

Electronic Theses and Dissertations, 2004-2019

2016

Modeling and Solving Large-scale Stochastic Mixed-Integer Problems in Transportation and Power Systems

Zhouchun Huang
University of Central Florida

 Part of the [Industrial Engineering Commons](#)
Find similar works at: <https://stars.library.ucf.edu/etd>
University of Central Florida Libraries <http://library.ucf.edu>

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

Huang, Zhouchun, "Modeling and Solving Large-scale Stochastic Mixed-Integer Problems in Transportation and Power Systems" (2016). *Electronic Theses and Dissertations, 2004-2019*. 5187. <https://stars.library.ucf.edu/etd/5187>

MODELING AND SOLVING LARGE-SCALE STOCHASTIC MIXED-INTEGER
PROBLEMS IN TRANSPORTATION AND POWER SYSTEMS

by

ZHOUCHUN HUANG
M.S. West Virginia University, 2013

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Industrial Engineering and Management Systems
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2016

Major Professor: Qipeng Zheng

© 2016 Zhouchun Huang

ABSTRACT

In this dissertation, various optimization problems from the area of transportation and power systems will be respectively investigated and the uncertainty will be considered in each problem. Specifically, a long-term problem of electricity infrastructure investment is studied to address the planning for capacity expansion in electrical power systems with the integration of short-term operations. The future investment costs and real-time customer demands cannot be perfectly forecasted and thus are considered to be random. Another maintenance scheduling problem is studied for power systems, particularly for natural gas fueled power plants, taking into account gas contracting and the opportunity of purchasing and selling gas in the spot market as well as the maintenance scheduling considering the uncertainty of electricity and gas prices in the spot market. In addition, different vehicle routing problems are researched seeking the route for each vehicle so that the total traveling cost is minimized subject to the constraints and uncertain parameters in corresponding transportation systems.

The investigation of each problem in this dissertation mainly consists of two parts, i.e., the formulation of its mathematical model and the development of solution algorithm for solving the model. The stochastic programming is applied as the framework to model each problem and address the uncertainty, while the approach of dealing with the randomness varies in terms of the relationships between the uncertain elements and objective functions or constraints. All the problems will be modeled as stochastic mixed-integer programs, and the huge numbers of involved decision variables and constraints make each problem large-scale and very difficult to manage. In this dissertation, efficient algorithms are developed for these problems in the context of advanced methodologies of optimization and operations research, such as branch and cut, benders decomposition, column generation and Lagrangian method. Computational experiments are implemented for each problem and the results will be present and discussed. The research carried out in this

dissertation would be beneficial to both researchers and practitioners seeking to model and solve similar optimization problems in transportation and power systems when uncertainty is involved.

ACKNOWLEDGMENTS

First and foremost, I would like to show my deepest gratitude to my supervisor, Dr. Qipeng Zheng, for his professional guidance, continuous encouragement and impressive kindness. It has been an absolute pleasure time to work with such a nice and resourceful advisor. His enthusiasm, religious attitude and keen observation toward scientific research deserve my great respect. His teaching and instruction enlightens me not only through my doctoral program but also in the future study. I am also grateful to all my lovely partners in our OR group who are always kind and helpful to me. I am proud of being part of the group with these hardworking graduate students and we spent a lot of pleasant time cooperating and sharing experience with each other.

I wish to extend my thanks to Dr. Eduardo Pasiliao and Dr. Vladimir Boginski for offering me research visitor position at Air Force Research Laboratory REEF where I spent two fantastic summers with many researchers from other universities and organizations. I would like to thank Dr. Petros Xanthopoulos, Dr. Jennifer Pazour and Dr. Ni-bin Chang for their help and effort for being the committee members, and providing me many valuable comments, advices and suggestions for my research and dissertation. From them I have learnt a lot. I also wish to acknowledge Dr. Andrew Liu and his research group who patiently trained me the parallel computation techniques and the use of the Purdue Rice cluster for solving one of the most challenging projects in the dissertation.

Last but not least, I would like to thank my family, specially my wife Xialing Chen, for their understanding, support and love in my life. To them I dedicate the dissertation.

TABLE OF CONTENTS

LIST OF FIGURES.....	ix
LIST OF TABLES.....	xi
CHAPTER 1: INTRODUCTION.....	1
CHAPTER 2: REVIEW OF ALGORITHMS FOR SOLVING STOCHASTIC MIXED-INTEGER PROGRAMS.....	7
CHAPTER 3: LARGE-SCALE STOCHASTIC MIXED-INTEGER PROBLEMS IN POWER SYSTEMS.....	12
3.1 Electricity Infrastructure Investment	12
3.1.1 Introduction	13
3.1.2 Literature Review	14
3.1.3 Model Description	17
3.1.4 Algorithm.....	28
3.1.5 Algorithm Implementation and Solution Assessment.....	35
3.2 Maintenance Scheduling of Natural Gas Powered Plants	40
3.2.1 Introduction	41

3.2.2	Problem definition	44
3.2.3	Solution approaches.....	50
3.2.4	Computational experiments and results	58

CHAPTER 4: LARGE-SCALE STOCHASTIC MIXED-INTEGER PROBLEMS IN TRANSPORTATION.....64

4.1	Stochastic Traveling Salesman Problem with Discretely Distributed Random Arc Costs	64
4.1.1	Introduction	64
4.1.2	Problem Formulation	67
4.1.3	Exact Solution Algorithms	71
4.1.4	Numerical Experiments and Results	80
4.2	Stochastic Traveling Salesman Problem with Continuously Distributed Random Arc Costs	84
4.2.1	Introduction	85
4.2.2	Literature Review	87
4.2.3	Problem Description	89
4.2.4	Algorithms for solving the STSP.....	92
4.2.5	Numerical Experiments and Results	103

4.2.6	Conclusions	109
4.3	Reliable Routing Problems under Uncertain Topology	109
4.3.1	Introduction	110
4.3.2	Problem Formulation	113
4.3.3	Exact methods for solving CCTSP and CCSPP	118
4.3.4	Numerical Experiment and Results	133
4.4	Optimal Scheduling of Rapid Transit Routes.....	143
4.4.1	Introduction	143
4.4.2	Problem Formulation	144
4.4.3	Numerical Experiments and Computational Results	147
CHAPTER 5: CONCLUSIONS.....		153

LIST OF FIGURES

Figure 1.1: A 7 bus power system	2
Figure 3.1: Structure of the Original Problem.....	28
Figure 3.2: Parallelization architecture for the electricity infrastructure investment problem	36
Figure 3.3: ToP contract modeled by two virtual reservoirs	44
Figure 3.4: Expected value solution vs. stochastic solution. Left: the expected values of electricity and gas prices, and their particular values in two arbitrary scenarios (scenarios 1 and 2); Right: the expected value solution as well as the stochastic solution respectively for scenarios 1 and 2.....	62
Figure 4.1: Using CVaR and its LB to evaluate the feasibility of current [RMP] solution..	83
Figure 4.2: Bound contraction process in the instances of 15×120 (left) and 15×90 (right) with $\alpha = 0.99$	83
Figure 4.3: Computational results of reliable routing problem in the 10×70 network.....	84
Figure 4.4: Developed cutting plane.	101
Figure 4.5: The change in objective function values of updated [RMP]. Input graph: 50 nodes with high density.	107

Figure 4.6: Probability distribution of total optimal cost as VaR or CVaR method is used (left); the process of VaR or CVaR controlling cost variances for the purpose of risk management in the CP algorithm (right). Note: The network with 50 nodes and high density is applied and the confidence level is 99%.....	108
Figure 4.7: The change in the optimal expected costs as the confidence level increases. Note: The network with 200 nodes and medium density is applied.	108
Figure 4.8: The special solution space tree for CCPLB Algorithm for a network with 4 arcs	123
Figure 4.9: A network under uncertain topology with arc failures. Starting node: a; Destination node: d.	131
Figure 4.10: The efficiency of each cutting plane algorithm in the instance of 30-d1 with $\alpha = 90\%$	138
Figure 4.11: Optimal cost vs. confidence level in the instance of 30-d3	139
Figure 4.12: The efficiency of each cutting plane algorithm in the instance of 50-d3 with $\alpha = 90\%$	142
Figure 4.13: Optimal cost vs. confidence level in the instance of 50-d3(The problem is infeasible while $\alpha \geq 98\%$.).....	142
Figure 4.14: The PRT system at WVU	148

LIST OF TABLES

Table 3.1:	Long-Term Related Parameters and Indices	19
Table 3.2:	Decision Variables in the Long-Term (Upper-Level) Model	19
Table 3.3:	Parameters, Sets and Indices for short-term UC model	24
Table 3.4:	Short-term Decision Variables	25
Table 3.5:	Test Instances	37
Table 3.6:	Direct vs Decomposition for the IEEE 7-bus test case.	38
Table 3.7:	Computational results for IEEE 7-, 57, and 118- bus systems.....	39
Table 3.8:	Comparison of stabilization techniques.....	40
Table 3.9:	The comparison of two formulations for [GMP]	60
Table 3.10:	Performance of various solution approaches.....	61
Table 3.11:	Deterministic vs. Stochastic.	63
Table 4.1:	Computational results for [TSP-R] by using Benders Decomposition algorithm.....	82
Table 4.2:	Input graphs.	103
Table 4.3:	Solution times, in CPU seconds, for solving the STSP with different algorithms.....	105
Table 4.4:	Summary of Parameters, Sets, and Variables.	115

Table 4.5:	The outgoing degree and probability ranges assumptions for CCTSP instances	134
Table 4.6:	Computational results for CCTSP	136
Table 4.7:	The outgoing degree and probability range assumptions for CCSPP instances.	140
Table 4.8:	Computational results for CCSPP	141
Table 4.9:	Summary of Parameters, Sets, and Variables.	145
Table 4.10:	PRT system characteristics.	148
Table 4.11:	Vehicle info in the PRT system.	149
Table 4.12:	Randomly generated customer demands for one hour.....	149
Table 4.13:	Number of customers not immediately served.....	150
Table 4.14:	Number of customers not served within the time window.	151

CHAPTER 1: INTRODUCTION

Extensive efforts have been made by scientists and researchers for optimizing the use of transportation and power systems. An electric power system is a network of electrical components, including generators, transmission lines, storage facilities and end users, to generate, transmit, supply and use electric power. Consider the 7-bus system shown in Figure 1.1, where there are 7 nodes and 10 arcs. Each node represents a region that supplies or uses the electricity, and power generators are located in those labeled with G . Various sources of power could be applied for the power generation, such as fossil fuel (including coal, gas and oil), nuclear energy, falling water and wind. The amount of electricity generated by each unit is restricted by its capacity. Each arc in the system is a transmission line, a cable that can carry electric energy and transmit it from one region to another. The vehicle routing problem (VRP), which seeks to minimize the transport cost, is among the most important transportation problems. The VRP was firstly proposed by Dantzig and Ramser in 1959 [34] in order to transport a number of customers with a fleet of vehicles in a minimum-cost way. All vehicles start and end at the same depot and each node except for the depot is visited exactly once by exactly one vehicle. Some side constraints may be additionally included, such as capacity restrictions, total time restrictions and time window requirement for visiting each node [77]. As a special case of the VRP when there is only one vehicle, the Traveling Salesman Problem (TSP) has also attracted a lot of research interest in operations research because of its close connection to other optimization problems and wide practical applications. Given a list of cities, the objective of TSP is to find the minimum-cost Hamiltonian cycle of visiting each city exactly once. Similar to the VRP and/or TSP, the shortest path problem (SPP) is also seeking the least-cost path within a network with deterministic arc costs and topology, while there is no need to visit each node and the origin node differs from the destination node. In contrast to the NP-hardness of VRP and TSP, the SPP can be solved in polynomial time with the well-known Dijkstra's algorithm [43].

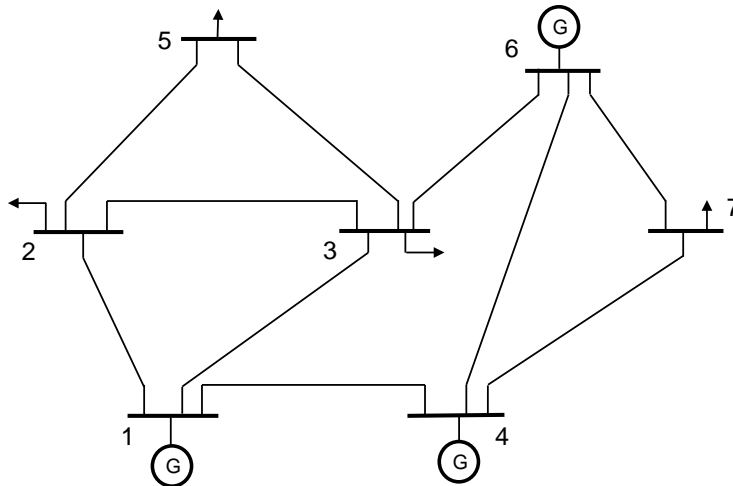


Figure 1.1: A 7 bus power system

While for many optimization problems that have been studied in these two areas are modeled to be deterministic, in reality most contain elements of uncertainty or randomness. For example, the cost of constructing a new generator would be unknown when the capacity expansion plans are made; A road could be disconnected due to inclement weather conditions or natural disasters; and the number of customers to be picked up at some location would be uncertain before a vehicle driver arrives there. Without considering the uncertainty in these cases, decision makers could obtain solutions that have much lower expected profits or are exposed of extremely high risks when the solutions are implemented in real applications. The stochastic programming has been widely used as a framework to model the problems in which the uncertainty is involved. Specifically, the uncertain parameters are assumed to be unknown before they are realized at some time point. Their probability distributions, however, are known or can be estimated in advance. By taking advantage of this statistical information, decision makers can develop a stochastic programming model aiming to find a solution which maximizes the expected profit or minimizes expected cost and also satisfies all (or most of all) the possible scenarios. The method of modeling the stochastic programming for a particular problem, however, could vary due to different circumstances and

requirements. For the two-stage unit commitment problem, for example, the first-stage decisions of switching generators on/off are firstly made, followed by the second-stage decisions of power dispatch which are subject to the unknown real-time customer demands. The goal is to minimize the current cost in the first stage plus the expected values of the future recourse actions, with all the scenarios of electric demands being taken into account. In other instances, however, managing and controlling the risk associated with the uncertainty would be desired. For example, consider the routing problem in the case that a city is struck by an earthquake or other disasters. A quick emergency response is then required to send the relief and rescue workers and humanitarian supplies to the damage zones via the most efficient route. The consequence of not considering the possibility that the roads may be broken due to uncertain factors such as aftershocks, heavy traffic and weather conditions, however, could be crucial as significant resources would be trapped and not be delivered in time. Therefore a more reliable route needs to be planned in advance to prevent the route failure with a desired level of confidence. The uncertainty of the network topology can then be considered and a chance constraint can be modeled for the purpose of risk management. In this dissertation, we develop different stochastic programming models and will show the variety of the approach in terms of managing the uncertainty. Both continuous and integer decision variables will be included in each model, and each problem would have a large scale because of the large numbers of variables and/or constraints. These make the problem extremely hard to solve, and some algorithms and strategies need to be developed to address the computational difficulty.

Each of the investigated optimization problems in this dissertation falls into either of the two categories, i.e., power systems and transportation. One of the most important problems in the electric power systems is the capacity expansion problem, which seeks the minimum-cost process of providing new facilities of similar type or expanding the capacities of existing ones over time to meet a rising demand for their services [37]. The planning of capacity expansion has been widely studied in operations research and more examples can be found in other areas, such as water resource

facilities, communication networks and heavy process industries. It has been suggested that uncertainty must be treated to achieve effective expansion planning [93], and uncertain parameters may include investment costs, operational costs and customer demands. Another important problem is the unit commitment problem, which refers to finding an optimal schedule for each of the facilities over a given period of time. The objective of the problem is to minimize the production cost of committing the units to supply the load. The unit commitment decision indicates which units are to be in use at each time point over a scheduling horizon [98] to meet real-time electricity demands. The problem might be modeled with deterministic approaches and a mix integer linear problem can be formulated to include the binary variables indicating the on/off status and start-up/shut-down action of each generating units, as well as the continuous variables representing the levels of generation and transmission. However in reality, the customer demands are not known in advance with certainty and thus a stochastic model would be more accurate to determine the unit commitment. The results in [135] indicates that significant savings can be achieved by including stochasticity. The first optimization problem we proposed for the power systems is an electricity infrastructure investment problem, which refers to the capacity expansion for the facilities in the electric power systems. The problem is modeled stochastic to address the uncertainty elements including investment costs, operational costs and demands. Unit commitment will be incorporated in the model so that the customer demands will be satisfied by well operating the expanded infrastructures.

The second optimization problem studied for the power systems is a long-term (1-3 years) operation, that is, the preventive maintenance scheduling for a power plant. The equipments in a power plant are generally under great pressure to minimize operating costs and to meet the electricity demands in real time. We aim to find the optimal schedule of periodically inspecting the equipments to minimize possible breakdowns and malfunctions during normal running and attain the reliability of the whole power system. The maintenance schedule affects many short and long-term operations-planning functions, including unit commitment and fuel scheduling [99]. On the oth-

er hand, these operations would also influence the maintenance schedule. For example, a natural gas power plant would prefer to schedule the inspection to the time periods when the fuel price is high and/or the electricity price is relatively low. Therefore we propose a model which integrates the natural gas purchase into the maintenance scheduling problem for power plants. We will also include stochasticity in the model to address the uncertainties of gas and electricity prices and the objective is to optimize the purchasing and scheduling for the purpose of profit maximization.

Three different stochastic TSPs are studied in this dissertation and various uncertain parameters will be respectively considered for each problem. The goal is to find the route with the minimum expected total travel cost while the risk associated with the uncertainty can be well managed. We also assume that the route is determined before the random travel costs are known with certainty and no route re-optimization is allowed due to time constraints or applicability; The priori optimal route must be followed once selected. As the total cost is just a summation of costs on all utilized arcs, we can use the mean costs of arcs to construct the objective function, producing a model equal to its deterministic counterpart. In the presence of the described uncertainty, however, it might not be appropriate to simply pursue the optimal expected total cost. For example, when the arc costs are uncertain and the costs of one or more selected arcs are higher than their expected values, the risk would exist and the total cost might exceed the limit that can be accepted. Under the uncertain network topology, on the other hand, the selected route might be heavily exposed to the risk that the route will fail. Thus, a solution with a higher expected cost but lower risk would be preferred over ones with much higher risk of being overcostly. The problem then cannot be modeled by a deterministic TSP without the risk management.

Of particular interest is the fact that many public transportation systems have been constructed and operated in urban areas to satisfy the high travel demand. However, a public system may cost a lot to maintain a satisfactory level of service for its customers and it is of primary concern that the public system is to be operated in the optimal way that the travel cost is minimized without

sacrificing the system performance. In addition to the three TSPs, therefore, we also investigate an optimal scheduling problem to address this concern for one of the public transport systems, in which all vehicles are electrically powered and provided point-to-point services to fulfill the real-time demand.

This dissertation would benefit other researchers and practitioners who seeks optimal solutions for similar problems in these two areas when the stochasticity or uncertainty is involved. Various advanced solution algorithms are developed and can be referred to for managing the intractability of the problems. The structure of the dissertation is as follows. In Chapter 2 we will briefly review related works in both areas of transportation and power systems. The models, optimization methodologies and solution algorithms for the proposed problems in power systems will be described in Chapter 3 and those for transportation problems will be discussed in Chapter 4. The last chapter concludes with a discussion of future work. In specific, the problems that are studied in the dissertation are listed as follows,

- (1) Electricity infrastructure investment problem.
- (2) Maintenance Scheduling problem for Natural Gas Powered Plants.
- (3) Stochastic TSP with each arc cost discretely distributed.
- (4) Stochastic TSP with each arc cost continuously distributed.
- (5) Reliable routing problems under uncertain topology.
- (6) Optimal Scheduling problem of personal rapid transit system

CHAPTER 2: REVIEW OF ALGORITHMS FOR SOLVING STOCHASTIC MIXED-INTEGER PROGRAMS

Solving large-scale stochastic mixed integer programming (SMIP) could be challenging and intractable. Many solution methods have been developed in the past to address the computational difficulty. In this chapter, we briefly review those that will be used as the foundation for solving our problems. Consider the general formulation in (2.1), where both X and Y might contain integrality restrictions on x and y , respectively.

$$\min \quad c^\top x + d^\top y \quad (2.1a)$$

$$\text{s.t.} \quad Ax + Dy \geq b \quad (2.1b)$$

$$x \in X, y \in Y \quad (2.1c)$$

L-shaped method [138] or Benders decomposition [14] have been applied to solve the problems where no integrality constraints are involved or integer variables exist only in the first stage. The basic idea is to build a master problem with the first-stage variables x only, and to formulate a subproblem evaluating the recourse function $d^\top y$ with the x being fixed in the second stage. The method has been successfully used for solving two stage SMIP problems when integer variables are included only in the first stage. Specifically in the Benders decomposition algorithm, the master problem and the subproblem for the general formulation (2.1) are modeled as follows,

$$[\text{MP:}] \quad \min \quad c^\top x + \theta \quad (2.2a)$$

$$\text{s.t.} \quad \theta \geq (b - Ax)^\top \hat{u}^p, \quad p = 1, 2, \dots, Q \quad (2.2b)$$

$$0 \geq (b - Ax)^\top \hat{u}^r, \quad r = 1, 2, \dots, Q' \quad (2.2c)$$

$$x \in X \quad (2.2d)$$

$$[\text{SP:}] \quad \min \quad d^\top y \quad (2.3a)$$

$$\text{s.t.} \quad Dy \geq b - A\hat{x} \quad (2.3b)$$

$$y \in Y \quad (2.3c)$$

where (2.2c) are optimality cuts and (2.2d) are feasibility cuts, and $\{\hat{u}^p, p = 1, 2, \dots, Q\}$ is the set of extreme points of the dual space of SP, and $\{u^r, r = 1, 2, \dots, Q'\}$ is that of extreme directions if the dual space is unbounded. Specifically, The MP is initialized by including only a subset of the extreme points and/or extreme ray. The value of θ is then obtained from solving the relaxed master problem (RMP) to form a lower bound for the recourse function. The objective function value of the second stage problem, i.e., SP, is an upper bound. An optimality cut is added to the RMP if there is any gap between the two bounds is achieved. Should the SP be infeasible the feasibility cut will be added. These cuts are iteratively added to the RMP until the converge of the bounds is achieved. When the uncertainty is involved, multiple subproblems could be formulated to evaluate the recourse cost functions corresponding to different scenarios, and the solution with the minimum expected cost over all the scenarios is desired. In this case, multiple cuts can be added in each iteration to accelerate the converge of the algorithm.

The second stage value function would become non-convex when there are integer decision variables in the second stage as discussed in [23]. This makes Benders decomposition [14] or generalized Benders decomposition [53] not readily applicable because of the duality gap of integer programs. Within the last two decades, a lot of research has been done to solve SIMPs with integer variables in the second stage. In [73], a decomposition-based branch-and-cut method is proposed, where both feasibility and optimality cuts are applied, for SMIP with pure binary variables in the first stage. In [30], a generalized L-shape method is proposed by generalized Benders decompo-

sition [53], where both Gomory cuts and branch-and-bound algorithm are applied. In [126] and [128], modified Benders decomposition methods are developed by sequentially convexifying the discrete subproblem using reformulation-linearization technique [125]. In [121], [105] and [104], decomposition methods are proposed for SMIP with random recourse and discrete second stage based on disjunctive programming [9]. Recently, a finite branch-and-cut solution algorithm is developed for SMIP with pure integer decision variables in the second stage, and a cutting plane framework for multi-stage stochastic integer program is studied in [59].

Column generation has been found as a promising method to deal with some multistage SMIP problems via using variable splitting techniques [122]. Assume in the general formulation (2.1) that Y is a nonempty and bounded polyhedral set. Let y^1, y^2, \dots, y^Q denote all the extreme points of the polytope Y . Note that each point $y \in Y$ can then be represented as a convex combination of these extreme points:

$$y = \sum_{i=1}^Q \lambda_i y^i, \quad \sum_{i=1}^Q \lambda_i = 1, \quad \text{and } \lambda_i \geq 0 \forall i$$

Then the original problem can be rewritten to the following problem:

$$[\text{MP:}] \quad \min \quad c^\top x + \sum_{i=1}^Q (d^\top y^i) \lambda_i \quad (2.4a)$$

$$\text{s.t.} \quad Ax + \sum_{i=1}^Q (Dy^i) \lambda_i \geq b \quad (2.4b)$$

$$\sum_{i=1}^Q \lambda_i = 1 \quad (2.4c)$$

$$x \in X, \lambda_i \geq 0 \quad \text{for } i = 1, 2, \dots, Q \quad (2.4d)$$

A general principle behind the column generation is that, because there are a huge number of columns (e.g., extreme points) in a problem, we want to avoid to deal with the extensive model. It is also unnecessary to enumerate all the columns since most of them would be nonbasic. We therefore include a subset of columns and formulate a restricted master problem first, and add only

the columns that can most likely improve its objective function value. Recall that in the simplex method we iteratively select a nonbasic variable with negative reduced cost to enter the basis. Similarly in the column generation, one can compute the most negative reduced cost by solving the following *subproblem* (SP) or *oracle*:

$$[\text{SP:}] \quad \min_{y \in Y} d^\top y - \hat{u}^\top Dy - \hat{v} \quad (2.5)$$

where \hat{u} and \hat{v} are dual solutions associated with the constraints (2.4c) and (2.4d), respectively. The column The idea of column generation was firstly proposed by [52], which dealt only implicitly with the non-basic variables included in a maximal multicommodity network flow problem. [35] extended the fundamental idea to a linear program by an appropriate generalization of the duality theorem. The technique was firstly applied by [54] to obtain an approximate solution for the well-known cutting-stock problem, the problem of filling an order at minimum cost for specified numbers of lengths of materials to be cut from given stock lengths of given cost. A branch-and-price algorithm was developed by embedding the column generation within the scheme of branch-and-bound algorithm to respectively solve a vehicle routing problem under time window constraints [42] and a generalized assignment problem [118], which were formulated as integer programs.

One of the most computationally useful ideas of the 1970s is the observation that many hard integer programming problems can be viewed as easy problems complicated by a relatively small set of side constraints [50]. Multistage stochastic SMIP problems, for instance, are very difficult to manage because of the nonanticipativity constraints. Dualizing these constraints produces Lagrangian problems that are easy to solve and even decomposable in many cases. For (2.1), the constraint (2.1c) can be dualized to form the following Lagrangian problem:

$$Z_D(u) = \min_{x \in X, y \in Y} c^\top x + d^\top y + u^\top (Ax + Dy - b) \quad (2.6)$$

The Lagrangian problem provides a lower bound (for minimization problems) on the optimal value of the original problem. The Lagrangian method can thus be used in place of a linear programming relaxation to provide bounds in a branch and bound algorithm. It is clear that the best choice for u would be an optimal solution to the dual problem:

$$Z_D = \max_u Z_D(u) \tag{2.7}$$

The drawback of the Lagrangian method is that only the lower bound of the original problem can be obtained. Some heuristic algorithms are therefore needed to obtain an implementable solution.

In the context of one or more of these algorithms, we develop the solution approaches with respect to each particular problem that is modeled in this dissertation. More details will be shown and discussed in following chapters.

CHAPTER 3: LARGE-SCALE STOCHASTIC MIXED-INTEGER PROBLEMS IN POWER SYSTEMS

3.1 Electricity Infrastructure Investment

A long-term problem of electricity infrastructure investment problem is proposed in this section to address the planning for capacity expansion in electrical power systems integration with hourly resolution of electricity markets' operation. The research objective of this study is to solve the long-term problem of electricity infrastructure expansion and clean energy integration with hourly resolution of electricity markets' operation through multiscale stochastic programming approaches. This research directly responds to the challenge of the new generation of electricity power systems, with high penetration of renewable energy, energy storage, electrical vehicles and demand response. Currently neither the current infrastructure nor the modeling tools are designed to handle the uncertainty arising from incorporating the new challenges. In this research, we will study and solve a multistage and multiscale stochastic programming model for electricity system expansion planning while considering all various features of the new generation electricity power systems. The model will minimize the costs from both expansion planning and daily generations operations. This model combine the long-term expansion model and the short-term unit commitment model, and actually co-optimize these two operations. However, this leads to a huge mixed integer programming problem. Decomposition algorithms, which decompose both sub models and integrate the sub decompositions, are developed to solve the problem.

3.1.1 Introduction

The electricity sector, while vital to support our modern society and economic growth, is also a main culprit for air pollution and natural resource depletion. Various renewable energy sources and new clean technologies have emerged in response to the grand challenges. To reach the new technologies' full environmental and economic benefits, they need to be integrated into the electricity grid.

The integration is of two folds. On one hand, large-scale network infrastructures are needed to connect clean generation resources to demand centers. As majority of U.S. wind and solar resources are located in sparsely-populated areas, new transmission lines are needed to reach high penetration of renewables. On the other hand, the leading renewable energy sources, wind and solar, together with demand-side resources such as plug-in hybrid vehicles (PHEVs), exhibit considerable variability in their electricity outputs (or consumption). Such variability requires additional generation resources and sophisticated system operations to ensure system reliability.

To address the integration issues, a coordinated planning process among various authorities and organizations is needed [107]. However, existing planning models are not designed to handle the mounting uncertainties from the increasing variable-output resources and unsettled environmental policies, or to model the exceedingly large-scale electricity system with great engineering details of system operation. As a result, only suboptimal solutions can be obtained in the sense that the infrastructure investments suggested by the models are not cost minimizing or the system is not sufficiently reliable to meet future demand.

To fill this research gap, [151] firstly proposed a next-generation planning model through a multistage, multiscale stochastic mixed integer programming (SMIP) approach, which combines the long-term generation, energy storage and transmission expansion and short-term unit commitment

problems. The model directly addresses two pressing issues faced by existing planning models: (i) endogenously accounting of various sources of uncertainty, and (ii) detailed modeling of engineering and operational characteristics of a power system. The resulting model will be tested using real data from U.S. regional electricity markets, which will lead to extremely large-scale stochastic mixed-integer problems of multi-million variables and constraints. The proposed SMIP model is discussed and explored in this dissertation. The multistage and multiscale features of the model, which makes it highly structured and decomposable, are fully exploited by the developed solution method: the nested decomposition algorithm with parallel computation. With the presence of integer variables in both the planning level and the unit commitment level of the SMIP problem, the solution approach is expected to greatly enhance computing speed compared to the algorithms that solve the deterministic equivalent problem of the SMIP.

3.1.2 Literature Review

3.1.2.1 Generation and Transmission Capacity Planning Expansion Models

A planning process is to identify infrastructure needs to serve future electricity demand reliably and economically while minimizing environmental impacts. There is an extensive literature on capacity expansion and long-term planning in general [3, 13, 38, 62, 83, 88, 112] and specific applications in energy systems [24, 46, 90, 127, 141]. Using optimization models for planning is also an established practice in industry over decades, with available industrial-scale models including NEMS [1], MARKAL[57], IPM [2], ReEDS [130] and META*Net [71], to name a few. The industry and early academic papers focus on the deterministic aspects of long-term planning. While the restructuring of U.S.'s electricity sector in recent years has introduced tremendous uncertainty and poses new challenges to the long-term planning process. Stochastic models are designed to address such challenges, with stochastic programming [129, 141, 146] and dynamic programming

[46, 97, 144] being the two primary approaches. Robust optimization is also used to focus on the reliability issue of the planning process [90]. Few of the long-term models (except for [144]), however, account for the detailed engineering operations of an electricity system. Such details become increasingly important as more variable-output resources such as wind and solar plants are entering the marketplace. Their impacts to a power system can only be accurately accounted and fully understood with a short-term model with detailed modeling of power plants' engineering characteristics. Such impacts in turn will have effects over long-term capacity expansion decisions. [144] introduces a novel modeling and solution approach through approximate dynamic programming and employs multiscale modeling to integrate long-term expansion and short-term operation models. It is in line with this research work, but our proposed model can provide finer resolutions of transmission expansion down to the substation level, which will be more informative to planning authorities than models that only have zonal resolutions. In addition, the proposed model can explicitly handle integer variables, which are important to model assets with significant sunk costs (such as coal and nuclear plants) and to model the flexibility of fossil fuel plants as how fast they can be turned on or shut down. Such flexibility is key in a power system with large penetration of variable-output resources and demand fluctuations (say, caused by PHEVs).

In addition to generation capacity expansion, transmission capacity expansion is a problem specific to electricity systems as it requires detailed accounting of physical rules of electricity over a network structure. [79] provides a comprehensive survey on this subject. We will consider both the location and voltage level of transmission capacity expansions with transmission losses in a direct-current (DC) network.

3.1.2.2 *Short-term Unit Commitment Models*

Unit commitment is one of the key optimization problems in power system operations and control. The objective of the unit commitment problem is to minimize the production cost from the generators to supply the load. This optimization problem is a mixed integer nonlinear programming (MINLP) problem in nature, with integer variables indicating the on/off statuses of generation units and technical linear constraints such as capacity limits, minimum on/off hours and ramping constraints [10, 123]. However, nonlinear fuel cost functions can be approximated very well by piecewise linear functions, which lead the problem to a mixed integer linear program (MILP). Various optimization techniques including priority list, dynamic programming, Lagrangian relaxation and MILP based methods have been used to solve the problem [63, 123]. A detailed survey of the methods in 1980s was given by [124]. Along with the dramatic improvement of MILP solvability, MILP based methods prevailed recently [63].

In the past several years, more advanced power system operations methods have been proposed to address the variability and uncertainty brought by uncertain demand and increasing penetration of renewable energy sources. Stochastic unit commitment has emerged as one of the most promising tools [12, 115, 137, 142, 143]. The idea of stochastic unit commitment is to capture the uncertainty and variability of the underlying factors by simulating a number of scenarios. By simulating (optimizing) the scenarios, the uncertainty can be represented to a large extent. However, the large number of scenarios also increases the computational requirement dramatically. More advanced optimization techniques need to be applied in these cases. Other than stochastic optimization, there are other modeling approaches by Markovian decision processes, such as [103], where the stochastic unit commitment is represented in the form of factored Markov decision process models. Also stochastic unit commitment models are used within agent based models of the electricity markets under uncertainties [145].

Most of current models only considered unit commitment in the master problem. In this research, the unit commitment problem is embedded in a long-term planning model as a lower level problem. Because binary variables are needed to model the commitment statuses of the units in the second stage (lower level) of the integrated model, the problem becomes a stochastic mixed integer program (SMIP) with discrete decision variables in both the first and the second stages. This problem is very difficult to solve directly by the state-of-the-art commercial optimization software when a large number of scenarios in the second stage are simulated.

