel times), and (iii) in both b & t.

We first generated a base case for the period 2 problem with associated values for $\overline{b_j'}$ and $\overline{t_{ij}'}$, and obtained Z_0 by solving problem **OS-MIP**. This solution with a value of Z_0= 619.17

represents the best robust solution obtainable for period 2 and is illustrated in Figure 6: it has four clinics (at locations 1, 2, 3, and 5) with two trips (0-1-2-0) and (0-3-5-0). The mobile clinics at locations 2, 3 and 5 also cover the populations at 7, (4,6) and (8,9) respectively. The cost of 619.17 is comprised of 290.50 in facility costs and 328.67 in operating costs.

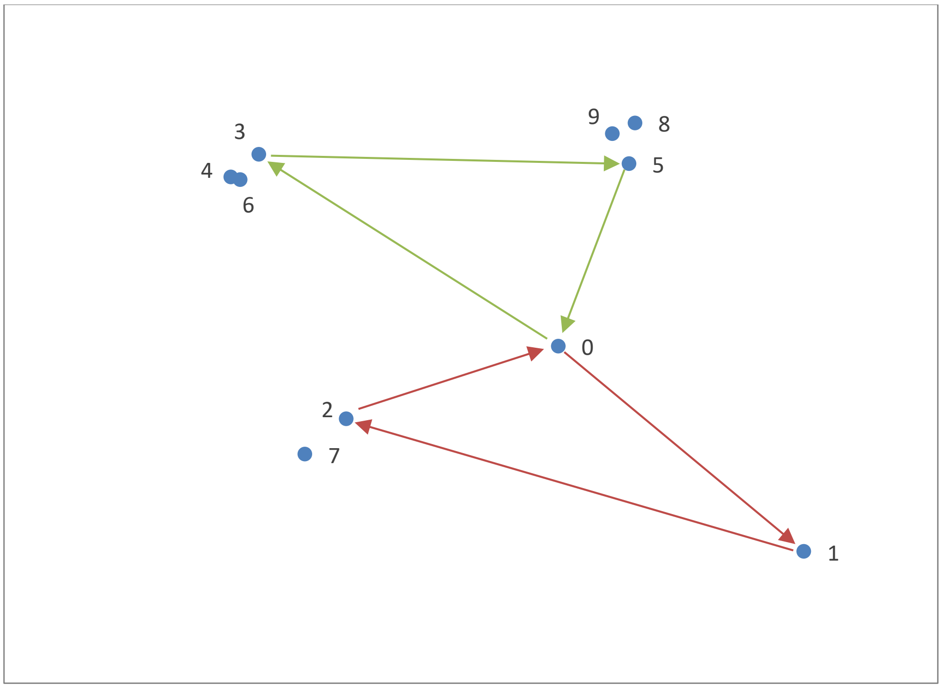

Figure 6 Optimal robust solution for period 2 problem

Next, for each of the three types of parameter changes (b, t, and b & t) we studied (a) small, (b) moderate and (c) large reductions in the initial estimates of the upper bounds before the first period ($\overline{b_j}$ and $\overline{t_{ij}}$). Specifically, for these three cases we assumed that $\overline{b_j}$ and $\overline{t_{ij}}$ were on average 20%, 80% or 150% larger than their values before the second period ($\overline{b_j'}$ and $\overline{t_{ij}'}$). In all cases, we first solve problem **TS-MIP-1** with the appropriate values of $\overline{b_j}$ and/or $\overline{t_{ij}}$ to obtain the robust solution for the first period, along with its value Z_1. We then fix clinic locations and their allocations to solve problem **TS-MIP-2** using $\overline{b_j'}$ and $\overline{t_{ij}'}$ for the bounds, and obtain the robust solution for the second period, along with its value Z_2. The results are listed in Table 18, and we discuss some of the insights that these offer.

Table 18 Example in Country D

	b			*t*			*b* & *t*		
Case	Small	Moderate	Large	Small	Moderate	Large	Small	Moderate	Large
Z_1	619.17	619.17	1003.68	651.29	714.19	778.49	693.39	831.13	1682.26
Z_2	619.17	619.17	817.49	630.32	630.32	642.03	619.35	627.30	923.71
ΔZ	0.00%	0.00%	18.48%	3.22%	11.74%	17.53%	10.68%	24.52%	45.09%
V	0.00%	0.00%	24.26%	1.77%	1.77%	3.56%	0.03%	1.29%	32.97%

Note: Theoretical best robust optimum for period 2 = Z_0=**619.17**

First, it may be seen that with the tighter bounds, the robust optimum for **TS-MIP-2** (=Z_2) shows improvement over that for **TS-MIP-1** (=Z_1) in seven of the nine cases, with the percentage improvement (ΔZ) being much more significant when the upper bounds get significantly tighter (i.e., large reductions). While these results are intuitive, it is interesting that the improvements are more pronounced with tighter time estimates as compared to tighter demand estimates (simultaneous reduction of uncertainty in both parameters further magnifies the savings).

Next, we look at the issue of what we could have achieved in the second period if we had been able to re-optimize locations and allocations. That is, we compare the robust optimum Z_2 from **TS-MIP-2** to its theoretical lowest value of Z_0=619.17. In particular, we compute the theoretical maximum percentage improvement possible in Z_2, i.e., the value of information (V) as given by **Definition 2**. It may be observed that for this example, the updated robust solution to the second period problem is actually very close to the theoretical best value in seven of the nine cases. The only instances where the value of information is high is when there are large reductions in the estimated upper bounds for demand, especially when there are simultaneous reductions in the upper bound on travel times. We visually illustrate some of the results in Figures 7 to 10.

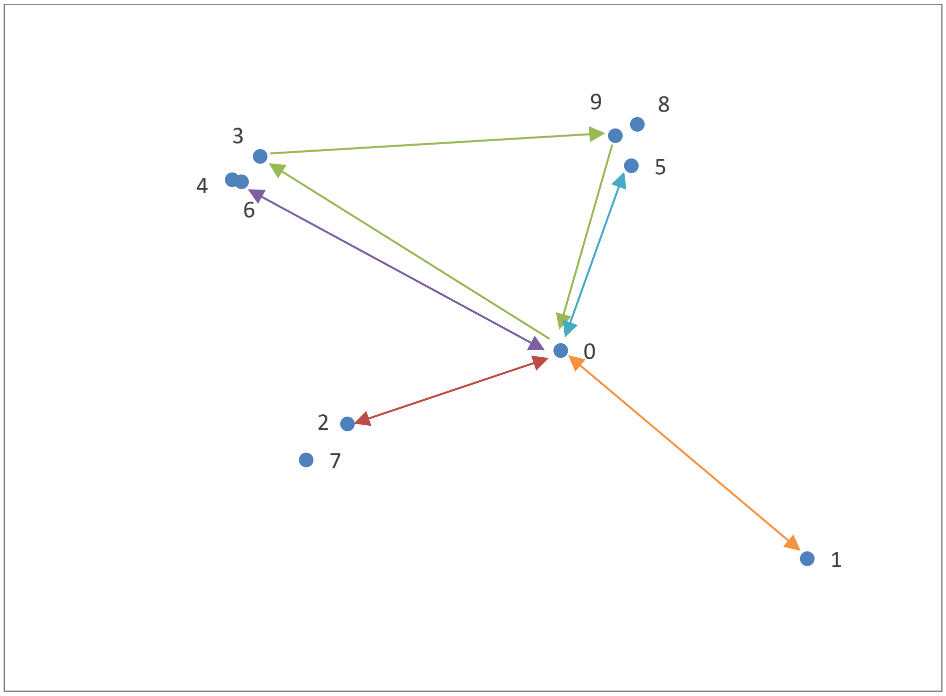

Figure 7 Solution in period 1 with initial demand estimates

First, consider demand estimates. If the revised estimates in the upper bounds are only slightly or moderately tighter (columns 1 and 2 in Table 18), we find the theoretical best solutions at the beginning and this does not change with revised estimates; thus there is no value to these revised estimates. In contrast, if there is a large reduction in the estimate from period 1 to period 2 (column 3 in Table 18) the situation is different. Figure 7 illustrates the solution to problem **TS-MIP-1** for the first period, with six clinics scheduled via five outreach trips covering 1, 2, (3, 9), 5, 6 respectively. Note that clinics at 2, 9 and 6 also cover the populations at locations 7, 8 and 4, respectively in this solution. As displayed in Table 18, this solution has a total cost of 1003.68.

Now, consider period 2 where we are constrained to maintain outreach clinics at these same six locations, but use the updated information on the worst-case demand to solve problem **TS-MIP-2**. This yields the updated solution shown in Figure 8 with two outreach trips (0, 1, 5, 9, 0) and (0, 3, 6, 2, 0) to cover the six clinics, and a total cost of 817.49. Locations 7, 8 and 4 are covered by the clinics at 2, 9 and 6, respectively. Note that the very loose initial upper bound for demand caused the robust solution to the first period to have more trips because a much larger demand could cause vehicle capacity constraints to be violated if the period 2 solution is adopted. In both solutions we have facility costs of 435.75 for the six open mobile clinics, but operating costs of 567.93 in the first period as opposed to 381.74 in the second.

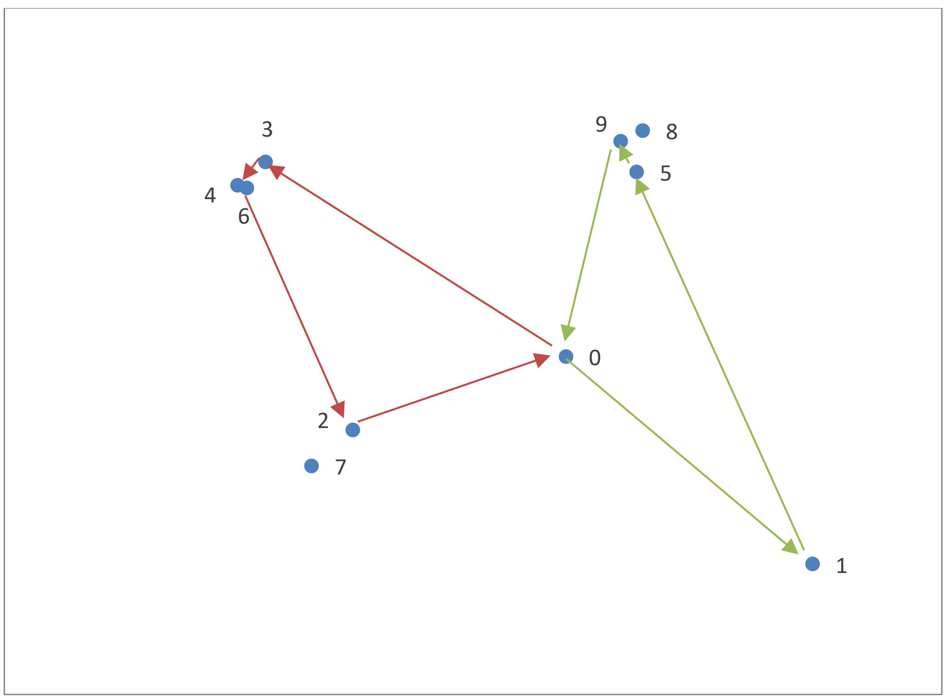

Figure 8 Updated solution in period 2: large reductions in demand estimates

Finally, note that if we had had the updated information prior to period 1, we would have obtained the best overall robust solution (shown in Figure 6), which is 24.26% better than the one from **TS-MIP-2**; this is the value of information in this instance.

Next we look at travel time; Figure 9 is for the case where upper bounds on travel times are tightened slightly or moderately, i.e., when the initial upper bound estimates were either 20% or 80% larger on average than their updated values in period 2 (columns 4 and 5 in Table 18). The robust solutions for both periods are identical in both cases, and the reductions from Z_1 to Z_2 (651.29 to 630.32, and 714.19 to 630.32, respectively) are only because $\overline{t'_{ij}} < \overline{t_{ij}}$. Also, the only difference between Figure 9 and the theoretical best design for period 2 shown in Figure 6 is that a clinic is located at 8 instead of 5. This is because $\overline{t_{05}}$ happens to be larger than $\overline{t_{08}}$, and thus in period 1, location 8 is preferable to location 5 in the solution to **TS-MIP-1**. When in period 2 the clinic is fixed at location 8 and we use $\overline{t'_{05}}$ and $\overline{t'_{08}}$ in Problem **TS-MIP-2**, the cost (Z_2=630.32) is 11.15 units higher than it would have been (Z_0=619.17) with the optimal location (i.e., V =1.77%).

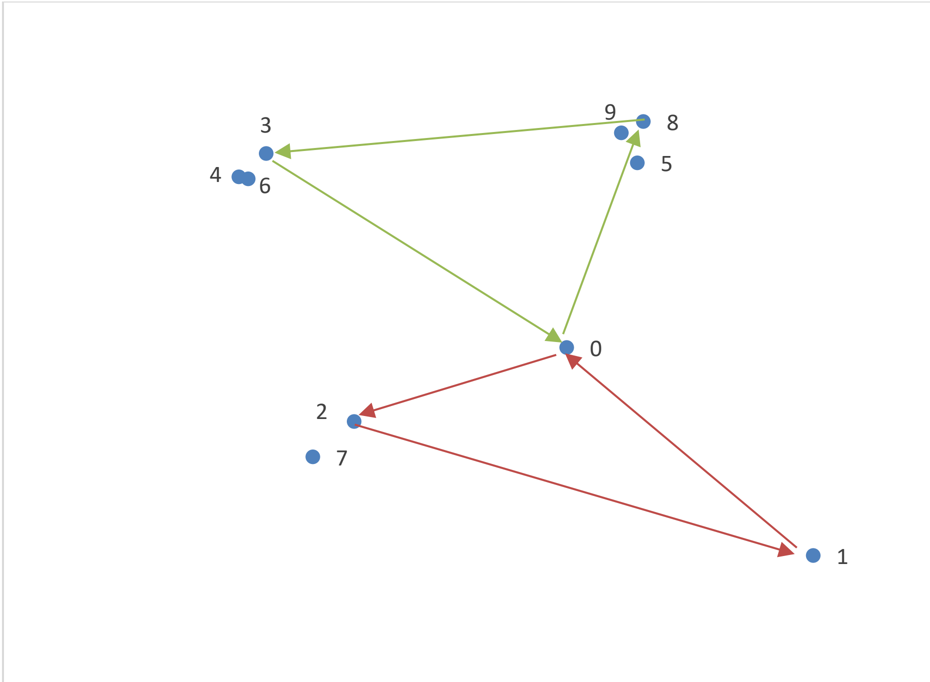

Figure 9 Solutions in periods 1 & 2: small/moderate reductions in travel time estimates

Figure 10 depicts the case where there is a large reduction (column 6 in Table 18) in the initial estimates of travel time ($\overline{t_{ij}}$ exceeds $\overline{t'_{ij}}$ by 150% on average). Again, the solutions are identical in both periods, and in contrast with the case when reductions in the bounds are small or moderate, a clinic is now assigned to location 7 instead of location 2. Using these locations as opposed to the optimal ones in Figure 6 yield a period 2 cost of Z_2=642.03 that is 22.86 units higher than the theoretical best value of Z_0 (with V =3.56%).