3.1.3 Model Description

We begin this section by presenting the stochastic programming model that integrates long-term planning and short-term unit commitment models. Both the long-term and short-term models are multistage decision problems in nature. However, the time scales for the decision-making are not the same. Infrastructure expansions are usually planned several years ahead due to their long lead time (development, construction, obtaining permits, etc). On the other hand, the time scales for unit commitment decisions are in hours (or even in minutes). This is so because to maintain system reliability, electricity generation and consumption must be balanced at all time. A system operator must ensure system reliability by running the unit commitment model repeatedly with updated system information to decide which units to dispatch to meet real-time demand and to manage the variability of electricity generation from intermittent resources. Since electricity transmission and generation capacity expansions are mainly driven by reliability needs, even the hourly-scale decisions can have critical impacts to long-term planning. Hence, to accurately study the infrastructure needs of an electricity grid to maintain reliability and to support high penetration of variable-output resources, it is imperative to incorporate the detailed unit commitment operations to the long-term model. The different decision-making time scales render the proposed multiscale modeling approach. In such an approach, long-term expansion decisions are represented in a multistage ‘here-

and-now’ model, which will provide a single system expansion plan with all future uncertainties in consideration. Short-term unit commitment decisions are derived from a ‘wait-and-see’ model, which provides the expected costs of various unit commitment schedules along different scenario paths. The next subsection presents the integrated stochastic long-term and short-term model.

3.1.3.1 Multiscale Long-term/Short-term Model

In the long-term planning/expansion process, investment decisions need to be made before future events and uncertainties are realized. We seek the optimal decisions that minimize total system costs regardless of the realization of future uncertainties, which are achieved through solving a multistage stochastic programming problem with nonantipativity constraints (here-and-now). We refer to this problem the upper-level (or decision-level) model in the integrated model to be presented. The decision variables in the upper-level model are the expansion (or retrofit) decisions of transmission lines, electricity power plants (both fossil fuel and renewable plants) and energy storage, while the major constraint is for the total installed capacity to exceed a certain percentage of the projected peak demand (i.e., the resource adequacy requirement). The parameters and variables related to the long-term model are summarized in Table 3.1 and 3.2, respectively. The parameters in Table 3.1 with (ω) indicating that they are random variables in the model.

The total system cost to be minimized consists of the total investment cost, incurred by infrastructure expansion decisions in the upper level, and total operation cost, incurred by the short-term unit commitment operation. The formulation of the total investment cost (denoted as TIC) in time period t is given below.

$$TIC^t(x; \omega) := \underbrace{\sum_{j=1}^N \sum_{y=1}^Y [VC_{jy}^t(\omega)x_{jy}^t + FC_{jy}^t(\omega)\sigma_{jy}^t]}_{\text{generation/storage investment costs}} + \underbrace{\sum_{l=1}^L \sum_{y=1}^Y [VC_{ly}^t(\omega)x_{ly}^0 + FC_{ly}^t(\omega)\sigma_{ly}^t]}_{\text{transmission investment costs}}. \quad (3.1)$$

Table 3.1: Long-Term Related Parameters and Indices

$t = 1, \dots, T$	time periods (or stages) in the upper-level model
$j = 1, \dots, N$	locations; ($\hat{j} = 1, \dots, \hat{N}$: reserve margin regions)
$y = 1, \dots, Y$	generation types (including storage) or transmission voltage type
$l = 1, \dots, L$	number of transmission line segments in a network (existing or potential)
$VC_{jy}^t(\omega)$	variable investment costs for generation capacity type y built at j in stage t ; [\$/MW]
$FC_{jy}^t(\omega)$	fixed investment costs for generation capacities type y built at j in stage t ; [\$]
$VC_{ly}^t(\omega)$	variable investment costs for line segment l with voltage type y built in t ; [\$/MW]
$FC_{ly}^t(\omega)$	fixed investment costs for line segment l with voltage type y built in t ; [\$]
M_{jy}^t	max capacity expansion of generation type y at location j in time t ; [MW]
M_{ly}^t	max expansion capacity of line segment l with voltage type y in time t ; [MW]
$PK_{\hat{j}}^t(\omega)$	peak demand in reserve margin \hat{j} in time t ; [MW]
$RM_{\hat{j}}^t$	reserve margin requirement for region \hat{j} in time t ; [%]
DF_{jy}	derating factors of generation unity/type y at location j ; [%]
$\delta(\omega)$	discount factor
$s = 1, \dots, S_t$	possible realizations of uncertainties (i.e., scenarios) at stage t
$a(t, s)$	the predecessor of scenario s in time t
π^{ts}	the probability of realization s in time t ; $\sum_s^{S_t} \pi^{ts} = 1$

Table 3.2: Decision Variables in the Long-Term (Upper-Level) Model

κ_{jy}^{ts}	cumulative generation capacity of type y at node j in t under the scenario s ; [MW]
κ_{ly}^{ts}	cumulative transmission capacity of segment l and voltage y in t under the scenario s ; [MW]
x_{jy}^{ts}	incremental capacity of generation type y built at node j in stage t under the scenario s ; [MW]
x_{ly}^{ts}	incremental line capacity of segment l and voltage type y built in t under the scenario s ; [MW]
σ_{jy}^{ts}	binary variable for generation capacity expansion
φ_{ly}^{ts}	binary variable for transmission expansion
ϕ_{ijy}^{ts}	binary variable indicating the existence of transmission line $i - j$ of voltage type y in t under s

The time scale t in the decision-level model is in years, which can be of different length over the planning horizon T . For example, t may represent every year at the beginning, with bigger intervals (5-year, 10-year, etc) towards the end of T . Within each t , we consider the hourly unit commitment (UC) process, which is to determine the most economic and reliable schedule to commit units to

meet the the demand in real time. We refer to the UC model as the lower-level (or recourse-level) model, in which the time scale is in hours. It is not practical, nor is it necessary, to optimize the UC operation over all the hours within a t . System operators in real-world typically solves the UC problem on a rolling basis from one-hour to at most one-week ahead [64]. We consider a weekly UC problem for the operational level. Let K_t denote the set of weeks within an decision-level time interval t . Then in each t we have $|K_t|$ unit commitment optimization problems. We assume that the weekly UC problems are independent across weeks. Such an assumption may cause discrepancy of unit commitment schedules at the end or beginning of a week. However, since we consider a long-term model over decades and our main interests are on the upper-level investment decisions, the solution distortion caused by the discrepancy should be negligible. An additional benefit of the assumption is that the weekly UC problems can be solved in parallel. The UC problem can be of a here-and-now type of multistage stochastic program as well, which would make the overall integrated problem impossible to solve. It may not be necessary though, as we do not really want to know which unit is operating at a certain hour many years from today. In stead, we are interested to know what the capacity mix and transmission infrastructure are like in the future. As a result, we propose to model the weekly UC problem as a wait-and-see-type stochastic program, and use Monte Carlo approach to simulate real time uncertainties such as demand fluctuation, wind and solar plants' intermittence, and forced outage events.

In a UC problem, the system operator minimizes total operational costs subject to a series system reliability and resource constraints. The detailed formulation of a UC model is presented in the next subsection. Here we define the total operation cost (denoted as TOC) in the objective function. Given an upper-level t , let z_t^k denote the vector of all variables in the lower-level UC problem with

$k \in K_t$, and let H_k be the set of hours in week k . Then the *TOC* function in week k is as follows.

$$TOC^k(z_1^k; \mu; \omega) := \underbrace{\sum_{h \in H_k} \sum_{j \in N} \sum_{g \in G_j \cup G'_j} [SC_{jg}^h \gamma_{jg}^h(\mu; \omega)]}_{\text{start-up costs}} + \underbrace{\sum_{h \in H_k} \sum_{j \in N} \sum_{g \in G_j \cup G'_j} C_g [p_{jg}^h(\mu; \omega)]}_{\text{generation costs}}, \quad (3.2)$$

where γ_g is a binary variable to determine if a unit g is turned on, and p_g is the real (versus reactive) power generated from unit g . SC_g represents the start-up costs for unit g and $C_g(\cdot)$ is a generic operation cost function, including fuel and other O&M costs. More detailed definition of UC-related sets, parameters and variables are in Table 3.3 and 3.4.

As the upper and lower-level models have different time scales, they have different sources of uncertainty as well. Let $\xi^t(\omega)$ denote a generic random vector¹ containing all the random variables in the decision-level. $\xi^t(\omega)$ is defined on the probability space $(\Omega_\omega, \Sigma_\omega^t, P_\omega)$. Let μ denote the underlying uncertainties in the unit-commitment-level problem, and we use the vector $\zeta^k(\mu)$ to represent lower-level's random variable with $k \in K_t$. $\zeta^k(\mu)$ is defined on $(\Omega_\mu, \Sigma_\mu^k, P_\mu)$.

¹The dimension of the vector depends on the sources of uncertainty to be modeled.

Now we can present the multiscale multistage stochastic programming model as follows.

$$\begin{aligned}
\min_{z_u} \mathcal{F}(z_u; \xi) &:= TIC^0(z_u^0) + \\
&+ E_{\xi^1} \left\{ \min_{z_u^1} \delta \left[TIC^1(z_u^1(\omega)) + \sum_{\mathbf{k} \in \mathbf{K}_1} \mathbf{E}_{\zeta^{\mathbf{k}}} \min_{z_l^k} TOC^k(z_l^k(\mu; \omega)) \right] + \dots \right. \\
&\quad \left. + E_{\xi^T | \xi^{T-1} \dots \xi^1} \left\{ \min_{z_u^T} \delta^T \left[TIC^T(z_u^T(\omega)) + \sum_{\mathbf{k} \in \mathbf{K}_T} \mathbf{E}_{\zeta^{\mathbf{k}}} \min_{z_l^k} TOC^k(z_l^k(\mu; \omega)) \right] \right\} \dots \right\}
\end{aligned} \tag{3.3}$$

s.t. Generation, transmission, storage capacity expansion accounting and upper bound constraints
Zonal resource adequacy constraint.

In (3.3), the notation $E_{\zeta^k} \min_{z_l^k} TOC^k(z_l^k(\mu; \omega))$ represents a wait-and-see type of unit commitment model because the minimization is done with each realization of ζ^k , and the expectation is then taken with respect to ζ^k , exactly as in a what-if type of analysis. The bolded expectation (and summation) operations in (3.3) highlight the difference of this model to a regular a multistage here-and-now-type stochastic program.

Suppose that the support of the random vector ξ^t (denoted as Ξ^t) is discrete and finite. It is shown in [129] that energy capacity expansion problems possess the special property called block separability [84], which allows to transform the multistage model (3.3) into a two-stage stochastic problem with recourse. By writing out the scenarios (a realization of ξ^t) explicitly, the reduced

two-stage problem is as follows.

$$\begin{aligned} \min_{z_u} \quad & TIC^0(z_u^0) + \delta \sum_{s=1}^{S_1} \pi^{s1} TIC^1(z_u^1) + \delta^T \sum_{s=1}^{S_T} \pi^{sT} TIC^T(z_u^T) \\ & + \delta \sum_{s=1}^{S_1} \pi^{s1} \mathcal{Q}^{1s}(z_u^0) + \delta^T \sum_{s=1}^{S_T} \pi^{sT} \mathcal{Q}^{Ts}(z_u^{T-1, a(T,s)}) \end{aligned} \quad (3.4a)$$

s.t. Capacity expansion accounting constraints

$$\kappa_{jy}^{ts} = \kappa_{jy}^{t-1, a(t,s)} + x_{jy}^{ts}, \quad \kappa_{ly}^{ts} = \kappa_{ly}^{t-1, a(t,s)} + x_{ly}^{ts}, \quad \forall j, y, l, s, t \quad (3.4b)$$

Capacity expansion upper bound constraints

$$x_{jy}^{ts} \leq \sigma_{jy}^{ts} M_{jy}^t, \quad x_{ly}^{ts} \leq \sigma_{ly}^{ts} M_{ly}^t, \quad \forall j, l, y, s, t \quad (3.4c)$$

Zonal resource adequacy constraint

$$\sum_{j \in \hat{j}} \sum_{y=1}^Y DF_{jy} \kappa_{jy}^{t, a(t,s)} \geq (1 + RM_{\hat{j}}^t) PK_{\hat{j}}^{ts}(\omega), \quad \forall \hat{j}, s, t, \quad \text{a.s.} \quad (3.4d)$$

Nonnegativity and binary constraint

$$\kappa_{jy}^{ts}, \kappa_{ly}^{ts} \geq 0, \quad \sigma_{jy}^{ts}, \sigma_{ly}^{ts} \in \{0, 1\}, \quad \forall j, l, y, s, t. \quad (3.4e)$$

where $\mathcal{Q}^{ts}(z_u^{ts})$ is the optimal value function of the lower-level unit commitment model. The nonanticipativity of the expansion decisions is implicitly enforced by the tracking parameter $a(t, s)$.

3.1.3.2 Unit Commitment Model

This subsection describes the details of the UC model in the lower-level problem within the integrated model (3.3). There are several features of the proposed model. First, we consider the unit commitment process that co-optimizes an energy and reserve market. The energy market is where generation units are dispatched to meet demands; while the reserve market runs back-up genera-

tion resources to maintain system reliability. The co-optimization mechanism has recently been adopted by several system operators in the U.S., and is believed to provide more economic value to flexible generators that can be used to compensate the variability of outputs from intermittent renewable resources. Second, we explicitly consider energy storage resources, which again offers flexibility to a system with large-scale variable-output resources. Third, we have model (nonlinear) transmission losses in a DC-approximation of an AC-transmission network. As certain renewable resources locate in the rural part of this country, long transmission lines are needed to connect such resources to the consumers, which will incur transmission losses that should not be ignored and may impact long-term expansion decisions. For the ease of arguments, the parameters and variables related to the lower-level UC problem are summarized in Table 3.3 and 3.4.

Table 3.3: Parameters, Sets and Indices for short-term UC model

P_{\min}^j	minimum amount of power generated by bus j
P_{\max}^j	maximum amount of power generated by bus j
\overline{SR}_t^j	spinning reserve of bus j at hour h
RU_h^j	ramp-up requirements of bus j at hour h
RD_h^j	ramp-down requirements of bus j at hour h
$D_h^j(\xi)$	real-time demand at bus j at hour h
Δ_h^{ij}	susceptance between bus i and bus j at hour h
θ_j	phase angle at hour h at generating unit j
G_j	set of existing generating units in bus j ; G'_j : set of potential units at j
N	set of generating nodes
A	set of all lines among buses in the network
L_g	required run time of generator g
T	set of all possible time periods t

For a given upper-level stage t , a week k , and a given upper-level scenario s , the unit commitment model is as follows.

$$\text{minimize} \quad \sum_{h \in H_k} \sum_{j \in N} \sum_{g \in G_j \cup G'_j} (SU_{jg}^t \gamma_{jg}^h) + \sum_{h \in H_k} \sum_{j \in N} \sum_{g \in G_j \cup G'_j} F_g(p_{jg}^t) \quad (3.5a)$$

Table 3.4: Short-term Decision Variables

α_{gh}^j	commitment decision on power generator unit g in hour h at bus j
p_{gh}^j	amount of power generated from power generator unit g in hour h at bus j
γ_{gh}^j	whether power generator unit g is turned on starting from hour h at bus j
p_{gh}^j	amount of power generated from generator unit g in hour h at bus j
p_h^j	total amount of power generated from generator bus j in hour h
f^{ij}	power flow between bus i and bus j
s_{gh}^j	spinning reserve of power generator unit g in hour h at bus j
q_h^j	actually power utilized from bus j at hour h
u_h^j	total power utilized from storage facilities of bus j at hour h
v_h^j	total power saved into storage facilities of bus j at hour h
r_h^j	total remaining power in storage facilities of bus j at the beginning of time hour h

subject to Generation unit availability constraint given the upper level expansion decision κ_{jg}^{ks}

$$\alpha_{jg}^t \leq \frac{\kappa_{jg}^{ks}}{m_{jg}}, \quad \forall g \in G'_j, j \in N, h \in H_k, \quad (3.5b)$$

Minimum up and down constraints of all generators

$$\alpha_{jg}^\tau - 1 \leq \alpha_{jg}^t - \alpha_{jg}^{t-1} \leq \alpha_{jg}^\tau, \quad \tau = t, \dots, \min\{t + L_g - 1, T\}, g \in G_j \cup G'_j, j \in N, h \in H_k, \quad (3.5c)$$

Startup action indicator constraint where γ_{jg}^t denote the startup action

$$\gamma_{jg}^h \geq \alpha_{jg}^t - \alpha_{jg}^{t-1}, \quad \forall g \in G_j \cup G'_j, j \in N, h \in H_k, \quad (3.5d)$$

Total Generation at bus/node j

$$\sum_{g \in G_j \cup G'_j} p_{jg}^t = p_j^t, \quad \forall j \in N, h \in H_k, \quad (3.5e)$$

Upper bound on the total of power dispatch and spinning reserve

$$0 \leq p_{jg}^h + s_{jg}^t \leq \begin{cases} P_g^{\max} \alpha_{jg}^t, & \forall g \in G_j, j \in N, h \in H_k, \\ \kappa_{jg}^{ks}, & \forall g \in G'_j, j \in N, h \in H_k, \\ M_{jg}^k \alpha_{jg}^t, & \forall g \in G'_j, j \in N, h \in H_k, \end{cases} \quad (3.5f)$$

Nodal spinning reserve constraint

$$\sum_{g \in G_j \cup G'_j} s_{gt}^j \geq SR_j^t, \quad \forall j \in N, h \in H_k, \quad (3.5g)$$

Maximum ramping up and down limit

$$p_{jg}^t - p_{jg}^{t-1} \leq RU_{jg}, \quad p_{jg}^{t-1} - p_{gt}^j \leq RD_{jg}, \quad \forall g \in G_j \cup G'_j, j \in N, h \in H_k, \quad (3.5h)$$

Nodal electrical power consumption from both generation and storage

$$q_j^h = \rho_j u_j^t + p_j^h - v_j^t, \quad \forall j \in N, h \in H_k, \quad (3.5i)$$

Limit of power consumed from the storage

$$u_j^t \leq r_j^t, \quad \forall j \in N, h \in H_k, \quad (3.5j)$$

Available power in storage

$$r_j^t = r_j^{t-1} + v_j^{t-1} - u_j^{t-1}, \quad \forall j \in N, h \in H_k, \quad (3.5k)$$

Power storage capacity

$$0 \leq r_j^t \leq R_j^{ks}, \quad \forall j \in N, h \in H_k, \quad (3.5l)$$

DC-OPF Kirchoff's Current Law (including power transmission loss)

$$\sum_{(i,j) \in A_i^+} f_{ij}^h - \sum_{(j,i) \in A_i^-} (f_{ji}^t - l_{ji}^t) = q_j^t - D_j^t, \quad \forall i \in N, h \in H_k, \quad (3.5m)$$

Calculating the power loss

$$l_{ij}^t = B_{ij} (f_{ij}^t)^2, \quad \forall (i, j) \in A \cup A', h \in H_k, \quad (3.5n)$$

DC-OPF Kirchoff's Voltage Law (including potential new lines)

$$|f_{ij}^t - \Delta_{ij}^t (\theta_i^t - \theta_j^t)| \leq \begin{cases} 0, & \forall (i, j) \in A, h \in H_k, \\ M(1 - \phi_{ij}^{ks}), & \forall (i, j) \in A', h \in H_k \end{cases} \quad (3.5o)$$

$$0 \leq f_{ij}^h \leq \begin{cases} M_{ij}, & \forall (i, j) \in A, h \in H_k, \\ M_{ij}^k \phi_{ij}^{ks}, & \forall (i, j) \in A', h \in H_k, \end{cases} \quad (3.5p)$$

Lower and upper bounds on power dispatch

$$P_g^{\min} \alpha_{jg}^t \leq p_{jg}^h \leq P_g^{\max} \alpha_{jg}^t, \quad \forall g \in G_j, j \in N, h \in H_k, \quad (3.5q)$$

Binary restriction on unihcommitmenhstatus

$$\alpha_{jg}^t, \gamma_{jg}^h \in \{0, 1\}, \quad \forall g \in G_j \cup G'_j, j \in N, \forall j \in N, h \in H_k, \quad (3.5r)$$

Nonnegativity restriction

$$p_j^t, q_j^t, r_j^t, u_j^t, v_j^t \geq 0, \quad \forall j \in N, h \in H_k, \quad (3.5s)$$

$$0 \leq \theta_j^t \leq 360^\circ, \quad j \in N, h \in H_k, \quad (3.5t)$$

where A is the set of the existing transmission lines, A' is the set of candidate transmission to be built, U_j is the set of existing generation units, G'_j is the set of new generation units/types built in the node (k, s) of the scenario tree. In (3.5b), m_{jg} is small value that is less than any possible accumulative capacity. Then unit g at j is available (α_{jg}^t can choose 1) if the right hand side is bigger than 1. γ_{gt} in (3.5d) and takes binary values 0 and 1, representing the startup action at power generator g at bus j at time t . The value of “1” for γ corresponds to a generator g turning on at time t at bus j , and the value zero indicates the turning off of this generator. Constraint (3.5f) calculate the spinning reserve of each generator, where last two cases are just a linearization of the previous right hand side $\kappa_{jg}^{ks} M_{jg}^k \alpha_{jg}^t$ for the newly added generators.

(3.5m) is the DC approximation of Kirchhoff's Current Law (KCL), where A_i^+ is the set of outgoing arcs from bus i and A_i^- is the set of incoming arcs to bus i since we model each power line by two arcs with opposite directions. (3.5o) is the DC approximation of Kirchhoff's Voltage Law (KCL). (3.5n) is the equation to calculate power loss, where $B^{ij} = \frac{R_{ij}}{(U_{ij})^2}$, and R_{ij} and U_{ij} are the resistance and voltage of this power transmission line respectively. This is because the power loss of transmission is equal to the resistance of the power line multiplied by the square of the current, which is equal to the transmitted power divided by the voltage, as follows, $l_t^{ij} = R_{ij} \times (f_t^{ij}/U_{ij})^2$. Even though this is a nonlinear function, we can use piecewise linear function to approximate

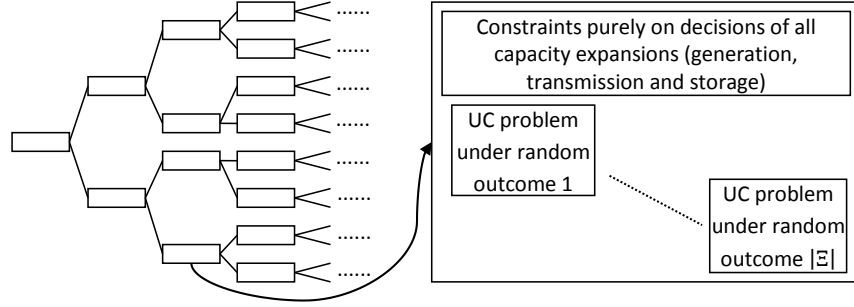


Figure 3.1: Structure of the Original Problem

it very well and use only continuous variables since it is a convex function. Also, in the objective function, $F_g(p_{jg}^t)$ is usually a quadratic function of positive second derivative (convex). We then can use a piecewise linear function to replace it, which will lead to very close to the real optimal value [63][150]. Then the deterministic UC problem is a mixed integer program with $2|T| \sum_{j \in N} \{|G_j| + |G'_j|\}$ binary variables, and the number of continuous variables are partly depending on the number of segments in the piecewise linear approximation.

3.1.4 Algorithm

The problems studied in this section are of multiple scales in terms of both time and uncertainties; this is because stochastic capacity expansion is considered in the upper level and stochastic unit commitment is embedded in the lower level, and they have planning horizons of different time scales and under different types of uncertainties. The structure of the original problem is shown in Figure 3.1. In this figure, the problem of each node in the scenario tree of the upper level is a stochastic mixed integer program by itself. This make the whole problem extremely hard to solve, since each subproblem of an outcome in the lower level is a mixed integer program, and the upper level is also multiple period stochastic mixed integer program.

Without the help of decomposition algorithm, an example of even a small size is difficult to solve directly by the start of the art commercial mixed integer optimization solvers, such as CPLEX, EXPRESS, etc., due to a huge number of integer variables and constraints. Just in the lower level, the total number of integer variables for unit commitment problems is $\frac{1-S^{K+1}}{1-S}|\Xi||T||G|$, where $G = \cup_{i \in N} \{G_i \cup H_i\}$. However, it is interesting to note that the UC problems will be separated to independent UC problems if the expansion planning decisions on generation, transmission and storage are fixed. This opens the opportunity to apply parallel computing decomposing this problem to $\frac{1-S^{K+1}}{1-S}|\Xi|$ independent individual UC problems and one stochastic expansion planning problem with linear constraints carrying the information of all UC problems.

3.1.4.1 Column Generation

The expansion planning (strategic level) is a multistage stochastic mixed integer program in its own right. We adopt variable splitting techniques to obtain a stronger formulation in the sense of having a tighter linear programming relaxation and to enable the nodal decomposition. The simplified reformulation is shown as follows,

$$\min_{x_n \in X_n, z_{mn} \in \{0,1\}} \sum_{n \in N} Prob_n(c_n x_n + \mathbb{E}_\xi d_n(\xi) y_n) \quad (3.6a)$$

$$\text{s.t.} \quad z_{mn} \leq x_m \quad m \in \mathcal{P}_n, n \in N \quad (3.6b)$$

$$\sum_{m \in \mathcal{P}_n} A_{mn} z_{mn} + B_n(\xi) y_n \geq b_n(\xi) \quad \forall n \in N, \xi \in \Xi \quad (3.6c)$$

where N is the set of nodes of the strategic-level scenario tree, and x_n and y_n are the strategic planning variables and operational scheduling variables at node n respectively, z_{mn} denotes the expansion made at node m for node n , \mathcal{P}_n is the set of nodes in the unique path from the root node to node n .

Although we have a tighter formulation for the strategic level SMIP, directly solving will still be prohibitive as it is not the convex hull and large scale. We adopt the Dantzig-Wolfe decomposition method to reformulate the large-scale intractable problem into a master problem and a set of subproblems to make the problem solvable. Specifically, the strategic-level primary master problem, [PMP], is shown as follows

$$\text{[PMP]:} \quad \min_{x_n \in X_n, \lambda_n^i \geq 0} \quad \sum_{n \in N} Prob_n \cdot c_n x_n + \sum_{n \in N} \sum_{i \in \mathcal{F}_n} Prob_n \cdot EGC_n^i \cdot \lambda_n^i \quad (3.7a)$$

$$\text{s.t.} \quad x_m \geq \sum_{i \in \mathcal{F}_n} z_{mn}^i \lambda_n^i \quad \forall m \in \mathcal{P}_n, n \in N, \quad \leftarrow \psi_{mn} \quad (3.7b)$$

$$\sum_{i \in \mathcal{F}_n} \lambda_n^i = 1 \quad \forall n \in N, \quad \leftarrow \psi_n^0 \quad (3.7c)$$

$$\lambda_n^i \geq 0 \quad \forall n \in N, i \in \mathcal{F}_n \quad (3.7d)$$

where \mathcal{F}_n is the set of all feasible expansion solutions for node n , ψ_{mn} and ψ_n^0 are the dual variables, and EGC_n^i is the expected generation cost of the whole year of \mathcal{T}_n of node n corresponding to expansion solution $z^i (z_{mn}^i; \forall m \in \mathcal{P}_n, n \in N)$, which are calculated by the following primary subproblem, [PSP], of node n as follows,

$$\text{[PSP}_n\text{]:} \quad \min_{z_{mn} \in \{0,1\}} \quad \sum_{m \in \mathcal{P}_n} \hat{\psi}_{mn} z_{mn} + \sum_{\xi \in \Xi_n} Prob_{n\xi} \cdot GC_{n\xi}(y_{n\xi}) - \hat{\psi}_n^0 \quad (3.8a)$$

$$\text{s.t.} \quad \sum_{m \in \mathcal{P}_n} A_{mn} z_{mn} + B_{n\xi} y_{n\xi} \geq b_{n\xi}, \quad \forall \xi \in \Xi_n \quad (3.8b)$$

$$y_{n\xi} \in Y_{n\xi}, \quad \forall \xi \in \Xi_n \quad (3.8c)$$

Note that there would be a extremely large number of feasible expansion plans in the set \mathcal{F}_n for each scenario node n . It is neither applicable nor necessary to include all the columns $\lambda_n^i, \forall n \in N, i \in \mathcal{F}_n$ in the [PMP]. We instead initialize the expansion problem with a subset of expansion plan columns. In specific each column represents the maximum capacity expansion that can be made for each infrastructure. The resulting restricted problem is then solved and an upper bound

on the total cost is obtained. All the excluded columns in the model are nonbasic. We evaluate them based on their reduced costs by solving the [PSP] to determine those eligible to enter the basis. We iteratively solve the two problems and update the restricted [PMP] by adding the columns with the most negative reduced cost, until the converge of the column generation is achieved.

In addition to avoiding solving the intractable SMIP model, another advantage of applying the column generation is that, the Dantzig-Wolfe reformulation can solve a much better lower bound than the linear relaxation. Moreover in our problem the coefficients and right hand sides are either 1's or 0's. This makes the problem pseudo-modularity, and integer solutions might be obtained from directly solving the linear master problem and consequently the branching can be avoided. In the instances that we tested, we can either directly achieve an implementable solution or one that provides a very reasonable approximation for the total cost.

The standard column generation algorithm, however, often exhibits slow convergence in particular for large and degenerate problems. The main reason lies in the unstable behavior of dual variables. To overcome this drawback, variations of the standard technique have been developed by stabilizing the dual variables of the master problem. Primal-dual column generation method (PDCGM) based on an interior point approach is used in [58] for the stabilization by preventing them from taking extreme values. For the stabilization techniques based on the simplex method, the stabilization can be obtained by penalizing the distance between the current and previous dual solutions, such as boxstep method [92], polyhedral penalty method [44] and bundle method [27]. A different approach to restrict large steps in the dual space does not need any modification to the restricted master problem, but convex combines the current and previous dual solutions. More details of the column generation and its stabilization can be referred to in [86, 87]. Here we apply at first the primal-dual column generation method (PDCGM) based on the interior point method. The CPLEX Barrier algorithm is used with the crossover parameter disabled to solve the master problem. We solve the problem to a certain relative gap and then use the bundle method thereafter to address

the tailing-off effect of the PDCGM. The bundle method stabilizes the dual variables via the use of Euclidean penalties, in which the restricted master problem is modified as,

$$\min_{x_n \in X_n} \sum_{n \in N} Prob_n \cdot c_n x_n + \sum_{n \in N} \sum_{i=1}^k Prob_n \cdot EGC_n^i \cdot \lambda_n^i + \sum_{n \in N} \sum_{m \in \mathcal{P}_n} (\psi_{mn}^c g_{mn} + \frac{t}{2} g_{mn}^2) \quad (3.9a)$$

$$\text{s.t.} \quad x_m - \sum_{i=1}^k z_{mn}^i \lambda_n^i + g_{mn} \geq 0, \quad \forall m \in \mathcal{P}_n, n \in N \quad (3.9b)$$

$$\sum_{i \in \mathcal{F}_n} \lambda_n^i = 1, \quad \forall n \in N \quad (3.9c)$$

where ψ_{mn}^c is the dual center selected in each iteration; g_{mn} represents a slack variable in the corresponding constraint; and t is the penalty coefficient. We will demonstrate in the computational results that the combination of these two methods exhibits the better performance than separately applying either stabilization method.

3.1.4.2 Integer L Shaped Method

Although the nodal decomposition is enabled by the Dantzig-Wolfe reformulation with the splitting variables, the set of primary subproblems (PSP) are still intractable due to the large number of scenarios in the operational level. Here We propose to further decompose each subproblem with the integer L shaped method or generalized Benders decomposition. Specifically we consider the expansion plan z as the first-stage decisions and the operations y as the second-stage ones, respectively. We then formulate the following secondary master problem as the master problem for the PSP,

$$[\text{SMP}_n] : \quad \min_{z_{mn} \in \{0,1\}} \sum_{m \in \mathcal{P}_n} \hat{\psi}_{mn} z_{mn} - \hat{\psi}_n^0 + \theta \quad (3.10a)$$

$$\text{s.t.} \quad \sum_{m \in \mathcal{P}_n} \alpha_{mn}^j z_{mn} + \beta^j \theta \geq \gamma^j, \quad j = 1, 2, \dots, J \quad (3.10b)$$

and a set of unit commitment problems are formulated to include a subset of scenarios and serve as the subproblems in the Benders decomposition method:

$$[\text{SP}_{n\xi}] : \quad \min \quad \text{Prob}_{n\xi} \cdot \text{GC}_{n\xi}(y_{n\xi}) \quad (3.11\text{a})$$

$$\text{s.t.} \quad B_{n\xi} y_{n\xi} \geq b_{n\xi} - \sum_{m \in \mathcal{P}_n} A_{mn} \hat{z}_{mn}, \quad \leftarrow \rho_{n\xi} \quad (3.11\text{b})$$

$$y_{n\xi} \in Y_{n\xi}, \quad \leftarrow \phi_{n\xi} \quad (3.11\text{c})$$

The constraints (3.10b) are feasibility and optimality cuts, which are added iteratively to approximate the value function of the unit commitment problems and include:

Benders feasibility cuts

$$0 \geq (b_{n\xi} - \sum_{m \in \mathcal{P}_n} A_{mn} z_{mn})^T \rho_{n\xi}^j + \phi_{n\xi}^j \quad (3.12)$$

Integer L shaped feasibility cuts: once an expansion is applied for a potential facility, a minimum capacity has to be reached. These cuts guarantee the feasibility of $[\text{SP}_{n\xi}]$ when its linear relaxation is feasible.

$$\sum_{m \in \mathcal{P}_n} A_{mn} z_{mn} \geq m\delta \quad (3.13\text{a})$$

$$M\delta \geq \sum_{m \in \mathcal{P}_n} z_{mn} \quad (3.13\text{b})$$

$$\delta \in \{0, 1\} \quad (3.13\text{c})$$

Total capacity requirement constraints: the total capacity of generators and storage facilities should exceed the maximum electrical demand

$$\sum_{m \in \mathcal{P}_n} \left(\sum_{g \in G} A_{mn}^g z_{mn}^g + \sum_{r \in R} A_{mn}^r z_{mn}^r \right) \geq \max\{D_\xi | \xi \in \Xi_n\} \quad (3.14)$$

Benders optimality cuts

$$\theta \geq (b_{n\xi} - \sum_{m \in \mathcal{P}_n} A_{mn} z_{mn})^\top \rho_{n\xi}^j + \phi_{n\xi}^j \quad (3.15)$$

Integer L shaped optimality cuts

$$\theta \geq (q_S - L) \left(\sum_{l \in S} z_l - \sum_{l \notin S} z_l - |S| + 1 \right) + L \quad (3.16)$$

The proposed nested decomposition algorithm, that is, solving the strategic-level master problem with column generation and the corresponding subproblems with integer L shaped method, is summarized as follows,

Algorithm 1 Nested decomposition algorithm for electricity infrastructure investment problem

- 1: Initialize: $UB(P)=+\infty$, $LB(P)=-\infty$, $LB'(P) = 0$, $i = 1$.
 - 2: **while** $UB(P) > LB(P)$ **do**
 - 3: Initialize the restricted primary master problem with $i \times N$ columns.
 - 4: Solve the reduced problem and update $UB(P)$ with its optimal objective function value (OFV).
 - 5: Get the dual solutions and model relaxed $SMP_n, \forall n \in N$.
 - 6: **for** $\langle n \in N \rangle$ **do**
 - 7: Initialize the bounds for the SMP_n : $UB(S)=+\infty$, $LB(S)=-\infty$.
 - 8: **while** $UB(S) > LB(S)$ **do**
 - 9: Solve the secondary master problem and update $LB(S)$ with the OFV.
 - 10: Model and solve the $SP_{n\xi}, \forall \xi \in \Xi_n$.
 - 11: Add Benders feasibility or optimality cuts to the SMP.
 - 12: **end while**
 - 13: Update: $LB'(P) += UB(S), i \leftarrow i + 1$.
 - 14: Generate new column λ_n^i and add them to the restricted PMP.
 - 15: **end for**
 - 16: $LB(P) = \max\{LB(P), LB'(P)\}$.
 - 17: **end while**
-

3.1.5 Algorithm Implementation and Solution Assessment

3.1.5.1 Parallel computation

As described in Section 3.1.4, the proposed electricity infrastructure investment problem is decomposed to a primary master problem and a set of primary subproblems via nodal decomposition; each primary subproblem is further decomposed to a secondary master and a set of subproblems via scenario decomposition. The nested decomposition breaks down the large-scale intractable problem into problems that are independent to each other. This allows us to apply the parallel computation to concurrently solve the problems simultaneously. The elapsed time for solving the problem can therefore largely reduced.

We perform the parallel computation in a computer cluster in which all the nodes have 20 processor cores, 64 GB of RAM, and 56 Gbps Infiniband interconnects. Each node runs Red Hat Enterprise Linux 6 (RHEL6) and use Moab Workload Manager 8 and TORQUE Resource Manager 5 as the portable batch system (PBS) for resource and job management. The parallel C++ code is compiled with Intel MPI. The parallelization for our problem is designed as shown in Figure 3.2. Specifically, the communicator, `MPI_COMM_WORLD`, is automatically created by the MPI (message passing interface) program and consists of all the processors that are initially called. The primary master problem (PMP) is assigned to one that is called master processor. In each iteration of the column generation, the dual solution is obtained from solving the PMP and sent to all other processors. A set of other communicators, `SUB_COMM`, are created by partitioning `MPI_COMM_WORLD` to N groups with respect to the scenario nodes. In each of these communicators, the corresponding secondary master problem (SMP) is assigned to one processor and the unit commitment subproblems are assigned to all other processors.

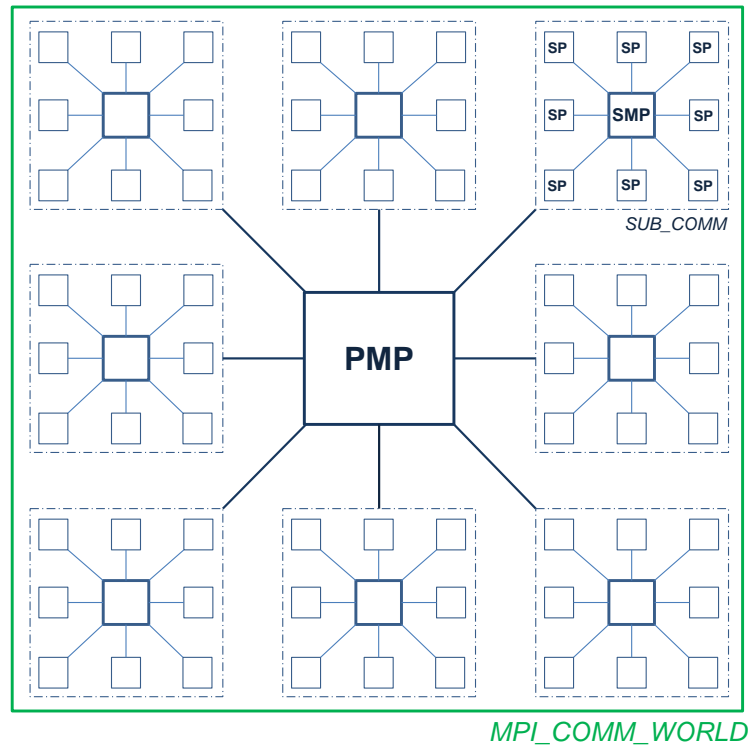


Figure 3.2: Parallelization architecture for the electricity infrastructure investment problem

3.1.5.2 Numerical Experiment and Computational Results

The developed algorithm for solving the electricity infrastructure investment problem is tested on various power systems respectively, including IEEE 7-, 57-, and 118-bus systems. For each system Table 3.5 shows the numbers of infrastructures, such as generators, renewable energy (RE) generators, transmission lines and storage facilities, to be expanded.

Table 3.5: Test Instances

System	Generators	RE generators	Transmission lines	Storage facilities
7-bus	4	1	10	2
57-bus	7	2	80	5
118-bus	54	13	186	10

We firstly test the IEEE 7-bus system to compare the performance of the proposed nested decomposition algorithm with that of directly solving the extensive SMIP model. The computational results are present in Table 3.6. The column ‘‘UC Scenarios’’ indicates the number of operational-level scenarios, and column ‘‘Stages’’ shows the number of time periods for making the strategic-level investment decisions. The wall-clock time of directly solving the SMIP model for each test instance with the commercial solver (CPLEX) is present in the column ‘‘Direct’’; and that of using the decomposition algorithm is shown in the column ‘‘Decomposition’’. The relative gap is given in the table if the problem has not been solved to optimality before the 4-hour time limit is reached.

Table 3.6: Direct vs Decomposition for the IEEE 7-bus test case.

UC Scenarios	Stages	Direct	Decomposition
20	2	43.24	348.06
	3	564.40	947.11
	4	6165.00	3166.51
	5	(4.09%)	13905.20
	6	(19.46%)	(3.07%)
100	2	1973.4	1951.3
	3	(20.68%)	5251.82
	4	(17.45%)	6791.32
	5	(32.51%)	13825.10
	6	(53.27%)	(3.54%)

Note: The percentage contained in each parenthesis represents the relative gap that is obtained when the 4 hour time limit is reached.

Our computational results show that the extensive models of small cases (2 or 3 stages with 20 UC scenarios) can be managed by the solver and exhibits the better performance than the decomposition algorithm. When the number of either strategic-level scenarios or operational-level scenarios increases, however, no converge but a considerably large gap is observed from solving the extensive model for each case with CPLEX within the time limit. With the decomposition algorithm, all the cases can be solved to either optimality or a very reasonable gap before the time limit is reached.

For larger power systems, such as IEEE 57- and 118- bus systems, the result of our experiments with ‘‘Direct’’ is that either no feasible solutions are found within the defined time limit or the prob-

lem is too large to be managed by the machine. The computational difficulty of each test instance, however, is well managed by the nested decomposition method. The parallelization results in a very reasonable wall-clock time for solving the case. Table 3.7 exhibits the computational results from solving the instances in all tested power systems. The column ‘‘Time/Gap’’ gives the total wall-clock time for solving each case to optimality while a percentage represents the gap obtained within the time limit. Column ‘‘Optimal’’ lists the optimal total expected costs for all problems.

Table 3.7: Computational results for IEEE 7-, 57, and 118- bus systems

Bus	UC Scenarios	Stages	Time/ Gap	Optimal
7	100	2	1951.3	5.58E+005
		3	5251.82	5.848E+05
		4	6791.32	6.34E+005
		5	13825.10	6.607E+05
		6	(3.54%)	7.123E+05
57	100	2	3160.12	1.08E+006
		3	6475.9	1.183E+06
		4	12826.0	1.27E+006
		5	(1.83%)	1.237E+06
		6	(5.19%)	1.388E+06
118	100	2	4406.93	5.054E+06
		3	8187.09	5.966E+06
		4	(8.39%)	6.90E+006
		5	(13.16%)	7.978E+06
		6	(20.19%)	9.932E+06

In addition, different stabilization methods are respectively applied to deal with the tailing-off effect of the column generation algorithm. Their performance are compared and present in Table 3.8. Specifically, the cases with 100 operational-level scenarios and different strategic-level stages in 118-bus system are tested, and columns ‘‘IPM’’, ‘‘Bundle’’ and ‘‘Combo’’ correspond to the computational results of using interior point method (IPM) based stabilization, bundle method and combination of the two methods, respectively. In particular for the combination, we at first

apply the CPLEX Barrier algorithm with “crossover” disabled to solve each problem to 15% gap and then use the bundle method to penalize the dual variables. Our experiments showed that more iterations of column generation can be implemented in the IPM stabilization than bundle method. This is because the master problem in this method is not modified and thus can be solved faster. However, the IPM exhibits the heavier tailing-off effect than the bundle method. We therefore use the combination of the two methods which has the better performance as shown in the table.

Table 3.8: Comparison of stabilization techniques

Bus	UC Scenarios	Stages	IPM	Bundle	Combo
		2	8451.07	4864.31	4406.93
		3	(10.31%)	12498.30	8187.09
118	100	4	(10.54%)	(7.30%)	8.39%
		5	(13.16%)	(24.63%)	(14.90%)
		6	(20.19%)	(29.50%)	(20.19%)

3.2 Maintenance Scheduling of Natural Gas Powered Plants

A profit maximization problem is present in this manuscript for natural gas fueled power plants taking into account gas contracting and the opportunity of purchasing and selling gas in the spot market as well as the maintenance scheduling. We consider the uncertainty of electricity and gas prices in the spot market, and model the problem as a multistage stochastic mixed integer program seeking the optimal dispatch for the plant over a time horizon. The multistage stochastic program is known to be intractable and integrity constraints involved make it more difficult to handle. To this respect, we apply the Lagrangian relaxation to enable the scenario-based decomposition and develop a heuristic algorithm to obtain a feasible solution at each iteration of solving Lagrangian

multipliers. Numerical experiments are performed and the results show the ability of the proposed methods in improving the manageability of the problem.

3.2.1 Introduction

Fossil fuels, including oil, coal and natural gas are the main resources for world energy supply. Natural gas has its advantage over other fuels as it is more environmentally friendly. More and more gas fueled power plants are being built in the United States due to its high productivity and low price. It is forecasted by U.S. Energy Information Administration (EIA) that the gap between the electricity generation of coal and that of natural gas will be diminishing because of environmental regulations such as the the U.S. Environmental Protection Agency's (EPA) implementation of Mercury and Air Toxics Standard (MATS), which is resulting in some coal plant retirements.

Maximizing the return on gas plant assets, however, becomes increasingly difficult. The main challenge to gas plant profitability comes from the volatility of spot gas prices. In addition to generating electricity, natural gas is also used as energy resource in other ways such as heating homes and business which would be highly affected by weather conditions. The demand and thus the spot price of natural gas tend to fluctuate more than other fuels. As a consequence, a fuel supply agreement is normally made between a fuel producer and a gas plant as the stabilization strategy to hedge the risk of the high volatility. Take-or-pay (ToP) contract is one of the most commonly practice fuel contract type for natural gas as it can not only reduce the risk of the supplier but also offer the purchaser (gas plant) the prices that are usually lower than market prices. In the contract, a purchaser of gas agrees to take a minimum quantity and pay the supplier at the contract price. Specifically, a volume of natural gas is established to define the maximum amount, M , that can be withdrawn from the supplier per month, and the plant must purchase at least $X\%$ of the amount in each month. The plant, however, can choose not to consume some of the purchased gas but

to "virtually" stored for future use within a specified period (e.g. one year). The contract also include an annual minimum withdraw percentage, $Y\%(Y \geq X)$, of the total annual volume based on the contracted monthly amount to determine the amount that the plant is obliged to purchase in each year. This means that if the cumulative purchase in the end of the year is less than $(Y\%)12M$ then the plant must purchase the difference. In addition to the contract, a gas plant would also consider to purchase and sell the gas in the spot market to sustain a higher degree of operational flexibility in the seek of profit maximization. It can purchase gas from the market for generation when the gas price is low and the electricity price is high; on the other hand it can withdraw the gas from the ToP supplier and sell it at the spot price to easily achieve financial benefits. It has been widely studied to determine the described items included in the contract. [60] considered an optimization problem to seek the optimal option value of the contract from the buyer's view, while [110] developed a pricing model to find the optimal offered price from the seller's view respecting the buyers's reaction to the offer. [69] proposed a bilevel optimization problem to address the fact that both agents would like to take optimal decisions which are related to each other. We consider the ToP clauses as input data and aim to find the optimal dispatch for a gas plant based on the given contract for profit maximization.

Maintenance is another important factor that affects the profit of a power plant. Equipment, such as turbines, combustors and electronic controllers in a power plant are generally under great pressure to minimize operating costs and to meet the electricity demands in real time. Periodically inspecting the equipment is thus necessary to minimize possible breakdowns and malfunctions during normal running and attain the reliability of the whole power system. Since maintenance would have the equipment be taken off from the system, resulting in no electricity generation until the completion of the maintenance, its schedule should be well planned to minimize the financial loss. In deterministic case, a gas plant can simply choose to implement the maintenance and shut down the plant when the electricity price is low and that of natural gas is high. In reality, however, the

maintenance scheduling could be very challenging due to various uncertain resources over a time horizon. A stochastic programming model was developed by [132] taking into account inter-area transfer limitations and stochastic reliability constraints and the Benders decomposition was applied as the solution approach. [29] considered uncertain electric demands and unit commitment was incorporated in their model to minimize the total cost of generation and maintenance.

In this manuscript, we present a profit maximization problem integrating the gas contracting and the opportunity of purchasing and selling gas in the spot market as well as the maintenance scheduling for natural gas fueled power plants, and denote it as [GMP]. Decisions made at each time period, such as the amount of gas withdrawn from suppliers with the gas contract and whether to implement the maintenance for each type of equipment, would affect those at the following periods, and therefore the problem is modeled as a multistage mixed integer program (MIP). The uncertainties of spot gas and electricity prices are considered in the problem, and the value of each uncertain parameter is unknown in advance until the beginning of each stage. We assume that the spot prices are not affected by the production of the thermal plant, which is reasonable since the plant generation capacity is significantly smaller than the total capacity of the power system in general. The gas plant considered could have different types of equipment each with a distinct maintenance interval within which at least one maintenance must be performed to attain the reliability for the plant.

Our paper provides several contributions. In Section 3.2.2, we introduce and define the proposed multistage stochastic problem with gas contract and maintenance scheduling for a natural gas fueled plant and formulate its mathematical model. To address the intractability of the multistage stochastic MIP problem, we apply the lagrangian relaxation algorithm to decompose the problem into subproblems each including a subset of the scenarios, and use the subgradient method to iteratively update the lagrangian dual. A heuristic method is developed to obtain a feasible solution in each iteration by taking advantage of the special structure of the model; the algorithms

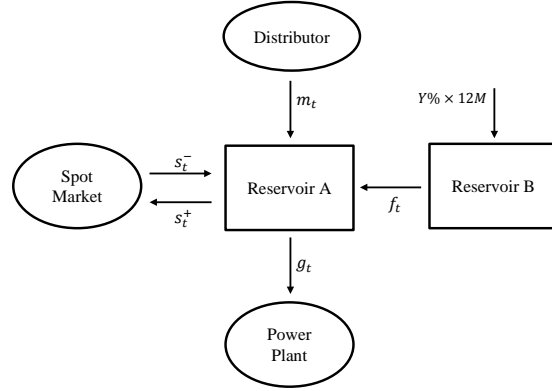


Figure 3.3: ToP contract modeled by two virtual reservoirs

are described in Section 3.2.3. Computational experiments and numerical results are presented in Section 3.2.4.

3.2.2 Problem definition

Over a finite time horizon, the objective of the proposed multistage stochastic optimization problem is to maximize the expected profit by optimize the dispatch of a natural gas fueled power plant taking into account the ToP contract as well as the maintenance scheduling. Let $t \in T$ denote the time periods on a monthly basis and $\xi \in \Xi$ represent the scenarios of uncertain parameters. The spot prices of electricity and natural gas are unknown in advance except for their probability distributions, and will be realized at the beginning of each time period.

A ToP gas contract can be modeled by two virtual reservoirs, A and B , as shown in Figure 3.3. Reservoir B receives the annual gas volume, i.e., $Y\% \times 12M$, which is pushed into the reservoir at the first month of each year and can be transferred to reservoir A in every month. Reservoir A receives both the gas transferred from B and that directly purchased from the distributors, and sends gas to the power plant for electricity generation. A mathematical model can then be formulated to

seek the profit maximization (i.e., cost minimization) for the power plant subject to flow balance constraints in the reservoir model:

$$\min \sum_{\xi \in \Xi} \text{Prob}^\xi \sum_{t \in T} \left[d_t^\xi (m_t^\xi + f_t^\xi) - h_t^\xi (g_t^\xi + s_t^{\xi-}) - p_t^\xi s_t^{\xi+} + q_t^\xi s_t^{\xi-} \right] \quad (3.17a)$$

$$\text{s.t. } a_t^\xi = a_{t-1}^\xi + m_t^\xi + f_t^\xi - s_t^{\xi+} - g_t^\xi, \quad \forall t \in T, \xi \in \Xi \quad (3.17b)$$

$$b_t^\xi = b_{t-1}^\xi - f_t^\xi, \quad \forall t \in T \setminus T_0, \xi \in \Xi \quad (3.17c)$$

$$X\%M \leq m_t^\xi + f_t^\xi \leq M, \quad \forall t \in T, \xi \in \Xi \quad (3.17d)$$

$$b_t^\xi = Y\%(12M) - f_t^\xi, \quad \forall t \in T_0, \xi \in \Xi \quad (3.17e)$$

$$b_t^\xi = 0, a_t^\xi = 0, \quad \forall t \in T_E, \xi \in \Xi \quad (3.17f)$$

$$a_t^\xi, b_t^\xi, g_t^\xi, s_t^{\xi+}, s_t^{\xi-}, m_t^\xi, f_t^\xi \geq 0 \quad \forall \xi \in \Xi, t \in T \quad (3.17g)$$

where all the parameters and decision variables are defined with respect to each scenario ξ , of which the probability is represented by Prob^ξ . The variables f_t^ξ and m_t^ξ denote the amount of gas transferred from B to A and that purchased from distributors at the time period t , respectively. The variables a_t^ξ and b_t^ξ indicate the gas level in reservoir A at the end of period t . The amount of gas used to power generation is denoted by g_t^ξ . $s_t^{\xi+}$ and $s_t^{\xi-}$ are the variables representing the gas sold to and bought from the spot market at time t , respectively. The parameters d_t defines the contracted gas price at time t , h_t^ξ is the profit gained from per unit gas when used to generate electricity, and p_t^ξ and q_t^ξ are the respective spot prices of selling out and buying in. The model aims to maximize the expected profit for the natural gas power plant over the time horizon T . The mass balance constraints for reservoir A are modeled by (3.17b), and the constraints (3.17c) and (3.17e) model the mass balance for reservoir B at the first month (T_0) of each year and the rest months, respectively. The constraint (3.17d) indicates that the amount of gas received by A is bounded below by the monthly ToP volume, $X\%M$, and above by M . Both reservoirs are restored to full capacity at the end of each year (T_E), which is defined by the constraint (3.17f).

Let I be the set of all equipment for which periodic maintenances are needed and $u_i (\forall i \in I)$ represents the maintenance cost for equipment i . Assume that Δ_i and σ_i represent the maximal length of time without maintenance (i.e., frequency) and the duration of the maintenance if applied for equipment i , respectively. The length of time period t for equipment i is denoted by $\delta_{i,t}$ ($\delta_{i,t} \ll \Delta_i, \forall i, t$). Each unit of equipment in the plant must be cyclicly inspected and the time interval between two consecutive maintenances must not exceed Δ_i for the reliability of the system. We define the variable $r_{i,t}$ that indicates the remaining time before next required maintenance at the end of period t for equipment i , and restrict it to be nonnegative to address this requirement. The remaining time $r_{i,t}$ decreases by $\delta_{i,t}$ for each period t until a new maintenance is implemented, which resets the time to maximum. We assume that the maintenance is made at the beginning of each time period for simplicity, and the following constraints are used to demonstrate the change in the maintenance status of equipment over the time horizon:

$$r_{i,t} \leq r_{i,t-1} - \delta_{i,t} + (\Delta_i + \sigma_i)z_{i,t}, \quad \forall t \in T, \quad (3.18a)$$

$$r_{i,t} \leq \Delta_i + \sigma_i - \delta_{i,t}, \quad \forall t \in T \quad (3.18b)$$

$$r_{i,t} \geq 0 \quad (3.18c)$$

We assume that the initial maintenance status for each unit of equipment is known and represented by $r_{i,0}$ ($r_{i,0} \geq 0$). $z_{i,t}$ is a binary variable defined as:

$$z_{i,t} = \begin{cases} 1, & \text{if a maintenance is scheduled in period } t \text{ for equipment } i, \\ 0, & \text{otherwise;} \end{cases}$$

The dispatch for the plant would be affected by the maintenance scheduling because its generator is turned off for the time length of σ_i during the maintenance. Therefore, the gas to be consumed for electricity generation, including that withdrawn from the supplier of the ToP contract and that

purchased from the spot market, would be reduced in the result of the maintenance, which is indicated by the constraint .

$$g_t + s_t^- \leq C(\delta_{i,t} - \sigma_i z_{i,t}) \quad \forall i \in I, \forall t \in T \quad (3.19)$$

where C is the gas consumption rate of the plant, determined by the power generation capacity. Here we assume that the profit can be obtained from every unit gas when used for power generation, which is reasonable because the amount of the electricity generated from the plant is significantly smaller than the total demand in the market. However, we note that our model is extendable to considering the impact of demand fluctuation by changing the length of each time period for each unit of equipment and the gas consumption rate.

To determine the optimal dispatch for the plant for the profit maximization, we take into account both the ToP contract and the maintenance scheduling and integrate them to create the first [GMP] model shown as follows,

$$\begin{aligned} \text{[GMP1]} \quad \min \quad & \sum_{\xi \in \Xi} \text{Prob}^\xi \sum_{t \in T} [d_t^\xi (m_t^\xi + f_t^\xi) - h_t^\xi (g_t^\xi + s_t^{\xi-}) - p_t^\xi s_t^{\xi+} + q_t^\xi s_t^{\xi-} \\ & - \sum_{i \in I} u_i z_{i,t}^\xi] \end{aligned} \quad (3.20a)$$

$$\text{s.t.} \quad (3.17b) - (3.17g) \quad (3.20b)$$

$$r_{i,t}^\xi \leq r_{i,t-1}^\xi - \delta_{i,t} + (\Delta_i + \sigma_i) z_{i,t}^\xi, \quad \forall \xi \in \Xi, \forall i \in I, \forall t \in T \quad (3.20c)$$

$$r_{i,t}^\xi \leq \Delta_i + \sigma_i - \delta_{i,t}, \quad \forall \xi \in \Xi, \forall i \in I, \forall t \in T \quad (3.20d)$$

$$g_t^\xi + s_t^{\xi-} \leq C(\delta_{i,t} - \sigma_i z_{i,t}^\xi) \quad \forall \xi \in \Xi, \forall i \in I, \forall t \in T \quad (3.20e)$$

$$r_{i,t}^\xi \geq 0, z_{i,t}^\xi \in \{0, 1\} \quad \forall \xi \in \Xi, t \in T, i \in I \quad (3.20f)$$

$$\mathbf{x}_t^\xi = \mathbf{x}_t^\eta, \quad \forall \xi \in \Xi, (\xi, \eta) \in S_{\xi,t}, t \in T \quad (3.20g)$$

The objective function in [GMP1] is equal to that in (3.17a) subtracted by the cost incurred by the

maintenance implementation. The constraint (3.20g) is non-anticipativity constraints to link the scenarios in the multistage stochastic problem and indicates that the decisions made at period t for scenario ξ should be the same as that for scenario η , and $S_{\xi,t}$ represents the set of scenarios that are indistinguishable with scenario ξ until the end of time period t .

The use of the remaining time variables and restricting them to be nonnegative to define the maintenance requirement, however, is limited as it involves more decision variables into the model specially in large-scale problems. We therefore propose the following theorem of applying the preprocessing on the input maintenance data to address this limitation.

Theorem 3.1. *The nonnegativity condition of each variable $r_{i,t}^\xi$ and thus the maintenance requirement are satisfied if and only if at least one maintenance is implemented during its previous $\phi_{i,t}$ time periods (including period t), such that $\sum_{\tau=0}^{\phi_{i,t}-1} \delta_{i,t-\tau} \leq \Delta_i + \sigma_i$ and $\sum_{\tau=0}^{\phi_{i,t}} \delta_{i,t-\tau} > \Delta_i + \sigma_i$.*

Proof. If any maintenance is performed during the described $\phi_{i,t}$ time periods, the earliest time for it would be at period $(t - \phi_{i,t} + 1)$, resulting in $r_{i,t-\phi_{i,t}+1} = \Delta_i + \sigma_i - \delta_{i,t-\phi_{i,t}+1}$. Then, $r_{i,t} \geq r_{i,t-\phi_{i,t}+1} - \sum_{\tau=0}^{\phi_{i,t}-2} \delta_{i,t-\tau} = \Delta_i + \sigma_i - \sum_{\tau=0}^{\phi_{i,t}-1} \delta_{i,t-\tau} \geq 0$, which proves the sufficiency condition.

If no maintenance is implemented during the periods, then the latest time to perform one would be at period $(t - \phi_{i,t})$, resulting in $r_{i,t-\phi_{i,t}} = \Delta_i + \sigma_i - \delta_{i,t-\phi_{i,t}}$. Then, $r_{i,t} \leq r_{i,t-\phi_{i,t}} - \sum_{\tau=0}^{\phi_{i,t}-1} \delta_{i,t-\tau} = \Delta_i + \sigma_i - \sum_{\tau=0}^{\phi_{i,t}} \delta_{i,t-\tau} < 0$, and thus the necessarily condition is probed. \square

An alternate [GMP] model is therefore developed with the remaining time variables being removed as shown below:

$$[\text{GMP2}] \quad \min \sum_{\xi \in \Xi} \text{Prob}^\xi \sum_{t \in T} c_t^\xi x_t^\xi \quad (3.21a)$$

$$\text{s.t.} \quad (3.20b), (3.20e), (3.20g) \quad (3.21b)$$

$$\sum_{\tau=0}^{\phi_{i,t}-1} z_{i,t-\tau}^{\xi} \geq 1 \quad \forall \xi \in \Xi, t \in T, i \in I \quad (3.21c)$$

$$z_{i,t}^{\xi} \in \{0, 1\} \quad \forall \xi \in \Xi, t \in T, i \in I \quad (3.21d)$$

Recall that \mathbf{x}_t^{ξ} is the vector of all variables at stage t for scenario ξ and \mathbf{c}_t^{ξ} is that of their respective coefficients in the objective function. The constraints (3.21c) is used to force that at least one maintenance is performed between time period $t - \phi_{i,t} + 1$ and t to maintain the reliability for each unit of equipment in the power plant. Note that the historical maintenance data, i.e., whether a maintenance was performed before the first time period, are known and can be used to construct the constraints in the cases of $t < \phi_{i,t}$. Although [GMP1] and [GMP2] share optimal solutions for the proposed problem, their linear programming (LP) relaxations differ from each other and thus different computational performance would be expected from solving the two models. The comparison of the LP relaxation tightness is demonstrated by Theorem 3.2.

Theorem 3.2. *A tighter LP relaxation is obtained from the reformulated [GMP2] model compared to the [GMP1].*

Proof. Assume that \hat{z} is an arbitrary feasible solution for maintenance obtained from solving the [GMP2] LP relaxation. Then we can easily generate a solution for the remaining time variables in [GMP1] such that $r_{i,t} = \min\{r_{i,t-1} - \delta_{i,t} + (\Delta_i + \sigma_i)\hat{z}_{i,t}, \Delta_i + \sigma_i - \delta_{i,t}\}, \forall i \in I, t \in T$, satisfying the constraints (3.20c) and (3.20d), with respect to each unit i . If, for time period t , $r_{i,t-\tau} = r_{i,t-\tau+1} - \delta_{i,t-\tau} + (\Delta_i + \sigma_i)\hat{z}_{i,t-\tau}, \forall \tau \in [0, \phi_{i,t} - 1]$, then the remaining time before next required maintenance for unit i at the end of this period satisfies:

$$\begin{aligned} r_{i,t} &= r_{i,t-\phi_{i,t}} - \sum_{\tau=0}^{\phi_{i,t}-1} \delta_{i,t-\tau} + (\Delta_i + \sigma_i) \sum_{\tau=0}^{\phi_{i,t}-1} \hat{z}_{i,t-\tau} \\ &\geq r_{i,t-\phi_{i,t}} + (\Delta_i + \sigma_i) \left(\sum_{\tau=0}^{\phi_{i,t}-1} \hat{z}_{i,t-\tau} - 1 \right) \end{aligned}$$

$$\geq r_{i,t-\phi_{i,t}}$$

For each unit i at each period t , the validity of $r_{i,t} \geq 0$ is therefore confirmed by recursively applying the inequations above.

If, on the other hand, there exists at least one remaining time variable takes the other value ($\Delta_i + \sigma_i - \delta_{i,t}$) during the $\phi_{i,t}$ periods, assume that the latest one is at time period k such that, $r_{i,k} = \Delta_i + \sigma_i - \delta_{i,k}$ and $r_{i,\tau} = r_{i,\tau-1} - \delta_{i,\tau} + (\Delta_i + \sigma_i)\hat{z}_{i,\tau}, \forall k < \tau \leq t$, then we would have, $r_{i,t} = \Delta_i + \sigma_i - \sum_{\tau=k}^t \delta_{i,\tau} + (\Delta_i + \sigma_i) \sum_{\tau=k+1}^t \hat{z}_{i,\tau} \geq 0$, satisfying the nonnegativity constraints.

Therefore we claim that any feasible solution obtained from the LP relaxation of [GMP2] is also feasible for that of [GMP1] and thus the theorem is proved. \square

3.2.3 Solution approaches

Multistage stochastic programs are known to be computationally intractable unless in very small instances. The plan for withdrawing natural gas from the supplier of the ToP contract and the schedule of maintenance for equipments need to be made months or even years ahead, and therefore the problem to be solved is generally in a moderate or large scale. The integer variables included at every stage make the problem more difficult to solve. Although the [GMP2] provides the stronger formulation than [GMP1], it is still extremely hard to solve the realistic-size problem directly using commercial MIP solvers, e.g., CPLEX. Some solution algorithms, therefore, need to be developed to increase the manageability of our problem.

Decomposition is a method of choice to handle the enormous size of the models in multistage stochastic programs. With an adaption of Benders decomposition (cf. [15]), the L-shaped method is developed by [139] and have been widely used to solve two-stage stochastic programs [e.g., [22,

31, 78]]. [85] extended it to multistage stochastic programs and proposed that a multistage stochastic program with block-separable recourse is equivalent to a two-stage stochastic program where the first-stage is the extensive form of the aggregate level problems and the second stage is that of the detailed level recourse functions. Consider the [GMP2] described in (3.21), the L-shaped method could be used to solve the problem by transforming it to a two-stage program, with the integer variables (i.e., whether to perform the maintenance) included in the first stage and the continuous variables (e.g., the amount of gas purchased from the spot market) in the second stage. The convergence is then obtained by iteratively generating feasibility cuts and optimality cuts from the second-stage problem and adding them to the first-stage relaxed problem. This decomposition of splitting the problem to a master problem and a subproblem, however, is very limited as the nonanticipativity constraints would still be included in both stages, resulting in the intractability of each problem. Another way of using Benders decomposition is to use it as a nested application to solve linear case, as in [21], and [109], using backward and forward induction. The backbone of this primal decomposition method is, however, the convexity of the problem at each stage, while the integrity requirements are included in our problem and feasibility and optimality cuts can no longer be obtained.

Extensive attention has been paid to dual decomposition methods for the multistage stochastic programs with discrete probability distribution of uncertainty. The dualization of nonanticipativity constraints enables scenario decomposition as in the use of augmented Lagrangian method (e.g.,[100]), and that of dynamic constraints results in nodal decomposition (e.g.,[133]). A non-zero duality gap is encountered due to the impact of integrity and [40] show that the scenario decomposition technique provides a better lower bound than the nodal decomposition. [50] summarize the percentage of problems for which a Lagrangian multiplier was discovered such that the upper bound and the lower bound converged.

In this section we dualize the nonanticipativity in the [GMP2] model to enable the scenario based

decomposition and formulate the Lagrangian dual problem to obtain a lower bound. Both the subgradient method and the primal simplex method with column generation are considered to solve the Lagrangian dual problem with different approaches to update the Lagrangian multipliers. A heuristic algorithm is developed to iteratively obtain an upper bound for the problem.

3.2.3.1 Lagrangian relaxation

Consider [GMP2], and we formulate the Lagrangian relaxation by dualizing the nonanticipativity constraints (3.20g) as follows,

$$\begin{aligned}
 [\text{LR}_\pi] \quad & \min \sum_{\xi \in \Xi} \text{Prob}^\xi \sum_{t \in T} \left(\mathbf{c}_t^\xi \mathbf{x}_t^\xi + \sum_{\eta: (\xi, \eta) \in S_{\xi, t}} \langle \boldsymbol{\pi}_{\eta, t}^\xi, \mathbf{x}_t^\xi - \mathbf{x}_t^\eta \rangle \right) & (3.22a) \\
 \text{s.t.} \quad & (3.20b), (3.20e), (3.21c), (3.21d) & (3.22b)
 \end{aligned}$$

where $\boldsymbol{\pi}_{\eta, t}^\xi$ denotes the Lagrange multipliers associated with the nonanticipativity constraints linking decisions for scenarios ξ and η at stage t . Note that the constraints (3.22b) can be partitioned into the sets with respect to the scenarios and the Lagrangian relaxation is decomposable into a set of subproblems. Let $Z_D(\boldsymbol{\pi})$ denote the optimal objective function value of [LR $_\pi$], which would provide a lower bound of the [GMP2]. It is then clear that the best choice of $\boldsymbol{\pi}$ would be an optimal solution to the Lagrangian dual problem:

$$[\text{LD}] \quad Z_D = \max_{\boldsymbol{\pi}} Z_D(\boldsymbol{\pi}) \tag{3.23}$$

We consider the column generation and subgradient methods as two respective schemes for determining the multiplier $\boldsymbol{\pi}$.