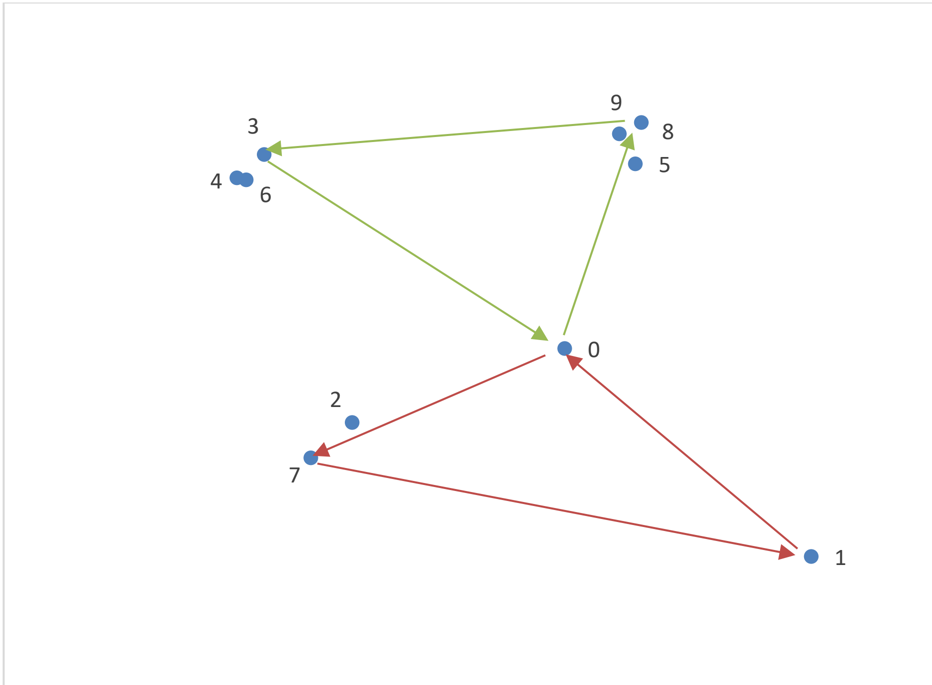

Figure 10 Solutions in period 1 & 2: large reductions in travel time estimates

With simultaneous changes in demand and travel times parameter estimates (columns 7, 8 and 9 in Table 18) similar types of results are obtained; figures are omitted in the interest of brevity.

We now present results from a larger set of instances across all four countries using the same methodology. The results for the value of information (V) are summarized in Tables 19, 20 and 21 based upon the demographic characteristics of the region. Table 19 shows results for examples from Country B and regions of Countries A, C, and D with high population densities (e.g., around their capital cities). Table 20 has instances from Countries A, C, and D where populations are moderately dense, while Table 21 covers larger, often remote regions in the same countries, with relatively sparse populations. Note that the examples in Table 21 have fewer population centers that are more sparsely distributed on a larger graph, while the examples in Table 19 have more population centers on a relatively small graph; the examples in Table 20 are in between these two extreme cases. Also note that smaller values for V in the tables indicate worst-case solutions that are relatively robust with our approach, while larger values indicate that having tighter, more accurate initial estimates of parameter bounds can result in more significant savings over the solutions from our approach. Insights that can be drawn from the computational results in these tables are discussed in detail in the following section.

Table 19 Value of information for examples in smaller, densely populated regions

	b			*t*			*b* & *t*		
	Small	Moderate	Large	Small	Moderate	Large	Small	Moderate	Large
A1	9.72%	17.71%	32.47%	0.00%	0.00%	0.00%	9.72%	17.71%	32.47%
B1	0.00%	22.87%	50.99%	0.38%	1.96%	1.96%	0.38%	23.35%	50.99%
B2	0.00%	15.47%	47.88%	0.05%	0.05%	0.05%	0.05%	15.47%	47.88%
B3	0.00%	19.76%	49.24%	0.00%	1.09%	2.22%	0.00%	20.02%	49.24%
B4	31.22%	31.70%	48.00%	0.00%	0.00%	31.87%	31.22%	31.70%	58.58%
B5	0.00%	17.45%	30.35%	0.26%	0.26%	0.26%	0.26%	17.51%	30.87%
B6	0.00%	15.54%	26.85%	0.00%	0.00%	1.14%	0.00%	15.63%	36.13%
C1	24.57%	38.75%	68.49%	0.00%	0.00%	0.00%	24.51%	38.82%	68.49%
C2	0.58%	24.08%	39.33%	0.58%	1.75%	2.04%	0.58%	25.19%	50.24%
D1	20.05%	41.38%	65.20%	0.27%	0.27%	2.62%	20.05%	42.57%	65.20%
Mean	8.61%	24.47%	45.88%	0.15%	0.54%	4.22%	8.68%	24.80%	49.01%
Median	0.29%	21.32%	47.94%	0.02%	0.15%	1.55%	0.48%	21.68%	49.74%

Table 20 Value of information for examples in moderately populated regions

	b			*t*			*b* & *t*		
	Small	Moderate	Large	Small	Moderate	Large	Small	Moderate	Large
A2	0.00%	17.98%	26.23%	0.01%	30.98%	26.23%	0.01%	26.23%	26.23%
A3	0.00%	0.00%	16.21%	0.00%	0.00%	0.00%	0.00%	0.00%	16.21%
A4	0.00%	0.14%	30.88%	0.00%	0.00%	0.00%	0.00%	10.83%	30.88%
C3	0.30%	0.30%	10.84%	0.30%	1.73%	2.02%	0.30%	0.30%	18.59%
C4	0.00%	9.52%	9.52%	0.00%	0.00%	8.95%	0.00%	9.52%	19.79%
D2	0.00%	11.69%	20.94%	0.00%	0.00%	0.00%	0.00%	12.12%	35.22%
D3	0.00%	0.00%	24.26%	1.77%	1.77%	3.56%	0.03%	1.29%	32.97%
D4	11.52%	11.52%	20.67%	0.00%	0.00%	20.67%	11.52%	11.52%	39.95%
D5	0.00%	13.14%	22.51%	0.00%	0.00%	0.00%	0.00%	13.14%	31.45%
D6	0.00%	0.00%	12.19%	1.69%	0.00%	2.58%	2.58%	0.00%	34.67%
Mean	1.18%	6.43%	19.42%	0.38%	3.45%	6.40%	1.44%	8.49%	28.60%
Median	0.00%	4.91%	20.80%	0.00%	0.00%	2.30%	0.00%	10.17%	31.16%

Table 21 Value of information for examples in larger, sparsely populated regions

	b			*t*			*b* & *t*		
	Small	Moderate	Large	Small	Moderate	Large	Small	Moderate	Large
A5	0.00%	0.00%	0.00%	2.42%	2.02%	2.02%	2.42%	2.02%	17.99%
A6	0.00%	0.12%	0.12%	0.12%	8.42%	8.42%	0.12%	8.42%	8.42%
A7	0.00%	6.98%	14.29%	0.00%	14.29%	14.29%	0.00%	14.29%	14.29%
A8	0.00%	0.00%	16.00%	0.00%	0.00%	0.00%	0.00%	0.00%	16.00%
C5	6.81%	6.81%	6.81%	0.28%	0.00%	0.00%	6.56%	6.56%	17.78%
C6	0.00%	0.00%	7.97%	0.00%	0.00%	0.00%	0.00%	7.97%	15.40%
C7	0.00%	5.51%	5.51%	0.00%	0.00%	5.51%	0.00%	5.51%	11.57%
C8	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
D7	0.00%	0.00%	4.91%	2.05%	4.91%	4.91%	2.05%	4.91%	11.25%
D8	0.00%	0.00%	0.00%	0.66%	0.96%	0.96%	0.00%	0.96%	0.96%
Mean	0.68%	1.94%	5.56%	0.55%	3.06%	3.61%	1.11%	5.06%	11.37%
Median	0.00%	0.00%	5.21%	0.06%	0.48%	1.49%	0.00%	5.21%	12.93%

4.7 Discussion and Conclusions

Tables 19, 20 and 21 show results for three different demographic characteristics: for each we list the mean as well as median values for each of the nine cases studied. We separately highlight instances with V values over 20% and those with values under 5% to indicate "large" and "small" values, respectively for the value of information. The entries not highlighted may be thought of as being in between. First, consider changes in estimates of only b or only t (the first two columns in each of the tables). Although exceptions do exist, we can draw the general conclusion that our approach is quite robust (i.e., V is small) in the following situations:

- Revisions are only in the travel time estimates, regardless of whether they are small, moderate or large: the value of information is under 5% in 79, and over 20% in only 4 out of the 90 instances corresponding to this situation (the nine columns - across the three tables - under "t").
- Revisions are only in the demand estimates and they are small: V is under 5% in 24 and over 20% in only 3 out of the 30 instances for this case (the three columns under "b" and "Small").
- Revisions are only in the demand estimates and they are moderate, but we are in larger areas with moderate to sparse population densities: V is under 5% in 12 of 20 instances (the two columns under "b" and "Moderate" in Tables 20 and 21).

Conversely, the costs in the worst-case scenario can be higher with our approach than they would be if we had perfect information in advance (i.e., V is much larger) under the following scenarios:

- There are large revisions in the bounds on demand: V is over 20% in 16 and under 5% in only 5 out of the 30 instances for this case, all of the latter for sparsely populated regions (the three columns under "b" and "Large").
- There are moderate revisions in densely populated regions: V is over 20% in 5 of the 10 instances for this case and never under 5% (column under "b" and "Moderate" in Table 19).

When we consider simultaneous changes in the estimated bounds for both travel time and demand estimate (the columns under "b & t"), the results are closely correlated with what happens when there are changes in demand alone, leading one to conclude that demand revisions constitute the primary factor and their effects overshadow revisions in travel time estimates.

Next, we conducted a set of separate, nonparametric, Wilcoxon signed-rank tests for each of the three different types of regions (d, m, s) to see if there were differences in the magnitude of the mean effects of the different types of changes (H_0: $\mu_{\cdot b} = \mu_{\cdot t}$ vs. H_1: $\mu_{\cdot b} > \mu_{\cdot t}$ and H_0: $\mu_{\cdot b,t} = \mu_{\cdot b}$ vs. H_1: $\mu_{\cdot b,t} > \mu_{\cdot b}$ for each of d, m, s). Note that in each of the six comparison we have 30 paired instances across which we study difference in means. The null hypotheses were strongly rejected (P-values all under 0.001) for five out of these six tests; the only case where there was no significant difference in the mean value of V was when comparing individual changes in estimates of b and t in sparse regions (μ_{sb} and μ_{st}) , which had a P-value of 0.66. In other words, (i) in dense and moderately populated regions, the value of V with changes in only b is significantly higher than that with changes in only t, and (ii) in all types of regions V is significantly higher with changes in both b & t as compared with changes in only b.

Finally, we look at the general demographic characteristics to study their effect. When we have large, sparsely populated regions (Table 21), our approach is quite robust: V is under 5% in 62 out of 90 instances, and always under 20%. When the population density is moderate (Table

20), our approach is still reasonably robust unless there are large changes in the estimated demand (as observed previously), in which case the value of V starts to increase. Finally, in smaller, densely populated areas (Table 19) the results can be much more sensitive to changes in demand and there is significant value to obtaining more precise estimates of demand. Given that there are 90 instances for each type of region we conducted two simple one-sided Z-tests for equality of means (H_0: $\mu_d = \mu_m$ vs. H_1: $\mu_d > \mu_m$ and H_0: $\mu_m = \mu_s$ vs. H_1: $\mu_m > \mu_s$). The null hypothesis is strongly rejected in favor of the alternative in both tests (P-values in both cases are under 0.0002), confirming that small, dense regions tend to have larger value of information than larger, moderately populated regions, and in turn, the latter yield larger value for V than large, sparsely populated regions.

Based on our computational study, we can draw two main conclusions. First, larger sparsely populated regions tend to have lower value of information, while the opposite is true for smaller more densely populated regions. We speculate that this is because when there are fewer population centers and they are relatively far apart and can serve relatively fewer neighboring population centers, a larger fraction of the locations are selected for outreach, and capacity is less of an issue with fewer people being served by each outreach trip. Thus, even with perfect information, there is relatively little opportunity to revise the initial plan even when demand and travel times estimates change. Conversely, in smaller, denser regions there are more dependencies between population centers and more people are served in each trip. Thus changes in population estimates, and to a lesser extent, travel times, have a significant effect: often, the best strategy could be different from the plan that we obtain because capacities might be exceeded or alternative solutions to the set covering problem yield shorter vehicle routes. Thus the value of obtaining accurate information is much higher, and the solution with our approach might not be as robust.

Second, it is better to focus more on obtaining more accurate population (demand) estimates than on travel time estimates. The latter have relatively low value of information and our approach is very robust even in smaller, densely populated regions with approximate estimates of these times. On the other hand, if demand estimates are too conservative (large) we could arrive at a strategy that results in locations that are not cost-effective after we get updated information; thus it is important to be able to get good estimates of demand in order for our approach to be robust.

In summary, this chapter presents a systematic way to plan for economical outreach operations by formulating the problem as a mixed integer program. It also studies the issues related to the typical uncertainties associated with estimating demand for vaccines and planning individual outreach trips and provides insights on where to focus attention if we are to follow a robust approach that plans for worst-case scenarios in order to comply with WHO-EPI guidelines to provide universal coverage.

5.0 Learning Combined Set Covering and Traveling Salesman Problem

In Chapter 4 we solved a combination of a set covering problem (SCP) and a vehicle routing problem with time windows (VRPTW). In this chapter we study a related but different combination of a Set Covering Problem and a Traveling Salesman Problem (TSP). The Traveling Salesman Problem is one of the most intensively studied combinatorial optimization problems due both to its range of real-world applications and its computational complexity. When combined with the Set Covering Problem, it raises even more issues related to tractability and scalability. In this chapter, we study a combined Set Covering and Traveling Salesman problem and provide a mixed integer programming formulation to solve the problem. Motivated by applications where the optimal policy needs to be updated on a regular basis and repetitively solving this via MIP can be computationally expensive, we propose a machine learning approach to effectively deal with this problem by providing an opportunity to learn from historical optimal solutions that are derived from the MIP formulation. We also present a case study using the World Health Organization's vaccine distribution chain, and provide numerical results with data derived from four countries in sub-Saharan Africa. Our results show that while the novel machine learning based mechanism generates high quality solution repeatedly for problems that resemble instances in the training set, it does not generalize as well on a different set of optimization problems. These mixed results indicate that there are promising research opportunities to use machine learning to achieve tractability and scalability. The remainder of this chapter is based this work (Yang & Rajgopal, 2020b).

5.1 Introduction and Literature Review

The Traveling Salesman Problem (TSP) is among the most intensively studied combinatorial optimization problems by both the operations research and the machine learning communities. It has been widely applied in areas such as planning, manufacturing, genetics, neuroscience, telecommunication, healthcare, supply chains and logistics (Applegate et al., 2011). The objective in TSP is to find the shortest route that visits each of a given set of locations and returns to the origin. We consider a TSP in combination with the Set Covering Problem (SCP), where the goal of the SCP in this context is to select an optimal subset of locations for facilities, so as to "cover" all locations in the original set of locations (Berman & Krass, 2002; Church & ReVelle, 1974). SCP is also a well-studied problem in the operations research community and has been broadly applied in areas such as facility planning, healthcare, and supply chains (Daskin & Dean, 2004).

TSP by itself is a hard problem. Exact algorithms such as branch-and-bound, cutting-plane and branch-and-cut methods often slow down when the number of nodes exceeds a few hundred (Applegate et al., 2011). Large-scale TSP is thus mainly solved by approximation algorithms and heuristics to find high quality solutions that are within 2–3% (say) of the optimum (Johnson et al., 2007; Johnson & Mcgeoch, 1997; Johnson & McGeoch, 2007; Rego et al., 2011). Some of the widely used TSP approximation algorithms and heuristics include Christofides' algorithm (Christofides, 1976), 2-opt moves (Croes, 1958), 3-opt moves (Bock, 1958; Lin, 1965), the Lin–Kernighan (LK) method (Christofides & Eilon, 1972; Lin & Kernighan, 1973), large-step Markov chains (Martin, Otto, & Felten, 1991; 1992), stem-and-cycle method (Pesch & Glover, 1997; Rego, 1998), and ant colony optimization (Dorigo & Gambardella, 1997). Interested readers can refer to

(Applegate et al., 2011; Gutin & Punnen, 2007; Purkayastha et al., 2020; Rego et al., 2011) for more details about TSP approximation algorithms and heuristics.

When combined with SCP, TSP takes on even more complexity. With a given set of facility locations, TSP only considers transportation costs and its binary decision variables only determine whether or not location i is followed by location j. When combined with SCP, the problem first needs to identify locations at which to open facilities, and then come up with a route that visits each such location before returning to the origin. In doing this it needs to consider an overall cost that is the sum of facility costs, customer assignment costs, and transportation costs. The binary decision variables determine whether or not a facility is open at location i, whether a location j is assigned to an open facility at location i, and whether location k is followed by location l in the TSP route. Clearly, the combined SCP and TSP problem raises more issues related to tractability and scalability than either SCP or TSP separately.