3.2.3.2 Column generation

Column generation has been a successful approach to solve large scale optimization problems with a long history ([55]). It begins with an initial Lagrangian multiplier obtained from solving a restricted Dantzig-Wolfe reformulation ([36]) including only a subset of columns, and iteratively solves a pricing problem to generate appropriate entering columns for determining the best multiplier. Consider the [GMP2], and the Dantzig-Wolfe reformulation can be viewed as the dual of the LD and is present as follows,

$$[\text{MP}] \quad \min \sum_{j=1}^Q \left(\sum_{\xi \in \Xi} \text{Prob}^\xi \sum_{t \in T} \mathbf{c}_t^\xi (\mathbf{x}_t^\xi)^j \right) \lambda_j \quad (3.24a)$$

$$\text{s.t.} \quad \sum_{j=1}^Q \left((\mathbf{x}_t^\xi)^j - (\mathbf{x}_t^\eta)^j \right) \lambda_j = \mathbf{0}, \quad \forall \xi \in \Xi, t \in T, (\xi, \eta) \in S_{\xi,t} \quad (3.24b)$$

$$\sum_{j=1}^Q \lambda_j = 1 \quad (3.24c)$$

$$\lambda_j \geq 0 \quad \forall j = 1, \dots, Q \quad (3.24d)$$

(explain more about the model, e.g., what does the lambda mean, etc.)

The column entering the basis of the MP on each iteration is obtained by solving the pricing problem, in which the objective function is to minimize the reduced cost. The pricing problem can be decomposed into a set of subproblems with respect to the scenarios $\xi \in \Xi$, shown as follows,

$$[\text{SP}(\xi)] \quad \min \sum_{t \in T} \mathbf{c}_t^\xi \mathbf{x}_t^\xi - \sum_{\eta: (\xi, \eta) \in S_{\xi,t}} \langle \hat{\boldsymbol{\pi}}_{\eta,t}^\xi, \mathbf{x}_t^\xi - \mathbf{x}_t^\eta \rangle - \hat{\rho} \quad (3.25a)$$

$$\text{s.t.} \quad a_t^\xi = a_{t-1}^\xi + m_t^\xi + f_t^\xi - s_t^{\xi+} - g_t^\xi, \quad \forall t \in T \quad (3.25b)$$

$$b_t^\xi = b_{t-1}^\xi - f_t^\xi, \quad \forall t \in T \setminus T_0 \quad (3.25c)$$

$$X \% M \leq m_t^\xi + f_t^\xi \leq M, \quad \forall t \in T \quad (3.25d)$$

$$b_t^\xi = Y \% (12M) - f_t^\xi, \quad \forall t \in T_0 \quad (3.25e)$$

$$b_t^\xi = 0, a_t^\xi = 0, \quad \forall t \in T_E \quad (3.25f)$$

$$a_t^\xi, b_t^\xi, g_t^\xi, s_t^{\xi+}, s_t^{\xi-}, m_t^\xi, f_t^\xi \geq 0 \quad \forall t \in T \quad (3.25g)$$

Recall that for scenario ξ , \mathbf{x}_t^ξ is the vector of all decision variables at stage t . $\hat{\boldsymbol{\pi}}_{\eta,t}^\xi$ represents the vector of current duals associated with the constraints (3.24b) and $\hat{\rho}$ is the dual solution of the convexity constraint (3.24c). The simplex-based column generation frequently exhibits "oscillation" and slow convergence. It is also time consuming to prove the optimality of a degenerate optimal solution of the MP. Various stabilization and acceleration techniques ([27, 44, 92]) have been developed to overcome the difficulty by restricting the distance travelled by the dual solution in the dual space of the restricted MP (RMP). As one of the most efficient and promising methods for non-smooth optimization, the bundle method is used for the column generation stabilization in this article by a quadratic penalty on the distance. More details of the method can be seen in [89].

3.2.3.3 Subgradient method

It is well known that $[\text{LR}_\pi]$ is concave, piecewise linear and subdifferentiable everywhere. The vector $(\mathbf{x}_t^\xi - \mathbf{x}_t^\eta)$ is a subgradient at any $\boldsymbol{\pi}_t^\xi$. The subgradient method can therefore be applied as a hill-climbing algorithm to obtain the best Lagrangian multipliers. Given an arbitrary initial Lagrangian multiplier, $\boldsymbol{\pi}^0$, and the sequence of positive scalar step sizes $\{t_k\}$, the sequence $\{\boldsymbol{\pi}^k\}$ is obtained iteratively by

$$\boldsymbol{\pi}^{k+1} = \boldsymbol{\pi}^k + t_k \nu(\mathbf{x}^k) \quad (3.26)$$

where \mathbf{x}^k is the vector of an optimal solution to $[\text{LR}_\pi^k]$, and $\nu(\boldsymbol{\pi}^k)$ is the subgradient at $\boldsymbol{\pi}^k$ which is determined by the violation of nonanticipativity at the k^{th} iteration:

$$\nu(\boldsymbol{\pi}^k) = [(\mathbf{x}_t^\xi - \mathbf{x}_t^\eta)^k, \forall \xi \in \Xi, t \in T] \quad (3.27)$$

Theoretically $Z_D(\boldsymbol{\pi}^k)$ will converge to its maximum Z_D if the conditions of $t_k \rightarrow 0$, $\sum_{k=0}^{\infty} t_k = \infty$ are satisfied. In practice, however, an alternate choice of step size is used most commonly and also in our study:

$$t_k = \lambda_k \frac{\bar{Z} - Z_D(\boldsymbol{\pi}^k)}{\|\nu(\boldsymbol{\pi}^k)\|^2} \quad (3.28)$$

where λ_k is a scalar satisfying $0 < \lambda_k \leq 2$ and \bar{Z} is an upper bound on Z_D , obtained by applying the heuristic algorithm described in Section 3.2.3.4. We initialize $\lambda = 2$ and successively halve it every n iterations where n is dependent on the problem size. The validation of this step size choice is provided by [61].

The difficulty encountered by the subgradient method is that there is no way to prove optimality of a Lagrangian multiplier, unless we obtain a feasible solution with its cost equals the Lagrangian function value of the multiplier. To resolve this difficulty, we terminate the method after a fixed number of iterations or a time limit, whichever is reached first. In each iteration, an primal solution \mathbf{x}^k is obtained by solving $[\text{LR}_\pi]$ and then used in the heuristic algorithm described in the following section to get a feasible solution satisfying the nonanticipativity.

In contrast, the convergence of [LD] is observable in the column generation as the upper bound can be obtained and iteratively updated with more columns entering the master problem. However, the number of rows in the master problem would equal to that of nonanticipativity constraints to be relaxed in (3.22), making the method hard to implement when the problem size increases. The performance of the column generation and subgradient method will be demonstrated and compared for some instances of our proposed [GMP] problem in the next section.

3.2.3.4 Heuristic algorithm

Although a better lower bound can be obtained from solving the Lagrangian dual problem (3.22) than using LP relaxation, it is very rare to discover a feasible solution satisfying the nonanticipativity constraints in the [GMP2] by using either the column generation or the subgradient algorithm. Note that all the uncertain parameters are included only in the objective function while none of them exists in any constraint. A heuristic algorithm is then developed to obtain feasible solutions by taking advantage of this special structure of the problem. The cost resulted from the feasible solution is also used as \bar{Z} in (3.28) to determine the step size at each iteration of the subgradient method.

Recall that the Lagrangian relaxation (3.22) can be decomposed into a set of subproblems with respect to the scenarios of uncertainty. Given any two scenarios ξ and η , and the stage t_i until the end of which the two scenarios are indistinguishable, let $\hat{\mathbf{x}}_\xi$ and $\hat{\mathbf{x}}_\eta$ represent the optimal solutions of their respective subproblems. Because no uncertain parameters are included in the constraints, both the coefficient matrices and right hand sides would be the same between the two scenario subproblems, resulting in the same feasible regions. The optimal solution of one subproblem is thus feasible but not necessarily optimal to the other due to the difference of the objective functions. Assume that $\hat{\mathbf{x}}_\xi^t = \hat{\mathbf{x}}_\eta^t, \forall t = 1, \dots, (t_i - 1)$ and $\hat{\mathbf{x}}_\xi^{t_i} \neq \hat{\mathbf{x}}_\eta^{t_i}$ (the solutions for the first $(t_i - 1)$ stages are the same while the difference is observed at stage t_i), violating the nonanticipativity. We apply one of the solutions, $\hat{\mathbf{x}}_\xi^{t_i}$ or $\hat{\mathbf{x}}_\eta^{t_i}$, to both subproblems so that the nonanticipativity is forced to be satisfied while the feasibility of other constraints are still attained. The expected cost of using either solution in both scenarios is computed and compared, and we greedily select the one with the lower cost.

The heuristic algorithm is then developed based on this idea of enforcing the nonanticipativity and implemented after each iteration of the column generation or subgradient method. Given the

Algorithm 2 Heuristic algorithm to get a feasible solution of [GMP]

- 1: Initialize the heuristic solution $\bar{\mathbf{x}} = \text{null}$ and the cost $\bar{Z} = +\infty$
- 2: Let $\hat{\mathbf{x}}$ be the optimal solution of [LR $_{\pi}$] and $\hat{\mathbf{x}}_{\xi}$ be the solution for scenario $\xi, \forall \xi \in \Xi$.
- 3: **for** $t \in T$ **do**
- 4: **for** $\xi \in \Xi$ **do**
- 5: Let $s \in S_{\xi,t}$ be the set of scenarios which are indistinguishable with scenario ξ at stage t .
- 6: Compute

$$\hat{Z}_S(\hat{\mathbf{x}}_s) = \sum_{\eta \in S_{\xi,t}} \text{Prob}^{\eta} \mathbf{c}_{\eta} \hat{\mathbf{x}}_s, \forall s \in S_{\xi,t} \quad (3.29)$$

- 7: Find s^* such that

$$\hat{Z}_S(\hat{\mathbf{x}}_{s^*}) = \arg \min_{s \in S_{\xi,t}} \hat{Z}_S(\hat{\mathbf{x}}_s)$$

- 8: Fix $\mathbf{x}_s^t = \hat{\mathbf{x}}_{s^*}^t, \forall s \in S_{\xi,t}$, in [LR $_{\pi}$].
 - 9: **end for**
 - 10: Solve the Lagrangian relaxation [LR $_{\pi}$].
 - 11: Set $\bar{\mathbf{x}}_{\xi}^t \leftarrow \hat{\mathbf{x}}_{\xi}^t$.
 - 12: **end for**
 - 13: Update the cost: $\bar{Z} = \min\{\bar{Z}, \sum_{\xi \in \Xi} \mathbf{c}_{\xi} \bar{\mathbf{x}}_{\xi}\}$.
-

solution of the Lagrangian relaxation $\hat{\mathbf{x}}$, a heuristic solution is obtained in the forward direction of time periods, i.e., from the first stage to the last one. Specifically for each period t , we partition all the scenarios into groups each representing the set of scenarios that are indistinguishable with each other at this stage. The expected cost as of applying each scenario solution ($\hat{\mathbf{x}}_s$) to all the scenarios in the same group is then computed as shown in (3.29). The scenario solution with the lowest expected cost is identified and its elements at the stage t is used as the heuristic solution. The algorithm can be summarized as follows,

We want to note that the performance of all the described algorithms could be improved by decomposing to bundles of scenarios rather than individual ones. Larger subproblems can be formulated to include multiple scenarios and the nonanticipativity links between the scenarios are not relaxed but treated directly as constraints in each subproblem, and the number of subproblems is thus reduced. Particularly for the column generation, significant decrease in the size of its master prob-

lem size can be achieved due to the decrease of the number of nonanticipativity constraints to be relaxed. However, the number of bundled scenarios must be well balanced with the increased computationally difficulty of the subproblems.

3.2.4 Computational experiments and results

3.2.4.1 Experiment setup

Numerical experiments and computational results are presented in this section on solving various instances of the [GMP] model seeking to maximize the profit for a natural gas fueled power plant taking into account the gas contracting and the opportunity of purchasing and selling gas in the spot market as well as the maintenance scheduling, subject to uncertain electricity and spot gas prices. The Lagrangian relaxation is applied to address the intractability of the problem by enabling scenario-based decomposition. The column generation and subgradient methods are respectively used to solve the Lagrangian dual and obtain a lower bound, and the results will be shown and compared. A heuristic algorithm is developed to obtain a feasible solution at each iteration, providing an upper bound for the problem.

An arbitrary power plant is defined with the gas consumption rate at each time period being determined by its capacity and used for the computational experiments. We assume that all the equipment in the plant can be categorized according to the time needed for accomplishing the maintenance for each of them. Various instances are generated to respectively include 10, 50 and 100 units of equipment for which the periodic maintenance need to be planned. The duration of each maintenance could be less than one week, between one and two weeks, or more than two weeks. We randomly generate the data of duration for each equipment from a uniform distribution on [1,7],[8,14], or [15,21] days to address this, respectively. The plant can either withdraw the

natural gas from the supplier with whom a ToP agreement is made or purchase the fuel in the spot market for generating electricity. The [GMP] is solved to obtain the optimal dispatch for the plant over a time horizon of 6 months or one year, resulting in 6- or 12-stage instances, respectively. The electricity and gas prices are randomly and respectively generated from uniform distributions of [10,12] and [2,6] (based on historical data from EIA; unit:dollars per MMBtu) to form a set of scenarios at each stage. The scenario statistics can be seen in Table 3.9.

All programs are implemented in Microsoft Visual C++ 2010 while calling the commercial MIP solver, CPLEX 12.5 (Concert Technology). All computational experiments are run in Microsoft Windows 7 Professional operating system on a Dell Desktop with Intel Core i7-2600 CPU 3.40GHz and 8GB RAM.

3.2.4.2 *Computational results and discussion*

The two MIP formulations to the problem, [GMP1] and [GMP2], are first solved with CPLEX and the comparison results are shown in Table 3.9. In the table, columns "OFV" show the objective function value associated with the best integer solution that is achieved for each instance within a time limit (1 hour), and the objective function value of the LP relaxation of each formulation is present in the columns named "LR". The computational time are recorded in seconds and displayed in the columns "time(sec)". It can be observed that the solutions with the same objective function values are obtained from the two different formulations for the small instances (6 stages), which confirms the equality between [GMP1] and [GMP2] demonstrated by Theorem 3.1. The higher values observed in the column "LR" of [GMP2] than that of [GMP1] for all instances show the tighter LP relaxation of the former formulation, which is accordant with Theorem 3.2. The stronger formulation renders [GMP2] the better performance. As shown in the table, the small instances can be solved to optimality at lower computational cost and the better solutions are obtained within

the time limit for the large instances, when the [GMP2] is applied.

Table 3.9: The comparison of two formulations for [GMP]

stages	scenarios	units	GMP1			GMP2		
			OFV	LR	time(sec)	OFV	LR	time(sec)
6	32	10	-1.806	-1.997	1.89	-1.806	-1.859	1.22
		50	-1.804	-1.996	6.38	-1.803	-1.857	2.59
		100	-1.800	-1.995	13.49	-1.800	-1.853	7.85
12	2048	10	-3.575	-3.983	3600.00	-3.576	-3.790	3600.00
		50	-2.024	-3.977	3600.00	-3.339	-3.748	3600.00
		100	-1.918	-3.970	3600.00	-2.987	-3.727	3600.00

Note: The time limit is set as 3600 seconds. The extensive forms of [GMP1] and [GMP2] can be solved to optimality with CPLEX in the small instances (6 stages) and "time(sec)" show the computational times; The objective function value (OFV, unit: 1×10^7) of the best feasible solution that is obtained within time limit is given otherwise. "LR" gives the LP relaxation of each formulation.

The computational results of using different solution algorithms in large instances (12 and 18 stages) are shown in Table 3.10, where columns "ub" and "gap" present the upper bounds and relative MIP gap obtained from various methods, respectively. It can be observed that, within a time limit (5 hours), the upper bounds (in the columns "ub") obtained from the developed heuristic are better than that from directly using CPLEX (column "Direct"). In addition, smaller lower bounds can be achieved by using the subgradient method or column generation, resulting in the smaller MIP gaps. Note that the comparison of different solution methods is not made for small-size problems (6 stages) as their extensive forms can be easily handled and thus no decomposition is needed. Alternatively, the proposed methods can be used by bundling all the scenarios and formulating only one subproblem, which would render them the same performance as directly using CPLEX.

Our results also show that the extensive forms become unmanageable when the problem size keeps growing to longer time horizon and thus more scenarios. The extremely large number of variables and constraints makes the machine run out of memory. The same difficulty is encountered by

the column generation. This is because its master problem has the same number of constraints as that of the nonanticipativity constraints in the [GMP], making the method hard to implement in the huge instances. The subgradient, on the other hand, is able to resolve the difficulty as it analytically updates the Lagrangian multipliers in each iteration after the decomposed subproblems are solved. The heuristic algorithm can also provide an upper bound for each instance by fixing the solutions in the forward direction of the time periods to enforce the nonanticipativity linking the scenarios among subproblems.

Table 3.10: Performance of various solution approaches.

stages	scenarios	units	Direct		Subgradient		Column generation	
			ub(e07)	gap	ub(e07)	gap	ub(e07)	gap
12	2048	10	-3.596	1.68%	-3.594	1.06%	-3.590	1.16%
		50	-3.404	7.23%	-3.460	1.38%	-3.459	1.38%
		100	-3.282	12.39%	-3.438	1.59%	-3.429	1.83%
18	131072	10	LoM		-4.737	14.89%	LoM	
		50	LoM		-4.332	21.44%	LoM	
		100	LoM		-4.140	25.06%	LoM	

Note: LoM represents that the machine is out of memory.

Recall that any solution that is feasible for one scenario is feasible for others as the uncertain parameters are included in the objective function only. Therefore an alternate approach the plant may have would be to assume the expected gas and electricity prices and always implement the dispatch according to these expected values, which however would result in significant loss by not considering the stochasticity in the model. For a 12-stage instance with 2048 scenarios, Figure 3.4 (*left*) shows two scenario representatives as well as the expected values of electricity and natural gas prices across all the scenarios. Our results of expected value solution indicate that the maintenance is scheduled to the 3rd and 8th stages when a small amount of electricity is generated due to the low expected price of electricity, as shown in Figure 3.4(*right*). Some natural gas is withdrawn from gas supplier and then sold to the market in the 3rd and 5th stages at relatively high expected prices.

According to our stochastic solution, on the other hand, different decisions could be made with

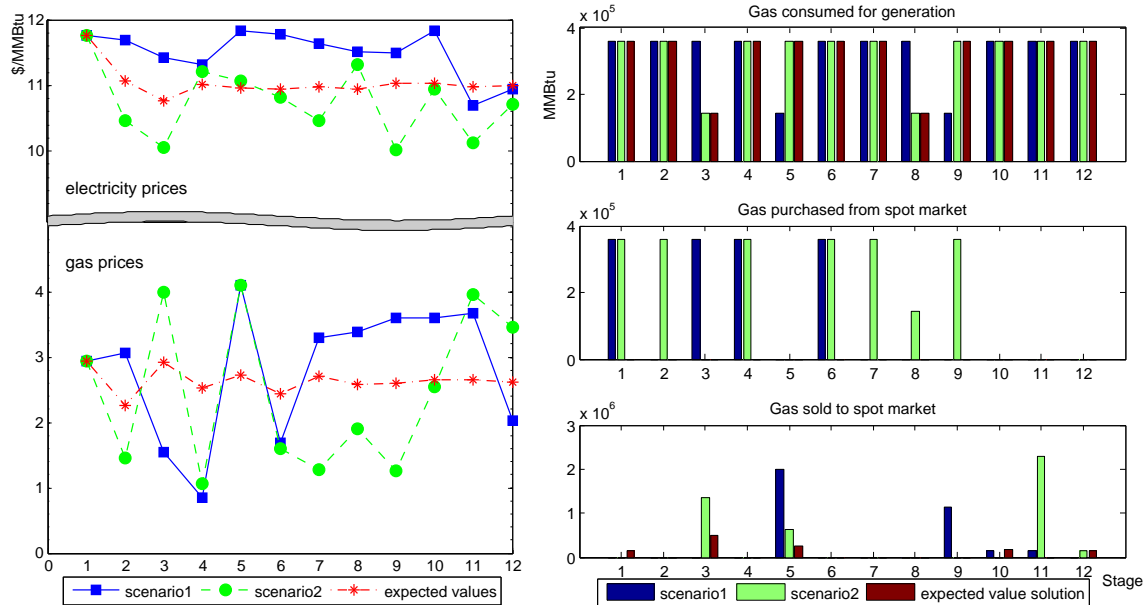


Figure 3.4: Expected value solution vs. stochastic solution. Left: the expected values of electricity and gas prices, and their particular values in two arbitrary scenarios (scenarios 1 and 2); Right: the expected value solution as well as the stochastic solution respectively for scenarios 1 and 2

respect to specific scenarios. As shown in Figure 3.4, the maintenance for scenario 1 is scheduled to the 5th when more profit can be achieved from selling the gas than generating electricity and 9th stages because of the low electricity price, while for scenario 2 it is scheduled to the 3rd and 8th stages. In contrast to the expected value solution in which no gas purchase is made, the stochastic solution indicates that a decent amount of gas can be bought from the spot market for power generation when the spot price is considerably low. The comparison between using the expected value solution and the stochastic solution for each instance is shown in Table 3.11, where the loss by not considering the stochasticity in the model is demonstrated by the relative difference between the profit gained from the two solutions.

Table 3.11: Deterministic vs. Stochastic.

stages	scenarios	units	deterministic($\times 10^7$) with expected value	stochastic($\times 10^7$)	difference
6	32	10	-1.650	-1.806	9.47%
		50	-1.648	-1.803	9.46%
		100	-1.644	-1.800	9.48%
12	2048	10	-3.158	-3.594	13.80%
		50	-3.020	-3.460	14.36%
		100	-2.998	-3.438	14.68%

CHAPTER 4: LARGE-SCALE STOCHASTIC MIXED-INTEGER PROBLEMS IN TRANSPORTATION

In this chapter, we aim to study various transportation problems, formulate their mathematical models, develop the optimization methodologies to respectively solve them and present the computational results.

4.1 Stochastic Traveling Salesman Problem with Discretely Distributed Random Arc Costs

Recently increasing attentions have been given to uncertainty handling in network optimization research. Along this trend, this section discusses the traveling salesman problem with discrete random arc costs while incorporating risk constraints. Minimizing expected total cost might not be enough because total costs of some realizations of the random arc costs might exceed the resource limit. To this respect, a model is present here incorporating risk constraints based on Conditional Value at Risk to evaluate those worst-cost scenarios. Exact solution methods are developed and applied on the risk-constrained traveling salesman problem. Numerical experiments are conducted, and the results show the ability of the proposed methods in reducing the computational complexity.

4.1.1 Introduction

Traveling salesman problem (TSP) is one of the most widely researched network optimization problems in operations research because of its wide applications and close connection to other optimization problems. Standard TSP assumes a deterministic network of nodes linked by arcs. Each of the arcs has a cost of travel, analogous to time or distance. TSP aims to find a Hamiltonian cycle that visits all nodes one and exactly one time with the minimal total cost. A brief survey

of the TSP, as well as integer linear programming (ILP) formulations of both the symmetrical and the asymmetrical TSP, can be found in [72]. Additionally, the survey reviewed different exact and heuristic methods employed to solve symmetrical and asymmetrical instances.

While in many applications, the TSP's parameters are modeled deterministically, in reality most contain elements of uncertainty. A variety of research efforts have been taken on TSP and other network optimization problems with various stochastic parameters. A mean-variance model was studied in [26] for the network flow problem with stochastic arc costs by minimizing the weighted average of the mean and variance of total traveling cost. Interval uncertain weights of edges were considered in a graph partitioning problem by [47] and robust optimization with decomposition algorithms were introduced to solve it. Nondeterministic distances between the nodes were considered in TSP by [81]. Uncertain arc capacities were modeled by [56] in a multiperiod dynamic network flow problem using scenario aggregation. Other uncertain parameters such as arc failures and node demand are also investigated, e.g., [20, 25, 95, 149].

Solving the stochastic TSP could be beneficial in many applications. Particularly for the military or military-related applications of TSP, e.g., [8, 82, 91, 106, 116, 147], many uncertain factors could exist in the route that is planned for a military manned or unmanned aerial vehicle to visit or monitor a series of targets and then to return to the base. When the traveling time of one or more arcs in the selected route is much longer than expected due to bad weather condition or enemy's presence and interference, the resources to support the mission (e.g. fuel in the tank, or battery level) might be exhausted before the mission is accomplished. By formulating and solving the problem from a stochastic perspective, we would consider the uncertainty and analyze the arc cost scenarios. The risk will then be measured and well controlled so that, at a specified confidence level, the mission can be accomplished before the resource is running out and the expected total travel cost or time is minimized as well.

In this dissertation a risk-constrained TSP model is proposed in which the arc costs are assumed to be discrete random variables. The model aims to minimize the expected total travel cost. Because the total cost is just a summation of costs on all utilized arcs, we can use the mean costs of arcs to construct the objective function, producing only a deterministic model. In the presence of the described uncertainty, however, it might not be appropriate to simply pursue the optimal expected total cost. When the costs of one or more arcs are higher than their expected values, the risk would exist and the total cost might exceed the limit that can be accepted. Thus, a solution with a higher expected cost but lower risk would be preferred over ones with much higher risk of being overcostly. The problem then cannot be modeled by a deterministic TSP after adding the risk requirements. Hence risk constraints are added in the expected total cost minimization problem so that the risk of the solution is well controlled at a specified confidence level with Conditional Value-at-Risk (CVaR), which is a common coherent risk measure. Although there exist many other risk measures (e.g., Value at Risk (VaR), Entropic Value at Risk, etc.), we choose to use CVaR because of its computational tractability, modeling flexibility and close relationship to VaR, which makes it a popular risk measure widely used in multiple areas (e.g., [6, 117, 148]). Different from our approaches, many other studies also have focused on the stochastic TSP with random arc costs. Solving stochastic routing problems were proposed by [140] via the sample average approximation method, where random samples are used to generate estimates for the expected objective function. With limited scenarios generated, however, the true optimal solution might not be achieved by this sampling method. A dynamic TSP is considered by [136] where arc costs are unknown ahead of time except its distribution but are revealed dynamically before the decision is executed. An approximate linear programming model was employed to help a salesman choose his next destination. A heuristic solution instead of an exact optimal solution was also applied in this case.

In this dissertation, an exact algorithm that is based on decomposition and scenario aggregation is

proposed to evaluate the risk (in terms of total cost) of each route by effectively analyzing a set of relevant scenarios. Benders feasibility cuts are developed to iteratively remove infeasible solutions with an unacceptable risk level. The proposed approach can avoid the enumeration of all possible scenarios and hence is able to deal with large-scale cases, which are supported by the numerical results. In addition to the application in the TSP, the algorithm could be further generalized to benefit other stochastic network optimization problems in which some information regarding the arcs/links can be modeled as discrete random variables.

The remaining of the section is organized as follows. Section 4.1.2 will give the formulation of risk-constrained traveling salesman problem. Exact solution algorithms based on Benders decomposition will be presented in Section 4.1.3. Section 4.1.4 will show the numerical experiments by using the proposed algorithms.

4.1.2 Problem Formulation

Given a network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where \mathcal{N} denotes the set of nodes and \mathcal{A} represents the set of arcs; we assume that the nodes are indexed by $i = 1, 2, \dots, n$.

A classical (deterministic) traveling salesman problem is shown as follows,

$$\text{[TSP]:} \quad \min \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} \quad (4.1a)$$

$$\text{s.t.} \quad \sum_{j:(i,j) \in \mathcal{A}} x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (4.1b)$$

$$\sum_{i:(i,j) \in \mathcal{A}} x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (4.1c)$$

$$w_i - w_j + nx_{ij} \leq n - 1, \quad 1 < i \neq j \leq n \quad (4.1d)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in \mathcal{A} \quad (4.1e)$$

$$w_j \geq 0, \quad \forall j(j > 1) \quad (4.1f)$$

where all parameters (e.g., arc cost c_{ij}) are assumed to be deterministic. The objective function (4.1a) in this formulation is the total travel cost, z , for the traveling salesman. Constraints (4.1b) and (4.1c) ensure that each node is visited exactly once and constraint (4.1d) is used to eliminate isolated subtours. Constraints (4.1e) and (4.1f) requires routing variables x_{ij} 's to be binary (with 1 meaning chosen and 0 otherwise) and node potentials w_j 's to be nonnegative respectively.

In this section, we aim to study a stochastic version of the classic TSP shown in (4.1), where we assume that the arc/link costs are random variables, denoted by C_{ij} . In addition, we assume that once the decision/route is made, the traveling salesman will not have further changes *en route*. This is reasonable for many real application cases such as disaster rescue routing, military mission routing, some commercial deliveries, etc., where real-time adjustment is costly, time consuming, or simply not possible. In this setting, the objective becomes the expected total cost (or travel time) as follows,

$$E(Z_{\mathbf{x}}) = E \left(\sum_{(i,j) \in \mathcal{A}} C_{ij} x_{ij} \right) = \sum_{(i,j) \in \mathcal{A}} E(C_{ij}) x_{ij} = \sum_{(i,j) \in \mathcal{A}} e_{ij} x_{ij} \quad (4.2)$$

where $Z_{\mathbf{x}}$, a random variable, is the random total cost given routing decisions $\mathbf{x} = [x_{ij}, (i, j) \in \mathcal{A}]^T$; E denotes the expectation function of a random variable; e_{ij} , a constant, is the expected cost of arc (i, j) .

Many of stochastic programming models are optimizing the expected values. In this case, the stochastic problem (simply minimizing total expected cost) is nothing different from a deterministic one, just using the mean cost for each arc. However, the optimal solution/route $\hat{\mathbf{x}}$ from such a program could be risky, i.e., that $Z_{\hat{\mathbf{x}}}$ could have a heavy-tailed distribution in favor of the worst cases. Risk-averse decision makers might prefer to avoid such type of decisions. Hence it is important to consider and control the risk when making decisions.

One way is to control the variance of $Z_{\mathbf{x}}$, which is one risk measure of the decision. In this respect, additional constraint, $Var(Z_{\mathbf{x}}) \leq \bar{v}$, can be added, or weighted average of the expectation

and variance of the total cost can be used as the objective function. The two approaches are not significantly different from the deterministic TSP shown in (4.1), as the former one simply includes one more constraint, and the latter one is as same as a deterministic case (using different constant costs), when the costs of each arc are independent. The main disadvantage of using variance as the risk measure is that it cannot provide many quantitative details on risk evaluations, e.g., the risky value, the risky probability (or confidence level), the maximum allowed total cost, etc. For example, the model cannot guarantee avoidance of decisions with possible extreme large total costs (over the cost threshold) by simply including variance either in the objective function or in the constraints.

However, those risk evaluation details are usually provided by many other risk measures, such as Value at Risk (VaR), Conditional Value at Risk (CVaR), Entropic Value at Risk (EVAR), etc. In this paper, we choose CVaR as the risk measure to fend off risks. Conditional Value at Risk is a coherent risk measure, which is popular in financial mathematics and risk management, and is also referred as Expected shortfall (ES), Average Value at Risk (AVaR), or Expected Tail Loss (ETL). Formulation of CVaR is tractable in terms of computation because it only involves linear constraints and continuous variables. In addition, solutions from a CVaR-based model also can provide information of VaR [113, 117], because CVaR is defined upon VaR. In our case, we would add the following constraints to the original TSP model (4.1) to control the α -level CVaR of total cost Z_x below a certain cost threshold, i.e., L .

$$\eta = \min_q \{q | P(Z_x \leq q) \geq \alpha\} \quad (4.3a)$$

$$\zeta = E[Z_x | Z_x \geq \eta] \leq L \quad (4.3b)$$

where η denotes the α -level VaR, i.e., $\text{VaR}_\alpha(Z_x)$, which is also the α -quantile of the random variable Z_x as in (4.3a); ζ represents $\text{CVaR}_\alpha(Z_x)$, which is bounded above by L as shown in (4.3b). Note that Z_x is a function of multiple random variables (the arc costs) and the routing

decisions, and then constraint (4.3) will require the selection of routes \mathbf{x} meet the threshold limit with a given confidence level, i.e., α .

We assume each individual arc cost, C_{ij} , follows a discrete probability distribution, which has a sample space $S_{ij} = \{c_{ij}^1, c_{ij}^2, \dots, c_{ij}^{|S_{ij}|}\}$. The whole sample/scenario space for the TSP problem (or the random vector $[C_{ij}, \forall (i, j) \in \mathcal{A}]^\top$) is $S = \prod_{(i,j) \in \mathcal{A}} S_{ij}$, in which each of the realization/outcome indexed by s is a combination of costs of all arcs $[c_{ij}(s), \forall (i, j) \in \mathcal{A}]^\top$, and is associated with probability P_s . Upon these assumptions and notations, by bounding the α -level CVaR, the Risk-Constrained TSP model is formulated as follows,

$$\text{[TSP-R]:} \quad \min \quad \sum_{(i,j) \in \mathcal{A}} e_{ij} x_{ij} \quad (4.4a)$$

$$\text{s.t.} \quad (4.1b) - (4.1f) \quad (4.4b)$$

$$\eta + \frac{1}{(1-\alpha)} \sum_{s \in S} P_s t_s \leq L \quad (4.4c)$$

$$t_s \geq \sum_{(i,j) \in \mathcal{A}} c_{ij}(s) x_{ij} - \eta, \quad \forall s \in S \quad (4.4d)$$

$$t_s \geq 0, \quad \forall s \in S \quad (4.4e)$$

where (4.4c) bounds $\text{CVaR}_\alpha(Z_{\mathbf{x}})$ (the left hand side) above by threshold L and η is a variable representing $\text{VaR}_\alpha(Z_{\mathbf{x}})$. t_s , a positive variable, represents the surplus (in terms of total cost) beyond the threshold for scenario s , and is 0 when the total cost $\sum_{(i,j) \in \mathcal{A}} c_{ij}(s) x_{ij}$ of scenario s is below the α -level quantile, i.e., $\text{VaR}_\alpha(Z_{\mathbf{x}})$. These special requirements and features are ensured by constraints (4.4d)-(4.4e).

Note that we need to obtain the surplus t_s for all possible scenarios within S as shown in (4.4d). In addition, all scenarios are involved in the summation to calculate $\text{CVaR}_\alpha(Z_{\mathbf{x}})$. This will be a hurdle for applying off-the-shelf commercial optimization solvers (e.g., CPLEX) directly. As discrete random costs are assumed on the arcs, the size of S grows exponentially as graph size (number of

arcs) increases. For example, if there are 2 possible costs for each arc, and then $|S| = 2^{|\mathcal{A}|}$, which means 1 million scenarios for a 20-arc instance, and 1 billion scenarios for a 30-arc instance. Most computers will not be able to handle the problems even in moderate-sized networks. Therefore, a better strategy needs to be developed to bypass the size hurdle so that [TSP-R] can be solved at an acceptable computational cost. First we need to face the fact that it is not possible to get all possible scenarios involved. Hence we need to resort some novel approach to reduce or aggregate the scenarios but still yield the exact optimal solution of [TSP-R]. Note that $\text{CVaR}_\alpha(Z_{\mathbf{x}})$ is also a function of \mathbf{x} , and only part of the elements are nonzeros in a feasible solution (a Hamiltonian cycle). This means only part of the elements are the sources of uncertainties, because zero elements (unselected arcs) do not contribute to the total cost. Hence this fact hints that a decomposition algorithm could be able to reduce the problem size significantly. In the following section, we will discuss the details of this approach and how this fact can be taken advantage to develop efficient exact solution algorithms based on decomposition.

4.1.3 Exact Solution Algorithms

Once a TSP solution (a Hamiltonian cycle) is found, the evaluation of its risk level only requires information of the arcs within the cycle and is easier than checking all possible scenarios. Hence it is enough to know the joint distribution of the costs of arcs in the Hamiltonian cycle. As many common assumptions, we also treat that the random arc costs are independent of each other.

4.1.3.1 Generating Benders feasibility cuts by solving subproblem

Benders decomposition (BD) in this situation is a good fit as it can separate TSP and risk evaluation. BD is a common row generation method which involves splitting a problem into a master problem and a subproblem and reaches the optimal solution iteratively. Here we utilize the same

idea to see by generating feasibility cuts to be incorporated in the solution space of the TSP while looking into a risk evaluation problem.

The original problem is relaxed first by eliminating the CVaR constraints to form the relaxed master problem [RMP] which includes variables $\mathbf{x} = \{x_{ij} | (i, j) \in \mathcal{A}\}$ and $\{w_i | i \in \mathcal{N} \setminus \{1\}\}$ only, shown as follows,

$$\text{[RMP]: } \min \left\{ \sum_{(i,j) \in \mathcal{A}} e_{ij} x_{ij} \mid (4.1b) - (4.1f), \sum_{(i,j) \in \mathcal{A}} b_{ij} x_{ij} \leq b_0 \right\}. \quad (4.5)$$

The [RMP] is almost identical to a standard TSP (except the additional feasibility cuts), which can be solved by the off-the-shelf commercial solvers and let the optimal solution be $\hat{\mathbf{x}} = \{\hat{x}_{ij} | (i, j) \in \mathcal{A}\}$.

Given the solution from a [RMP], $\hat{\mathbf{x}}$, the remaining task is to evaluate the risk associated with the route defined by the solution. Subproblem [SP] evaluates the risk by calculating its corresponding α -level CVaR to determine the feasibility of the route. Note that only continuous variables are involved in the [SP], which is the following feasibility check problem.

$$\text{[SP]: } \min 0 \quad (4.6a)$$

$$\text{s.t. } \eta + \frac{1}{(1-\alpha)|\bar{S}|} \sum_{s \in \bar{S}} t_s \leq L \quad (4.6b)$$

$$t_s + \eta \geq \sum_{(i,j) \in \mathcal{A}} c_{ij}(s) \hat{x}_{ij}, \forall s \in \bar{S} \quad (4.6c)$$

$$t_s \geq 0, \forall s \in \bar{S} \quad (4.6d)$$

$$\eta \geq 0 \quad (4.6e)$$

where, for convenience, we assume all arc costs' sample spaces are equally big and follows discrete uniform distribution. \bar{S} denotes the set of all related scenarios when $\hat{\mathbf{x}}$ is given, and $|\bar{S}| = 2^{|\mathcal{N}|}$ since any [RMP] solution would include exactly $|\mathcal{N}|$ arcs, assuming that the cardinality of the

sample spaces are 2. Then for each arc, there are a high cost h_{ij} and a low cost l_{ij} . Let v and $\mathbf{u} = \{u_i | i = 1, 2, \dots\}$ be the dual variables associated with the constraints (4.6b) and (4.6c), respectively. Then the dual subproblem [DSP] is given as follows,

$$\text{[DSP]} \quad \max \quad \sum_{s \in \bar{S}} \sum_{(i,j) \in A} c_{ij}(s) \hat{x}_{ij} u_s - Lv \quad (4.7a)$$

$$\text{s.t.} \quad u_s - \frac{1}{(1-\alpha)|\bar{S}|} v \leq 0, \quad \forall s \in \bar{S} \quad (4.7b)$$

$$\sum_{s \in \bar{S}} u_s - v \leq 0 \quad (4.7c)$$

$$u_s \geq 0, \quad \forall s \in \bar{S} \quad (4.7d)$$

$$v \geq 0 \quad (4.7e)$$

A lower bound (LB) of [TSP-R] is obtained by solving the [RMP]. An upper bound (UB) of [TSP-R] is then given by the summation of the LB and the objective function value of [SP], which is solved with $\hat{\mathbf{x}}$ fixed to be the optimal solution of [RMP]. Apparently the UB is always equal to LB, which means the optimal condition is always satisfied in regards to optimal solutions of [RMP]. However, when the [RMP] generates the solutions which render the [SP] infeasible, which means the CVaR of route exceeds loss of limit, the solutions are then infeasible to [TSP-R]. It can be observed that the [DSP] cannot be infeasible since the trivial solution $\mathbf{u} = 0, v = 0$ is always feasible. Therefore, the infeasible [SP] would correspond to an unbounded [DSP], and infeasibility cuts need to be developed to remove these solutions.

It's easy to calculate the total travel cost of scenario s , $c_s = \sum_{(i,j) \in A} c_{ij}(s) x_{ij}$, when the route is known. Sort the indices of these scenario costs so that $c_1 \geq c_2 \geq \dots \geq c_{|S|}$, and $\mathcal{C} = [c_s, \forall s \in S]^T$. Let $b = d^{-1} = \frac{1}{(1-\alpha)|S|}$ and $\lfloor d \rfloor$ and $\lceil d \rceil$ represent the integer part of d and $d + 1$, respectively. Then we have the following theorem.

Theorem 4.1.

$$b \sum_{s=1}^{\lfloor d \rfloor} \sum_{(i,j) \in \bar{A}} c_{ij}(s)x_{ij} + (1 - b\lfloor d \rfloor) \sum_{(i,j) \in \bar{A}} c_{ij}(\lceil d \rceil)x_{ij} - L \leq 0 \quad (4.8)$$

is a valid Benders feasibility cut to the [TSP-R], where \bar{A} denotes a set of arcs included in the given route.

Proof. Note that only linear programming (LP) is considered in the [DSP] model with all the integer variables determined by [RMP]. In addition, the feasible region of the dual formulation does not depend on the solution of [RMP]. An extreme direction can be found: $\mathbf{r} = (\mathbf{r}_u, r_v) = (b, b, \dots, b, (1 - b\lfloor d \rfloor)b, 0, \dots, 0, 1)$ where the number of b is $\lfloor d \rfloor$. Based on Bender decomposition, when the [DSP] has an unbounded feasible region, the Benders feasibility cut

$$\mathcal{C}\mathbf{r}_u - Lr_v \leq 0 \quad (4.9)$$

has to be satisfied to make the [DSP] bounded. □

As discussed before, the [DSP] is unbounded if an infeasible solution is obtained. The feasibility of a solution can be determined by comparing the CVaR at a specified probability α with the cost limit L . Note that the CVaR is exactly equal to the objective function value of a reformulated subproblem, which has the objective function as the left hand side of (4.6b) and constraints of (4.6c)-(4.6e). The dual of this subproblem is formed as the [DSP] with the term Lv removed in (4.7b) and v fixed to be 1 in (4.7c) and (4.7d). By solving the reformulated dual subproblem we have:

$$\text{CVaR}_\alpha(Z_x) = b \sum_{s=1}^{\lfloor d \rfloor} c_s + (1 - b\lfloor d \rfloor)c_{\lceil d \rceil}. \quad (4.10)$$

It can be observed that the equation (4.10) is in accordance with the definition of CVaR: the ex-

pected cost of worst scenarios.

Apparently if a route obtained from [RMP] has $\text{CVaR}_\alpha(Z_x) > L$, the inequality (4.8) is violated and thus the [DSP] is unbounded. As in the BD algorithm, the feasibility cut of (4.9), i.e. (4.8) will be generated and added to [RMP]. While being used in our problem, this method would encounter two difficulties. One is in regard to determining the feasibility of a solution. In order to do this, we compute the CVaR value of its total cost based on the sets of scenarios which have the worst costs, as shown in (4.10). When the size of problem increases, however, it would take a plenty of computational time to obtain the exact CVaR value. In a 20-node network, for example, any Hamiltonian route would contain exactly 20 arcs and there are 2^{20} cost scenarios in total. Assume that the confidence level is 90%. It's then needed to construct as many as $2^{20} \times (1 - 90\%) \approx 104858$ worst-cost scenarios and calculate their expected cost to obtain the CVaR. The other difficulty with the method is that, it is impractical to enumerate the cost of each arc in the worst scenarios, which is required to generate the coefficients in the feasibility cut (4.8). To overcome these difficulties, we propose another method to more efficiently evaluate the feasibility of [RMP] solutions and generate corresponding feasibility cuts.

4.1.3.2 Alternative valid Benders feasibility cuts (TBC) using an LB of CVaR

We note that a [RMP] solution would be infeasible if an LB of its CVaR exceeds the limit L . We construct a set of scenarios from \bar{S} as follows to generate this LB: The worst-cost scenario occurs when high costs are spent on all the arcs in a given route, and the corresponding total cost is $c = \sum_{(i,j) \in \bar{\mathcal{A}}} h_{ij}$, where $\bar{\mathcal{A}}$ denotes the set of arcs included in the route. Other scenarios are generated by taking high costs for a subset of $\bar{\mathcal{A}}$ and low costs for others. For example, $|\mathcal{N}|$ scenarios can be found when exact one arc has low cost and the others has high costs, and $\binom{|\mathcal{N}|}{2}$ scenarios are produced when low costs are spent on two arcs. So on so forth and $\lceil d \rceil$ scenarios can

then be generated by selecting up to k ($k < |\mathcal{N}|$) arcs at low costs.

Note that the expected cost of these scenarios would be no more than that of the worst scenarios, i.e. CVaR. Let $\delta_{ij} = h_{ij} - l_{ij}$ be the difference between high and low cost for each arc. Assume that the arc from node i to node j has low cost in β_{ij} out of $\lceil d \rceil$ scenarios and high cost in the others. The following LB of CVaR is then generated to determine the infeasibility of a solution:

$$\text{LCVaR}_\alpha(Z_{\mathbf{x}}) = \frac{1}{\lceil d \rceil} \sum_{(i,j) \in \bar{\mathcal{A}}} (\lceil d \rceil h_{ij} - \beta_{ij} \delta_{ij}) \leq \text{CVaR}_\alpha(Z_{\mathbf{x}}) \quad (4.11)$$

An alternative feasibility cut is then developed as follows:

Theorem 4.2.

$$\frac{1}{\lceil d \rceil} \sum_{(i,j) \in \bar{\mathcal{A}}} (\lceil d \rceil h_{ij} - \beta_{ij} \delta_{ij}) x_{ij} - L \leq 0 \quad (4.12)$$

is a valid Benders feasibility cut to our problem.

Proof. As described in the Section 4.1.3.1, the Benders feasibility cut (4.9) has to be satisfied to make the [DSP] bounded or the [SP] feasible. Reorder the indices of \mathcal{C} so that $c_1, c_2, \dots, c_{\lceil d \rceil}$ represent the costs of the scenarios that are constructed for the LCVaR. The feasibility cut (4.12) can then be derived from (4.9). When a [RMP] solution has $\text{LCVaR}_\alpha(Z_{\mathbf{x}}) > L$, the inequality (4.12) is violated and will be added to the [RMP]. \square

Although the LCVaR is not exactly equal to the expected cost of the highest-cost scenarios, its good approximation of the CVaR can still be applied to efficiently determine if a solution is infeasible or not.

The calculation of the exact CVaR is necessary if and only if the LCVaR of its cost does not exceed L but the exact CVaR does. Due to the difficulty of generating the coefficients in (4.8), we generate

the following combinatorial cut for the purpose of route elimination:

$$\sum_{(i,j) \in \bar{\mathcal{A}}} x_{ij} \leq |\mathcal{N}| - 1 \quad (4.13)$$

Apparently the arcs in the set $\bar{\mathcal{A}}$ are not allowed to be included in one Hamiltonian route in the presence of (4.13). Although this combinatorial cut would be expected to suffer low efficiency, our computational experiments in the next section demonstrate that it is not necessary to calculate the exact CVaR in most iterations while the LCVaR can be alternatively used to evaluate solutions more efficiently and generate the stronger cut (4.12).

4.1.3.3 Improved Benders feasibility cuts (IBC)

The Benders feasibility cut described above is constructed based on the set of arcs included in the given route. The improvement of the cut can be made by introducing the other binary variables regarding the other set of arcs.

Theorem 4.3.

$$\frac{1}{\lceil d \rceil} \sum_{(i,j) \in \bar{\mathcal{A}}} (\lceil d \rceil h_{ij} - \beta_{ij} \delta_{ij}) x_{ij} + \sum_{(i',j') \in \mathcal{A} \setminus \bar{\mathcal{A}}} l_{i'j'} x_{i'j'} - L \leq 0 \quad (4.14)$$

is a valid improved Benders feasibility cut to our problem.

Proof. Assume that a route satisfies the constraint (4.8), and the total $\lceil d \rceil$ scenarios is generated for computing the LB of CVaR as follows: For the arc which is also included in the presolved route, i.e. $(i, j) \in \bar{\mathcal{A}}$, low cost occurs in β_{ij} out of $\lceil d \rceil$ scenarios and high cost in the others. For the other arcs, i.e. $(i', j') \in \mathcal{A} \setminus \bar{\mathcal{A}}$, low cost occur in all $\lceil d \rceil$ scenarios. The expected cost of these generated scenarios is also no more than CVaR. Thus the cut (4.14) would be also satisfied for the route. \square

4.1.3.4 Implementing Benders Decomposition in the Branch-and-cut scheme

The BD algorithm can be implemented in two ways with the developed Benders feasibility cuts (TBC and IBC). One is regard to the traditional BD approach. Specifically, the [RMP] is solved with the branch and cut algorithm to optimality in each iteration. The solution is optimal to the [TSP-R] should the CVaR requirement be satisfied; otherwise, a feasibility cut is generated and added to the [RMP]. Within this method, however, a lot of candidate (binary) solutions would be revisited in the search tree of solving [RMP] in each iteration. Additionally, the optimal solution of our original problem would probably have been visited and discarded for many times, since its objective function value is usually worse than that of the current [RMP]. Another approach is now generally performed for the BD algorithm in the solvers, like CPLEX, by using callback functions. Along the branch-and-cut process of solving the [RMP], every candidate solution in the search tree is evaluated according to the CVaR requirement. The feasible one with the objective value better than that of the current incumbent is considered as the new incumbent and produces a new upper bound (UB). Otherwise, a TBC or IBC is added if any infeasible candidate solution is found with an unsatisfactory CVaR, giving the problem a new LB. The convergence of the LB and UB would then lead to the optimal solution.

The second approach that keeps the BD algorithm in only one search tree would be expected to be more efficient than the first one. This has also been confirmed by our computational test. In the next section, we only report the computational results of implementing the BD algorithm in the second way. The method is summarized as follows,

Algorithm 3 BD algorithm for TSP with CVaR constraints.

- 1: Set $UB = +\infty$, $LB = -\infty$ and the optimal route $\mathbf{x}^* = null$
 - 2: **while** ($UB - LB > 0$) **do**
 - 3: Solve [RMP] and obtain a candidate integer solution: $\hat{\mathbf{x}} = \{\hat{x}_{ij} | (i, j) \in \mathcal{A}\}$ with the objective value \hat{z} .
 - 4: $\bar{\mathcal{A}} = \{\text{the set of arcs included in the route defined by } \hat{\mathbf{x}}\}$
 - 5: Compute the LB of CVaR, i.e. $LCVaR_\alpha(Z_{\hat{\mathbf{x}}})$, with (4.11).
 - 6: **if** $LCVaR_\alpha(Z_{\hat{\mathbf{x}}}) > L$ **then**
 - 7: Reject the incumbent and add the Benders feasibility cut (4.12) or (4.14) to the model.
 - 8: Update LB by solving the updated [RMP].
 - 9: **else**
 - 10: Compute the exact CVaR value.
 - 11: **if** $CVaR_\alpha(Z_{\hat{\mathbf{x}}}) \leq L$ **then**
 - 12: Accept $(\hat{\mathbf{x}}, \hat{z})$ as the new incumbent and set $UB \leftarrow \hat{z}$, $\mathbf{x}^* \leftarrow \hat{\mathbf{x}}$.
 - 13: **else**
 - 14: Add the combinatorial cut (4.13) to the [RMP] to remove the current infeasible solution.
 - 15: **end if**
 - 16: **end if**
 - 17: **end while**
-

4.1.4 Numerical Experiments and Results

4.1.4.1 Experiment Setup

Numerical experiments and computational results are presented in this section on solving the [TSP-R]. The described BD algorithm is used to solve the problem in a set of directed graph networks with different numbers of nodes and arcs. The two different cuts, traditional Benders feasibility cut (TBC) and improved Benders feasibility cut (IBC), are applied in the algorithm to respectively solve the problems. The computational cost of using these two cuts will be shown and compared.

A set of graphs with different sizes of 5, 10, 15 and 20 nodes are used for the computational experiments. By assigning different outgoing degrees n_i to each node i , we form respective graphs with low, medium and high densities. Then the tail nodes are randomly and repeatedly generated for each node until the degree is fulfilled. We manually connect all the nodes in each graph with arcs to form a cycle route to avoid the existence of isolated nodes. The expected traveling cost of each arc is randomly generated from a uniform distribution on [30,40] for all sizes of graphs. In addition, we randomly generate a value between 1 and 29 for each arc representing the deviation between its low cost or high cost and the expected cost. The levels of confidence are set as $\alpha = 0.90, 0.95$ and 0.99 , respectively. The limits of cost are set as 250, 400, 600 and 800 for the graphs with the sizes of 5, 10, 15 and 20 nodes, respectively.

All programs are implemented in Microsoft Visual C++ 2010 while calling CPLEX 12.5 (Concert Technology). All computational experiments are run in Microsoft Windows 7 Professional operating system on a Dell Desktop with Intel Core i7-2600 CPU 3.40GHz and 8GB RAM.

4.1.4.2 Computational Results

The computational results obtained by using different methods or feasibility cuts are shown in Table 4.1, where computational time (in columns named “Time”) are recorded in seconds, and OFV denotes the optimal objective value.

In the table, column “DIRECT” shows the results when we call CPLEX to solve [TSP-R] directly with no decomposition strategy applied. We can quickly obtain the optimal solution in a small-sized network up to 15 arcs. However, the problem size becomes too large to be handled by our computer as the number of arcs is increased to 20 or more.

In contrast, the problem in all the generated networks can be solved at a reasonable cost by applying the the BD algorithm. By checking the LB of CVaR for current [RMP] optimal solution to generate TBC or IBC to the [RMP] and computing the exact CVaR to generate combinatorial cuts, the BD algorithm can iteratively solve the problem to obtain the optimal solution satisfying the cost limit requirement given a specified confidence level. Columns “BD-TBC” and “BD-IBC” in Table 4.1 show the computational results based on applying Traditional Benders cuts (TBC) discussed in Section 4.1.3.2 and Improved Benders cuts (IBC) in Section Section 4.1.3.3 respectively. Figure 4.1 shows the lower bounds of CVaR values in the process of solving the problem in a 15×150 network (15 nodes and 150 arcs) at the confidence level of 0.99. In most iterations, the LB of CVaR can be used to more efficiently evaluate the current solution and generate a corresponding cut. The exact CVaR is calculated only when its LB is found below the limit of total cost as shown in the figure at 18th, 31st, and 40th iterations.

The optimal objective function value (OFV) of the reliable routing problem in each network is also shown to demonstrate the lowest cost we could spend in satisfying the requirement of specified confidence level.

Table 4.1: Computational results for [TSP-R] by using Benders Decomposition algorithm

Nodes	Arcs	Confidence Level	DIRECT		BD-TBC		BD-IBC		
			Time(sec)	OFV	Time(sec)	OFV	Time(sec)	OFV	
5	10	0.90	0.078	*	0.078	*	0.09	*	
		0.95	0.109	*	0.094	*	0.09	*	
		0.99	0.109	*	0.14	*	0.08	*	
	15	0.90	41.996	180	0.156	180	0.12	180	
		0.95	115.59	*	0.203	*	0.19	*	
		0.99	87.605	*	0.187	*	0.16	*	
		0.90	*		0.093	171	0.09	171	
		20	0.95	*		0.374	173	0.41	173
			0.99	*		0.234	173	0.39	173
	10	50	0.90	*		0.765	350	1.107	350
			0.95	*		1.904	*	1.544	*
			0.99	*		1.794	*	1.246	*
70		0.90	*		1.28	335	1.111	335	
		0.95	*		1.716	341	1.31	341	
		0.99	*		11.591	349	5.819	349	
90		0.90	*		0.203	317	0.156	317	
		0.95	*		0.64	319	0.78	319	
		0.99	*		1.966	326	3.651	326	
15		90	0.90	*		1.155	498	1.154	498
			0.95	*		13.937	513	12.745	513
			0.99	*		1158.05	*	137.467	*
	120	0.90	*		1.732	489	2.028	489	
		0.95	*		1.779	489	1.935	489	
		0.99	*		22.822	500	16.437	500	
	150	0.90	*		1.092	470	1.139	470	
		0.95	*		1.217	470	1.295	470	
		0.99	*		6.287	476	10.111	476	
	20	100	0.90	*		64.436	695	138.731	695
			0.95	*		10809.5	*	3534.05	*
			0.99	*		2501.97	*	186.716	*
200		0.90	*		173.466	621	172.366	621	
		0.95	*		52.213	621	52.182	621	
		0.99	*		11.482	623	10.514	623	
300		0.90	*		61.121	619	61.457	619	
		0.95	*		16.395	619	16.317	619	
		0.99	*		214.372	625	2061.54	625	

* Problem size is too large to be solved by our computer

* The problem is infeasible

Figure 4.2 shows the process of bound contraction in the instances of 15×120 and 15×90 at the confidence level of 0.99, respectively. The UB of the problem is initialized as $+\infty$ and changed while the first incumbent solution is found. Then CPLEX keeps updating the incumbent until the

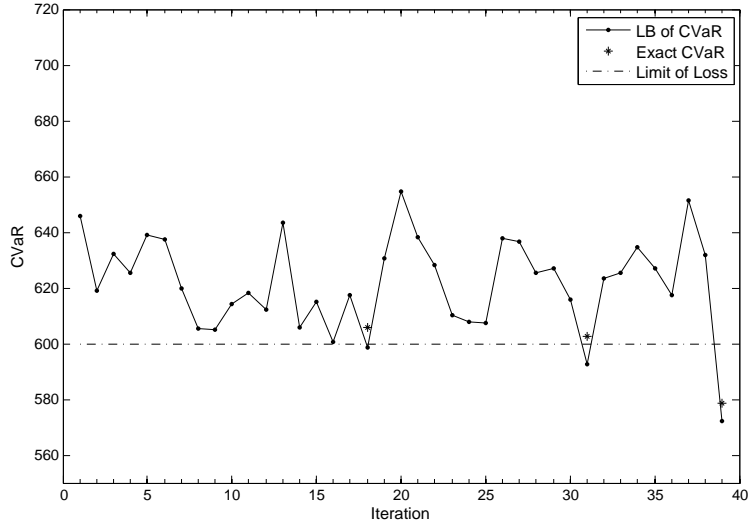


Figure 4.1: Using CVaR and its LB to evaluate the feasibility of current [RMP] solution

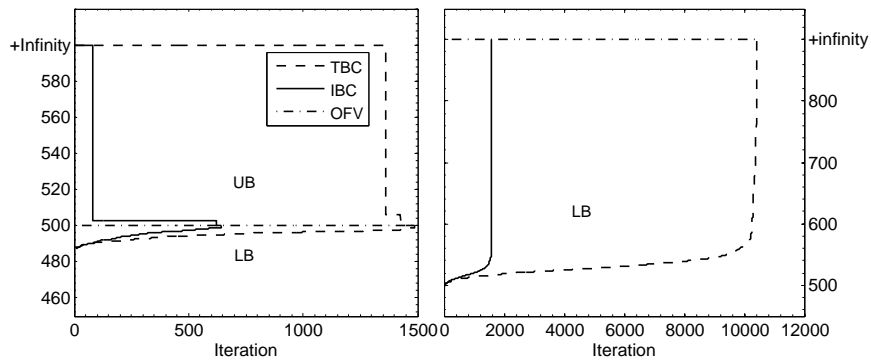


Figure 4.2: Bound contraction process in the instances of 15×120 (left) and 15×90 (right) with $\alpha = 0.99$

optimal solution is obtained. Although no obvious difference of the computational time between the use of TBC and IBC is observed, the higher efficiency of the IBC can still be confirmed by its fewer-iteration convergence. Our results show that the developed IBC can also identify those infeasible problems (e.g. in the 15×90 and 20×100 networks) more efficiently than the TBC. This is because the improved cuts can remove the infeasible integer solutions and reduce the search space in a faster way.

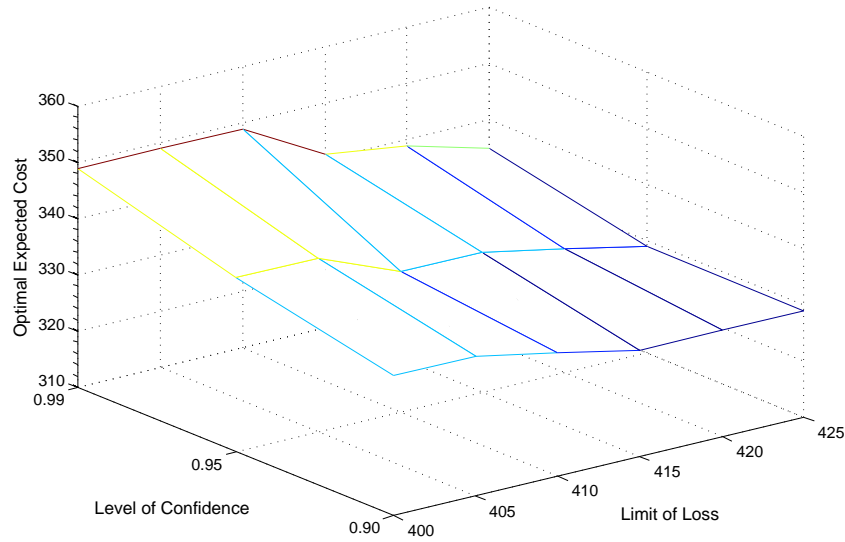


Figure 4.3: Computational results of reliable routing problem in the 10×70 network

Figure 4.3 shows the relation among the optimal expected cost, limit of loss and confidence level in the generated 10×70 graph. Given a confidence level, the route with less expected total cost can be selected when more resource, i.e. extended limit of loss, is available. Also, higher cost would be expected to investigate on the routes with higher reliability. The problem turns out to be infeasible when it is impossible to reach the required reliability with the available resource.

4.2 Stochastic Traveling Salesman Problem with Continuously Distributed Random Arc Costs

This section considers the risk-constrained stochastic traveling salesman problem with random arc costs. In the context of stochastic arc costs, the deterministic traveling salesman problem's optimal solutions would be ineffective because the selected route might be exposed to a greater risk where the actual cost can exceed the resource limit in extreme scenarios. We present a stochastic model of the traveling salesman problem that incorporates risk management. Value at Risk and

Conditional Value at Risk are respectively applied to measure and control the risk experiencing overly costly scenarios. We propose a novel cutting plane method to find the minimum-cost route in the stochastic environment while the risk level of the route is controlled by bounding the risk measures. Computational experiments are conducted to demonstrate the properties of the proposed models and the performance of the proposed cutting plan algorithm.

4.2.1 Introduction

Traveling Salesman Problem (TSP) has been extensively studied in operations research because of its close connection to other optimization problems and wide practical applications. Given a list of cities, the objective of TSP is to find the minimum-cost Hamiltonian cycle of visiting each city exactly once. The history and early works on the TSP can be seen in [33, 51, 77, 119], while recent reviews are found in [7, 94, 108].

While a number of TSP's parameters such as node and arc failures have been modeled as random variables to address various uncertainties in real-world problems, we consider travel cost as the uncertain element in this paper. In the stochastic context, the random cost of each arc is assumed to follow a given/konwn probability distribution. The stochastic TSP, seeking the minimal expected total cost, is essentially a deterministic TSP which uses the expected cost for each arc because of the homogeneity and additivity of the expectation function. However, the expectation only tells one aspect of the total travel cost. For any two routes with the same expected total cost, the variabilities of travel costs can be very different. For example, it is better to take a road with a lower speed limit but few traffic jam other than a road that has a higher speed limit but is frequently subject to constructions and accidents, given that the two road has the same expected travel time. In the presence of stochastic travel costs with such different variabilities, simply minimizing the expected total travel cost or time would be inadequate in these cases. When the costs of one or more arcs

have very large variances, it would increase the risk that the total cost exceeds the acceptable limit. A solution with a slightly higher expected cost but a much lower risk would be preferred other than the one with a low cost but much higher risk of being overcostly.

In this section, we study a risk-constrained stochastic TSP model to help find the route with the least expected total cost while the risk of exceeding the budget is controlled at a certain level. We assume that a directed graph is available and the travel cost/time incurred to move from one node to another is an independent random variable following a distribution with fixed parameters, i.e., distinct mean and variance. All the other parameters in the graph are considered deterministic. In addition, we assume that the route is determined before the random travel costs are known with certainty and no re-route optimization is allowed due to time constraints or applicability; the *a priori* optimal route must be followed once selected. To be general and as the first step, from the various continuous distribution families (e.g., gamma, exponential, student's t, etc.), we choose to use normal distribution for the random arc cost because it has been the most widely applied of all distributions and developed as a standard of reference for many statistic problems. Moreover, we will discuss how our models and solutions algorithm are actually general and can be used to treat cases with other probability distributions.

For controlling the risk associated with the route, one way is to control the fluctuations of its total travel cost by modeling a variance-constrained TSP. However, the main disadvantage of this approach is that it cannot provide quantitative details on risk evaluations such as the risky value and the risky probability (or confidence level), etc., to perform the risk management. For example, the model cannot quantify how bad the cost or loss could be in a planned route. As the cost of each arc is independently and normally distributed, we will demonstrate in Section 4.2.3 that the total travel cost is also a random variable with a normal density and the risk management in the stochastic TSP can be achieved by controlling the possible heavy tail in the total cost distribution. With the ability of quantifying worst-cost scenarios for the outcomes of a random variable, value at risk (VaR) and

conditional value at risk (CVaR) can be applied as another risk measures for the stochastic TSP. Both VaR and CVaR are commonly used in many engineering areas involving uncertainties, such as military, airspace, and finance [117]. Given a confidence level $\alpha \in (0, 1)$, the VaR at confidence level α is given by the smallest number l such that the probability that the loss exceeds l is at most $(1 - \alpha)$. By calculating the expectation/average of the outcomes that exceed VaR, CVaR provides a more conservative measure to manage the risk of having extreme loss or being overcostly.

We consider our main contributions as follows. We introduce a risk-constrained stochastic TSP with independent arc costs normally distributed, and address its advantages in real-world applications over deterministic TSP. We model the risk-constrained stochastic TSP to control the risk of planned routes by using explicit risk measures, i.e., VaR or CVaR. In addition, we study cutting plane methods for exact solution algorithms to solve the risk-constrained stochastic TSP, and compare its computational performance with other methods including linear reformulation and general branch and cut algorithms. In addition to the application in the TSP, the algorithm could be further generalized to benefit other stochastic network optimization problems in which some information regarding the arcs/links can be modeled as normally distributed random variables.

As follows is the structure of this section. In Section 4.2.2 we review related works. Section 4.2.3 models the mathematical formulation of the STSP with a risk constraint included to bound the VaR or CVaR. Various exact algorithms are derived in Section 4.2.4 for solving the problem. In Section 4.2.5, we introduce our numerical experiment setup and report computational results.

4.2.2 Literature Review

Researchers have studied the TSP under various uncertain parameters. [65] considers uncertain demand from the nodes, and allow the agent to skip nodes with no demand, and aims to find a priori tour with the minimal expected travel cost. [76] formulate this a priori TSP with uncertain

demands as a two-stage stochastic program and solve it using a branch-and-cut approach.

The TSP with random arc cost have also been widely researched. [81] and [18] consider the distances between the nodes nondeterministic. [70] apply the random link TSP with large node size in statistical physics where each arc is weighted as a Boltzmann factor. [68] model vehicle routing problems in which both arc travel times and node service times are stochastic. A branch-and-cut scheme embedded in a Monte Carlo sampling-based procedure is proposed as their solution method. A dynamic TSP with stochastic arc costs are proposed by [136] to allow the salesman to observe outgoing arc realizations at each city before deciding what place to visit next. A probabilistic TSP with deadlines is studied in [28] and time constraints in the context of stochastic customer presence is addressed.

Related studies to our work can be seen in [66], [134], and [32], which also considers a stochastic TSP with independent and normally distributed arc times. Their objective is to maximize the probability of completing the tour by a deadline, which would be inappropriate when it is unnecessary to achieve the lowest risk. We address in the following sections that the expected total cost could be largely increased as the probability increases a little. By using the risk management tool, VaR or CVaR, we can obtain the minimum expected cost route with the risk controlled at a desired level. Another related work is in [77] which considers vehicle routing problems with stochastic service and travel times. A chance constrained model is proposed to minimize the routing costs while ensuring that the probability of the total cost exceeding a given number is at most equal to a confidence level. A general branch and cut algorithm is described to solve the problem. Specifically, if a violation is detected at a candidate integer solution, i.e. the total cost exceeding budget, route or subtour elimination constraints are added to exclude the infeasible solution. Although the method can solve moderate problems to optimality, we demonstrate in the following sections that it would suffer from low efficiency of eliminating infeasible routes in large networks.

The pros and cons of VaR and CVaR for risk management are discussed by [117] regarding stability, simplicity, or specific problems, etc. [45] present an overview of VaR and its usage in finance and optimization. [114] optimize VaR and CVaR simultaneously. [120] present a two-stage stochastic integer program using CVaR. In this paper we model VaR or CVaR constraints to estimate and control the risk associated with the optimal route planned for the traveling salesman. Our model can reduce to deterministic TSP when the risk management constraint is relaxed.