In Section 5.2 we formulate the combined Set Covering and Traveling Salesman Problem as a mixed integer program (MIP). Solving this MIP can theoretically provide us with the optimal solution, but the run time explodes exponentially as the problem size increases. In addition, every time we re-solve the MIP model using new inputs and parameters, it typically starts from scratch and there is often no obvious method to incorporate information from historical solutions. Therefore, in any application where the above combinatorial problem needs to be solved repeatedly over time with different input values, an approach that relies on MIP can become computationally expensive. For example, in an application that we describe in Section 5.4, the problem needs to be solved and the solution needs to be updated every month over hundreds of locations.

If a training dataset can be derived from a set of historically solved problems, machine learning (ML) could be a natural candidate to effectively deal with this repeated combinatorial situation. The proposed work aims at providing a tractable and scalable learning-based approach to solve the combined SCP and TSP problems. We first formulate this problem as a mixed integer programming model and discuss relevant issues in Section 5.2. We then propose a learning-based mechanism to efficiently deal with this problem in Section 5.3. In Section 5.4, we provide an illustration of our approach with an application to mobile clinics and vaccination outreach that arises in the context of the World Health Organization's (WHO) Expanded Programme on Immunization (EPI), and present numerical results using data derived from the WHO and four countries in sub-Saharan Africa. Finally, Section 5.5 discusses some related issues and Section 5.6 summarizes this chapter.

Several recent studies have been conducted by both the operations research and the machine learning communities to try and incorporate machine learning into combinatorial optimization problems. Many of these studies aim at end-to-end learning methods that train the ML model from discrete optimization problems and directly output solutions from the input instance. For instance, Vinyals et al. introduced the pointer network (Vinyals, Fortunato, & Jaitly, 2015) as a recurrent neural network (RNN) that sequentially takes all the nodes in the graph as input and outputs the TSP route using a mechanism that is similar to the graph attention mechanism (Veličković et al., 2018) that is normally used to focus only on a subset of the input. Using a similar model, Bello et al. trained a reinforcement learning model and defined its reward signals as tour lengths (Bello et al., 2016). Instead of using RNN to process the input, Kool and Welling utilized graph neural networks (GNN) (Scarselli et al., 2009) after adding attention to establish a similar model (Kool & Welling, 2019). GNN can also be derived to learn the node selection policy (Dai et al., 2017). Using a different approach, Nowak et al. approximated a double stochastic matrix (Nowak at el., 2018) and Emami and Ranka used Sinkhorn Policy Gradient (Emami & Ranka, 2018) in the GNN output in order to characterize the permutation. Kaempfer and Wolf developed a Multiple Traveling Salesmen Problem solver using Permutation Invariant Pooling Networks (Kaempfer & Wolf, 2018). Bronstein et al. overviewed geometric deep learning problems with several applications, solutions, difficulties, and future directions (Bronstein et al., 2017).

Machine learning can also be used to retrieve meaningful properties of optimization problems, or even alongside the optimization as part of the optimization algorithm. This class of approaches include learning to branch (Alvarez, Louveaux, & Wehenkel, 2017; Balcan et al., 2018; He, Iii, & Eisner, 2014; Khalil et al., 2016; Khalil et al., 2017; Lodi & Zarpellon, 2017), learning to cut (Baltean-Lugojan et al., 2018; Tang, Agrawal, & Faenza, 2019), learning when to use Dantzig-Wolf decomposition (Kruber, Lübbecke, & Parmentier, 2017), learning how to disaggregate the problem (Yang et al., 2020), learning where to linearize a mixed integer quadratic problem (Bonami, Lodi, & Zarpellon, 2018), learning tactical solutions under imperfect information (Larsen et al., 2019), and learning as a modeling tool (Lombardi & Milano, 2018). These studies, including those end-to-end learning approaches, often face feasibility, modeling, scaling, and data generation challenges, and are still mostly in the exploratory stages (Bengio, Lodi, & Prouvost, 2018).

On the other hand, deep learning, as a sub-field of machine learning, has advanced dramatically with the growth of large datasets and computational power, and has led to breakthroughs on a variety of tasks including speech recognition, machine translation, objective detection, and computer vision (Lecun, Bengio, & Hinton, 2015). Deep learning often outperforms other learning algorithms when exploring high dimensional spaces and large datasets (Deng & Yu, 2014; Goodfellow, Bengio, & Courville, 2016). Many researchers have been using deep learning to solve combinatorial optimization problems. In fact, many of the approaches mentioned above belong to this category; these include Pointer Networks (Bello et al., 2016; (Vinyals, Fortunato, & Jaitly, 2015), GNN (Dai et al., 2017; Kool & Welling, 2019; Nowak at el., 2018; Scarselli et al., 2009; Veličković et al., 2018), Sinkhorn Policy Gradient (Emami & Ranka, 2018), Permutation Invariant Pooling Networks (Kaempfer & Wolf, 2018), Geometric learning (Bronstein et al., 2017), and others (Baltean-Lugojan et al., 2018; Kruber, Lübbecke, & Parmentier, 2017; Larsen et al., 2019; Tang, Agrawal, & Faenza, 2019). This paper aims at providing one of the early approaches for an end-to-end learning algorithm for a particular combinatorial optimization problem via deep learning.

5.2 The MIP Formulation

The Combined Set Covering and Traveling Salesmen Problem can be formulated using the following mathematical programming model:

Parameters:

n: Total number of targeted population centers

i: Index of locations. $1 \leq i \leq n$ for targeted customer locations; $i = 0, n+1$ for the origin

f_i: Fixed cost of running a facility at location i

c_{ij}: Variable cost of assigning a location j to a facility at location i; $c_{ii} = 0$

p: Average transportation cost per unit of distance

d_{ij}: Distance between location i and location j $\left(\text{with } d_{ii} = 0 \text{ and } d_{ij} = d_{ji}\right)$

D: Maximal coverage distance (MCD)

$a_{ij} \in \{0,1\}$: 1 if location *j* is within a distance *D* from location *i*

Variables:

$X_{ij} \in \{0,1\}$: 1 if location *j* is assigned to facility at location *i*, 0 otherwise;

$Y_i \in \{0,1\}$: 1 if there is a facility at location i, 0 otherwise

$Z_{ij} \in \{0,1\}$: 1 if location *i* is followed by location *j* on the trip route

U_i: Cumulative number of stops visited during a trip when arriving at facility location *i*

<u>Program MIP-3</u>:

$$Min \sum_{1 \le i \le n} f_i Y_i + \sum_i \sum_{1 \le j \le n} c_{ij} X_{ij} + \sum_i \sum_j p d_{ij} Z_{ij} \tag{5-1}$$

subject to

$$X_{ij} \le a_{ij} \qquad for\ \forall i, j \tag{5-2}$$

$$X_{ij} \le Y_i \qquad for\ \forall 0 \le i \le n, 1 \le j \le n \tag{5-3}$$

$$\sum_{i \le n} X_{ij} = 1 \qquad for\ \forall 1 \le j \le n \tag{5-4}$$

$$\sum_j Z_{0j} = 1 \tag{5-5}$$

$$\sum_j Z_{j0} = 0 \tag{5-6}$$

$$\sum_i Z_{i(n+1)} = 1 \tag{5-7}$$

$$\sum_i Z_{(n+1)i} = 0 \tag{5-8}$$

$$\sum_j Z_{ij} = \sum_j Z_{ji} \qquad for\ \forall 1 \leq i \leq n, \tag{5-9}$$

$$\sum_j Z_{ij} = Y_i \qquad for\ \forall 1 \leq i \leq n \tag{5-10}$$

$$U_0 = 0 \tag{5-11}$$

$$1 \leq U_i \leq \sum_i Y_i \qquad for\ \forall i \geq 1 \tag{5-12}$$

$$U_i - U_j + MZ_{ij} \leq M - Y_j \qquad for\ \forall i, j \tag{5-13}$$

$$Z_{ii} = 0 \qquad for\ \forall i \tag{5-14}$$

$$X_{ij} \in \{0, 1\} \qquad for\ \forall i, j \tag{5-15}$$

$$Y_i \in \{0, 1\} \qquad for\ \forall i \tag{5-16}$$

$$Z_{ij} \in \{0, 1\} \qquad for\ \forall i, j \tag{5-17}$$

$$U_i \geq 0 \qquad for\ \forall i \tag{5-18}$$

Note that the objective function (5-1) minimizes the sum of the facility location, assignment, and transportation costs. Constraint set (5-2) ensures that a location can only be assigned to a facility that is within the maximal coverage distance (MCD), in which case $a_{ij} = 1$, and 0 otherwise. Constraint set (5-3) ensures that the assignment is to an existing facility. Constraint set (5-4) ensures that each location is assigned to exactly one facility. These three sets of constraints define a typical Set Covering Problem.

The next few sets of constraints relate to the Traveling Salesman Problem. Note that node 0 denotes the beginning node of a trip and node (n+1) is the final node of a trip; both represent the origin. Constraint sets (5-5) and (5-6) imply that the trip departs from the origin (0) exactly once, while Constraint sets (5-7) and (5-8) imply that the trip enters back into the origin (n+1) exactly once. Constraint set (5-9) ensures that the flow that enters and departs any location i is balanced. Constraints (5-5) – (5-9) thus ensure that the trip is indeed a (0)-(n+1) path.

Constraint set (5-10) states that the trip enters and departs a location i exactly once if there is a facility at this location (i.e., $Y_i = 1$). Constraint set (5-11) – (5-13) is the MTZ subtour elimination constraints introduced by Miller, Tucker, and Zemlin (Miller, Zemlin, & Tucker, 1960). Note that if j follows i in the TSP route, $Z_{ij} = 1$ implies $Y_j = 1$ and $U_j \geq U_i + Y_j > U_i$. Suppose there exists a subtour $(i, j, \ldots i)$, with $i \neq 0, n+1$. Then, $U_j > U_i > U_j$ leads to a contradiction. Note that if location i is not a part of the trip (in which case $Y_i = 0$ by Constraint set (5-10)), the values of U_i are irrelevant to the problem as long as they satisfy the constraints. Constraint sets (5-14) – (5-18) are self-explanatory.

Having to solve this MIP on a regular basis can be computationally expensive or even impossible when n is large. In the next section, we introduce a machine learning approach that leverages the MIP to solve this problem.

5.3 Proposed Learning Mechanism

We start by formally defining the learning models to solve the combined SCP and TSP, and discuss how we iteratively update the model parameters. Suppose the optimal solution to **MIP-3** is given by the vector $(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$. The proposed framework is to establish two supervised machine learning models that can be trained to find this vector given a set of problem parameters. In particular, given a graph $\boldsymbol{G}$ along with its associated location and cost information, **Model 1** will be used to derive $(\boldsymbol{X}^*, \boldsymbol{Y}^*)$. That is, **Model 1** is trained to provide the portion of the optimal solution corresponding to the set covering problem, while also accounting for the TSP cost that this results in (in addition to the location and assignment costs). Recall that Y_j, as a component of $\boldsymbol{Y}$, is a binary variable that represents whether the facility in a particular location j is open or not. **Model 1** is trained to generate each component Y_j of the vector $\boldsymbol{Y}$ as a value between 0 and 1 that corresponds to the probability that in the optimum solution, the facility at location j is open. Denoting this vector of probabilities by $\boldsymbol{g}(\boldsymbol{Y})$ we could define **Model 1** via the following map:

Model 1 (SCP predictor**)**: $\boldsymbol{G} \rightarrow \boldsymbol{g}(\boldsymbol{Y})$

Note that $\boldsymbol{g}(\boldsymbol{Y})$ is used to derive a vector $\widehat{\boldsymbol{Y}}$ that is a prediction of $\boldsymbol{Y}$, and also a vector $\widehat{\boldsymbol{X}}$ that is a prediction of assignments given by $\boldsymbol{X}$. We will discuss this later in Section 5.3.4.

On the other hand, **Model 2** establishes the relationship between $(\boldsymbol{G}, \boldsymbol{Y})$ and $(\boldsymbol{Z})$, i.e., it is trained to use the graph described by $\boldsymbol{G}$ and the facility locations defined by $\boldsymbol{Y}^*$ to determine the optimal TSP sequence in $\boldsymbol{Z}^*$. In particular, given $\boldsymbol{Y}$, **Model 2** is trained to generate a set of values between 0 and 1 for each Z_{ij} that correspond to the probability that location i is followed by location j in the optimal sequence. Defining these probabilities by matrix $\boldsymbol{p}(\boldsymbol{Z})$, we could define **Model 2** via the following map:

Model 2 (TSP predictor**)**: $(\boldsymbol{G}, \boldsymbol{Y}) \rightarrow \boldsymbol{p}(\boldsymbol{Z})$

In the following sections we provide the detailed methodology to train each of **Model 1** and **Model 2**, and discuss how the models can be used to derive a combined solution that is guaranteed to be feasible.

5.3.1 Data Generation and Labeling

To begin with, we generate a data set **Train** with graph information $\boldsymbol{G}$ that contains candidate facility locations and all relevant cost information. We also generate an example data set **Test** with graph information $\boldsymbol{G'}$. For evaluation purposes, the data sets $\boldsymbol{G'}$ and $\boldsymbol{G}$ should be drawn from the same overall pool, i.e., the elements of these two data sets, although different, resemble each other in some sense. Finally, we generate another data set **Test$_{\text{new}}$** with graph information $\boldsymbol{G^{new}}$ that represents significantly different problems that are unseen in data sets **Train** and **Test**.

We then use the MIP formulation that is proposed in Section 5.2 to solve the optimization problem on each of the instances in $\boldsymbol{G}$, $\boldsymbol{G'}$, and $\boldsymbol{G^{new}}$. For each instance, we use the optimal solution $(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$ to generate the corresponding vectors $\boldsymbol{g}(\boldsymbol{Y}^*)$ and $\boldsymbol{p}(\boldsymbol{Z}^*)$; note that for these instances $\boldsymbol{g}(\boldsymbol{Y}^*)$ and $\boldsymbol{p}(\boldsymbol{Z}^*)$ have values of 0 or 1 for all components since the "probabilities" from our optimal solutions are either 0 or 1. We also document each of the individual cost components in the optimal solution, namely, the optimal facility location cost, assignment cost, and TSP cost.

5.3.2 Training Machine Learning Model 1 (SCP Predictor)

In this step, we establish a Neural Network on **Train** that maps $\boldsymbol{G}$ to $\boldsymbol{g}(\boldsymbol{Y})$, where the j^{th} element of $\boldsymbol{g}(\boldsymbol{Y})$ is the probability that $Y_j^* = 1$. From a pure optimization perspective, we would ideally like this value to be 1 or 0, depending on whether the facility at j is open or closed. Our model is trained via back propagation, and the hyperparameters in the learning process, such as the network architecture, the number of hidden layers and hidden units, the learning rate, the mini-batch size, activation functions and the number of epochs are tuned iteratively. To accomplish this, we further split dataset **Train** into a training set that comprises 90% of the elements in **Train** and a validation set that comprises the remaining 10% of the data set. Within an iteration, the hyperparameters are fixed and the model is trained using the training set with a pre-set number of epochs (one of the hyperparameters). At the end of each iteration, we obtain a model corresponding to the current set of hyperparameters. We then evaluate this model on the validation set. The hyperparameters are then altered for the next iteration based on how the model performs on the validation set. If overfitting is observed, i.e., the model performs significantly better on the training set and worse on the validation set, then regularization layers (typically Dropout layers) are added, and the number of activations and layers are reduced. On the other hand, if underfitting is observed, i.e., the model performs poorly on both sets, then more layers and activations are added. This process is repeated until no significant improvements are observed.

It is important to note that in general, there is no guarantee that the integer values obtained by directly rounding the fractional $\boldsymbol{g}(\boldsymbol{Y})$ will lead to the optimal solution of **MIP-3**. In fact, these integer values might not even lead to feasible solution. In Section 5.3.4 we will present a detailed discussion on how to convert $\boldsymbol{g}(\boldsymbol{Y})$ into a feasible Combined SCP and TSP solution.