4.2.3 Problem Description

Given a network $G = (N, A)$, where N denotes the set of nodes indexed by i or $j = 1, 2, \dots, n$ and A represents the set of arcs. In this paper we aim to study the risk-constrained TSP where the cost of each arc (i, j) is independently and normally distributed and denoted by C_{ij} . The objective is to minimize the expected total cost as follows,

$$\mathbb{E}(Z_{\mathbf{x}}) = \mathbb{E} \left(\sum_{(i,j) \in A} C_{ij} x_{ij} \right) = \sum_{(i,j) \in A} \mathbb{E}(C_{ij}) x_{ij} = \sum_{(i,j) \in A} e_{ij} x_{ij} \quad (4.15)$$

where $Z_{\mathbf{x}}$ is a random variable representing the random total cost given routing decisions $\mathbf{x} = [x_{ij}, (i, j) \in A]^T$; C_{ij} is the random variable normally distributed with mean e_{ij} and variance v_{ij} , \mathbb{E} denotes the expectation function of a random variable. Without the requirement of risk controlling, therefore, the stochastic TSP minimizing the expected total travel cost/time can be modeled by its deterministic counterpart as follows,

$$\text{[DTSP]:} \quad \min \sum_{(i,j) \in A} e_{ij} x_{ij} \quad (4.16a)$$

$$\text{s.t.} \quad \sum_{j:(i,j) \in A} x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (4.16b)$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (4.16c)$$

$$\sum_{i \in S, j \in S} x_{ij} \leq |S| - 1, \quad S \subsetneq N, |S| > 1 \quad (4.16d)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (4.16e)$$

where all parameters except the arc costs are assumed to be deterministic. Constraints (4.16b) and (4.16c) ensure that each node is visited exactly once and constraint (4.16d) is used to eliminate isolated subtours. Constraints (4.16e) require routing variables x_{ij} 's to be binary (with 1 meaning chosen and 0 otherwise).

Note that a feasible route for the TSP would include a set of arcs to form a Hamiltonian cycle. As described above, the costs of the arcs in the network are random variables which have normal probability densities with distinct means and variances. The probability distribution of the random total cost, $Z_{\mathbf{x}}$, can be obtained from the summation of costs of all the arcs included in the route. For independent normally distributed arc costs, we have the following remark regarding $Z_{\mathbf{x}}$.

Remark 4.1. *In the stochastic TSP with arc costs independently and normally distributed, the total travel cost has a normal probability density with the mean and variance respectively equal to the summation of the expected value and variance of the costs incurred in all included arcs, i.e., that*

$$Z_{\mathbf{x}} \sim \mathcal{N} \left(\sum_{(i,j) \in A} e_{ij} x_{ij}, \sum_{(i,j) \in A} v_{ij} x_{ij} \right)$$

Note that for normally distributed random variables, both VaR and CVaR are proportional to the standard deviation (see [114]). Given a TSP solution (a Hamiltonian cycle), we can express the the VaR and CVaR of its total travel cost in terms of means and variances on individual arcs as follows,

$$\text{VaR}_{\alpha}(Z_{\mathbf{x}}) = \min_l \{l | P(Z_{\mathbf{x}} \leq l) \geq \alpha\} = \sum_{(i,j) \in A} e_{ij} x_{ij} + k_1(\alpha) \left(\sum_{(i,j) \in A} v_{ij} x_{ij} \right)^{\frac{1}{2}} \quad (4.17)$$

$$\text{CVaR}_{\alpha}(Z_{\mathbf{x}}) = E[Z_{\mathbf{x}} | Z_{\mathbf{x}} \geq \text{VaR}_{\alpha}(Z_{\mathbf{x}})] = \sum_{(i,j) \in A} e_{ij} x_{ij} + k_2(\alpha) \left(\sum_{(i,j) \in A} v_{ij} x_{ij} \right)^{\frac{1}{2}} \quad (4.18)$$

where $\text{VaR}_\alpha(Z_x)$ denotes the α -level VaR, i.e., the α -quantile of the random variable Z_x ; $\text{CVaR}_\alpha(Z_x)$, i.e., the α -level CVaR, equals the average of worst-case costs that exceeds the α -level VaR. $k_1(\alpha)$ and $k_2(\alpha)$ are constants and their values are determined by following two equations,

$$\begin{aligned} k_1(\alpha) &= \sqrt{2}\text{erf}^{-1}(2\alpha - 1) \\ k_2(\alpha) &= (\sqrt{2\pi}\text{exp}(\text{erf}^{-1}(2\alpha - 1))^2(1 - \alpha))^{-1} \end{aligned}$$

where $\text{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-\tau^2} d\tau$ is the error function based on normal distribution, and $\text{erf}^{-1}(t)$ is its corresponding inverse function.

In addition to the [DTSP], we would model VaR or CVaR constraints for risk-constrained stochastic TSP to fend off the risk of having extreme loss or being overcostly. Specifically, the VaR constraint, $\text{VaR}_\alpha(Z_x) \leq L$, is modeled so that the probability of the total cost exceeding a certain cost threshold, L , is below the confidence level α ; the CVaR constraint, $\text{CVaR}_\alpha(Z_x) \leq L$, is modeled to control the α -level CVaR of total cost below the threshold L .

A nonlinear programming model, [STSP-NLP], is then formed to find the minimum-expected-cost route with the risk of loss being managed by the either VaR or CVaR constraint.

$$\begin{aligned} \text{[STSP-NLP]} \quad \min \quad & \sum_{i=1}^n \sum_{j=1}^n e_{ij}x_{ij} & (4.19a) \end{aligned}$$

$$\text{s.t.} \quad (4.16b) - (4.16e) \quad (4.19b)$$

$$\sum_{(i,j) \in A} e_{ij}x_{ij} + k(\alpha) \left(\sum_{(i,j) \in A} v_{ij}x_{ij} \right)^{\frac{1}{2}} \leq L \quad (4.19c)$$

The objective function (4.19a) in this formulation also seeks to minimize the expected total cost for the problem. The constraints (4.19b), which are equivalent to that in the [DTSP], are used to ensure that a Hamiltonian cycle is planned for the traveling salesman. The risk management

constraint (4.19c) is included where the parameter $k(\alpha)$ is equal to $k_1(\alpha)$ when the VaR technique is applied and $k_2(\alpha)$ for CVaR, respectively.

4.2.4 Algorithms for solving the STSP

We note that the difficulty of solving the proposed [STSP-NLP] mainly comes from the risk management constraint (4.19c), without which the problem would reduce to the deterministic TSP. In this manuscript we will propose and discuss various strategies to deal with this constraint in order to address the difficulty. First we consider a special case for the constraint, that is, the variances of arc costs are proportional to the means and propose Theorem 4.4 which indicates that in this case the problem becomes a mixed integer linear program (MILP).

Theorem 4.4. *Let $f(\mathbf{x}) = \sum_{(i,j) \in A} e_{ij}x_{ij} + k(\alpha) \left(\sum_{(i,j) \in A} v_{ij}x_{ij} \right)^{\frac{1}{2}} - L : \mathbf{x} \in \mathbb{R}$. Then $f(\mathbf{x}) = 0$ defines a hyperplane in the affine space of $\mathbf{x} \in \mathbb{R}$, if $\frac{v_{ij}}{e_{ij}} = h$, where h is a scalar constant.*

Proof. Substitute v_{ij} with he_{ij} and we have, $f(\mathbf{x}) = \sum_{(i,j) \in A} e_{ij}x_{ij} + k(\alpha)\sqrt{h} \left(\sum_{(i,j) \in A} e_{ij}x_{ij} \right)^{\frac{1}{2}} - L$. Solve $f(\mathbf{x}) = 0$ and we can obtain

$$\sum_{(i,j) \in A} e_{ij}x_{ij} = L + [k(\alpha)^2h - k(\alpha)\sqrt{4Lh + k(\alpha)^2h^2}]/2 \quad (4.20)$$

which defines a hyperplane. □

It can be observed that the hyperplane (4.20) defined by the risk management constraint is a contour of the objective function in our proposed risk-constrained stochastic TSP. Therefore, the optimal solution of [DTSP] would be optimal to the [STSP-NLP] if the solution is within the half space below the hyperplane, and otherwise the [STSP-NLP] would be infeasible.

When the special case is not applied, on the other hand, the feasible region of the [STSP-NLP] LP relaxation is then neither concave nor convex as indicated by Theorem 4.5. It is then unlikely to adopt the idea of approximating a convex set with a joint half-space combination, as used in the Kelley's cutting plane method [67], to solve the problem.

Theorem 4.5. $f(\mathbf{x}) = \sum_{(i,j) \in A} e_{ij}x_{ij} + k(\alpha) \left(\sum_{(i,j) \in A} v_{ij}x_{ij} \right)^{\frac{1}{2}} - L : \mathbf{x} \in \tilde{P}$ is a concave function, where \tilde{P} is the feasible region of [DTSP] LP relaxation.

Proof. Let $\mathbf{v} = [v_{ij}, (i, j) \in A]^T$. The Hessian matrix of the function $f(x)$ is,

$$H(\mathbf{x}) = -\frac{k(\alpha)}{4} \left(\sum_{(i,j) \in A} v_{ij}x_{ij} \right)^{-\frac{3}{2}} \mathbf{v}\mathbf{v}^T$$

the following observation is straightforward:

$$\mathbf{x}^T H(\mathbf{x}) \mathbf{x} < 0, \quad \forall \mathbf{x} \in \tilde{P}$$

which indicates that $H(\mathbf{x})$ is negative definite and thus $f(\mathbf{x})$ is concave. □

In the rest of this paper we will consider that the variances of arc costs are not proportional to the means and propose and compare various strategies to address the difficulty of the proposed problem.

4.2.4.1 A Linearization Approach

The first method we present here is to develop and solve a mixed integer linear programming (MILP) model by linearizing the risk management constraint in the [STSP-NLP]. Since we only have binary variables, we can take advantage of properties of bilinear and quadratic functions of

binary variables. First, we move the expectation on the left of nonlinear constraint (4.19c) to the right. Second, we square both sides as they are all nonnegative. These two steps will create bilinear and quadratic terms, i.e., $x_{ij}x_{i'j'}$ and x_{ij}^2 respectively. The bilinear term can be linearized by introducing additional binary variables $y_{ij,i'j'}$ while enforcing constraints (4.21d)–(4.21e). The quadratic term is equivalent to the binary variable itself. Hence, we can transform the [STSP-NLP] to an equivalent mixed integer linear program, [STSP-MILP], formulated as follows,

$$\text{[STSP-MILP]} \quad \min \sum_{i=1}^n \sum_{j=1}^n e_{ij}x_{ij} \quad (4.21a)$$

$$\text{s.t.} \quad (4.16b) - (4.16e) \quad (4.21b)$$

$$\sum_{(i,j) \in A} a_{ij}x_{ij} - 2 \sum_{(i,j) \in A} \sum_{(i',j') \in A'} e_{ij}e_{i'j'}y_{ij,i'j'} \leq L^2 \quad (4.21c)$$

$$x_{ij} - y_{ij,i'j'} \geq 0 \quad \forall (i,j) \in A, (i',j') \in A' \quad (4.21d)$$

$$x_{i'j'} - y_{ij,i'j'} \geq 0 \quad \forall (i,j) \in A, (i',j') \in A' \quad (4.21e)$$

$$\sum_{(i,j) \in A} e_{ij}x_{ij} \leq L \quad (4.21f)$$

$$y_{ij,i'j'} \geq 0 \quad \forall (i,j) \in A, (i',j') \in A' \quad (4.21g)$$

where $a_{ij} = k^2(\alpha)v_{ij} + 2Le_{ij} - e_{ij}^2$ is a constant determined by the arc (i,j) , cost threshold or budget limit L and the confidence level α , and $A' = A \setminus \{(i,j)\}$ denotes a set of arcs so that $A' \cup \{(i,j)\} = A$ and $A' \cap \{(i,j)\} = \emptyset$. The risk management constraint (4.19c) in the [STSP-NLP] is linearized and replaced by the constraints (4.21c)-(4.21g) with a new set of continuous variables y included. Note that we add constraint (4.21f) to ensure the right side (when we take the square) is a nonnegative value. This is in accordance with the original model, because any solution with $\sum_{(i,j) \in A} e_{ij}x_{ij} > L$ would violate the risk constraint (4.19c).

It can be easily observed that in the [STSP-MILP], the total number of constraints (4.21d) and (4.21e) equals $|A| * (|A| - 1)$ and would largely increase with the number of arcs. For small-sized networks, many off-the-shelf commercial MILP optimization solvers (e.g., CPLEX) can efficiently

solve the model to optimality. When the problem size keeps growing, however, a huge number of constraints would be generated, and it would take an unacceptable computational cost even just to obtain a feasible route.

4.2.4.2 The Branch and Cut Algorithm

A general branch and cut algorithm is introduced in [77] to solve chance constraint vehicle routing problems with stochastic travel times. The problems are firstly solved with the chance constraint relaxed. Integer solutions are evaluated, and marked as illegal routes while the relaxed constraint is violated. A cut is then added to the problem to eliminate the illegal solution. Here we utilize the same idea to iteratively generate route elimination constraints by separating TSP and risk evaluation. Specifically, the [STSP-NLP] is first relaxed by removing the risk management constraint (4.19c) to form the relaxed master problem [RMP] as follows,

$$\mathbf{[RMP]} \quad \min \sum_{(i,j) \in A} e_{ij} x_{ij} \quad (4.22a)$$

$$\text{s.t.} \quad (4.16b) - (4.16e) \quad (4.22b)$$

$$\sum_{(i,j) \in A} b_{ij} x_{ij} \leq b_0 \quad (4.22c)$$

The [RMP] is almost identical to the deterministic TSP [DTSP] except the additional route elimination constraints (4.22c), which can be solved by the off-the-shelf commercial solvers. Given the solution from a [RMP], $\hat{\mathbf{x}} = \{\hat{x}_{ij} | (i, j) \in A\}$, the risk evaluation is performed with the route defined by the solution. Note that when the TSP solution or a Hamiltonian route is known, the VaR and CVaR of the random total cost $Z_{\mathbf{x}}$ can be easily obtained by (4.17) and (4.18), respectively. When the VaR or CVaR exceeds the cost threshold, the route is marked to be infeasible and we

generate the following combinatorial cut for the purpose of route elimination:

$$\sum_{(i,j) \in \bar{A}} x_{ij} \leq |N| - 1 \quad (4.23)$$

where \bar{A} represents the set of arcs included in the route and $|N|$ is the number of nodes in the input graph. The validity of the cut (4.23) is straightforward. Specifically, $|N|$ arcs would be included in any TSP route to form a Hamiltonian cycle; when the route is infeasible according to the risk management constraint, the arcs in the set \bar{A} are not allowed to be simultaneously included in one Hamiltonian route in the presence of (4.23); any other solutions with at least one of x_{ij} 's ($(i, j) \in \bar{A}$) equal to 0, would not be eliminated by the combinatorial cut.

A branch and cut algorithm can then be applied to solve the STSP. Specifically, in the branch-and-cut scheme of solving the [RMP], every candidate solution in the search tree is evaluated according to the VaR or CVaR requirement. The feasible one with the objective value better than that of the current incumbent is considered as the new incumbent and produces a new upper bound (UB). Otherwise, the combinatorial cut (4.23) is added should the VaR or CVaR constraint be violated, giving the problem a new LB. The convergence of the LB and UB would then lead to the optimal solution. The algorithm is summarized as follows,

Algorithm 4 Branch and Cut algorithm.

- 1: Set $UB = +\infty, LB = -\infty$ and the optimal route $\mathbf{x}^* = null$
 - 2: **while** $(UB - LB > 0)$ **do**
 - 3: Solve the [RMP], and get a candidate integer solution $\hat{\mathbf{x}} = \{\hat{x}_{ij} : (i, j) \in A\}$ and its objective function value \hat{z} .
 - 4: Compute VaR or CVaR by the (4.17) or (4.18), respectively.
 - 5: **if** $VaR_\alpha(Z_{\hat{\mathbf{x}}}) \leq L$ or $CVaR_\alpha(Z_{\hat{\mathbf{x}}}) \leq L$ **then**
 - 6: Accept $(\hat{\mathbf{x}}, \hat{z})$ as the new incumbent and set $UB \leftarrow \hat{z}, \mathbf{x}^* \leftarrow \hat{\mathbf{x}}$.
 - 7: **else**
 - 8: Reject the incumbent and add the combinatorial cut (4.23) to the model.
 - 9: Update LB by solving the updated [RMP].
 - 10: **end if**
 - 11: **end while**
-

This branch and cut algorithm could be expected to solve small-sized and some moderately-sized network problems to optimality by iteratively removing the infeasible nodes in the branching tree. While in large-sized networks, there could be a huge number of infeasible routes needed to be assessed and eliminated before reaching the optimal route with desired reliability, which renders the method very low efficiency.

4.2.4.3 A Cutting Plane Algorithm

In this paper we mainly propose a cutting plane (CP) algorithm to more efficiently remove those “risky” routes that violate the risk management constraint by taking advantage of the structure of the constraint. Given an arbitrary lower bound of the original problem which is denoted as \bar{z} , a valid cut can be generated as described in the following proposition.

Proposition 4.1.

$$\sum_{(i,j) \in A} v_{ij} x_{ij} \leq [(L - \bar{z})/k(\alpha)]^2 \quad (4.24)$$

is a valid cut to the STSP.

Proof. Denote the optimal objective value of STSP as z^* and apparently we have $z^* = \sum_{(i,j) \in A} e_{ij} x_{ij} \geq \bar{z}$. The cut (4.24) is more relaxed than the constraint (4.19c) and is then valid to be included in our problem. \square

It can be observed that the strength of cutting plane (4.24) increases as its right hand side decreases (i.e., the lower bound \bar{z} increases). The cut is the strongest when $\bar{z} = z^*$ but invalid when $\bar{z} > z^*$. A bisection method can therefore be considered to find the target right hand side so that the strongest and valid cut is achieved. Specifically, a window $[l, r]$ is first initialized with $l = [(L - \bar{z})/k(\alpha)]^2$ and $r = [(L - \hat{z})/k(\alpha)]^2$ to define the interval for the binary search, where \bar{z} is the optimal cost and \hat{z} is the highest cost of the deterministic TSP, respectively. Assume that the [DTSP] feasible region is close and bounded, then both \bar{z} and \hat{z} are finite numbers. The cut with the right hand side equal to the middle value within the window is then evaluated, that is, $\sum_{(i,j) \in A} v_{ij} x_{ij} \leq (l + r)/2$, by solving the [RMP] with this cut being added. If a feasible solution is obtained from the [RMP] and satisfies the risk management constraint, then the search continues on the left half of the window and the solution provides an UB for the STSP; The right half of the window is also eliminated when the [RMP] is infeasible; The search continues on the right half and a LB is obtained, on the other hand, should the solution obtained from [RMP] violates the VaR or CVaR requirement. This process is repeatedly performed until $UB - LB < \varepsilon$.

While the convergence of UB and LB is expectable from the bisection method by iteratively eliminating half interval of the searching window, the performance of the method is limited because a large number of iterations would be required when the window is considerably large. To address

this limitation, we develop a CP method to iteratively add a route elimination constraint to (4.22c), which continuously decreases the size of the [RMP] feasible region until its optimal solution satisfies the VaR or CVaR requirement, and obtain the following proposition.

Proposition 4.2. *The STSP can be solved within a finite number of iterations by iteratively solving the [RMP] and adding the cut (4.24), with \bar{z} equal to the optimal objective function value of [RMP], as a route elimination constraint in (4.22c).*

Proof. Let \bar{x}^ν denote the optimal solution of the [RMP] in the ν^{th} iteration and \bar{z}^ν be the corresponding objective function value, providing an LB for the STSP. Apparently the solution is also optimal to the STSP should it satisfy the constraint (4.19c). Otherwise, the [RMP] is updated with the cut $\sum_{(i,j) \in A} v_{ij} x_{ij} \leq [(L - \bar{z}^\nu)/k(\alpha)]^2$ being added, which would cut off the solution \bar{x}^ν as well as other points that violate the cut from the [RMP] feasible region. Because of the closeness and boundedness of the [RMP] feasible region, a finite number of cuts need to be generated to continuously reduce the size of the feasible region until the optimal solution to the STSP is achieved. □

As the route elimination constraint (4.24) is invalid if \bar{z} is replaced with an upper bound, we are unable to incorporate the cut in the branch and cut scheme. Instead of evaluating the candidate integer solutions in the branch and cut algorithm, the CP algorithm iteratively measure the risk of [RMP] optimal solutions by computing their VaRs or CVaRs. The solution is optimal to the STSP should the VaR or CVaR requirement be satisfied; otherwise, the cut (4.24) is generated and added to the [RMP] to be solved in the next iteration. The method is summarized as follows,

Algorithm 5 Cutting Plane Method

- 1: **while** (1) **do**
 - 2: Solve the [RMP], and get the optimal solution $\bar{x} = \{\bar{x}_{ij} : (i, j) \in A\}$ and its objective function value \bar{z} .
 - 3: Compute VaR or CVaR by the equations (4.17) or (4.18), respectively
 - 4: **if** $\text{VaR}_\alpha(Z_{\bar{x}}) \leq L$ or $\text{CVaR}_\alpha(Z_{\bar{x}}) \leq L$ **then**
 - 5: The current solution is optimal and break the loop.
 - 6: **else**
 - 7: Generate the cut (4.24) and add it to the [RMP].
 - 8: **end if**
 - 9: **end while**
-

We note that the cut (4.24) used in the CP method has a higher dimension ($\dim = |A| - 1$) than the combinatorial cut (4.23) ($\dim = |N| - 1$). In addition, it eliminates from solution space not only the current [RMP] optimal solution with unsatisfying risk level but also those with the cost variance exceeding its right hand side value. The strength of the proposed cutting plane is therefore expected to be improved compared to the combinatorial cut. Let $\text{conv}(P)$ represent the convex hull of the [RMP] feasible region and \bar{x} be the optimal solution violating the risk management constraint with the objective function value \bar{z} . Then the developed cut is added as illustrated in Figure 4.4. A larger lower bound \bar{z}' can be obtained in the next iteration, resulting in a stronger cut ($10'$) as shown in the figure. We repeatedly solve the [RMP] and generate the cuts until achieving the optimal route x^* which satisfies the VaR or CVaR requirement.

It can be observed that the CP algorithm seeks to find the optimal route with the risk well managed by controlling the variance of the random total travel cost. However, the method differs from modeling a variance-constrained TSP, that is, simply including a variance constraint (i.e., $\text{Var}(Z_x) \leq v^*$) to the deterministic TSP, in both measuring and controlling the risk. Specifically,

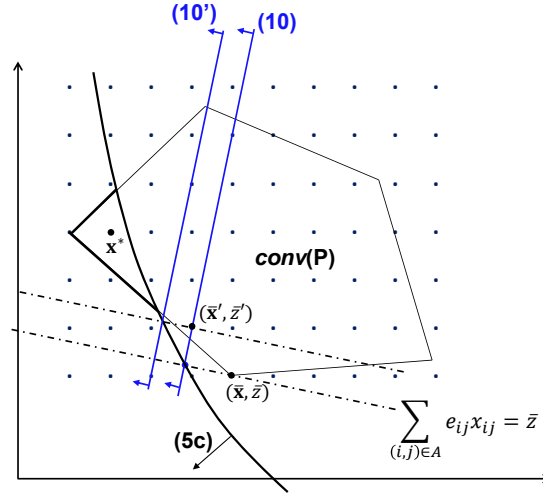


Figure 4.4: Developed cutting plane.

Note: $conv(P)$ is the convex hull of the feasible region of [RMP]; (4.19c) represents the concave set defined by the risk management constraint. \bar{x} is the optimal solution of the current [RMP] and \bar{z} is the corresponding cost; The cut (4.24) is generated and added to the [RMP], which is then solved to obtain a new optimal solution \bar{x}' with a larger cost \bar{z}' , leading to a stronger cut (10').

the variance-constrained TSP measures the risk by evaluating the cost fluctuations or how much the cost deviates from its expected outcome, but what decision makers are really concerned would be how bad the worst-cost scenarios could be, which on the other hand can be quantified by the VaR or CVaR technique in the CP algorithm. While in both methods the risk associated with the route is controlled by bounding the variance of the travel cost, the threshold, v^* , cannot be appropriately determined in the variance-constrained model according to the information on risk evaluations. The CP method, however, can well analyze the risk evaluation details, including the confidence level, the mean and variance of total cost of the route evaluated, and cost limit in each iteration, and repeatedly updates and reduces the threshold until the optimal route is eventually found with the α -VaR or α -CVaR of its total cost controlled below the cost limit.

Compared to the cut (4.24) controlling only the variance for the purpose of risk management, here we propose another cut to manage the risk by taking into account both the expected value and the

variance of total travel cost, as stated in the following proposition.

Proposition 4.3. *Let \bar{z} denote the optimal cost of the [RMP], then the cut*

$$\sum_{(i,j) \in A} [e_{ij} + k(\alpha)^2 v_{ij} / (L - \bar{z})] x_{ij} \leq L \quad (4.25)$$

is valid and stronger than the cut (4.24).

Proof. The risk management constraint (4.19c) can be reformulated to

$$\sum_{(i,j) \in A} e_{ij} x_{ij} + k(\alpha) \left(\sum_{(i,j) \in A} v_{ij} x_{ij} \right) \left(\sum_{(i,j) \in A} v_{ij} x_{ij} \right)^{-\frac{1}{2}} \leq L$$

According to (4.24), the validity of the inequality (4.25) can be confirmed by substituting the second $\sum_{(i,j) \in A} v_{ij} x_{ij}$ above with $[(L - \bar{z})/k(\alpha)]^2$. Assume that an arbitrary point $\tilde{\mathbf{x}} \in \text{conv}(P)$ is removed from solution space by the cut (4.24), which indicates that $\sum_{(i,j) \in A} v_{ij} \tilde{x}_{ij} > [(L - \bar{z})/k(\alpha)]^2$. It is straightforward that $\sum_{(i,j) \in A} e_{ij} \tilde{x}_{ij} \geq \bar{z}$ since \bar{z} is the optimal cost within $\text{conv}(P)$. Therefore the solution $\tilde{\mathbf{x}}$ also violates the inequality (4.25), proving that the cut (4.25) is stronger than (4.24). \square

Although a stronger route elimination constraint can be generated from the cut (4.25), the updated [RMP] still provides a LB for the STSP because of the validity of the constraint to the proposed problem. Therefore Proposition 4.2 and Algorithm 5 are both valid when (4.25) is used in place of the weaker cut (4.24), while the better performance of the CP method can be expected.

4.2.5 Numerical Experiments and Results

Numerical experiments and results of different methods are presented in this section on solving the STSP. The algorithms are coded in Microsoft Visual C++ linked with CPLEX 12.5. All the programs are run in Microsoft Windows 7 Professional operating system on a Dell Desktop with Intel Core i7-2600 CPU 3.40GHz and 8GB RAM.

4.2.5.1 Input data generation.

We consider different settings to precisely evaluate the performance of each algorithm. The STSP are modelled in a set of networks which are generated with different numbers of nodes and arcs. For each size of 10-,50-,100- and 200-nodes, we respectively produce the networks with low, medium and high densities. In order to do so, we randomly generate an integer value from three different value ranges, as shown in 4.2 and assign it as the outgoing degree to each node. To provide an initially connected graph, we manually generate a cycle route that uses all nodes in N .

Table 4.2: Input graphs.

Nodes	Density of outgoing degrees		
	Low	Medium	High
10	4-6	6-8	8-9
50	5-15	20-30	35-45
100	10-20	45-55	80-90
200	20-30	90-110	160-180

The mean and variance of arc costs in the graphs are produced to define their normal distributions. Specifically, the mean cost of each arc is randomly generated in the range of [30,40] and the variance in [1,1600]. The levels of confidence are set as $\alpha = 0.90, 0.95$ and 0.99 , respectively.

4.2.5.2 Computational results.

The computational results obtained by using different methods on the STSP are shown in Table 4.3, where computational times are recorded in seconds. We would note that the only difference between the STSP with VaR and CVaR requirements regards to the constant $k(\alpha)$, whose value would not significantly affect the performance of each algorithm. Therefore, in Table 4.3 we only report the results of solving the problem with the CVaR constraint included to compare the algorithms present in Section 4.2.4.

In the table, columns “Node” and “Density” describe the size of each input graph for the STSP and column “L” gives each problem the cost threshold which is arbitrarily determined. Column “Linearize” shows the results when we call CPLEX to solve the linearized model [STSP-MILP]. For the small-sized networks with 10 nodes and various densities, the number of arcs is at most 90 according to Table 4.2. We can solve the problems to optimality at accepted computational costs. When the network grows to 50 nodes or more, hundreds or thousands of arcs would be included and the number of constraints (4.21d) and (4.21e) would be enormous. As a consequence, the problem size would become too large to be handled by our computer.

Column “Branch and cut” shows the computational time of using the branch and cut algorithm as described in Section 4.2.4.2. No significant improvement is observed compared to calling CPLEX to solve the [STSP-MILP]. The algorithm can solve small-sized networks at low computational costs but is limited while the network size or the level of confidence increases. The reason is that, along the process of branch and bound to solve the relaxed problem, the generated combinatorial cut is not efficient enough to prune infeasible solutions in the search tree. Our computational results show that in most networks even a feasible solution cannot be found after exhaustive search (after 3600 seconds).

Table 4.3: Solution times, in CPU seconds, for solving the STSP with different algorithms.

Node	L	Density	α	Linearize	B&B&C	CP(I)		CP(II)	
						Itn	time(sec.)	Itn	time(sec.)
10	450	Low	0.9	2.762	0.203	2	0.318	2	0.312
			0.95	5.882	5.819	3	0.487	3	0.375
			0.99	5.569	5.959	2	1.344	2	0.281
		Medium	0.9	2.964	0.109	1	0.135	1	0.14
			0.95	5.912	0.14	2	0.281	2	0.296
			0.99	138.461	1178.2	3	0.398	2	0.296
		High	0.9	9.079	0.156	1	0.638	1	0.296
			0.95	631.244	-	5	1.255	3	0.608
			0.99	628.198	-	3	0.661	2	0.375
50	1800	Low	0.9	-	-	5	4.852	2	0.767
			0.95	-	-	3	3.083	2	0.768
			0.99	-	-	2	1.344	2	0.568
		Medium	0.9	-	4.555	2	4.375	2	4.496
			0.95	-	-	4	10.897	4	9.139
			0.99	-	-	11	31.56	6	17.463
		High	0.9	-	0.811	1	2.02	1	2.589
			0.95	-	-	2	4.162	2	4.94
			0.99	-	-	6	31.766	6	27.581
100	3300	Low	0.9	-	-	3	4.995	2	6.505
			0.95	-	-	2	6.061	2	7.285
			0.99	-	-	2	3.836	2	4.165
		Medium	0.9	-	-	7	186.582	7	102.539
			0.95	-	-	6	90.691	3	28.439
			0.99	-	-	4	48.683	2	10.826
		High	0.9	-	-	2	19.484	2	17.503
			0.95	-	-	6	164.509	3	64.631
			0.99	-	-	10	274.526	5	115.676
200	6500	Low	0.9	-	-	5	73.742	2	19.032
			0.95	-	-	4	49.948	2	15.803
			0.99	-	-	3	19.299	2	21.118
		Medium	0.9	-	-	2	42.369	2	71.979
			0.95	-	-	2	170.582	2	47.284
			0.99	-	-	6	563.774	4	170.889
		High	0.9	-	-	2	101.782	2	136.383
			0.95	-	-	2	103.124	2	97.61
			0.99	-	-	2	244.25	2	368.985

Note: The dash in the table indicates that no feasible solution is obtained within 3600 sec.

The performance of the cutting plane algorithm is shown in columns “CP(I)” and “CP(II)” in the table with the cuts (4.24) and (4.25) being respectively applied. The number of iterations as well as the computational time of the method are shown in the columns “Itn” and “time”. Compared to the

branch and cut algorithm that is expected to obtain the optimal solution by visiting the search tree only once, the cutting plane method would visit the tree for more times before finding the optimal solution. The generated cutting planes, however, is much more efficient than the combinatorial cut and can quickly remove infeasible solutions. Our computational results show that only a few iterations are needed to find the optimal solution and the problem in each generated network with up to 200 nodes can be solved at a reasonable cost. The better computational performance obtained from “CP(II)” for most instances confirms the higher strength of the cut (4.25).

The process of using the CP method is well demonstrated in Figure 4.5. The optimal objective values (OFV) of the [RMP] in each iteration is shown when the VaR and CVaR are respectively used for the risk management. The network with 50 nodes and medium density is applied and the level of confidence is set as 99%. The CVaR technique is known to be more conservative than VaR; this means that at the same confidence level, the higher total travel cost would be expected to select the route that satisfies the CVaR requirement than VaR. In addition, the more conservative method (i.e., CVaR) would expect more iterations in the CP algorithm before reaching optimality as shown in the figure. However, this is not the necessary scenario as the route elimination cuts (4.24) and (4.25) generated in each iteration of the algorithm is stronger when the CVaR is applied.

Figure 4.6 shows the change in the probability distributions of total travel cost when the VaR or CVaR risk management is applied to the STSP, respectively. It can be observed that the route solved from the deterministic TSP [DTSP] is highly exposed to the risk that the actual cost can exceed the cost limit. The risk is well managed as the stochastic model is applied while expected total cost is not significantly increased as shown in the figure. The probability that the total travel cost exceeds the cost limit is controlled below the confidence level α by the VaR requirement. When the CVaR is applied, the risky level is further reduced so that the average of the percentage, $1 - \alpha$, of worst cases is bounded by the cost limit. Along the process of running the CP algorithm, the change in the standard deviations (square root of variances) of total costs is represented by the

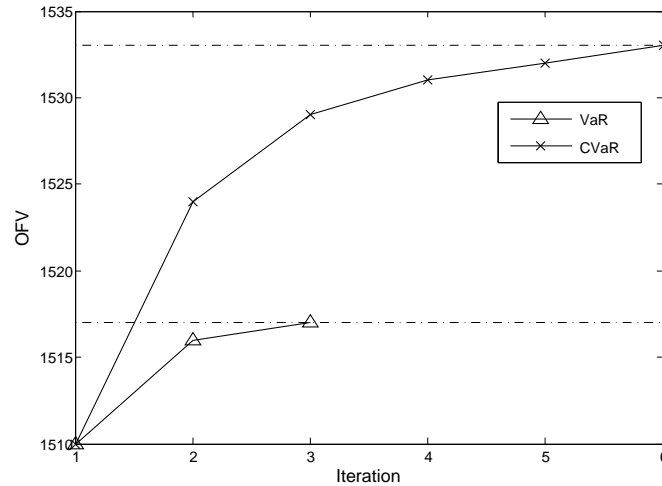


Figure 4.5: The change in objective function values of updated [RMP]. Input graph: 50 nodes with high density.

dash lines in Figure 4.6(right) while the VaR and CVaR techniques are respectively applied. Each solid line in the figure shows the decrease of VaR or CVaR achieved from the CP method until the cost limit (1800) is crossed.

Figure 4.7 shows the relation between the optimal expected cost and confidence level in the generated network with 200 nodes and medium density. Higher cost would be expected to investigate on the routes with higher reliability. The problem turns out to be infeasible when it is impossible to reach the required reliability with the available resource. The expected total cost is largely increased as the probability increases a little from 97% to 98%, which indicates that it would be inappropriate to maximize the probability of completing the tour by a deadline (or a cost limit) in our case.