5.3.3 Training Machine Learning Model 2 (TSP Predictor)

We train another Neural Network on **Train** that maps $(\boldsymbol{G}, \boldsymbol{Y})$ to $\boldsymbol{p}(\boldsymbol{Z})$, where element (i, j) of $\boldsymbol{p}(\boldsymbol{Z})$ is the probability that $Z^*_{ij} = 1$. That is, it uses the graph information described by $\boldsymbol{G}$ and the facility locations defined by $\boldsymbol{Y}$ to generate a matrix of probabilities $\boldsymbol{p}(\boldsymbol{Z})$ corresponding to the optimal TSP solution. The training process for **Model 2** also follows the same approach described in Section 5.3.2 for **Model 1** of dividing **Train** into a training and validation set with similar hyperparameter tuning over a predetermined number of epochs. In the training process, we use the optimal vector $\boldsymbol{Y}^*$ with its associated TSP path since the objective here is to train this model to find the optimal path for a given set of locations on a given graph. In the next section we discuss how to utilize **Model 2** to generate a TSP route from $\boldsymbol{p}(\boldsymbol{Z})$ that visits each open location.

5.3.4 Obtaining a Feasible Solution to MIP-3

In Section 5.3.2 and Section 5.3.3, we developed probability vectors $\boldsymbol{g}(\boldsymbol{Y})$ and probability matrix $\boldsymbol{p}(\boldsymbol{Z})$. In this section, we use these results to derive a feasible solution to the combined SCP/TSP. Recall that $\boldsymbol{g}(\boldsymbol{Y})$ corresponds to the probabilities associated with opening a facility at each location. For each example, we define a threshold α and let $\hat{Y}_i = 1$ if $g(Y_i) \geq \alpha$ and $\hat{Y}_j = 0$ for all j with $g(Y_i) < \alpha$, i.e., we use α to convert the vector of probabilities $\boldsymbol{g}(\boldsymbol{Y})$ into the discrete estimator $\hat{\boldsymbol{Y}}$ for use as the SCP solution. Note that in order to ensure that $\hat{\boldsymbol{Y}}$ is feasible, we need to ensure that (i) every location i either has an open facility, or (ii) is assigned to a location j with an open facility that is within a distance D from it, i.e., $\hat{Y}_j = 1$ with $a_{ij} = 1$. We use traversal search in **Algorithm 4** to go over all examples and locations to check if either of these two conditions is meet and force any location i that does not meet either to have an open facility. Note that all locations with open facilities are obviously assigned to the facility there, i.e., $\hat{X}_{ii} = 1$ if $\hat{Y}_i = 1$, and any location where there is no open facility is assigned to the feasible open facility location that leads to the smallest assignment cost. The algorithm is summarized in Table 22.

Table 22 Algorithm 4 to obtain SCP solution

Input:

Optimization parameters a_{ij}^{exp}, d_{ij}^{exp}, c_{ij}^{exp}, f_i^{exp} for each example exp and location i, j

Machine learning **Model 1** result $g^{exp}(Y_i)$ for each example exp and location i

Parameter α

---

for exp in index of $\boldsymbol{G}$, $\boldsymbol{G'}$, or $\boldsymbol{G^{new}}$:

    for i in Index of locations:

        let $\hat{X}_{ij}^{exp}$ = 0 for all j in index of locations

        if $g^{exp}(Y_i) < \alpha$:

            if $\exists$ $\boldsymbol{K}$ such that $\forall$ $k \in \boldsymbol{K}$, a_{ij}^{exp} = 1 and $g^{exp}(Y_i) > \alpha$:

                let $\hat{Y}_i^{exp}$ = 0

                let $\hat{X}_{ji}^{exp}$ = 1 where $j = \min_{l \in \boldsymbol{L}} l$, and $\boldsymbol{L} = \operatorname{argmin}_{k \in \boldsymbol{K}} c_{ki}^{exp}$

            else:

                let $\hat{Y}_i^{exp}$ = 1

                let $\hat{X}_{ii}^{exp}$ = 1

        else:

            let $\hat{Y}_i^{exp}$ = 1

            let $\hat{X}_{ii}^{exp}$ = 1

Output:

SCP solution $\hat{\boldsymbol{X}}$ and $\hat{\boldsymbol{Y}}$ with $\hat{X}_{ij}^{exp}$, $\hat{Y}_i^{exp}$ for each example exp and location i, j

---

Proposition 6: $\hat{\boldsymbol{X}}$ and $\hat{\boldsymbol{Y}}$ that are generated by **Algorithm 4** satisfy Constraint sets (5-2) – (5-4) in **MIP-3**.

Proof: The outer for loop iterates through each of all examples in our **Train**, **Test**, and **Test$_{\mathbf{new}}$** dataset, which corresponds to an optimization problem. The inner *for* loop further iterates all facility location i, whose $g^{exp}(Y_i)$ should be either $< \alpha$ or $\geq \alpha$ that are represent by the outer if/else conditional statement. In the first part of the inner *if/else* statement, Constraint set (5-2) and

(5-3) are satisfied by the condition, i.e., X_{ji}^{exp} = 1 only if a_{ij}^{exp} = 1 and $g^{exp}(Y_i) > \alpha$ (which leads to $\hat{Y}_i^{exp}$ = 1 in the second part of the outer *if/else* conditional statement). In all other parts, Constraint sets (5-2) and (5-3) are also satisfied by ensuring that $\hat{X}_{ii}^{exp}$ = 1 with $\hat{Y}_i^{exp}$ = 1 and a_{ii}^{exp} = 1. Additionally, throughout the inner and outer *if/else* conditional statement, there exists one and only one j such that X_{ji}^{exp} = 1. Note that j is the location with smallest assignment cost (and with the smallest index if there are multiple such j), or j = i.

□

So far we have obtained $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$ that constitute a feasible solution to the SCP. Next, we input ($\boldsymbol{G}$, $\widehat{\boldsymbol{Y}}$**)** ($\boldsymbol{G'}$ or $\boldsymbol{G^{new}}$ in **Test** and **Testnew** respectively) into **Model 2** to obtain $\boldsymbol{p}(\boldsymbol{Z})$. Note that we do not input the true optimal solution $\boldsymbol{Y}^*$ into **Model 2**, because $\boldsymbol{Y}^*$ is assumed to be unknown when conducting a valid end-to-end evaluation and comparison. In order to derive from $\boldsymbol{p}(\boldsymbol{Z})$ a feasible set of TSP routes given by $\widehat{\boldsymbol{Z}}$, the routes are required to visit each of the open location in $\widehat{\boldsymbol{Y}}$ once, and only once. Note that the routes have to start from the origin (indexed by 0) and return to the origin (indexed by n+1). We start the transformation process with node 0, and look at $\boldsymbol{S^{exp}}$, the set of open facilities in an instance exp with $\{i$: $\hat{Y}_i^{exp}$ = 1 for all $i \leq n\}$. We find the node j in $\boldsymbol{S^{exp}}$ with the largest probability $\hat{p}^{exp}(Z_{0j})$ (and with the smallest index if there are multiple such j), i.e., j is the most promising stop to follow node 0 in the optimal TSP route. We let $\hat{Z}_{0j}$ = 1 and remove j from $\boldsymbol{S^{exp}}$. We continue with a similar process while adding open facilities to the TSP route until $\boldsymbol{S^{exp}}$ is empty. We summarize this algorithm in Table 23.

Table 23 Algorithm 5 to obtain TSP solution

Input:

Machine learning **Model 2** result $\hat{p}^{exp}(Z_{ij})$ for each example exp and locations i, j

Set of open facility $\boldsymbol{S}^{\boldsymbol{exp}} = \{i: \hat{Y}_i^{exp} = 1 \text{ for all } i \leq n\}$

---

for exp **in** index of $\boldsymbol{G}$, $\boldsymbol{G}'$, and $\boldsymbol{G}^{\boldsymbol{new}}$:

let $\hat{Z}_{ij}^{exp} = 0$ for all i and j in index of locations

let $i = 0$

let $\boldsymbol{S}^{\boldsymbol{exp}} = \boldsymbol{S}^{\boldsymbol{exp}} - \{i\}$

while $\boldsymbol{S}^{\boldsymbol{exp}} \neq \phi$:

let $\hat{Z}_{ij}^{exp} = 1$ where $j = \min_{l \in \boldsymbol{L}} l$, and $\boldsymbol{L} = \operatorname{argmax}_{k \in \boldsymbol{S}^{\boldsymbol{exp}}} \hat{p}^{exp}(Z_{ik})$

let $\boldsymbol{S}^{\boldsymbol{exp}} = \boldsymbol{S}^{\boldsymbol{exp}} - \{j\}$

let $i = j$

let $\hat{Z}_{i(n+1)}^{exp} = 1$

Output:

TSP solution $\hat{\boldsymbol{Z}}$ with $\hat{Z}_{ij}^{exp}$ for each example exp and location i, j

---

Proposition 7: $\hat{\boldsymbol{Z}}$ that are generated by **Algorithm 5** satisfy Constraint sets (5-5) – (5-14) in **MIP-3**.

Proof: The outer *for* loop iterates through each of the examples in our **Train**, **Test**, or **Test$_{\text{new}}$** dataset, which corresponds to an optimization problem. Lines 1, 2 and 3 in the *for* loop ensures that Constraint sets (5-5) and (5-6) are satisfied, and the last line ensures that Constraint set (5-7) and (5-8) are satisfied. Note that we have $\hat{Z}_{0j}^{exp} = 1$ and $\hat{Z}_{i(n+1)}^{exp} = 1$ where j is the node selected by the first line in the first iteration in the while loop, while i is the last node selected when the while loop terminates. Constraint sets (5-9) and (5-10) are ensured by iteratively selecting one, and only one j throughout the iterations in the *while* loop and assign $\hat{Z}_{ij}^{exp} = 1$; the j then becomes the next i and the next j is then selected similarly until all node in $\boldsymbol{S}^{\boldsymbol{exp}}$ are visited. Therefore, $\forall 1 \leq i \leq n$, we have at iteration k, if i in $\boldsymbol{S}^{\boldsymbol{exp}}$ = $\{i$: $\hat{Y}_i^{exp}$ = 1 for all $i \leq n\}$, $\sum_j Z_{ij}$ = Z_{ij} = 1 = Y_i, and in the previous iteration, $\sum_j Z_{ji}$ = $Z_{i^{k-1}j^{k-1}}$ = $Z_{i^{k-1}i}$ = 1, where i^{k-1} and j^{k-1} are the previous i and j. On the other hand, if $i \notin \boldsymbol{S}^{\boldsymbol{exp}}$, $\sum_j Z_{ij}$ = $\sum_j Z_{ji}$ = Y_i = 0. Recall that Constraint sets (5-11) – (5-13) is the MTZ constraints that ensure that no subtour is present in the solution, where a subtour is defined as a (0)-(n+1) path that does not visit all of the open facilities in $\boldsymbol{S}^{\boldsymbol{exp}}$. Note that according to the definition of $\boldsymbol{S}^{\boldsymbol{exp}}$, node ($n$+1) can never be inside $\boldsymbol{S}^{\boldsymbol{exp}}$. Thus, the *while* loop will never set $\hat{Z}_{i(n+1)}^{exp}$ = 1 if there exists another node in $\boldsymbol{S}^{\boldsymbol{exp}}$. Constraint set (5-14) is also ensured because we set $\boldsymbol{S}^{\boldsymbol{exp}}$ = $\boldsymbol{S}^{\boldsymbol{exp}}$ – $\{j\}$ in previous iteration, and the next i (precious j) is already excluded from the selection.

□

We conclude this section with the following result:

Proposition 8: $\widehat{\boldsymbol{X}}$, $\widehat{\boldsymbol{Y}}$, and $\widehat{\boldsymbol{Z}}$ together constitute a feasible solution to the Combined Set Covering and Traveling Salesman Problem that is formulated as **MIP-3**.

Proof: From **Proposition 6** and **Proposition 7**, Constraint sets (5-2) – (5-14) are satisfied, and Constraint sets (5-15) – (5-18) are satisfied because $\widehat{\boldsymbol{X}}$, $\widehat{\boldsymbol{Y}}$, and $\widehat{\boldsymbol{Z}}$ only contains binary values. Note that the convenience variable U_i in **MIP-3** is not given here and is unnecessary.

□

5.3.5 Evaluating the End-to-end Mechanism

We use the overall cost that is defined by the objective function (5-1) in **MIP-3** as the basis for evaluating the performance of our end-to-end ML-based mechanism to solve the Combined SCP and TSP problem. In the next section, we provide a detailed demonstration of how our approach performs by using a case study that is based upon the World Health Organization's vaccine distribution chain, and provide numerical results with data derived from four countries in sub-Saharan Africa.

5.4 An Illustration: Mobile Clinics and Outreach Operations

As we saw in Chapter 4, residents in remote locations in many low- and middle-income countries often have no (or limited) direct access to clinics and hospitals, and outreach is typically utilized to raise immunization rates. A set of these remote population centers are chosen for locating mobile clinics and a team of clinicians and support personnel set up these mobile clinics periodically to vaccinate people in the immediate surrounding area. As discussed in Chapter 4 there are a limited number of mathematical programming models to help determine optimal outreach strategies (e.g., Lim et al., 2016; Mofrad, 2016), but none of these models looks at the problem on an ongoing basis even though outreach in practice is done at regular intervals of time, and the underlying mathematical models are required to be solved repeatedly because the same plan is not followed each time. In Chapter 4 we presented a quantitative model that considers updated model parameters and obtains revised plans for subsequent planning periods using a two-period Robust approach. We presented a method to economically plan for outreach and provided management insights on where to focus more attention, but when solving the mathematical model, our approach starts from scratch every time that the MIP needs to be solved. Thus, there are no mechanisms to learn from historical optimization solutions. We now discuss a variation of the outreach problem of Chapter 4, but with slightly different assumptions, and use this as the basis for a combined SCP and TSP formulation.

5.4.1 Problem Development

We consider a version of the outreach problem from the previous chapter, with the main difference being that we only have a single outreach trip. For example, one may think of this as covering a subset of the population centers for a monthly outreach trip. A medical team departs from an existing clinic at a district center in a truck or van, while carrying vaccines in cold boxes along with related supplies. The team then sets up at one or more mobile clinic location(s) to vaccinate residents in that area as well as residents from nearby areas that are within the maximum coverage distance. In the outreach trip, the team goes to each of the locations sequentially and eventually returns to the original depot.

We take three sets of decisions into consideration: 1) choosing locations of mobile clinics as a subset of all targeted population centers to be covered; 2) assigning population centers to mobile clinics that are within the maximum coverage distance to that mobile clinic (each mobile clinic could serve multiple population centers, but a population center can only be assigned to one mobile clinic); and 3) vehicle trips that ensure that all mobile clinic locations are visited once and only once within some suitable planning horizon (e.g., 1 month). Within each planning horizon, only one vehicle trip is assumed to be undertaken and the vehicle must depart from a fixed depot and return to that depot after it visits all mobile clinic locations. The vehicles utilized in outreach trips are typically trucks or vans with several coolers or cold boxes, and since our target populations are in remote and sparsely populated area, we assume that these vehicles are not capacitated in terms of how much vaccine can be carried. Therefore, we assume that each trip could carry necessary clinical and support personnel along with the sufficient amount of vaccine for the location(s) that are served by the trip. In cases that the required vaccine volume is larger than the capacity, a larger vehicle is acquired or a shorter planning horizon can be leveraged to conduct more outreach trip throughout the year.

As opposite to the problem discussed in Chapter 4 that only consider two components of cost, three components of costs are considered in planning the outreach operation. First, we consider direct cost associated with running a mobile clinic at a particular location that includes the setup at the outreach site, labor costs for vaccination operations onsite, the cost of renting or obtaining space and storage devices, energy consumptions cost, and any other local cost. We also consider the cost of assigning population centers to a mobile clinic. This cost includes the cost of moving targeted newborns from other population centers without a mobile clinic, incentives paid to them to have them visit a mobile clinic, and estimated social and healthcare cost associated with people not visiting mobile clinics due to relative long travel distances. The assignment cost can be formulated as a linear function of distance from population centers to mobile clinic, although this is not crucial. Lastly, we consider trip-related cost that includes vehicle depreciation or vehicle rental costs, fuel costs, hourly wages/allowances paid to the team and driver. Note that this cost is assumed to be proportional to the duration of the trip, and we thus utilize an average cost per hour based on the trip duration to quantify it. In summary, the total cost is determined by the selected mobile clinics locations, distance from population centers to mobile clinics, and the route taken by the vehicle on the outreach trip.

This process can be viewed as an example of a combination of the Set Covering Problem and the Traveling Salesman Problem as discussed in Section 5.1. The selection of mobile clinics can be viewed as the selection of facility locations in $\boldsymbol{Y}$, assigning population centers to the selected mobile clinics can be considered via assignment variables $\boldsymbol{X}$**,** and the route to visit mobile clinics can be viewed as the TSP route $\boldsymbol{Z}$. Moreover, because the demand, traffic and road conditions in these targeted zones are typically unstable, it can be challenging to obtain an accurate set of estimates for these problem parameters ahead of time. This situation was discussed in detail in Section 4.4 and it would be ideal to determine a flexible strategy over every successive planning period. The outreach model thus needs to be solved periodically with similar parameter and inputs, where each instance corresponds to a graph with one depot and multiple population centers. The learning-based mechanism described in Section 5.3 therefore constitutes a promising direction from which to approach this problem, and we thus apply the methodology illustrated in Section 5.3 to solve this problem and present the numerical results using data derived from the WHO and four countries in sub-Saharan Africa.

5.4.2 Numerical Results

To present numerical results using the mobile clinic and outreach operations example, we selected 30 unique sets of targeted populations across different parts of four countries in sub-Saharan Africa, and for each such set, we generated the node set of locations and 1,000 different examples using different combinations of cost data. This generated a total of 30,000 different instances. We ran these 30,000 examples on **MIP-3** using Gurobi over a period of several weeks, and obtained the optimal solution vector $(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$ for each instance. We also documented each cost component (facility location cost, assignment cost, and TSP cost) in each of the optimal solutions.

We then randomly picked 5 distinct sets of population centers from the 30 we started with and assigned all of the corresponding 5,000 examples associated with the 5 graphs in this set to dataset $\textbf{Test}_{\textbf{new}}$. For the remaining 25 sets of population centers and the corresponding 25,000 instances, we randomly split these into 22,500 examples for dataset **Train** and 2,500 examples for dataset **Test**. Note that instances in **Train** and **Test** are drawn from the same pool of locations, so that the same population center and its associated graph could appear in both datasets, albeit with different cost information. However, the graphs in **Test$_{\text{new}}$** are all different from those in **Train** and **Test**. In other words, the instances in **Train** and **Test** bear some resemblance to each other, unlike **Test$_{\text{new}}$**, whose instances come from a completely different set of population centers and their graphs. After utilizing the method introduced in Section 5.3.2 and 5.3.3 to train the two machine learning models on **Train** with the data set split into training and validation sets and parameter tuning after each iteration, we obtain the final model parameters. Given any instance, we then use machine learning **Model 1** to obtain $\boldsymbol{g}(\boldsymbol{Y})$ followed by **Algorithm 4** in Section 5.3.4 to obtain SCP solution $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$. We then feed $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$ into machine learning **Model 2** to obtain $\boldsymbol{p}(\boldsymbol{Z})$ followed by **Algorithm 5** to obtain TSP solution $\widehat{\boldsymbol{Z}}$.

We use the datasets **Test** and **Test$_{\text{new}}$** to evaluate how well the machine learning models (along with the two algorithms for obtaining a final solution) perform on data that is distinct from the data used to train. To evaluate the end-to-end performance of the approach, we compare the total cost of the solution obtained with the minimum cost obtained by solving **MIP-3**. We started with a value of 0.5 for the threshold α described in Section 5.3.4 since this would appear to be a natural value for it if we interpret each element of $\boldsymbol{g}(\boldsymbol{Y})$ as the probability that the corresponding variable is equal to 1. We summarize the numerical results for each component of cost in Table 24. For each component and data set, the entry in Table 24 is the ratio of the sum of the costs for that component across all instances in the data set obtained from our approach, and the sum of the optimal costs for that component across the same instances obtained by solving the MIP (expressed as a percentage).

Table 24 Cost comparison of each component in Train, Test and Test_{new}

	# examples	Facility cost	Assignment cost	TSP cost	Total Cost
Train	22,500	100.09%	100.25%	102.02%	101.20%
Test	2,500	100.34%	100.42%	102.36%	101.49%
Test_{new}	5,000	147.56%	58.77%	158.80%	146.33%

Since examples in **Train** and **Test** are drawn from a common pool of population centers, while examples in **Test_{new}** are from a completely different set of population centers, it is natural that the approach will yield better results with instances in **Test** than with instances in **Test_{new}**. The numerical results from data set **Test** should give us a fair measurement of the performance of the mechanism on future examples from node sets that have been solved before albeit with different cost parameters. On the other hand, data set **Test_{new}** presents a more "tough" test to measure the model's generality, because these instances use node sets that were never part of the training process and thus no actual information on historical solutions is given to the learning mechanism.

As shown in Table 24, with instances in **Train** and **Test** our approach is able to generate heuristic solutions that are on average about 1% more expensive than the optimal solution from solving **MIP-3**. Because the mechanism utilized a train-validation splitting procedure within the backend training and hyperparameter tuning process, overfitting is not observed, and the mechanism preforms similarly on both data sets.

On the other hand, when dealing with optimization problems with a different pool of population centers that are not part of the training process, the model is not able to generate solutions that are as good. In particular, it appears that the machine learning approach performs notably worse on the **Test_{new}** set, and leads to facility costs that are 47.56% higher than the optimal cost. When it comes to new datasets that it has never seen, **Model 1** is not as good at identifying the correct combination of facility locations to cover all population centers, and consequently **Algorithm 4** turn out to be more "conservative" and opens a number of additional locations to ensure complete coverage, leading to higher facility costs. However, since more mobile clinics are selected, the corresponding total assignment costs are only 41.23% of what they are at the optimum, because on average, with more clinic locations there are fewer population centers assigned to each clinic and population centers now get assigned to clinics that are closer. More clinics also lead to outreach trip costs that are 58.80% higher, since there are now more stops to visit in the final solution. Overall, the cost is around 46% higher than the optimal total cost on average. Note that in this study, it is more common to have higher facility and transportation cost, because vaccine has to be transported and stored in a very narrow range of temperature, and often times special types of storage devices and vehicles have to be used. In the optimal solution, **MIP-3** thus tries to select fewer clinic locations, with each serving more population centers, as opposed to having more clinic locations serving fewer population centers. If the mechanism would be used in a scenario with more balanced cost components, we might expect it to perform better with the **Test$_{\mathbf{new}}$** data set.

One observation in solving via this mechanism that needs to be mentioned is run time. Although the training procedure can take several hours with the 22,500 observations, when it actually comes to predicting probabilities and translating these into a feasible solution for a specific instance, the time is negligible. On the other hand, solving the problem via **MIP-3** can take much more time, and in applications where an immediate solution is desirable, the proposed mechanism is more favorable over **MIP-3** once training has been completed. Overall, the mechanism is able to generate high quality results repeatedly for problems that resemble instances in the training set, but when it encounters totally new problems, it generates more expensive heuristic solutions but that are guaranteed to be feasible.

5.4.3 Selecting a Value for the Parameter α

In Section 5.4.2 we discussed the numerical results when parameter α is set to a natural value of 0.5 and is consistent throughout the implementation of **Algorithm 4** on **Train**, **Test** and **Test$_{\mathbf{new}}$**. However, as illustrated in Table 24, the total cost of our solutions on **Test$_{\mathbf{new}}$** is significantly higher than the total cost based on the optimal solution. In particular, the facility cost in our solutions is higher than the optimum, and more facilities then lead to longer TSP routes, which in turn are also higher than the optimum, and these together overwhelm the reductions in assignment costs. To further study parameter α and its impact on the total cost, we conducted a sensitivity analysis on parameter α. Table 25 reports each component of cost across instances in **Test$_{\mathbf{new}}$**, similar to what we had in Table 24 for $\alpha = 0.5$, but with different values of parameter α from 0.1 to 0.9 in **Algorithm 4**. Note that α is set to be consistent throughout each of the scenarios from $\alpha = 0.1$ to $\alpha = 0.9$.

Table 25 Sensitivity analysis of parameter α on $\mathbf{Test_{new}}$

α	Facility cost	Assignment cost	TSP cost	Total Cost
0.1	158.62%	49.62%	168.46%	154.92%
0.2	154.35%	52.90%	164.60%	151.51%
0.3	151.94%	54.97%	162.58%	149.69%
0.4	149.64%	56.96%	160.50%	147.88%
0.5	147.56%	58.77%	158.80%	146.33%
0.6	145.72%	60.52%	157.14%	144.90%
0.7	143.97%	62.24%	155.67%	143.59%
0.8	141.60%	64.63%	153.61%	141.79%
0.9	138.48%	67.75%	151.08%	139.52%

It can be seen that as α increases from 0.1 to 0.9, the predicted total cost in $\mathbf{Test_{new}}$ improves from being 54.92% higher to being 39.52% higher than the optimal solution, with a decrease in facility cost and TSP cost from 58.62% and 68.46% higher to 38.48% and 51.08% higher, respectively. This indicates that **Algorithm 4** appears to perform better with higher α values when the initial number of facility locations is relatively small. One possible explanation for this is that often times when two population centers are close to each other and also resemble each other in terms of their demand, from a machine learning perspective, **Model 1** would tend to predict similar probabilities for locations at both. In situations where these probabilities are relatively large and α is relatively small, **Algorithm 4** yields a solution with facilities open at both locations. However, in practice we would only want a facility to be open at one of these locations; the other one could be covered by the open facility if the distance between the two is within the

MCD, and many other population centers might be close enough to be served by either location. On the other hand, with a higher α, the likelihood of this happening is smaller because there will tend to be fewer open locations overall, thus reducing the likelihood of this type of duplication. Furthermore, when the probabilities are on either side of α, but the difference is small (e.g.,0.89 vs. 0.91, with α=0.9), higher values of α could prevent both from being opened (in our example, only one facility will be opened, whereas both would be open with α = 0.8.

To further study this, we treated α as a decision variable, and ran multiple threads of the process in Section 5.3.4 with different values of α simultaneously. Here, for each specific instance in **Test$_{\textbf{new}}$**, we picked the value of α that yields the minimum total cost (α^*_{exp}), and then calculate the corresponding cost components. In examining the values of α^*_{exp} that yielded the lowest total costs, we found that in 2,036 of the 5,000 instances (~41%), the choice of a value for α made no difference at all. This is because the majority of the probabilities in these instances are in the range (0, 0.1), or (0.9, 1.0), so that any value of α between 0.1 and 0.9 would give the same solution. In many other instances, there was a range of values for α^*_{exp} that yielded the same solution; in fact, in over 75% of the instances there were at least three (consecutive) values for α^*_{exp} that were optimal.

To understand this better, Figure 11 shows the count of different values of α that are optimal across the 5,000 instances in **Test$_{\textbf{new}}$**, noting again that in general we have multiple optimal of α^*_{exp} for almost all instances. Figure 11 also shows that our solutions improve as α increases and in 4,577 instances (~92%) the value of α=0.9 yielded the best solution from using our approach. In Table 26 we further compare each individual component of cost to the optimum, similar to what we had in Table 26. The first row in Table 26 summarizes costs when we pick $\alpha = \alpha^*_{exp}$ for each instance. Overall, our results indicate that simply picking a value of α=0.9 provides us with virtually the same result (139.52% from the last row of Table 25) as picking the best possible value for α (=α^*_{exp}) for each instance (138.75% from the first row of Table 26).

As a matter of interest we also show in each subsequent row of Table 26 cost comparisons from different values of α from 0.1 to 0.9, but only for those instances for which $\alpha = \alpha^*_{exp}$. Again, comparing the last row of Table 25 with the first and last rows of Table 26 shows that a value of 0.9 for α yields the best results for the vast majority of instances.

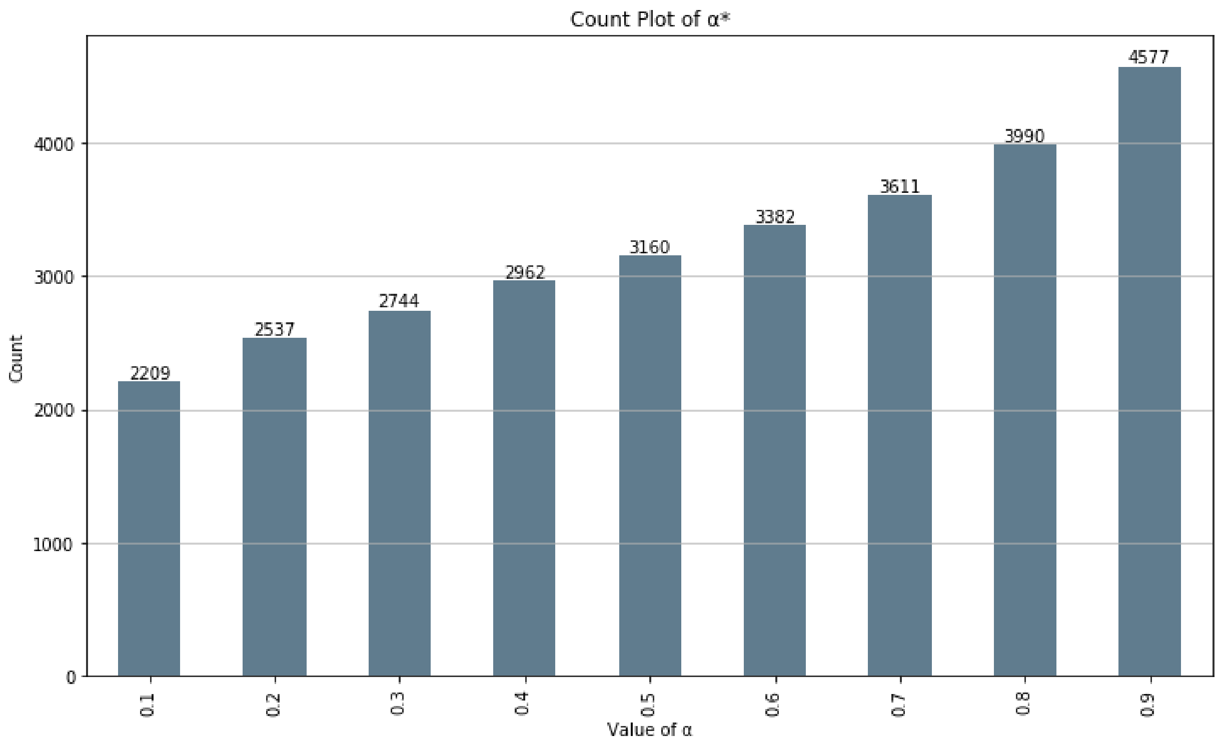

Figure 11 Count of examples in Testnew for which α^*_{exp} is best

Table 26 Cost comparison of each component in Testnew with different α^*_{exp}

α	Facility cost	Assignment cost	TSP cost	Total Cost
Overall	138.79%	66.82%	149.63%	138.75%
0.1	143.48%	58.94%	155.68%	143.79%
0.2	142.96%	59.64%	155.15%	143.20%
0.3	142.77%	60.55%	154.76%	142.87%
0.4	142.61%	61.08%	154.31%	142.56%
0.5	142.17%	61.83%	153.86%	142.16%
0.6	141.58%	62.54%	153.19%	141.67%
0.7	140.75%	63.67%	152.47%	141.00%
0.8	139.94%	64.99%	151.73%	140.30%
0.9	138.80%	66.86%	150.57%	139.32%

Note that in theory, α^*_{exp} could also be treat as a component of the dependent variables and can thus be modeled and predicted via a machine learning model. We will discuss this along with some other future research directions in Section 5.5.