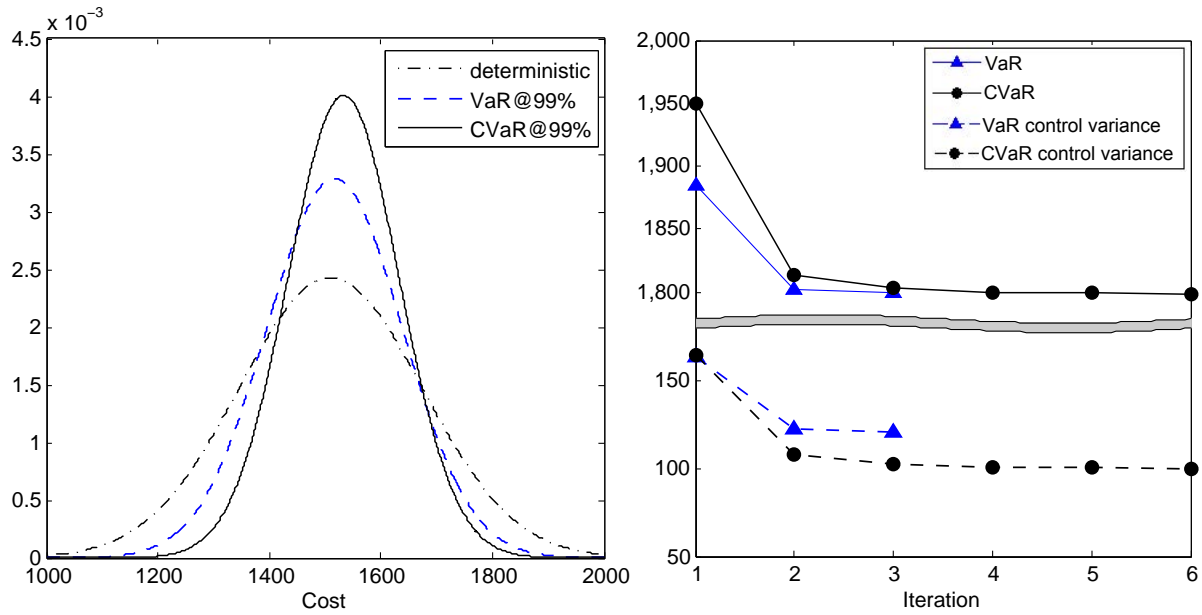


Figure 4.6: Probability distribution of total optimal cost as VaR or CVaR method is used (left); the process of VaR or CVaR controlling cost variances for the purpose of risk management in the CP algorithm (right). Note: The network with 50 nodes and high density is applied and the confidence level is 99%.

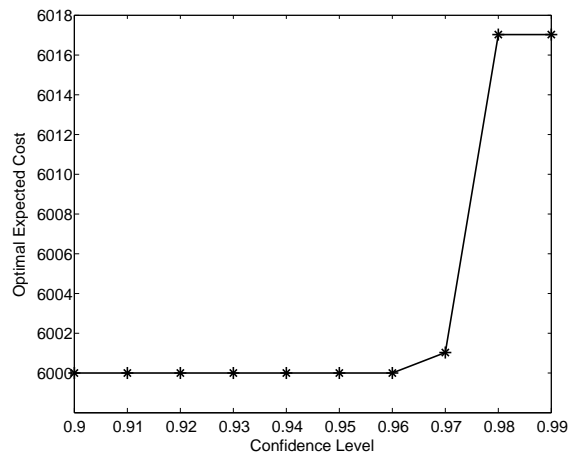


Figure 4.7: The change in the optimal expected costs as the confidence level increases. Note: The network with 200 nodes and medium density is applied.

4.2.6 Conclusions

In this paper, we proposed a STSP with the travel cost of each arc independently and normally distributed. A nonlinear programming model is formulated for the STSP to find the route with minimum expected cost while the risk is controlled at a confidence level. Both VaR and CVaR were respectively applied as the risk measures to evaluate the reliability of selected routes. A CP algorithm was developed to address the computational difficulty of the problem, and our computational results demonstrated that it can solve the STSPs in all the generated networks at reasonable computational costs and has overwhelming performance beyond the linearization method and the branch and cut algorithm.

4.3 Reliable Routing Problems under Uncertain Topology

Network routing problems are often modeled with the assumption that the network structure is deterministic, though it is often subject to uncertainty in many real-life scenarios. In contrast, we study the traveling salesman and the shortest path problems with uncertain topologies modeled by arc failures. We present the formulations that incorporate chance constraints to ensure reliability of the selected route considering all arc failure scenarios. Due to the computational complexity and large scales of these stochastic network optimization problems, we consider two cutting plane methods and a Benders decomposition algorithm to solve them. We also consider to solve the reformulations of the problems obtained by taking the logarithm transformation of the chance constraints. Numerical experiments are performed to obtain results for comparison among these proposed methods.

4.3.1 Introduction

In this paper, we consider two stochastic routing problems: the traveling salesman problem (TSP) and the shortest path problem (SPP), with deterministic arc costs but uncertain topologies. TSP and SPP have been extensively studied and used in a wide range of applications, while both of them could face the challenges from the uncertainty of arc existence. A road that links two cities might become impassable due to severe accidents or road constructions. Inclement weather might cause significant delays or cancellations of flights between airports. A wired connection might physically break between computers and hence cut off the communication. It is easy to name quite a few more similar cases involving uncertain topology or connectivity. In the presence of such uncertainty, the solutions of deterministic TSP/SPP could be un-implementable or lead to tremendous losses if they are implemented in practice. In this perspective, we propose chance-constrained stochastic TSP/SPP models and develop exact solution algorithms to find the optimal routes that are reliable enough when experiencing the uncertain topologies. In each chance-constrained model, we assume that the route are determined before the topology is known with certainty and after knowing the topology no re-routing is allowed due to time constraints or applicability.

Both TSP and SPP are among the most widely researched network optimization problems because of their close connections to other optimization problems and practical applications. In deterministic cases, the objective of each problem is to seek the least-cost route given a network with certain arc costs and topology. In the TSP, the route (also a cycle) should pass each and every node exactly once (cf. [72]), while the SPP, which solves the optimal path from the origin to destination, does not have such requirement (cf. [5]). The parameters of the networks in many real-world applications are usually uncertain. Many researchers have devoted efforts to handle the uncertainty issues from different perspectives. [19] try to find the optimal location of an agent's home in the stochastic TSP with nondeterministic arc travel costs. [65] considers uncertain demands from the

nodes, allows the agent to skip nodes with no demand, and aims to find an *a priori* a tour of all nodes with minimum expected travel cost. [74] formulate this *a priori* TSP with uncertain demands as a two-stage stochastic program and solve it using the integer L-shaped method ([75]). In addition to the stochastic TSP, researchers have also studied the SPP under various uncertain network parameters. [96] proposes a method to calculate the shortest expected distance when the arc travel times are independently distributed and the method can simultaneously compute the path reliability. [131] define path optimality with the probability that a path is shorter than any other path in the network. [11] use an approximation approach based on the combination of dynamic programming and heuristics to solve the SPP with random arc costs and additional budgetary constraints. [111] use a dynamic programming algorithm to solve the SPP in which the arc costs are accumulated while the graph is traversed. [48] study a formulation based on the adaptive feedback control for the SPP with correlated arc costs. [102] develop a label-correcting algorithm and a pseudo-polynomial approximation scheme for the SPP with on-time arrival probability.

In contrast to these works on uncertain arc costs or node demands, we study the two routing problems (i.e., TSP and SPP) with uncertain network topologies. Our models assume that any arc exists with a certain probability, and there are only two possible scenarios for a route connected by a set of arcs: success or failure. The nonexistence or failure of any single arc within the selected route *a priori* would result in the failure of the entire route due to the lack of recourse. Our objective is to measure the risk of failure and to find the lowest-cost route with an acceptable level of reliability, with the assumption that the Bernoulli trial of each arc's existence is independent of each other for convenience. Similar uncertain topology is studied by [25] and [152], while both of them investigate risk-constrained stochastic network flow problems and modeled them as linear programs in which the arc flows are continuous rather than binary. In particular, they both aim to find the minimum-cost flows that meet the risk requirement of losing flows as some arcs might be unavailable, where conservative upper bound loss and exact flow loss are respectively used.

However, in this paper, the flow loss is discrete (either 1 or 0), because the route is not applicable (the flow loss being 1) if there is any arc becoming unavailable while implementing the routing decision and no loss would be incurred otherwise. Another difference between this paper and the aforementioned two is that this paper considers complete information (i.e., $2^{|A|}$ scenarios) of the arc failure uncertainty while the aforementioned two use sampled scenarios (i.e., a subset of $\{0, 1\}^A$). In this paper, we design cutting plane algorithms and reformulations to handle the exponentially growing number of scenarios.

Our models are useful in situations where both the travel cost and the reliability are important to the decision makers. Consider the case that a city is struck by some natural disasters, as described in Chapter 1, where a more reliable route needs to be planned in advance to prevent the route failure with a desired level of confidence. In addition we would note that there is always a tradeoff between minimizing the total cost and maximizing the reliability. Solving our proposed models would benefit decision makers to find a good balance in between. By setting different confidence levels, the corresponding optimal costs can be obtained and therefore the tradeoff between the total cost and the risk level can be achieved. We will show in the following sections that the cost could be significantly saved by slightly decreasing the route reliability, and on the other hand, the risk could be largely reduced by selecting another route with only a little higher total cost.

While considering independent failures of all arcs, the number of scenarios of a reliable routing problem would exponentially increase as the number of arcs in the input network increases, and the problem of a large-sized or even moderate-sized network would be very difficult to be solved. [140] propose solving stochastic routing problems via the sample average approximation method, where random samples are used to generate estimates for the expected objective function. With limited scenarios generated, however, this sampling method would not expect to find the exact optimal solution 100% of the time. In this paper, we aim to develop exact solution algorithms for the routing problems under uncertain topologies.

We summarize the contributions of this paper as follows: (1) We introduce and model the reliable routing problems with arc failures and address its real-world applications; (2) We develop exact algorithms for solving the chance-constrained routing problems, e.g., stochastic TSP and SPP; (3) In order to avoid directly dealing with the exponentially growing number of scenarios, we take advantage of implicit scenario enumeration, scenario aggregation and disaggregation and design efficient cutting plane algorithms to solve the problems; (4) We reformulate the problems by taking the logarithm transformation of chance constraints and replacing them with a knapsack constraint; and (5) We apply the sensitivity analysis while perturbing the level of confidence to show the tradeoff between the total cost and the reliability.

Following is the structure of the section. In Section 4.3.2, we present mathematical formulations of the chance-constrained TSP and SPP with uncertain topologies. Exact algorithms are then developed in Section 4.3.3 to solve the proposed problems; their efficiencies are also discussed. In Section 4.3.4, we introduce our numerical experiment settings and report computational results. We compare the proposed algorithms in solving the reliable routing problems.

4.3.2 Problem Formulation

Given a network $G = (N, A)$, where N denotes the set of nodes and A represents the set of arcs, we assume that the nodes are indexed by $i = 1, 2, \dots, n$ for convenience. As we are focusing on routing problems in this paper, we use a binary variable, x_{ij} , to denote whether arc (i, j) is chosen as a part of the route (1: yes; 0: no). In this paper, we assume that every arc has only two states: normal operation or failure (with probability p_{ij}), therefore making it a Bernoulli trial. Since we assume that all arc failures are independent to each other, the whole sample space S for all possible topologies (structure of the problem's network) is $\{0, 1\}^A$. Each scenario is indexed by s and the occurring probability is denoted by Prob^s . In order to model this uncertainty, we

introduce a parameter Y_{ij}^s to denote whether the arc (i, j) fails in scenario $s \in S$ (1: yes; 0: no), and then $\text{Prob}^s = \prod_{(i,j) \in A} (p_{ij})^{Y_{ij}^s} (1 - p_{ij})^{1 - Y_{ij}^s}$. In the stochastic setting, we assume that the route will fail if any arc within the route fails. Hence we introduce another binary variable l^s to denote whether the route will fail in scenario s (1: yes; 0: no). Because both x_{ij} and Y_{ij}^s only take binary values, we can use the following constraint to define the route failure indicator,

$$x_{ij}Y_{ij}^s \leq l^s, \quad \forall (i, j) \in A, s \in S,$$

where the left hand side is true (being 1) if and only if the arc is not only selected in the route but also fails. As the left hand side only takes binary values, the binary requirement on l^s can be relaxed. Once we can determine the state of each possible route for any scenario, we can easily evaluate the risk indicating how likely the selected route will fail if the decision is implemented, by calculating $\sum_{s \in S} \text{Prob}^s l^s$. In particular, we will define our stochastic routing problems in the two following subsections. For convenience, sets, parameters and variables are summarized in Table 4.4.

Table 4.4: Summary of Parameters, Sets, and Variables.

Sets and Parameters	
N	The set of all nodes in the network, indexed by $i = 1, 2, \dots, n$
A	The set of all arcs in the network, indexed by (i, j)
S	The set of all possible scenarios of the network topology s , i.e., $\{0, 1\}^A$
Y_{ij}^s	The arc-status parameter which equals 1 if arc (i, j) fails in scenario s
p_{ij}	The parameter denoting the failure probability of arc from node i to node j
Prob^s	The probability that the scenario s will occur
α	The constant denoting the level of confidence that the path will be successful
c_{ij}	The cost of travelling from node i to node j
Decision Variables	
x_{ij}	The binary variable that equals 1 if travelling from node i to node j
u_j	The node potential variable that eliminates subtours
l^s	The binary variable that equals 1 if the selected route fails in scenario s

4.3.2.1 Chance-constrained TSP

The chance-constrained TSP (CCTSP) can be formulated as follows:

$$\text{[CCTSP]} \quad \min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.26a)$$

$$\text{s.t.} \quad \sum_{j:(i,j) \in A} x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (4.26b)$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (4.26c)$$

$$u_i - u_j + nx_{ij} \leq n - 1, \quad 1 < i \neq j \leq n \quad (4.26d)$$

$$x_{ij} = \{0, 1\}, \quad \forall (i, j) \in A \quad (4.26e)$$

$$u_j \geq 0, \quad \forall j (j > 1) \quad (4.26f)$$

$$x_{ij} Y_{ij}^s \leq l^s, \quad \forall (i, j) \in A, s \in S \quad (4.26g)$$

$$\sum_{s \in S} \text{Prob}^s l^s \leq 1 - \alpha \quad (4.26h)$$

The objective function (4.26a) in this formulation seeks to minimize the cost for the problem. The constraints (4.26b)-(4.26d) are identical to the standard TSP constraints, where (4.26b) and (4.26c) ensure that each node is visited exactly once and the Miller-Tucker-Zemlin constraint (4.26d) is used to exclude subtours (cf. [41]). While alternative models could be used for the subtour elimination, in this paper we mainly focus on the comparison of different cutting planes in dealing with the uncertainty in the problem.

Chance constraints are modelled in (4.26g) and (4.26h). Specifically, as mentioned previously, the left hand side of constraint (4.26h) calculates the probability that the route fails by summing up the occurrence probability of all failure scenarios. By restricting this probability less than or equal to $(1 - \alpha)$, we can obtain the optimal and reliable route, of which the failure probability is controlled at a desired level.

4.3.2.2 *Chance-constrained SPP*

Assuming node 1 and n as the origin and destination nodes respectively, the chance-constrained SPP (CCSPP) can be formulated as follows:

$$\text{[CCSPP]} \quad \min \quad z = \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.27a)$$

$$\text{s.t.} \quad \sum_{j: (1,j) \in A} x_{1j} = 1 \quad (4.27b)$$

$$\sum_{i:(i,n) \in A} x_{in} = 1 \quad (4.27c)$$

$$\sum_{j:(i,j) \in A} x_{ij} - \sum_{j:(j,i) \in A} x_{ji} = 0, \quad i = 2, 3, \dots, n-1 \quad (4.27d)$$

$$x_{ij} = \{0, 1\}, \quad \forall (i, j) \in A \quad (4.27e)$$

$$x_{ij} Y_{ij}^s \leq l^s, \quad \forall (i, j) \in A, s \in S \quad (4.27f)$$

$$\sum_{s \in S} \text{Prob}^s l^s \leq 1 - \alpha \quad (4.27g)$$

The objective function (4.27a) is identical to that of the CCTSP (4.26a). Constraints (4.27b)-(4.27d) comprise the SPP. The first two constraints ensure that only one arc on the path comes from the origin node (4.27b) and one arc goes to the destination node (4.27c). Next, constraint (4.27d) balances the arcs entering and leaving each node in the network that is neither the origin nor destination node. The final two constraints (4.27f)-(4.27g) form the chance constraints as explained previously.

4.3.2.3 Computational Challenges

As is mentioned in the introduction, we assume that the arc failures are independent to each other in both CCTSP and CCSPP. Let $|A|$ represent the number of arcs in the given network. The total number of scenarios is then equal to $2^{|A|}$ since there are two outcomes for each arc, “success” and “failure”. In the result, the number of constraints of type (4.26g) or (4.27f), which is equal to $|A| \cdot 2^{|A|}$, would be exponentially increased with the number of arcs. For example, the number of the constraints increases from 160 as $|A| = 5$, to 10,240 as $|A| = 10$, and to around 20 million as the network includes only 20 arcs. Most computer solvers will not be able to handle the problems even in a moderate-sized network. In this paper, we propose exact algorithms to solve the reliable routing problems at reasonable computational costs.

4.3.3 Exact methods for solving CCTSP and CCSPP

Note that the reliability or feasibility of a given route can be easily evaluated via computing its failure probability, as shown in Theorem 4.6. The validity of the theorem can be proved by the fact that the selected route would fail unless all the included arcs succeed, which has the probability equal to the product of their success probabilities. Let $\hat{\mathbf{x}}$ be a routing solution to the routing problems, and $\hat{A} = \{(i, j) \in A | \hat{x}_{ij} = 1\}$, and $L(\hat{\mathbf{x}})$ or $L_{\hat{A}}$ be the Bernoulli random variable of the route failure, with 1 being a failure and 0 being a success.

Theorem 4.6. *The failure probability of a given route is*

$$\text{Prob}(L_{\hat{A}} = 1) = 1 - \prod_{(i,j) \in \hat{A}} (1 - p_{ij}).$$

The route is unreliable or infeasible if the probability greater than $1 - \alpha$.

Dealing directly with the exponentially growing number of chance constraints (4.26g-4.26h) and (4.27f – 4.27g) will be computationally intractable. In this paper, we consider to replace these direct chance constraints with feasibility cutting planes (i.e., (4.28c), (4.29c)) which are not as many but still can remove all the infeasible routes and we formulate the master problems (MP) for both the proposed problems as follows:

$$\text{[MP-CCTSP]} \quad \min \quad z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (4.28a)$$

$$\text{s.t.} \quad (4.26b) - (4.26f) \quad (4.28b)$$

$$\sum_{(i,j) \in A} \eta_{ij}^{\xi} x_{ij} \leq \beta^{\xi} \quad \xi = 1, 2, \dots, \Xi \quad (4.28c)$$

$$\begin{aligned}
\text{[MP-CCSPP]} \quad \min \quad & z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} & (4.29a) \\
\text{s.t.} \quad & (4.27b) - (4.27e) & (4.29b) \\
& \sum_{(i,j) \in A} \mu_{ij}^{\omega} x_{ij} \leq \nu^{\omega} \quad \omega = 1, 2, \dots, \Omega & (4.29c)
\end{aligned}$$

The constraints (4.28b) and (4.29b) are identical to those in deterministic routing problems. We include a set of constraints (4.28c) and (4.29c), where η_{ij}^{ξ} and μ_{ij}^{ω} are the coefficients of corresponding variables, and β^{ξ} and ν^{ω} are the right hand side values, respectively. Different algorithms will be applied to generate these constraints cutting away all the infeasible or risky routes which are however feasible for the deterministic TSP or SPP. Algorithms' performance will be compared and discussed in a later section.

Building the constraints (4.28c) and (4.29c), however, is challenging because 1) the number of infeasible routes may be too huge to be enumerated, and 2) it is hard to find the similarity among these infeasible routes to generate a few of cuts that can remove all the routes. However, actually it is not needed to initially include all of these constraints in the problems. With them being removed, we can form relaxed master problems (RMP) for both CCTSP and CCSPP. The initial RMP, which is identical to the deterministic routing problems, are smaller and easier to be solved. We then iteratively implement the callbacks and add each of the removed "lazy constraints" as soon as it becomes violated until the convergence between the upper bound and the lower bound of the problem is achieved. In this paper, we propose the following methods to construct and add the cutting planes in different ways.

4.3.3.1 Combinatorial Cutting Plane

The first method presented here is a combinatorial cutting plane (CCP) method. While the mixed integer linear programming (MILP) RMP is being solved with the branch-and-bound-and-cut algorithm, the feasibility of any incumbent candidate (i.e., a route) can be evaluated according to Theorem 4.6. A combinatorial cut will be added to the model should the route be exposed to an unacceptable risk (i.e., a loss probability higher than $1 - \alpha$), as follows,

$$\sum_{(i,j) \in \hat{A}} x_{ij} \leq |\hat{A}| - 1, \quad (4.30)$$

where, as mentioned earlier, \hat{A} is set of arcs representing the currently selected route. Otherwise the new incumbent will be accepted, resulting in an updated upper bound (UB) of the problem.

The following statement describes the algorithm of CCP method.

Algorithm 6 Combinatorial cutting plane algorithm

```
1: Set  $UB = +\infty, LB = -\infty$ 
2: while ( $UB > LB$ ) do
3:   Solve the RMP model and get a candidate incumbent:  $\hat{v}x = \{x_{ij} : (i, j) \in A\}$  and  $\hat{z}$ 
4:   Let  $\hat{A} =$  The set of arcs included in the path defined by the incumbent.
5:   Calculate  $P =$  the failure probability of the incumbent.
6:   if  $P \leq 1 - \alpha$  then
7:     Accept  $(\hat{x}, \hat{z})$  as the new incumbent and set  $UB \leftarrow \hat{z}, x^* \leftarrow \hat{x}$ .
8:   else
9:     Reject the incumbent and add the combinatorial cut to the model:
10:    
$$\sum_{(i,j) \in \hat{A}} x_{ij} \leq |\hat{A}| - 1.$$

11:    Update LB with the new cut added to RMP
12:   end if
13: end while
```

The CCP algorithm could be applied as an accessible method for moderate-sized network problems by iteratively removing the “infeasible” incumbent candidate in the solution tree of RMP. While in large-sized networks, there would be an extremely large number of incumbents in the solution tree to be visited and evaluated before reaching the optimal and feasible one. The weak combinatorial cut cannot efficiently remove these routes, and thus a slow convergence of UB and lower bound (LB) is expected.

4.3.3.2 Combinatorial Cutting Plane using Local Branching

One of the major reasons for the slow converges of CCP method is that the combinatorial cutting plane (4.30) only cuts off one feasible routing solution (i.e., \hat{A}) at each iteration. In order to ac-

celerate the speed of combinatorial cutting plane algorithm, we can seek to add multiple or tighter cuts at each iterations. One way to find multiple infeasible solution is through local branching (i.e., checking the local solutions, which may be infeasible, around the current solution). We name this method Combinatorial Cutting Plane Algorithm using Local Branching (CCPLB). In order to facilitate the discussion, we need to first introduce the following definition, Minimal Infeasible Set (MIS).

Definition 1. A Minimal Infeasible Set, Ω , is a collection of arcs such that setting $x_{ij} = 1, \forall (i, j) \in \Omega$ will violate the chance constraints (4.26g–4.26h) or (4.27f-4.27g), and excluding any arc from Ω will satisfy the chance constraints.

In this method, instead of only eliminating the current infeasible node from the solution tree, the CCPLB method removes the minimal infeasible sets (MIS), allowing the algorithm more efficient to cut risky routes away.

In order to find the MIS for the network problems in each iteration, we need to resort to a specially designed solutions space tree, which is constructed in the following manner. Considering that each arc could be included or not, there are totally $2^{|A|}$ potential solutions (feasible and infeasible) in our problem. The tree consists of all these $|A|$ -tuples each of which represent one node. All the arcs in the network are reordered in the descendant order of failure probabilities. Each tuple (i.e., a node) is then formed to contain $|A|$ binary numbers and each number indicates whether the corresponding arc is selected or not (1:yes; 0:no). The left most digit (i.e., binary number) is representing the arc with the highest failure probability and the right most digit is corresponding to the arc with the lowest failure probability. There are $|A|$ levels and the γ^{th} level represents all the routes of exactly γ arcs being included ($\gamma \in \Gamma = \{\text{the set of all levels in the tree}\}$). In addition, the tuples that have all the “1”s except for the rightmost “1” in the same positions are branched from the same parent tuple and form a group. They are placed in the tree so that for any neighbour child nodes the failure probabilities of the routes in the left are greater than those in the right as

in Remark 2. From another angle, performing a depth-first search from right to left on this tree actually is equivalent to counting binary numbers from 0 to $2^{|A|} - 1$. A specially designed solution space tree for a network with 4 arcs is shown in Figure 4.8.

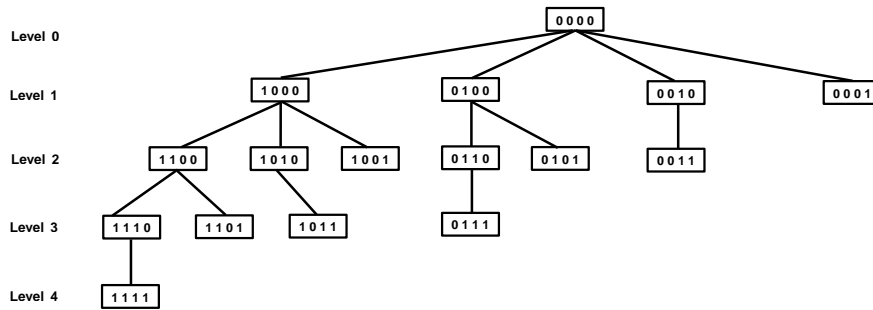


Figure 4.8: The special solution space tree for CCPLB Algorithm for a network with 4 arcs

Remark 1. In the special solution space tree, the failure probabilities of child tuples (the deeper level) are always higher than their ancestor nodes.

Remark 2. For any two consecutive nodes/tuples in the same group (i.e., from the same parents), the failure probability of the left node is higher than the one on the right.

Remark 3. If the current solution, \hat{x} , from relaxed master problem, does not satisfy the chance constraints, then solutions corresponding to the sibling node(s)/tuple(s) on the left side of the tuple/node based on \hat{x} also does not meet the chance constraints.

Algorithm 7 Combinatorial cutting plane with local branching

```
1: Set  $UB = +\infty, LB = -\infty$ 
2: while ( $UB > LB$ ) do
3:   Solve the RMP model and get a candidate incumbent  $\hat{\mathbf{x}}$  and  $\hat{z}$ , and find the corresponding
   tuple  $\hat{c}$ .
4:   Let  $\hat{G} = \{\text{The set of tuples in the group which contains } \hat{c}\}$ .
5:   Let  $\hat{c}_p = \text{the parent tuple of } \hat{c}$ .
6:   Compute  $P_{\hat{c}} = \text{the failure probability of the tuple } \hat{c}$ .
7:   if  $P_{\hat{c}} \leq 1 - \alpha$  then
8:     Accept  $(\hat{\mathbf{x}}, \hat{z})$  as the new incumbent and set  $UB \leftarrow \hat{z}, \mathbf{x}^* \leftarrow \hat{\mathbf{x}}$ .
9:   else
10:    Reject the incumbent. Compute  $P_{\hat{c}_p} = \text{the failure probability of } \hat{c}_p$ .
11:    while  $P_{\hat{c}_p} > 1 - \alpha$  do
12:      Set  $\hat{c} \leftarrow \hat{c}_p$ , and update  $\hat{G}$ .
13:    end while
14:    Add MIS Cuts based on all the infeasible tuples  $C_B \subset \hat{G}$ .
15:  end if
16: end while
```

Assume that an integer solution $\hat{\mathbf{x}}$ was obtained by solving the RMP and the solution corresponds to the tuple (1011) in the tree. Then the failure probability of the tuple is calculated as in Theorem 4.6. If the probability is smaller than the allowed level $1 - \alpha$, $\hat{\mathbf{x}}$ is accepted as an incumbent. Otherwise, the feasibility of the local tuples will be checked and more constraints will be added to the relaxed master problem. In general, the parent tuple (1010) will be firstly considered. If (1010) is feasible, a series of combinatorial cuts will be generated from the tuple (1011) and the infeasible sibling tuples (if there is any). On the other hand, if (1010) is infeasible, we will continue to go up

until a feasible tuple is obtained. As a result, a series of MIS will be found by locally exploring the solution space of RMP by taking advantage of the special solution space tree, and the combinatorial cuts will be added to the problem to remove these MIS's. The algorithm is described as Algorithm 7.

4.3.3.3 Benders Decomposition Method

As is discussed in previous section, the binary restriction on l^s can be removed because there are already binary requirements on x_{ij} and failure status Y_{ij}^s . Once x_{ij} is fixed, the feasibility checking problem (chance constraints) is reduced to checking the feasibility of a linear program. Hence it is natural to consider another well-known and powerful algorithm, Benders Decomposition. Benders decomposition (BD) is a common constraint generation method which also splits a problem into a relaxed master problem and a subproblem (SP) and reaches the optimal solution by iteratively adding cutting planes. Here we utilize the same idea to generate feasibility cuts which will be incorporated in the solution space of the routing problems by looking into a risk evaluation problem. Also, the original problem is relaxed first by eliminating the chance constraints in the CCTSP and CCSPP. Because the subproblem is a feasibility checking problem, we can simply choose a fixed-value objective function (e.g., 0). Hence the subproblem is formulated as follows,

$$\text{[SP]} \quad \min \quad 0 \quad (4.31a)$$

$$\text{s.t.} \quad l^s \geq \hat{x}_{ij} Y_{ij}^s, \quad \forall (i, j) \in A, s \in S \quad (4.31b)$$

$$\sum_{s \in S} \text{Prob}^s l^s \leq 1 - \alpha \quad (4.31c)$$

$$l^s \geq 0, \quad \forall s \in S \quad (4.31d)$$

Let π and u_{ij} be the dual variables associated with the constraints (4.31b) and (4.31c) respectively.

Then the dual problem of [SP] is formulated as follows,

$$\text{[DSP]} \quad \max \sum_{s \in S} \sum_{(i,j) \in A} \hat{x}_{ij} Y_{ij}^s u_{ij}^s + (\alpha - 1)\pi \quad (4.32a)$$

$$\text{s.t.} \quad \sum_{(i,j) \in A} u_{ij}^s - \text{Prob}^s \pi \leq 0 \quad \forall s \in S \quad (4.32b)$$

$$u_{ij}^s \geq 0 \quad \forall (i,j) \in A \quad \forall s \in S \quad (4.32c)$$

$$\pi \geq 0 \quad (4.32d)$$

Note that [DSP] is always feasible as it has a homogeneous form, i.e., all right hand sides are 0. Hence, the only occasion of [SP] being infeasible is when [DSP] has recession directions or extreme directions. Recall that we can easily determine whether a solution \hat{x} satisfies the chance constraints through Theorem 4.6. Once we know \hat{x} is making [SP] infeasible, we can use a recession or extreme direction of [DSP] to construct a feasibility cut to be added into the relaxed master problem in order to eliminate this current infeasible solution and hope to cut off some other infeasible solutions. Hence, next, we will focus on how to efficiently construct a recession or extreme direction of [DSP] without solving any optimization problem.

4.3.3.4 *Constructing a Benders' Feasibility Cut via Scenario Aggregation without Solving Any Optimization Problem*

In the objective function (4.32a), the coefficients of u_{ij}^s is 1 if and only if both \hat{x}_{ij} and Y_{ij}^s are assigned as 1. This motivates us to focus on special arcs instead of all arcs when solving [DSP]. To facilitate the discussion, we first introduce the following definition of reordered arc set based on the current solution from [RMP], \hat{x} .

Definition 2. Assume that \hat{x} is a solution of the [RMP], which represents a route including k arcs. Let m denote the total number of arcs in the given network. An **\hat{x} -reordered arc set** is

a set in which arcs $1(\hat{\mathbf{x}}), 2(\hat{\mathbf{x}}), \dots, k(\hat{\mathbf{x}})$ corresponds to those included in the route, \hat{A} , and the remaining $m - k$ arcs are given random orders after k . Equivalently, the followings hold for the un-ordered sets that $\{1(\hat{\mathbf{x}}), 2(\hat{\mathbf{x}}), \dots, k(\hat{\mathbf{x}})\} = \{(i, j) \in A | \hat{x}_{ij} = 1\}$ and $\{k + 1(\hat{\mathbf{x}}), \dots, m(\hat{\mathbf{x}})\} = \{(i, j) \in A | \hat{x}_{ij} = 0\}$.

Because handling all scenarios individually is very time consuming, we aggregate the scenarios in the sample space $\{0, 1\}^A$ into m subsets (i.e., $\mathcal{S}_{1(\hat{\mathbf{x}})}, \mathcal{S}_{2(\hat{\mathbf{x}})}, \mathcal{S}_{3(\hat{\mathbf{x}})}, \dots, \mathcal{S}_{m(\hat{\mathbf{x}})}$) in the following way:

- (a) the first subset includes all the scenarios in which arc $1(\hat{\mathbf{x}})$ fails, i.e., $Y_{1(\hat{\mathbf{x}})}^s = 1$, and apparently the occurring probability of this subset is equal to $\text{Prob}^{\mathcal{S}_{1(\hat{\mathbf{x}})}} = p_{1(\hat{\mathbf{x}})}$;
- (b) the second subset includes all the scenarios in which arc $1(\hat{\mathbf{x}})$ succeeds and arc $2(\hat{\mathbf{x}})$ fails, i.e., $Y_{1(\hat{\mathbf{x}})}^s = 0, Y_{2(\hat{\mathbf{x}})}^s = 1$, and the occurring probability of this subset is $\text{Prob}^{\mathcal{S}_{2(\hat{\mathbf{x}})}} = (1 - p_{1(\hat{\mathbf{x}})})p_{2(\hat{\mathbf{x}})}$;
- (c) the third subset include all scenarios where arc $1(\hat{\mathbf{x}})$ and $2(\hat{\mathbf{x}})$ succeed but arc $3(\hat{\mathbf{x}})$ fails, i.e., $Y_{1(\hat{\mathbf{x}})}^s = Y_{2(\hat{\mathbf{x}})}^s = 0, Y_{3(\hat{\mathbf{x}})}^s = 1$, and the probability of this subset is $\text{Prob}^{\mathcal{S}_{3(\hat{\mathbf{x}})}} = (1 - p_{1(\hat{\mathbf{x}})})(1 - p_{2(\hat{\mathbf{x}})})p_{3(\hat{\mathbf{x}})}$;
- (d) we perform this selection process until the m^{th} subset, which would includes the scenarios in which arcs $1(\hat{\mathbf{x}}), 2(\hat{\mathbf{x}}), \dots, m - 1(\hat{\mathbf{x}})$ succeed and arc $m(\hat{\mathbf{x}})$ fails and its probability is $\text{Prob}^{\mathcal{S}_{m(\hat{\mathbf{x}})}} = \prod_{t=1}^{m-1(\hat{\mathbf{x}})} (1 - p_{t(\hat{\mathbf{x}})})p_{m(\hat{\mathbf{x}})}$.