5.5 Discussion

There are several limitations from this initial work that provide us with future research directions. First, as discussed at the end of Section 5.4.3, we could learn the optimal threshold parameter α^*_{exp} in **Algorithm 4** via machine learning, and include it as a component of the output in **Model 1**: **Model 1′:** $\boldsymbol{G} \rightarrow \boldsymbol{g}(\boldsymbol{Y}), \alpha$, i.e., **Model 1′** not only establishes the probability that in the optimum solution, the facility at location j is open, but also the threshold to consider opening the facility at location j in **Algorithm 4**. Note that the training process of **Model 1$'$** must be iterative, because the α^* is unknown until the different threads of the whole mechanism including **Model 2** is completed in parallel.

Second, as discussed in Section 5.4.3, **Model 1** tends to predict similar probability when two nearby population centers resemble each other. With a small α, we could end up with both having open facilities, and this results in much higher facility costs then necessary. This leads to another promising direction in the future to utilize techniques such as Convolutional Neural Networks (CNN) to ensure that the Set Covering model considers the correlation between population centers.

Third, there could be a mechanism with feedback between the models, as opposed to training each independently. As a new direction, we could leverage the information that is available within each machine learning model to help with improvement across both models. For example, the TSP cost that is calculated from **Model 2** can be viewed as a function of the output of **Model 1 (**which is the SCP solution). If the TSP cost that is incurred by a particular SCP solution can be utilized as intermediate information when training **Model 1**, it is possible that **Model 1** could generate better SCP solutions that leads to smaller overall cost. One way of doing this might be to initially use an approximation of the TSP cost to help speed up the learning process. Overall, this direction requires a more sophisticated methodology to implement.

Fourth, in Section 5.4, because all examples considered in the demonstration have a similar number of population centers (around 10), we utilized a standard, fully connected neural network in the implementation of **Model 1** and **Model 2**. This places some limitations on the generality of the mechanism and the possibility of utilizing the mechanism to solve problems with potentially more nodes. In many real-world circumstances, the number of facility locations could vary over a wider range and could be unknown before the start of the training process. This points to the possibility of using more general, sequential neural networks whose dimension of input is not fixed (unlike with the standard version we used). For example, using Recurrent Neural Networks (RNN) such as Long Short Term Memory (LSTM) networks could potentially alleviate this problem. In developing RNN, we can use a structure similar to LSTM that is widely used in Natural Langrage Processing and time-series analysis. Information that is related to a particular node in our graph can be fed into the RNN sequentially, and the predicted probabilities related to this node along with the information of the next node are then fed into the next unit of the RNN, until the information across all nodes is fed into the model and all prediction results are obtained.

Fifth, in this work the training procedure to establish the machine learning models is conducted on the entire training dataset before the prediction process. However, in most situations, especially when inputs are coming in as a flow or when optimal solutions may change dynamically over time because of changes in parameter values, the machine learning models also need to be updated over time. One natural way is to retrain the models on the entire training set over time after a given set of planning horizon, but this could be time consuming. Alternatively, we could define a preselected number of inputs as a "batch" and once we receive a whole batch of new input, we update the parameters of the machine learning models sequentially by batch. This technique is often referred to as online learning, or incremental learning, and is widely used when near real-time inputs are present.

Lastly, from an implementation standpoint, we would like to seek more real-world circumstances that are applicable for the mechanism. There are often two challenges. First, to obtain mechanisms that are more consistent and general, it is necessary to have a different number of nodes, and a wider range of different parameter distributions. The historical optimization solution should also be available to train the machine learning model. Secondly, there must be a need to have the optimization model solved repeatedly and rapidly.

5.6 Summary

This chapter aims to provide an early exploration and experiments in leveraging a machine learning algorithm to solve a difficult combinatorial optimization problem. We first study the combined Set Covering and Traveling Salesmen problem, and formulate this problem as a Mixed Integer Program. When the optimization problem needs to be solved on a regular basis with similar input values, it starts from scratch as new inputs and parameters are given. This could lead to high computational expense as well as tractability issues. To address this, we introduce a machine learning based mechanism to solve this problem by leveraging historical optimal solutions. The mechanism can be utilized to efficiently generate heuristic solutions via two machine learning models that are dedicated to solving the Set Covering Problem and the Traveling Salesmen Problem separately, but are aimed at minimizing overall cost. We discuss data generation and preprocessing, model training, and how to generate feasible solutions using the machine learning results. We also discuss how to compare the overall cost of the machine learning based mechanism to the optimal cost that is generated by the optimization formulation. We then present a detailed case study for the World Health Organization's vaccine distribution chain, and provide numerical results with data derived from four countries in sub-Saharan Africa after several train-test-evaluation iterations. Based on the computational performance observed, the machine learning based mechanism appears to be a promising way to achieve tractability and scalability without significantly sacrificing solution quality, but it still requires significant further development and should supplement the current exploratory approaches of incorporating machine learning with optimization.

6.0 Summary

Vaccination has been proven to be the most effective method to prevent illness, disability and death from infections. It is estimated that 2 to 3 million deaths are averted each year because of vaccines (World Health Organization, 2019b) and over the years, significant levels of coverage have been achieved. However, it is estimated that an additional 1.5 million deaths could be avoided annually if global vaccination coverage could improve further, and that even in the 21st century there are still almost 20 million infants worldwide who lack access to routine immunization services and remain at risk for vaccine-preventable diseases (World Health Organization, 2019c). This problem is especially pronounced in low and middle-income countries (LMICs) (Gavi, 2019), where some of the contributors to the problem include high costs, competing health priorities, lack of resources, inadequate infrastructure, poor monitoring and supervision, rigid distribution structures, and even vaccine hesitancy such as complacency, convenience and confidence (de Oliveira, Martinez, & Rocha, 2014; Gavi, 2019; Hotez, Nuzhath, & Colwell, 2020; Jarrett et al., 2015; MacDonald et al., 2015; Shen, Fields, & McQuestion, 2014; Yadav, et al., 2014; Zaffran, 1996).

The World Health Organization (WHO) established the Expanded Programme on Immunization (EPI) in 1974 with the goal of providing universal access to all important vaccines for all children (Bland & Clements, 1998). The program was further expanded with the formation of the Global Alliance for Vaccines and Immunization (Gavi) in 2000 to accelerate access to new vaccines in the poorest countries. EPI and Gavi together have successfully contributed to saving millions of lives worldwide by reducing mortality and even largely eliminating some diseases like polio and measles (Gavi, 2020; World Health Organization, 2013).

With the help of international organizations and new technological developments, many vaccines can now be obtained at low cost and in mass quantities. However, shipping, storing and delivering vaccines in a cost-efficient fashion while ensuring that vaccines are reliably available to end-users remains a major challenge. In particular, in many LMICs vaccines are usually distributed via a hierarchical legacy medical network, with locations and shipping routes of this network often determined by political boundaries and history. The overarching goal of this dissertation is to ensure that every child has access to vaccines, and along with this, in most LMICs the objective is to design a system that can be operated without the need for sophisticated logistics personnel and at minimum cost.

We discuss three specific problems. The first focuses on redesigning an improved vaccine network. As an alternative to the current structure, the network for vaccine distribution could be separated from the current legacy health network, while using some appropriate subset of these facilities and with vaccine flows along routes that differ from the current ones. However, the operation of the network cannot deviate from established WHO guidelines and needs to be simple because of the relative lack of sophisticated vaccine management abilities in LMICs. The main consideration in redesigning the vaccine network includes decisions on the choice of the best set of intermediate hubs from the set of current distribution center locations, obtaining optimal replenish frequencies for each hub, deciding on hub-to-hub connections and the clinic allocations to each hub, determining the actual vaccine flow along all connections, and lastly, selecting the types of storage and transportation devices to use at each location and along each flow path. The redesigned network is not required to follow the three, four or five tiered hierarchical structure that is currently the norm, nor are they required to follow the replenishment policy associated with a particular tier.

In Chapter 2, we present a mixed-integer optimization model and develop a column generation based algorithm to solve the MIP model and an iterative heuristic that cycles between solving restrictions of the original problem. We also present numerical tests using real data from different LMICs to study the performance of the algorithms. Because of the computational expense and tractability issues of the optimization model, in Chapter 3 we further present a novel MIP-based disaggregation-and-merging algorithm that is based on the simple observation that changes to the structure in a part of the network are unlikely to have a significant effect on the structure in other parts that are far away. The algorithm thus uses a divide-and-conquer approach to intelligently generate and solve a sequence of MIPs. Extensive tests based on real-world data derived from four different countries in sub-Saharan Africa show that it yields solutions that are optimal or within 0.5% of the best cost where optimality can be verified, and for large instances that are impossible to solve optimally, it is uniformly robust and yields good solutions in a few minutes.

In addition to the suboptimal structure and operations of a vaccine distribution network, another situation that could cause low vaccination rates is when resources are limited and there are population centers without access to direct clinic services. In this case, an approach known as *outreach* is typically utilized where a team of clinicians and support personnel is sent from an existing clinic to visit one or more of these locations to vaccinate residents in their immediate surrounding area. In Chapter 4 we focus on the problem of outreach and present a systematic way to plan for economical outreach operations by formulating the problem as a mixed integer program. We also study the issues related to the typical uncertainties associated with estimating demand for vaccines and planning individual outreach trips and provides insights on where to focus attention if we are to follow a robust approach that plans for worst-case scenarios in order to comply with WHO-EPI guidelines to provide universal coverage.

Finally, we provide an early exploration and experiments in leveraging a machine learning algorithm to solve a difficult combinatorial optimization problem. We look at the outreach problem in Chapter 4 from a different viewpoint and reformulate it as a problem combining the Set Covering Problem and the Traveling Salesmen Problem in Chapter 5, and formulate this problem as a Mixed Integer Program. When the optimization problem needs to be solved on a regular basis with similar input values, it starts from scratch as new inputs and parameters are given. This could lead to high computational expense as well as tractability issues. To address this, we introduce a machine learning based mechanism to solve this problem by leveraging historical optimal solutions. The mechanism can be utilized to efficiently generate high quality heuristic solution via two machine learning models that are dedicated to solving the Set Covering Problem and the Traveling Salesmen Problem separately, but are aimed at minimizing overall cost. We then present a detailed case study for the World Health Organization's vaccine distribution chain, and provide numerical results with data derived from four countries in sub-Saharan Africa with the outreach problem as an example. Based on results observed, the machine learning based mechanism appears to be a promising way to achieve tractability and scalability without significantly sacrificing solution quality, but it still requires significant further development and should supplement the current exploratory approaches of incorporating machine learning with optimization.

While we have addressed a diverse set of issues in optimizing the design and operations of WHO-EPI vaccine distribution chain, there are still open research topics in this area that need to be addressed. This includes the incorporation of uncertainty in the network design problem, more sophisticated vaccine inventory management policies, improvement of alternative outreach policies that can be standardized in the field, and more sophisticated machine learning based mechanisms for enhanced model performance and generality. Please see each chapter for more detailed discussions on future research directions.

In summary, with the encouraging solution quality and computational performance observed in extensive tests based on real-world data derived from four different countries in sub-Saharan Africa, this dissertation has provided additional mathematical models to analyze the issues raised and has developed powerful algorithms to solve these problems.

Bibliography

Acosta, D., Hendrickx, S., & McKune, S. (2019). The livestock vaccine supply chain: Why it matters and how it can help eradicate peste des petits Ruminants, based on findings in Karamoja, Uganda. *Vaccine*. Elsevier Ltd. https://doi.org/10.1016/j.vaccine.2019.09.011

Aggarwal, C. C.. (2015). Data Mining: The Textbook. *Springer International Publishing*, 746. https://doi.org/10.1007/978-3-319-14142-8

Alvarez, A. M., Louveaux, Q., & Wehenkel, L. (2017). A machine learning-based approximation of strong branching. *INFORMS Journal on Computing*, *29*(1), 185–195. https://doi.org/10.1287/ijoc.2016.0723

Applegate, D. L., Bixby, R. E., Chvátal, V., & Cook, W. J. (2011). *The traveling salesman problem: A computational study*. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press. https://doi.org/10.5860/choice.45-0928

Assi, T. M., Brown, S. T., Djibo, A., Norman, B. A., Rajgopal, J., Welling, J. S., … Lee, B. Y. (2011). Impact of changing the measles vaccine vial size on Niger's vaccine supply chain: A computational model. *BMC Public Health*, *11*(1), 425. https://doi.org/10.1186/1471-2458-11-425

Assi, T. M., Brown, S. T., Kone, S., Norman, B. A., Djibo, A., Connor, D. L., … Lee, B. Y. (2013). Removing the regional level from the Niger vaccine supply chain. *Vaccine*, *31*(26), 2828–2834. https://doi.org/10.1016/j.vaccine.2013.04.011

Balcan, M. F., Tyavis, D., Sandholm, T., & Vitercik, E. (2018). Learning to branch. *35th International Conference on Machine Learning, ICML 2018*, *1*, 582–618.

Balseiro, S. R., Loiseau, I., & Ramonet, J. (2011). An Ant Colony algorithm hybridized with insertion heuristics for the Time Dependent Vehicle Routing Problem with Time Windows. *Computers and Operations Research*, *38*(6), 954–966. https://doi.org/10.1016/j.cor.2010.10.011

Baltean-Lugojan, R., Bonami, P., Misener, R., & Tramontani, A. (2018). Selecting cutting planes for quadratic semidefinite outer-approximation via trained neural networks. *Technical Report, CPLEX Optimization, IBM*. Retrieved from http://www.optimization-online.org/DB_FILE/2018/11/6943.pdf

Baxby, D. (1999). Edward Jenner's inquiry; a bicentenary analysis. *Vaccine*, *17*(4), 301–307. https://doi.org/10.1016/S0264-410X(98)00207-2

Bello, I., Pham, H., Le, Q. V., Norouzi, M., & Bengio, S. (2016). Neural Combinatorial Optimization with Reinforcement Learning. *5th International Conference on Learning Representations, ICLR 2017 - Workshop Track Proceedings*.

Bengio, Y., Lodi, A., & Prouvost, A. (2018). Machine Learning for Combinatorial Optimization: a Methodological Tour d'Horizon. *arXiv:1811.06128*. Retrieved from http://arxiv.org/abs/1811.06128

Bent, R., & Hentenryck, P. Van. (2006). A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. *Computers and Operations Research*, *33*(4), 875–893. https://doi.org/10.1016/j.cor.2004.08.001

Berman, O., & Krass, D. (2002). The generalized maximal covering location problem. *Computers and Operations Research*, *29*(6), 563–581. https://doi.org/10.1016/S0305-0548(01)00079-X

Bland, J., & Clements, C. J. (1998). Protecting the world's children: the story of WHO's immunization programme. In *World health forum 1998; 19 (2)* (pp. 162–173).

Blanford, J. I., Kumar, S., Luo, W., & MacEachren, A. M. (2012). It's a long, long walk: accessibility to hospitals, maternity and integrated health centers in Niger. *International Journal of Health Geographics*, *11*(1), 24. https://doi.org/10.1186/1476-072X-11-24

Bock, F. (1958). *An algorithm for solving '"traveling-salesman"' and related network optimization problems*.

Bol, K. F., Aarntzen, E. H. J. G., Pots, J. M., Olde Nordkamp, M. A. M., van de Rakt, M. W. M. M., Scharenborg, N. M., … Schreibelt, G. (2016). Prophylactic vaccines are potent activators of monocyte-derived dendritic cells and drive effective anti-tumor responses in melanoma patients at the cost of toxicity. *Cancer Immunology, Immunotherapy*, *65*(3), 327–339. https://doi.org/10.1007/s00262-016-1796-7

Bonami, P., Lodi, A., & Zarpellon, G. (2018). Learning a classification of mixed-integer quadratic programming problems. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 10848 LNCS, pp. 595–604). Springer Verlag. https://doi.org/10.1007/978-3-319-93031-2_43

Braaten, S., Gjønnes, O., Hvattum, L. M., & Tirado, G. (2017). Heuristics for the robust vehicle routing problem with time windows. *Expert Systems with Applications*, *77*, 136–147. https://doi.org/10.1016/j.eswa.2017.01.038

Bräysy, O. (2003). A reactive variable neighborhood search for the vehicle-routing problem with time windows. *INFORMS Journal on Computing*. https://doi.org/10.1287/ijoc.15.4.347.24896

Bräysy, O., Dullaert, W., & Gendreau, M. (2004). Evolutionary algorithms for the Vehicle Routing Problem with Time Windows. *Journal of Heuristics*. https://doi.org/10.1007/s10732-005-5431-6

Bronstein, M. M., Bruna, J., Lecun, Y., Szlam, A., & Vandergheynst, P. (2017). Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, *34*(4), 18–42. https://doi.org/10.1109/MSP.2017.2693418

Campbell, J. F., & O'Kelly, M. E. (2012). Twenty-five years of hub location research. *Transportation Science*, *46*(2), 153–169. https://doi.org/10.1287/trsc.1120.0410

Chang, Y., Brewer, N. T., Rinas, A. C., Schmitt, K., & Smith, J. S. (2009). Evaluating the impact of human papillomavirus vaccines. *Vaccine*, *27*(32), 4355–4362. https://doi.org/10.1016/j.vaccine.2009.03.008

Chen, J. F. (2008). A heuristic for the capacitated single allocation hub location problem. In *Lecture Notes in Electrical Engineering* (Vol. 5 LNEE, pp. 185–196). https://doi.org/10.1007/978-0-387-74905-1_14

Chen, S.-I. (2012). *Modeling the WHO-EPI Vaccine Supply Chain in Low and Middle Income Countries*. University of Pittsburgh. Retrieved from http://d-scholarship.pitt.edu/12832/

Chen, S.-I., Norman, B. A., Rajgopal, J., Assi, T. M., Lee, B. Y., & Brown, S. T. (2014). A planning model for the WHO-EPI vaccine distribution network in developing countries. *IIE Transactions (Institute of Industrial Engineers)*, *46*(8), 853–865. https://doi.org/10.1080/0740817X.2013.813094

Christofides, N. (1976). *Worst-Case Analysis of a New Heuristic for the Travelling Salesman Problem.* Pittsburgh.

Christofides, Nicos, & Eilon, S. (1972). Algorithms for Large-scale Travelling Salesman Problems. *Journal of the Operational Research Society*, *23*(4), 511–518. https://doi.org/10.1057/jors.1972.79

Church, R., & ReVelle, C. (1974). The maximal covering location problem. *Papers of the Regional Science Association*, *32*(1), 101–118. https://doi.org/10.1007/BF01942293

Contreras, I., Díaz, J. A., & Fernández, E. (2009). Lagrangean relaxation for the capacitated hub location problem with single assignment. *OR Spectrum*, *31*(3), 483–505. https://doi.org/10.1007/s00291-008-0159-y

Cordeau, J. F., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*. https://doi.org/10.1057/palgrave.jors.2601163

Correia, I., Nickel, S., & Saldanha-da-Gama, F. (2010). Single-assignment hub location problems with multiple capacity levels. *Transportation Research Part B: Methodological*, *44*(8–9), 1047–1066. https://doi.org/10.1016/j.trb.2009.12.016

Costa, M. G, Captivo, M. E., & Clímaco, J. (2008). Capacitated single allocation hub location problem-A bi-criteria approach. *Computers and Operations Research*, *35*(11), 3671–3695. https://doi.org/10.1016/j.cor.2007.04.005

Croes, G. A. (1958). A Method for Solving Traveling-Salesman Problems. *Operations Research*, *6*(6), 791–812. https://doi.org/10.1287/opre.6.6.791

Dai, H., Khalil, E. B., Zhang, Y., Dilkina, B., & Song, L. (2017). Learning combinatorial optimization algorithms over graphs. *Advances in Neural Information Processing Systems*, *2017-Decem*(Nips), 6349–6359.

Daskin, M S, & Dean, L. K. (2004). Location of Health Care Facilities. In F. Sainfort, M. Brandeau, & W. Pierskalla (Eds.), *Operations Research and Health Care: A Handbook of Methods and Applications* (pp. 43–76). Springer US. https://doi.org/10.1007/b106574

Daskin, Mark S., Snyder, L. V., & Berger, R. T. (2005). Facility location in supply chain design. In A. Langevin & D. Riopel (Eds.), *Logistics Systems: Design and Optimization* (pp. 39–65). Springer US. https://doi.org/10.1007/0-387-24977-X_2

De Backer, B., Furnon, V., Shaw, P., Kilby, P., & Prosser, P. (2000). Solving vehicle routing problems using constraint programming and metaheuristics. *Journal of Heuristics*. https://doi.org/10.1023/A:1009621410177

De Boeck, K., Decouttere, C., & Vandaele, N. (2019, August 9). Vaccine distribution chains in low- and middle-income countries: A literature review. *Omega (United Kingdom)*. Elsevier Ltd. https://doi.org/10.1016/j.omega.2019.08.004

de Camargo, R., de Miranda, G., & Ferreira, R. P. M. (2011). A hybrid Outer-Approximation/Benders Decomposition algorithm for the single allocation hub location problem under congestion. *Operations Research Letters*, *39*(5), 329–337. https://doi.org/10.1016/j.orl.2011.06.015

de Camargo, R. S., & Miranda, G. (2012). Single allocation hub location problem under congestion: Network owner and user perspectives. *Expert Systems with Applications*, *39*(3), 3385–3391. https://doi.org/10.1016/j.eswa.2011.09.026

de Oliveira, M. F. S., Martinez, E. Z., & Rocha, J. S. Y. (2014). Factors associated with vaccination coverage in children <5 years in Angola. *Revista de Saude Publica*, *48*(6), 906–915. https://doi.org/10.1590/S0034-8910.2014048005284

Deng, L., & Yu, D. (2014). Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing*, *7*(3–4), 197–387. https://doi.org/10.1561/2000000039

Dorigo, M., & Gambardella, L. M. (1997). Ant colonies for the travelling salesman problem. *BioSystems*, *43*(2), 73–81. https://doi.org/10.1016/S0303-2647(97)01708-5

Earnshaw, S. R., Hicks, K., Richter, A., & Honeycutt, A. (2007). A linear programming model for allocating HIV prevention funds with state agencies: A pilot study. *Health Care Management Science, 10*(3), 239–252. https://doi.org/10.1007/s10729-007-9017-8

Emami, P., & Ranka, S. (2018). Learning Permutations with Sinkhorn Policy Gradient. *arXiv:1805.07010*. Retrieved from http://arxiv.org/abs/1805.07010

Ernst, A. T., & Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, *86*, 141–159. https://doi.org/10.1023/A:1018994432663

Farahani, R. Z., Hekmatfar, M., Arabani, A. B., & Nikbakhsh, E. (2013). Hub location problems: A review of models, classification, solution techniques, and applications. *Computers and Industrial Engineering*, *64*(4), 1096–1109. https://doi.org/10.1016/j.cie.2013.01.012

Ferguson, N. M., Keeling, M. J., Edmunds, W. J., Gani, R., Grenfell, B. T., Anderson, R. M., & Leach, S. (2003, October 16). Planning for smallpox outbreaks. *Nature*. Nature Publishing Group. https://doi.org/10.1038/nature02007

Figliozzi, M. A. (2010). An iterative route construction and improvement algorithm for the vehicle routing problem with soft time windows. *Transportation Research Part C: Emerging Technologies*, *18*(5), 668–679. https://doi.org/10.1016/j.trc.2009.08.005

Figliozzi, M. A. (2012). The time dependent vehicle routing problem with time windows: Benchmark problems, an efficient solution algorithm, and solution characteristics. *Transportation Research Part E: Logistics and Transportation Review*, *48*(3), 616–636. https://doi.org/10.1016/j.tre.2011.11.006

Fiore, A. E., Bridges, C. B., & Cox, N. J. (2009). Seasonal Influenza Vaccines (pp. 43–82). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-92165-3_3

Fooks, A. R., Banyard, A. C., & Ertl, H. C. J. (2019). New human rabies vaccines in the pipeline. *Vaccine*, *37*, A140–A145. https://doi.org/10.1016/j.vaccine.2018.08.039

Frazer, I. H. (2014). Development and Implementation of Papillomavirus Prophylactic Vaccines. *The Journal of Immunology*, *192*(9), 4007–4011. https://doi.org/10.4049/jimmunol.1490012

Fu, L. Y., Bonhomme, L. A., Cooper, S. C., Joseph, J. G., & Zimet, G. D. (2014, April 7). Educational interventions to increase HPV vaccination acceptance: A systematic review. *Vaccine*. Elsevier BV. https://doi.org/10.1016/j.vaccine.2014.01.091

Gavi. (2019). Annual Progress Report 2018. Retrieved from https://www.gavi.org/progress-report

Gavi. (2020). Our impact. Retrieved from https://www.gavi.org/programmes-and-impact/our-impact

Geoffrion, A. M., & Graves, G. W. (1974). Multicommodity Distribution System Design by Benders Decomposition. *Management Science*, *20*(5), 822–844.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Gutin, G., & Punnen, A. (2007). *The Traveling Salesman Problem and its Variations*. *... of Combinatorial Optimization: Problems and ...*. Boston, MA: Springer. https://doi.org/10.1007/b101971

He, H., Iii, H. D., & Eisner, J. (2014). *Learning to Search in Branch-and-Bound Algorithms. NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, (pp. 3293-3301)

Heaton, A., Krudwig, K., Lorenson, T., Burgess, C., Cunningham, A., & Steinglass, R. (2017, April 19). Doses per vaccine vial container: An understated and underestimated driver of performance that needs more evidence. *Vaccine*. Elsevier Ltd. https://doi.org/10.1016/j.vaccine.2016.11.066

Hill, A. N., & Longini, I. M. (2003). The critical vaccination fraction for heterogeneous epidemic models. *Mathematical Biosciences*, *181*(1), 85–106. https://doi.org/10.1016/S0025-5564(02)00129-3

Homberger, J., & Gehring, H. (1999). Two evolutionary metaheuristics for the vehicle routing problem with time windows. *INFOR Journal*. https://doi.org/10.1080/03155986.1999.11732386

Homberger, J., & Gehring, H. (2005). A two-phase hybrid metaheuristic for the vehicle routing problem with time windows. In *European Journal of Operational Research* (Vol. 162, pp. 220–238). https://doi.org/10.1016/j.ejor.2004.01.027

Hotez, P. J., Nuzhath, T., & Colwell, B. (2020, April 1). Combating vaccine hesitancy and other 21st century social determinants in the global fight against measles. *Current Opinion in Virology*. Elsevier B.V. https://doi.org/10.1016/j.coviro.2020.01.001

Jacobson, S. H., Sewell, E. C., Deuson, R., & Weniger, B. G. (1999). An integer programming model for vaccine procurement and delivery for childhood immunization: A pilot study. *Health Care Management Science*, *2*(1), 1–9. https://doi.org/10.1023/A:1019011106198

Jarrett, C., Wilson, R., O'Leary, M., Eckersberger, E., Larson, H. J., Eskola, J., … Schuster, M. (2015, August 14). Strategies for addressing vaccine hesitancy - A systematic review. *Vaccine*. Elsevier Ltd. https://doi.org/10.1016/j.vaccine.2015.04.040

Johnson, D. S., Gutin, G., McGeoch, L. A., Yeo, A., Zhang, W., & Zverovitch, A. (2007). Experimental Analysis of Heuristics for the ATSP. In G. Gutin & A. Punnen (Eds.), *The Traveling Salesman Problem and Its Variations* (pp. 445–487). Boston, MA: Springer, Boston, MA. https://doi.org/10.1007/0-306-48213-4_10

Johnson, D. S., & Mcgeoch, L. A. (1997). The Traveling Salesman Problem: A Case Study in Local Optimization. In E. H. L. Aarts & J. K. Lenstra (Eds.), *Local Search in Combinatorial Optimization* (pp. 215–310). New York: John Wiley & Sons.

Johnson, D. S., & McGeoch, L. A. (2007). Experimental Analysis of Heuristics for the STSP. In G. Gutin & A. Punnen (Eds.), *The Traveling Salesman Problem and Its Variations* (pp. 369–443). Boston, MA: Springer, Boston, MA. https://doi.org/10.1007/0-306-48213-4_9

Kaempfer, Y., & Wolf, L. (2018). Learning the Multiple Traveling Salesmen Problem with Permutation Invariant Pooling Networks. *arXiv:1803.09621*, (i), 1–22. Retrieved from http://arxiv.org/abs/1803.09621

Kaplan, E. H., Craft, D. L., & Wein, L. M. (2002). Emergency response to a smallpox attack: The case for mass vaccination. *Proceedings of the National Academy of Sciences of the United States of America*, *99*(16), 10935–10940. https://doi.org/10.1073/pnas.162282799

Kaplan, E. H., & Merson, M. H. (2002). Allocating HIV-Prevention Resources: Balancing Efficiency and Equity. *American Journal of Public Health*, *92*(12), 1905–1907. https://doi.org/10.2105/AJPH.92.12.1905

Khalil, E. B., Bodic, P. Le, Song, L., Nemhauser, G., & Dilkina, B. (2016). Learning to branch in mixed integer programming. *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, 724–731.

Khalil, E. B., Dilkina, B., Nemhauser, G. L., Ahmed, S., & Shao, Y. (2017). Learning to run heuristics in tree search. *IJCAI International Joint Conference on Artificial Intelligence*, *2*, 659–666. https://doi.org/10.24963/ijcai.2017/92

Kilby, P., Prosser, P., & Shaw, P. (1999). Guided Local Search for the Vehicle Routing Problem with Time Windows. In S. Voß, S. Martello, I. H. Osman, & C. Roucairol (Eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization* (pp. 473–486). Boston, MA: Springer US. https://doi.org/10.1007/978-1-4615-5775-3_32

Klose, A., & Drexl, A. (2005). Facility location models for distribution system design. *European Journal of Operational Research*, *162*(1), 4–29. https://doi.org/10.1016/j.ejor.2003.10.031

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2015). A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. *Computers and Operations Research*, *64*, 11–27. https://doi.org/10.1016/j.cor.2015.05.004

Kool, W., & Welling, M. (2019). Attention, Learn to Solve Routing Problems! *arXiv:1803.08475v3*, 1–25.

Kratica, J., Milanović, M., Stanimirović, Z., & Tošić, D. (2011). An evolutionary-based approach for solving a capacitated hub location problem. In *Applied Soft Computing Journal* (Vol. 11, pp. 1858–1866). https://doi.org/10.1016/j.asoc.2010.05.035

Kruber, M., Lübbecke, M. E., & Parmentier, A. (2017). Learning when to use a decomposition. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10335 LNCS*, 202–210. https://doi.org/10.1007/978-3-319-59776-8_16

Kumar, S. N., & Panneerselvam, R. (2012). A Survey on the Vehicle Routing Problem and Its Variants. *Intelligent Information Management*, *04*(03), 66–74. https://doi.org/10.4236/iim.2012.43010

Larsen, E., Lachapelle, S., Bengio, Y., Frejinger, E., Lacoste-Julien, S., & Lodi, A. (2019). Predicting Tactical Solutions to Operational Planning Problems under Imperfect Information. *arXiv:1901.07935*, 1–19. Retrieved from http://arxiv.org/abs/1901.07935

Lecun, Y., Bengio, Y., & Hinton, G. (2015, May 27). Deep learning. *Nature*. Nature Publishing Group. https://doi.org/10.1038/nature14539

Lee, B. Y., Assi, T.-M., Rookkapan, K., Wateska, A. R., Rajgopal, J., Sornsrivichai, V., … Burke, D. S. (2011). Maintaining Vaccine Delivery Following the Introduction of the Rotavirus and Pneumococcal Vaccines in Thailand. *PLoS ONE*, *6*(9), e24673. https://doi.org/10.1371/journal.pone.0024673

Lee, B. Y., Assi, T. M., Rajgopal, J., Norman, B. A., Chen, S. I., Brown, S. T., … Burke, D. S. (2012). Impact of introducing the pneumococcal and rotavirus vaccines into the routine immunization program in Niger. *American Journal of Public Health*, *102*(2), 269–276. https://doi.org/10.2105/AJPH.2011.300218

Lee, B. Y., Assi, T. M., Rookkapan, K., Connor, D. L., Rajgopal, J., Sornsrivichai, V., … Burke, D. S. (2011). Replacing the measles ten-dose vaccine presentation with the single-dose presentation in Thailand. *Vaccine*, *29*(21), 3811–3817. https://doi.org/10.1016/j.vaccine.2011.03.013

Lee, B. Y., Norman, B. A., Assi, T. M., Chen, S. I., Bailey, R. R., Rajgopal, J., … Burke, D. S. (2010). Single versus multi-dose vaccine vials: An economic computational model. *Vaccine*, *28*(32), 5292–5300. https://doi.org/10.1016/j.vaccine.2010.05.048

Lenstra, J. K., & Kan, A. H. G. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, *11*(2), 221–227. https://doi.org/10.1002/net.3230110211

Liesegang, T. J. (2009). Varicella zoster virus vaccines: Effective, but concerns linger. *Canadian Journal of Ophthalmology*, *44*(4), 379–384. https://doi.org/10.3129/i09-126

Lim, J. (2016). *Improving the design and operation of WHO-EPI vaccine distribution networks*. University of Pittsburgh, Pittsburgh, PA. Retrieved from http://d-scholarship.pitt.edu/28696/

Lim, J., Claypool, E., Norman, B. A., & Rajgopal, J. (2016). Coverage models to determine outreach vaccination center locations in low and middle income countries. *Operations Research for Health Care*, *9*, 40–48. https://doi.org/10.1016/j.orhc.2016.02.003

Lim, J., Norman, B. A., & Rajgopal, J. (2017). Process Redesign and Simplified Policies for More Effective Vaccine Inventory Management. *EMJ - Engineering Management Journal*, *29*(1), 17–25. https://doi.org/10.1080/10429247.2016.1277446

Lim, J., Norman, B. A., & Rajgopal, J. (2019). Redesign of vaccine distribution networks. *International Transactions in Operational Research*. https://doi.org/10.1111/itor.12758

Lin, S., & Kernighan, B. W. (1973). An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research*, *21*(2), 498–516. https://doi.org/10.1287/opre.21.2.498

Lin, Shen. (1965). Computer Solutions of the Traveling Salesman Problem. *Bell System Technical Journal*, *44*(10), 2245–2269. https://doi.org/10.1002/j.1538-7305.1965.tb04146.x

Lodi, A., & Zarpellon, G. (2017). On learning and branching: a survey. *Top*, *25*(2), 207–236. https://doi.org/10.1007/s11750-017-0451-6

Lombardi, M., & Milano, M. (2018). Boosting Combinatorial Problem Modeling with Machine Learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence* (pp. 5472–5478). https://doi.org/10.24963/ijcai.2018/772

Longini, I. M., Ackerman, E., & Elveback, L. R. (1978). An optimization model for influenza A epidemics. *Mathematical Biosciences*, *38*(1–2), 141–157. https://doi.org/10.1016/0025-5564(78)90023-8

MacDonald, N. E., Eskola, J., Liang, X., Chaudhuri, M., Dube, E., Gellin, B., … Schuster, M. (2015). Vaccine hesitancy: Definition, scope and determinants. *Vaccine*, *33*(34), 4161–4164. https://doi.org/10.1016/j.vaccine.2015.04.036

Malande, O. O., Munube, D., Afaayo, R. N., Annet, K., Bodo, B., Bakainaga, A., … Musyoki, A. M. (2019). Barriers to effective uptake and provision of immunization in a rural district in Uganda. *PLOS ONE*, *14*(2), e0212270. https://doi.org/10.1371/journal.pone.0212270

Martin, O. C., Otto, S. W., & Felten, E. W. (1991). Large-step Markov chains for the traveling salesman problem. *Complex Systems*, *5*(3), 299–326. Retrieved from https://www.complex-systems.com/abstracts/v05_i03_a03/

Martin, O., Otto, S. W., & Felten, E. W. (1992). Large-step markov chains for the TSP incorporating local search heuristics. *Operations Research Letters*, *11*(4), 219–224. https://doi.org/10.1016/0167-6377(92)90028-2

Melo, M. T., Nickel, S., & Saldanha-da-Gama, F. (2009). Facility location and supply chain management - A review. *European Journal of Operational Research*, *196*(2), 401–412. https://doi.org/10.1016/j.ejor.2008.05.007

Miller, C. E., Zemlin, R. A., & Tucker, A. W. (1960). Integer Programming Formulation of Traveling Salesman Problems. *Journal of the ACM (JACM)*, *7*(4), 326–329. https://doi.org/10.1145/321043.321046

Mirchandani, P. B. (1990). The p-median problem and generalizations. In P. Mirchandani & R. Francis (Eds.), *Discrete Location Theory* (pp. 55–117). New York: John Wiley and Sons Inc.

Mofrad, M. H. (2016). *Optimizing Vaccine Clinic Operations in Low and Middle Income Countries*. University of Pittsburgh, Pittsburgh, PA. Retrieved from http://d-scholarship.pitt.edu/27262/

Mofrad, M. H., Garcia, G. G. P., Maillart, L. M., Norman, B. A., & Rajgopal, J. (2016). Customizing immunization clinic operations to minimize open vial waste. *Socio-Economic Planning Sciences*, *54*, 1–17. https://doi.org/10.1016/j.seps.2015.10.007

Mofrad, M. H., Maillart, L. M., Norman, B. A., & Rajgopal, J. (2014). Dynamically optimizing the administration of vaccines from multi-dose vials. *IIE Transactions (Institute of Industrial Engineers)*, *46*(7), 623–635. https://doi.org/10.1080/0740817X.2013.849834

Norman, B. A., Rajgopal, J., Lim, J., Gorham, K., Haidari, L., Brown, S. T., & Lee, B. Y. (2015). Modular vaccine packaging increases packing efficiency. *Vaccine*, *33*(27), 3135–3141. https://doi.org/10.1016/j.vaccine.2015.04.091

Nowak, A., Villar, S., Bandeira, A. S., & Bruna, J. (2018). REVISED NOTE on LEARNING QUADRATIC ASSIGNMENT with GRAPH NEURAL NETWORKS. *2018 IEEE Data Science Workshop, DSW 2018 - Proceedings*, 229–233. https://doi.org/10.1109/DSW.2018.8439919

O'Brien, K. L., & Nolan, T. (2019, October 3). The WHO position on rabies immunization – 2018 updates. *Vaccine*. Elsevier Ltd. https://doi.org/10.1016/j.vaccine.2018.10.014

Ouma, P. O., Maina, J., Thuranira, P. N., Macharia, P. M., Alegana, V. A., English, M., … Snow, R. W. (2018). Access to emergency hospital care provided by the public sector in sub-Saharan Africa in 2015: a geocoded inventory and spatial analysis. *The Lancet Global Health*, *6*(3), e342–e350. https://doi.org/10.1016/S2214-109X(17)30488-6

Pesch, E., & Glover, F. (1997). TSP ejection chains. *Discrete Applied Mathematics*, *76*(1–3), 165–181. https://doi.org/10.1016/S0166-218X(96)00123-0

Potvin, J. Y., & Bengio, S. (1996). The vehicle routing problem with time windows part II: Genetic search. *INFORMS Journal on Computing*. https://doi.org/10.1287/ijoc.8.2.165

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers and Operations Research*. https://doi.org/10.1016/S0305-0548(03)00158-8

Purkayastha, R., Chakraborty, T., Saha, A., & Mukhopadhyay, D. (2020). Study and Analysis of Various Heuristic Algorithms for Solving Travelling Salesman Problem—A Survey. In J. Mandal & S. Mukhopadhyay (Eds.), *Proceedings of the Global AI Congress 2019. Advances in Intelligent Systems and Computing, vol 1112* (pp. 61–70). Springer, Singapore. https://doi.org/10.1007/978-981-15-2188-1_5

Qureshi, A. G., Taniguchi, E., & Yamada, T. (2009). An exact solution approach for vehicle routing and scheduling problems with soft time windows. *Transportation Research Part E: Logistics and Transportation Review*, *45*(6), 960–977. https://doi.org/10.1016/j.tre.2009.04.007

Rajgopal, J., Connor, D. L., Assi, T. M., Norman, B. A., Chen, S. I., Bailey, R. R., … Lee, B. Y. (2011). The optimal number of routine vaccines to order at health clinics in low or middle income countries. *Vaccine*, *29*(33), 5512–5518. https://doi.org/10.1016/j.vaccine.2011.05.044

Rego, C. (1998). Relaxed tours and path ejections for the traveling salesman problem. *European Journal of Operational Research*, *106*(2–3), 522–538. https://doi.org/10.1016/S0377-2217(97)00288-9

Rego, C., Gamboa, D., Glover, F., & Osterman, C. (2011). Traveling salesman problem heuristics: Leading methods, implementations and latest advances. *European Journal of Operational Research*, *211*(3), 427–441. https://doi.org/10.1016/j.ejor.2010.09.010

Reil, S., Bortfeldt, A., & Mönch, L. (2018). Heuristics for vehicle routing problems with backhauls, time windows, and 3D loading constraints. *European Journal of Operational Research*, *266*(3), 877–894. https://doi.org/10.1016/j.ejor.2017.10.029

Şahin, G., & Süral, H. (2007). A review of hierarchical facility location models. *Computers and Operations Research*, *34*(8), 2310–2331. https://doi.org/10.1016/j.cor.2005.09.005

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, *20*(1), 61–80. https://doi.org/10.1109/TNN.2008.2005605

Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/3-540-49481-2_30

Shen, A. K., Fields, R., & McQuestion, M. (2014). The future of routine immunization in the developing world: Challenges and opportunities. *Global Health Science and Practice*. Johns Hopkins University Press. https://doi.org/10.9745/GHSP-D-14-00137

Shen, Z. M., Coullard, C., & Daskin, M. S. (2003). A Joint Location-Inventory Model. *Transportation Science*, *37*(1), 1–121.

Snyder, L. V., & Daskin, M. S. (2005). Reliability models for facility location: The expected failure cost case. *Transportation Science*, *39*(3), 400–416. https://doi.org/10.1287/trsc.1040.0107

Taghipourian, F., Mahdavi, I., Mahdavi-Amiri, N., & Makui, A. (2012). A fuzzy programming approach for dynamic virtual hub location problem. *Applied Mathematical Modelling*, *36*(7), 3257–3270. https://doi.org/10.1016/j.apm.2011.10.016

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J. Y. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*. https://doi.org/10.1287/trsc.31.2.170

Tang, Y., Agrawal, S., & Faenza, Y. (2019). Reinforcement Learning for Integer Programming: Learning to Cut. *arXiv:1906.04859*. Retrieved from http://arxiv.org/abs/1906.04859

Thangiah, S. R., Nygard, K. E., & Juell, P. L. (1990). GIDEON: A genetic algorithm system for vehicle routing with time windows. In *Proceedings of the Conference on Artificial Intelligence Applications*. https://doi.org/10.1109/caia.1991.120888

Toth, P., & Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. *MOS-SIAM Series on Optimization*. https://doi.org/doi:10.1137/1.9781611973594

United Nations. (2017). World Population Prospects. Retrieved November 26, 2019, from https://population.un.org/wpp/DataQuery/

Veličković, P., Casanova, A., Liò, P., Cucurull, G., Romero, A., & Bengio, Y. (2018). Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N., & Rei, W. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research*. https://doi.org/10.1287/opre.1120.1048

Vinyals, O., Fortunato, M., & Jaitly, N. (2015). Pointer networks. In *Advances in Neural Information Processing Systems* (Vol. 2015-January, pp. 2692–2700). Neural information processing systems foundation.

Wolff, G. G. (2020). Influenza vaccination and respiratory virus interference among Department of Defense personnel during the 2017–2018 influenza season. *Vaccine*, *38*(2), 350–354. https://doi.org/10.1016/j.vaccine.2019.10.005

World Health Organization. (1977). Expanded Programme on Immunization: study of the feasibility coverage and cost of maintenance immunization for children by district mobile teams in Kenya. *The Weekly Epidemiological Record*, *52*, 197–204. Retrieved from https://www.epistemonikos.org/documents/bb71c8cf48fc95ccee982a369c5a5b842c51f847/

World Health Organization. (2009). Prequalified Devices and Equipment. Retrieved from http://apps.who.int/immunization_standards/vaccine_quality/pqs_catalogue/categorylist.aspx?cat_type=device

World Health Organization. (2013). The Expanded Programme on Immunization. Retrieved from http://www.who.int/immunization/programmes_systems/supply_chain/benefits_of_immunization/en/

World Health Organization. (2017). Global Vaccine Action Plan 2011-2020. Retrieved from https://www.who.int/immunization/global_vaccine_action_plan/GVAP_doc_2011_2020/en/

World Health Organization. (2018). Health on wheels: mobile clinic brings vital care to displaced Yazidis in Iraq. Retrieved June 1, 2020, from https://medium.com/@who/health-on-wheels-mobile-clinic-brings-vital-care-to-displaced-yazidis-in-iraq-a02f750df314

World Health Organization. (2019a). Delivering health in some of the worlds' worst crises through mobile clinics and medical teams. Retrieved June 1, 2020, from https://www.who.int/news-room/feature-stories/detail/delivering-health-in-some-of-the-worlds-worst-crises-through-mobile-clinics-and-medical-teams

World Health Organization. (2019b). Immunization. Retrieved November 25, 2019, from http://www.who.int/topics/immunization/en/

World Health Organization. (2019c). Immunization coverage. Retrieved May 10, 2020, from https://www.who.int/en/news-room/fact-sheets/detail/immunization-coverage

World Health Organization. (2020a). *Highlights from the Meeting of the Strategic Advisory Group of Experts (SAGE) on Immunization*. Retrieved from https://www.who.int/immunization/sage/meetings/2020/april/SAGE_Highlights_MarchApril_2020.pdf?ua=1

World Health Organization. (2020b). Immunization policy and strategies. Retrieved May 10, 2020, from https://www.who.int/immunization/programmes_systems/policies_strategies/en/

Wu, T. H., Low, C., & Bai, J. W. (2002). Heuristic solutions to multi-depot location-routing problems. *Computers and Operations Research*, *29*(10), 1393–1415. https://doi.org/10.1016/S0305-0548(01)00038-7

Yadav, P., Lydon, P., Oswald, J., Dicko, M., & Zaffran, M. (2014). Integration of vaccine supply chains with other health commodity supply chains: A framework for decision making. *Vaccine*, *32*(50), 6725–6732. https://doi.org/10.1016/j.vaccine.2014.10.001

Yang, Y., Bidkhori, H., & Rajgopal, J. (2020). Optimizing vaccine distribution networks in low and middle-income countries. *Omega*. https://doi.org/10.1016/J.OMEGA.2020.102197

Yang, Y., & Rajgopal, J. (2019). Outreach Strategies for Vaccine Distribution: A Two-Period Robust Approach. *arXiv:1908.10465*. Retrieved from http://arxiv.org/abs/1908.10465

Yang, Y., & Rajgopal, J. (2020a). An Iterative Cyclic Algorithm for Designing Vaccine Distribution Networks in Low and Middle-Income Countries. In Z. Anisic, B. Lalic, & D. Gracanin (Eds.), *Proceedings on 25th International Joint Conference on Industrial Engineering and Operations Management – IJCIEOM. IJCIEOM 2019. Lecture Notes on Multidisciplinary Industrial Engineering.* (pp. 512–521). Springer, Cham. https://doi.org/10.1007/978-3-030-43616-2_54

Yang, Y., & Rajgopal, J. (2020b). Learning Combined Set Covering and Traveling Salesman Problem. *arXiv:2007.03203*. Retrieved from http://arxiv.org/abs/2007.03203

Yu, B., Yang, Z. Z., & Yao, B. Z. (2011). A hybrid algorithm for vehicle routing problem with time windows. *Expert Systems with Applications*, *38*(1), 435–441. https://doi.org/10.1016/j.eswa.2010.06.082

Zaffran, M. (1996). Vaccine transport and storage: environmental challenges. *Developments in Biological Standardization*, *87*, 9—17. Retrieved from http://europepmc.org/abstract/MED/8853997