The following Theorem 4.7 summarizes our scenario aggregation scheme.

Theorem 4.7. $\mathcal{S}_{1(\hat{\mathbf{x}})}, \mathcal{S}_{2(\hat{\mathbf{x}})}, \dots, \mathcal{S}_{m(\hat{\mathbf{x}})}$ is a partition of the sample space $\{0, 1\}^A$, where

$$\mathcal{S}_{t(\hat{\mathbf{x}})} = \left\{ (Y_1, Y_2, \dots, Y_m) \in \{0, 1\}^A \mid \begin{array}{l} Y_\tau = 0, \tau = 1, \dots, t - 1; Y_t = 1; Y_\tau \in \{0, 1\}, \tau = t + 1, \dots, m \end{array} \right\} \quad (4.33)$$

In addition, $\text{Prob}^{S_{t(\hat{\mathbf{x}})}} = p_{t(\hat{\mathbf{x}})} \prod_{\tau=1}^{t-1} (1 - p_{\tau(\hat{\mathbf{x}})})$.

Upon the scenario aggregation, we construct a recession direction as described in Theorem 4.8 for the [DSP].

Theorem 4.8. *If the [SP] is infeasible, then its dual problem [DSP] is unbounded along the recession direction $\mathbf{r} = [\mathbf{r}_u \quad \mathbf{r}_\pi]^\top = [r_{ij}^s, (i, j) \in A, s \in S, \quad 1]^\top$, where*

$$\forall s \in S, (i, j) \in A, \quad r_{ij}^s = \begin{cases} \text{Prob}^s, & \text{if } (i, j) \equiv t(\hat{\mathbf{x}}) \text{ and } s \in \mathcal{S}_{t(\hat{\mathbf{x}})}, \\ 0, & \text{otherwise;} \end{cases} \quad (4.34)$$

Proof. As is discussed previously, [DSP] constraints are homogeneous and $\mathbf{0}$ is an obvious feasible point. Hence when [SP] is infeasible, [DSP] is unbounded. An arbitrary point along direction \mathbf{r} starting from $\mathbf{0}$ is $\delta \mathbf{r} = (\delta \mathbf{r}_u, \delta \mathbf{r}_\pi)$ where δ is an arbitrary positive number. $\delta \mathbf{r}$ certainly satisfies constraint (4.32c) and (4.32d). Plugging this point into the left hand side (4.32b), for any scenario $s \in S$, we obtain the following,

$$\delta \left(\sum_{(i,j) \in A} r_{ij}^s - \text{Prob}^s \right) \quad (4.35)$$

Any $s \in S$ belongs to one and only one subset among $\mathcal{S}_{1(\hat{\mathbf{x}})}, \mathcal{S}_{2(\hat{\mathbf{x}})}, \dots, \mathcal{S}_{m(\hat{\mathbf{x}})}$ as they are a partition of S according to Theorem 4.7. Suppose that $s \in \mathcal{S}_{t(\hat{\mathbf{x}})}$. Based on the definition of \mathbf{r} , i.e., (4.34), there is at most one arc, $(i, j) \equiv t(\hat{\mathbf{x}})$, of which the decision variable r_{ij}^s is assigned as Prob^s . Hence $\sum_{(i,j) \in A} r_{ij}^s \leq \text{Prob}^s$. Therefore, (4.35) is always less than 0 since $\delta > 0$. This concludes that any point $\delta \mathbf{r}$ is feasible ($\delta > 0$) for [DSP].

In the following, we will show that the objective value of the point $\delta \mathbf{r}$ is positive. Plugging this point into the objective function, we obtain the following,

$$\sum_{s \in S} \sum_{(i,j) \in A} \hat{x}_{ij} Y_{ij}^s r_{ij}^s - (1 - \alpha) = \sum_{t=1(\hat{\mathbf{x}})}^{m(\hat{\mathbf{x}})} \sum_{s \in \mathcal{S}_t} \hat{x}_t Y_t^s \text{Prob}^s - (1 - \alpha)$$

$$\begin{aligned}
&= \sum_{t=1(\hat{\mathbf{x}})}^{k(\hat{\mathbf{x}})} \text{Prob}^{S_t} - (1 - \alpha) = \sum_{t=1(\hat{\mathbf{x}})}^{k(\hat{\mathbf{x}})} p_t \prod_{\tau=1}^{t-1} (1 - p_\tau) - (1 - \alpha) \\
&= \left(1 - \prod_{(i,j) \in \hat{A}} (1 - p_{ij}) \right) - (1 - \alpha) > 0
\end{aligned}$$

In above, the first equality sign is due to definition (4.34); the second equality sign is due to the definition of $\mathcal{S}_{1(\hat{\mathbf{x}})}, \mathcal{S}_{2(\hat{\mathbf{x}})}, \dots, \mathcal{S}_{m(\hat{\mathbf{x}})}$; the third equality sign is due to the definition of Prob^{S_t} in Theorem 4.7; the greater-than inequality sign is due to the definition of failure probability in Theorem 4.6 and the fact that [SP] with $\hat{\mathbf{x}}$ is infeasible. Therefore, because for any $\delta > 0$ the objective value is positive and proportional to δ , \mathbf{r} is a recession direction of [DSP], and we can use it to generate a feasibility cut that at least will eliminate the current solution, $\hat{\mathbf{x}}$. \square \square

With the newly obtained recession ray, we can construct a feasibility cut. In the following proposition, we will show a concise form of this cut.

Proposition 4.4.

$$\sum_{t=1(\hat{\mathbf{x}})}^{m(\hat{\mathbf{x}})} \left[\prod_{\tau=1}^t (1 - p_{\tau-1}) \right] p_t x_t \leq 1 - \alpha \tag{4.36}$$

is a valid Benders feasibility cut to our reliable routing problems.

Proof. This proof is very similar to the proof showing \mathbf{r} is a solution with positive objective value. We just need to plug $\mathbf{r} = [\mathbf{r}_u, 1]$ into the objective function in [DSP], and show the aggregated form (4.36) based on the definition of $\mathcal{S}_{1(\hat{\mathbf{x}})}, \mathcal{S}_{2(\hat{\mathbf{x}})}, \dots, \mathcal{S}_{m(\hat{\mathbf{x}})}$ and Theorem 4.7 and 4.8. \square

In the definition of $\hat{\mathbf{x}}$ -reordered arc set, we mainly distinguish arcs with $\hat{x}_{ij} = 1$ from arcs with $\hat{x}_{ij} = 0$. Then the $\hat{\mathbf{x}}$ -reordered arc set can be divided into two parts: the first part containing all arcs with $\hat{x}_{ij} = 1$ and second part containing all arcs with $\hat{x}_{ij} = 0$. Within each part, the arcs

can be randomly sorted. However, with different order with the two parts, we can derive cuts with different strengths. Note that, (4.36) has a fixed right hand side and positive coefficients on the left hand side. Hence the larger are the coefficients, the stronger the constructed cuts will be. This is actually controlled by the order of arcs within the \hat{x} -reordered arc set. Usually we sort the first part by the failure probabilities of arcs in \hat{A} in descending order because of the way the coefficients are calculated in (4.36). Similarly we will sort the second part. Upon the aggregation scheme, the recession ray construction method, and cutting planes, we can implement the Benders decomposition in Algorithm 8.

Algorithm 8 Benders decomposition

- 1: Set $UB = +\infty, LB = -\infty$
 - 2: **while** ($UB > LB$) **do**
 - 3: Solve the RMP model and get a candidate incumbent: $\hat{x} = \{x_{ij} : (i, j) \in A\}$ and \hat{z}
 - 4: Let $\hat{A} =$ The set of arcs included in the path defined by the incumbent.
 - 5: Calculate $P =$ the failure probability of the incumbent.
 - 6: **if** $P \leq 1 - \alpha$ **then**
 - 7: Accept (\hat{x}, \hat{z}) as the new incumbent and set $UB \leftarrow \hat{z}, x^* \leftarrow \hat{x}$.
 - 8: **else**
 - 9: Reject the incumbent and add the Benders feasibility cut (4.36) to the model.
 - 10: Update LB with the new cut added to RMP
 - 11: **end if**
 - 12: **end while**
-

4.3.3.5 Discussion on Cutting Planes' Strength

The performance of each algorithm presented above is dependent upon how efficiently the added lazy constraints can eliminate infeasible routes. We consider the following example to illustrate

the comparison of the proposed algorithms. Suppose that we are given a network with nodes and arcs denoted by letters and numbers respectively, as shown in Figure 4.9. Arc failure probabilities are shown inside the parentheses. A reliable shortest path is requested departing from the starting node and arriving at the destination one. A 98% confidence level is required.

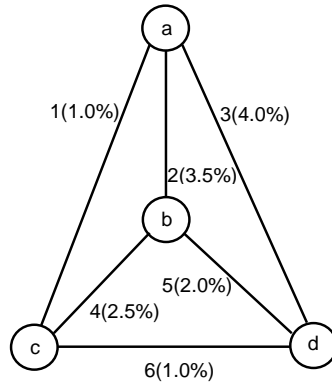


Figure 4.9: A network under uncertain topology with arc failures. Starting node: a; Destination node: d.

Assume that in the process of branch-and-cut search, we are currently at the incumbent which represents the path of 2-4-6. We can then compute its failure probability, $P = 1 - (1 - 0.035)(1 - 0.02)(1 - 0.01) = 6.4\% > 2\%$, which renders the incumbent infeasible. With the CCP algorithm, we generate the combinatorial cut, $x_2 + x_4 + x_6 \leq 2$, to remove the current solution from the search tree. Higher efficiency is observed from CCPLB method, in which we first find the corresponding tuple: 010101, in the probability tree. Then we keep going up until arriving at the tuple whose parent is feasible: 010000. Finally we generate the following MIS cuts: $x_3 \leq 0$ and $x_2 \leq 0$. In addition of only removing the current infeasible solution as in the CCP, the CCPLB algorithm identifies all the infeasible paths containing arcs 2 or 3 and removes them from the search tree with the MIS cuts. We notice that there are only two possible paths left: 1-4-5 and 1-6. With the BD algorithm, we generate the following Benders feasibility cut : $0.035x_2 + 0.0193x_4 + 0.009457x_6 + 0.009362x_1 + 0.037075x_3 + 0.013347x_5 \leq 0.02$. With the failure probability of each arc involved, the cut can identify more infeasible routes than the current one itself. For example, neither arc 2

nor 3 can be selected in the optimal route according to the cut. Moreover, the combinations of some arcs are also excluded, such as arcs 1,4 and 5. In the result, the only possible path of 1-6 is left in the search tree. Hence, we expect that the BD algorithm would have a better performance than the others to obtain the reliable shortest path.

4.3.3.6 Compact Reformulation

In [16], it is shown that integer programming problems under probabilistic constraints involving discrete distributions can be reformulated as integer problems with knapsack constraints. In [17] and [4], the logarithm is taken for both sides of chance constraints in set covering problems and a compact formulation is then obtained with the chance constraints being replaced by a knapsack constraint. Here we will use the method to both the CCTSP and CCSPP, as demonstrated in Theorem 4.9.

Theorem 4.9. *The chance constraints (4.26g – 4.26h) and (4.27f – 4.27g) are identical to the following knapsack constraint,*

$$\sum_{(i,j) \in A} x_{ij} \log(1 - p_{ij}) \geq \log(\alpha) \quad (4.37)$$

Proof. According to the facts that a route would fail unless all the included arcs succeed and that the arcs not included in the route will not affect its risk level, it is straightforward to rewrite the chance constraints (4.26g – 4.26h) and (4.27f – 4.27g) as,

$$P(L_{\hat{A}} = 0) = \prod_{(i,j) \in \hat{A}} (1 - p_{ij})^{x_{ij}} \geq \alpha$$

which can be reformulated to (4.37) by taking the logarithm transform. □

The compact formulations for the CCTSP and CCSPP can then be written as follows,

$$\text{[CF-CCTSP]} \quad \min \quad z = \sum_{i=1}^n \sum_{j=1}^n c_{ij}x_{ij} \quad (4.38a)$$

$$\text{s.t.} \quad (4.26b) - (4.26f), (4.37) \quad (4.38b)$$

$$\text{[CF-CCSPP]} \quad \min \quad z = \sum_{i=1}^n \sum_{j=1}^n c_{ij}x_{ij} \quad (4.39a)$$

$$\text{s.t.} \quad (4.27b) - (4.27e), (4.37) \quad (4.39b)$$

The cutting plane methods proposed in Sections 4.3.3.1-4.3.3.3 are all about reducing the feasible set of a given optimization problem by cutting risky solutions away, whereas the compact formulation extends the original model with an additional knapsack constraint. Both approaches have pros and cons, which were discussed by [49] in an uncertain set covering problem. In the next section, we will computationally compare the practical behaviors of all the proposed methods for dealing with the uncertainty in the two routing problems, i.e., CCTSP and CCSPP.

4.3.4 Numerical Experiment and Results

Numerical experiments and results of different algorithms are presented in this section on solving both the CCTSP and the CCSPP. The algorithms are coded in Microsoft Visual C++ linked with CPLEX 12.5. All the programs are run in Microsoft Windows 7 Professional operating system on a Dell Desktop with Intel Core i5-2400 CPU 3.10GHz and 4GB RAM.

4.3.4.1 CCTSP Experimental Setup

In our computational experiments of solving CCTSP, we generate instances of the directed graph $G = (N, A)$ in different sizes with 20, 30 and 50 nodes. For every node $i \in N$, we randomly pick an integer between the minimum and the maximum outgoing degrees required for every node in G , of which the values are indicated in Table 4.5 for the corresponding graph sizes and density levels (Density 1, 2 and 3). This integer value represents a target degree of node i . We then repeatedly generate nodes adjacent to every node i from all nodes in $N \setminus \{i\}$ until reaching the degree target. After this, to provide an initially connected graph we manually add arcs connecting each pair of nodes if they are not connected, checked by using a search algorithm.

Table 4.5: The outgoing degree and probability ranges assumptions for CCTSP instances

Node	Density 1	Density 2	Density 3	Probability Range
20	[4, 6]	[6, 8]	[9, 11]	[0, 1.5%]
30	[4, 6]	[6, 8]	[9, 11]	[0, 1.0%]
50	[4, 6]	[9, 11]	[14, 16]	[0, 0.5%]

For each arc $(i, j) \in A$, the cost c_{ij} is randomly generated based on a uniform distribution in $[1, 20]$ for all instances. Because the number of arcs in a TSP route is equal to that of nodes in the graph, we tune the ranges of uniform distributions for generating failure probabilities, as shown in Table 4.5 for corresponding graph sizes, to ensure the existence of feasible solutions. (We also tried to solve the problem with higher failure probabilities, and it turned out that either the problem is infeasible or the optimal cost is unrealistically high.) In Section 4.3.4.2 we will show the experimental results for each instance and for each value of confidence level $\alpha \in \{0.85, 0.90, 0.95\}$. The algorithm for generating our random instances is summarized as follows,

Algorithm 9 Random instance generation

- 1: Initialize the set of arcs: $A = \emptyset$.
 - 2: Determine the outgoing degree range $[d_l, d_u]$ and the failure probability range $[p_l, p_u]$ according to Table 4.5.
 - 3: **for** each node $i \in N$ **do**
 - 4: Randomly generate an integer $d \sim U(d_l, d_u)$.
 - 5: Let the set $\Gamma = \{i\}$.
 - 6: **while** (d) **do**
 - 7: Generate an arc (i, j) after randomly picking an integer $j \in N \setminus \Gamma$.
 - 8: Update the sets: $A = A \cup \{(i, j)\}$, $\Gamma = \Gamma \cup \{j\}$.
 - 9: $d \leftarrow d - 1$.
 - 10: **end while**
 - 11: **end for**
 - 12: **for** each arc $(i, j) \in A$ **do**
 - 13: Generate the arc cost $c_{ij} \sim U(1, 20)$.
 - 14: Generate the failure probability $p_{ij} \sim U(p_l, p_u)$.
 - 15: **end for**
-

4.3.4.2 CCTSP Computational Results

The computational results of proposed methods are shown in Table 4.6 on the CCTSP, where the first column gives the graph instances that have been generated and tested (e.g., 20-d1 represents the graph with 20 nodes and density level 1). Column “ α ” indicates the values of confidence levels for each instance and “OFV” represents the objective function value or optimal cost for each graph size and each value of α . The columns of CCP, CCPLB, BD and CF show the computational time of solving the CCTSP with corresponding algorithms.

Our computational experiments showed that our machine is unable to handle the problem while calling the MILP solver to directly solve the extensive formulation (4.26a)-(4.26h) of each CCTSP instance. On the other hand, most of test instances including large-sized ones can be solved at a reasonable cost by the three proposed cutting plane methods, including CCP, CCPLB and B-D algorithms. All of these methods obtain optimal solutions by evaluating the risk of candidate incumbents while iteratively adding different cuts to reduce solution spaces. The CCP method

Table 4.6: Computational results for CCTSP

Instance	α	OFV	CCP	CCPLB	BD	CF
20-d1	0.85	99	0.858	0.952	1.451	0.66
	0.90	165	-	1096.27	1.919	1.131
	0.95	o	-	9.469	0.234	0.016
20-d2	0.85	91	0.094	0.093	0.218	0.287
	0.90	101	2.34	2.808	0.811	0.385
	0.95	190	-	-	8.111	1.044
20-d3	0.85	88	0.14	0.14	0.312	0.686
	0.90	100	1818.38	352.119	5.522	1.771
	0.95	153	-	-	1646.97	1.674
30-d1	0.85	172	0.218	0.218	0.335	1.015
	0.90	212	184.982	50.356	4.212	1.312
	0.95	o	-	-	0.415	0.024
30-d2	0.85	154	0.733	0.81	1.345	1.307
	0.90	165	4.954	14.067	3.89	1.147
	0.95	o	-	-	4.844	0.458
30-d3	0.85	120	0.687	0.687	0.933	1.507
	0.90	130	6867.76	2529.88	19.906	1.167
	0.95	191	-	-	4660.22	1.832
50-d1	0.85	233	0.297	0.277	0.242	0.348
	0.90	268	1273.47	403.988	6.305	1.488
	0.95	o	-	-	1.248	0.483
50-d2	0.85	232	0.218	0.218	0.405	0.624
	0.90	240	-	-	53.576	2.013
	0.95	382	-	-	-	1.358
50-d3	0.85	217	0.296	0.297	0.296	1.614
	0.90	217	32.65	7.098	4.056	1.328
	0.95	289	-	-	-	1.612

Note: The columns of CCP, CCPLB, BD, and CF show the computational time of solving the CCTSP with the respective algorithms, where a dash indicates that the time limit of 3 hours is reached. OFV represents the objective function value of each instance, where a circle indicates that the instance is infeasible.

prunes infeasible incumbents with combinatorial cuts. It can be observed that the efficiency of this method is limited when the value of confidence level is increased from 0.85 to 0.95 due to the weakness of the cut generated in each iteration. Our results show that it takes a lot of computational time to reach the optimal solutions in the instances with larger-sized networks or high confidence level requirements. Moreover, the algorithm is unable to find any route that can satisfy the reliability requirement even after an exhaustive search (for 3 hours) for some instances (e.g.,

20-d1 with $\alpha = 0.90$).

By evaluating the subsets locally branched from candidate incumbents, the CCPLB would expect to generate much stronger cuts in each iteration than the CCP and thus cost less computational time. However, this algorithm is still limited when the graph size or the requirement of reliability is increased. The BD algorithm, which produces Benders feasibility cut in each iteration, successfully solve the problem in most of the instances that we have generated. It takes acceptable computational time to find the reliable route with optimal cost for each instance and the method exhibits much higher efficiency than the CCP and CCPLB. Similar to the CCPLB algorithm, the BD algorithm generates the Benders feasibility cuts that removes not only the current infeasible solution but also other sets which violate cuts as discussed in the last section. Figure 4.10 compares the efficiency of the cutting plane algorithms for the instance of “30-d1” and the confidence level of 90%. It takes only a few seconds for the lower bound and upper bound in the BD algorithm to converge, while longer time is observed when the CCP or CCPLB method is applied. As the network size and value of confidence level keep growing, however, the algorithm still struggles to solve the problem (e.g. 50-d3 with $\alpha = 0.95$).

Computational results show that the compact formulation is able to solve all the CCTSP instances with a reasonable average computing time. For the instances in which the confidence level is relatively low (e.g. 0.85) such that the addition of a limited number of cuts is enough to cut risky routes away, the three cutting plane methods exhibit better performance than the compact formulation. As confidence levels are increased, which leads to increasingly difficult problems, many candidate incumbents violate the reliability requirement and thus the callback function is repeatedly executed within the MILP solver. With the modeled knapsack condition, the compact formulation is able to more efficiently prune away these infeasible incumbents and thus is preferable to the cutting plane methods. Furthermore, it is also interesting to note that for both BD algorithm and compact formulation, a large speedup is achieved in the infeasible cases as a smaller number of iterations

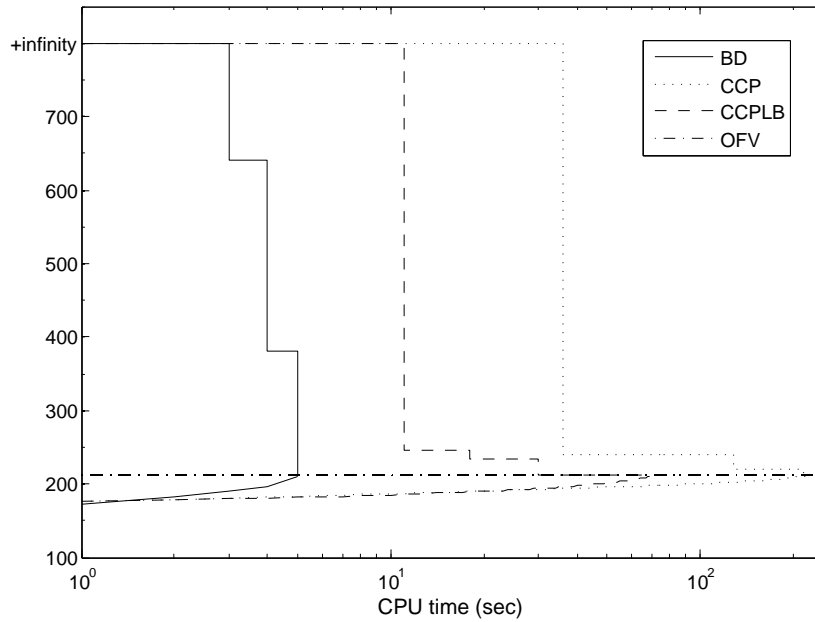


Figure 4.10: The efficiency of each cutting plane algorithm in the instance of 30-d1 with $\alpha = 90\%$

would be enough to prove that no reliable route exists.

In addition to solving the CCTSP at a reasonable computational cost for either small-sized or large-sized network, decision makers can benefit more from solving the proposed model. An important one is the sensitivity analysis regarding the change in confidence levels. Figure 4.11 shows the optimal cost that a TSP route in the graph instance of “30-d3” can have to satisfy the requirement of each confidence level.

As expected, more cost would be needed to increase the route reliability. But sometimes the cost can be dramatically decreased by reducing the confidence level even slightly. As shown in the figure, the total cost is reduced from 230 to 160 when the confidence level drops from 96% to 94%. According to these data, a decision maker probably would like to spend less by taking an acceptable risk rather than simply pursuing the predetermined reliability requirement. On the other hand, the reliability that the traveling salesman can achieve can be enhanced a lot via spending a

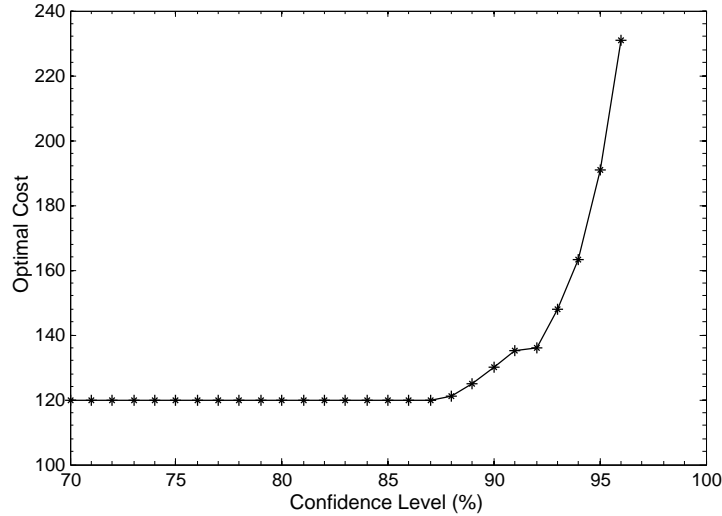


Figure 4.11: Optimal cost vs. confidence level in the instance of 30-d3

little more to select an alternate route. For instance, it costs less than 1% more to increase the level of confidence from 85% to 90%. Another information demonstrated by the data is the feasibility of the problem. Our results show that it is impossible to reach the confidence level $\geq 97\%$ no matter how much resource is investigated in the given 30-d3 network. In this case, the decision maker could alternatively select the last safe solution before infeasibility in practice.

4.3.4.3 CCSPP Experimental Setup

The same approach is applied as in the CCTSP to generate all the CCSPP instances of the directed graph $G = (N, A)$, in different sizes with 20, 50 and 100 nodes. For each graph size, Table 4.7 shows the ranges of outgoing degree for each density level and the failure probability associated with each arc, respectively. The cost of each arc is randomly generated from a uniform distribution in $[1, 20]$. Again, a set of different confidence levels $\alpha \in \{0.85, 0.90, 0.95\}$ are respectively applied in each instance, and to provide an initially connected graph and ensure the existence of paths between origins and destinations, we manually add arcs connecting each pair of nodes if they are

not connected.

Table 4.7: The outgoing degree and probability range assumptions for CCSPP instances

Node	Density 1	Density 2	Density 3	Probability Range
20	[2, 3]	[3, 4]	[6, 8]	[0, 4.0%]
50	[3, 4]	[4, 6]	[9, 10]	[0, 2.0%]
100	[3, 4]	[4, 8]	[7, 10]	[0, 1.0%]

4.3.4.4 SPP Computational results

Similar computational results are observed for solving the CCSPP with the presented algorithms, as shown in Table 4.8. Reliable shortest paths can be obtained in small-sized graphs when no specific strategy is used. When the network is lightly enlarged, the number of scenarios that need be generated is exponentially increased to make the problem unsolvable.

Our experiments show that the CCP method performs well in the instances with low reliability requirement. As the network size and confidence level value increase, however, it takes considerably longer or even unacceptable computational cost to achieve the optimal solutions. The CCPLB method performs much better in some of the instances but is still limited in the others.

The BD algorithm, which costs much shorter time to reach the optimal solution for either small-sized or large-sized networks, exhibits the better performance to obtain the reliable shortest paths. The comparison between cutting plane methods is well demonstrated by Figure 4.12, which shows the bound contraction process for each algorithm for the instance of “50-d3” and the confidence level of 90%. Our results confirmed the ability of developed Benders feasibility cuts in effectively removing a large number of infeasible routes from the solution tree than the combinatorial cuts.

Table 4.8: Computational results for CCSPP

Instance	α	OFV	CCP	CCPLB	BD	CF
20-d1	0.85	140	5.514	1.123	0.218	0.218
	0.90	○	9.116	0.312	0.249	0.187
	0.95	○	8.643	0.266	0.14	0.031
20-d2	0.85	67	0.272	0.427	0.218	0.125
	0.90	106	114.452	0.764	0.156	0.125
	0.95	○	-	-	0.063	0.187
20-d3	0.85	41	0.171	0.187	0.156	0.219
	0.90	46	0.827	0.234	0.265	0.234
	0.95	○	-	-	0.359	0.234
50-d1	0.85	115	0.374	0.515	0.437	0.375
	0.90	163	-	-	1.217	0.25
	0.95	○	-	3.189	0.249	0.327
50-d2	0.85	95	0.265	0.296	0.858	0.406
	0.90	118	-	178.012	3.541	0.265
	0.95	○	-	1552.29	1.217	0.405
50-d3	0.85	40	0.343	0.359	0.343	0.327
	0.90	48	13.19	7.613	1.045	0.375
	0.95	72	-	-	32.279	0.344
100-d1	0.85	160	0.271	0.303	0.281	0.297
	0.90	165	5.611	1.872	0.452	0.358
	0.95	○	-	-	10.254	0.702
100-d2	0.85	88	0.39	0.359	0.405	0.39
	0.90	99	26.029	3.844	1.622	0.453
	0.95	188	-	-	-	1.155
100-d3	0.85	89	0.39	0.359	0.421	0.499
	0.90	97	29.063	66.346	8.611	1.326
	0.95	158	-	-	-	0.546

Note: The columns of CCP, CCPLB, BD, and CF show the computational times of solving the CCSPP with the respective algorithms, where a dash indicates that the time limit of 3 hours is reached. OFV represents the objective function value of each instance, where a circle indicates that the instance is infeasible.

By iteratively evaluating incumbent solutions, however, the BD method needs to generate a large number of cutting planes to the solution space of RMP before reaching the optimal solution. With the chance constraint transformed to the knapsack constraint, the compact formulation of each CCSPP instance can be solved with with a reasonable average computational cost as shown in the table.

Figure 4.13 shows the smallest cost that a path from the origin to the destination in the graph of

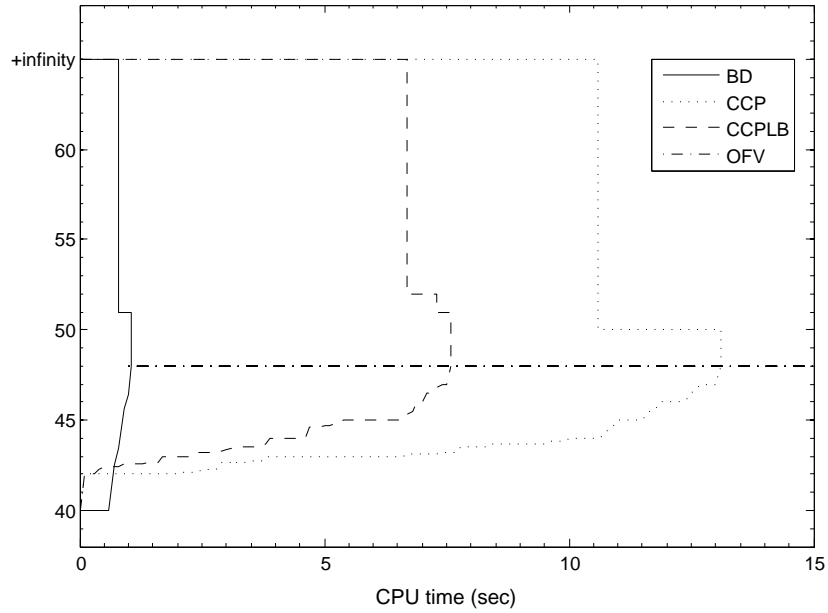


Figure 4.12: The efficiency of each cutting plane algorithm in the instance of 50-d3 with $\alpha = 90\%$

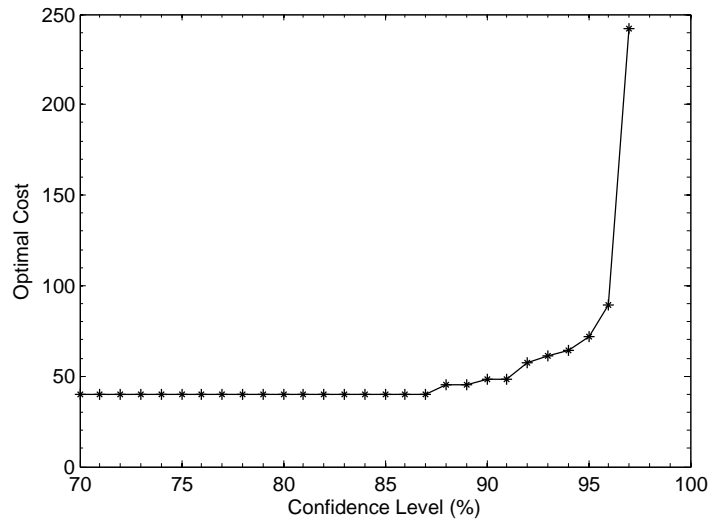


Figure 4.13: Optimal cost vs. confidence level in the instance of 50-d3(The problem is infeasible while $\alpha \geq 98\%$.)

“50-d3” can have for each confidence level, which is similar to the result of our CCTSP model. The information of the tradeoff between the cost and reliability obtained from solving our model

would be helpful for making the decisions accordingly.

4.4 Optimal Scheduling of Rapid Transit Routes

Personal rapid transit (PRT) is a public transport mode featuring small automated vehicles on a network of specially built guideways. In this section, we will study a PRT system in which the point-to-point transportation service is provided for customers by a fleet of vehicles that are driven by electricity. A dynamic vehicle routing problem with soft time window requirement is modeled for solving the optimal schedule of the system. Numerical experiments and computational results will be present and demonstrated.

4.4.1 Introduction

PRT is a public transport mode. A fleet of vehicles are distributed in a set of stations which are connected by specially built guideways, and provide customers direct service from origin to destination. The system is operated on-demand and the real-time responses need to be made upon customer requests. A few PRT systems are operational, including the oldest one in Morgantown, WV since 1975, one at Masdar City in UAE since 2010, the Ultra PRT system at London Heathrow Airport since 2011 and another one in South Korea since 2014.

In spite of the great need of scheduling optimal routes for public transport systems, there are few studies that have been conducted in this area until recently. A dynamic scheduling model for Beijing urban rail transit network with capacity constraints is developed in [153]. In [39], the transit routing problem is studied for computing all Pareto-optimal journeys in a dynamic public transit network for multiple criteria, such as arrival time and number of transfers. A heuristic method is discussed in [101] for the urban transit routing problem. [80] develops three lower bounds on

achievable mean passenger waiting time, one based on queuing theory, one based on the static problem, in which it is assumed that perfect information is available, and one based on a Markov Decision Process model. In this study, we focus on finding the routes for all the vehicles based at various stations in the PRT system so that the overall routing cost is minimized. This is actually in accordant with the definition of the classic vehicle routing problem (VRP). In addition to the typical VRP settings, we consider soft time windows while considering the cost for the routing. Specifically, each customer should be picked up at the origin station during a specific time frame and a penalty would be applied otherwise.

We aim to model a dynamic routing problem for the PRT system respecting the fact that the future customer demand are unknown with certainty and will be revealed dynamically. At the time point when the problem is being solved, the requests made during its past time period are collected and the schedule of each vehicle in future time periods are determined accordingly. All the vehicles are identical and have finite capacity. Each vehicle is driven by electricity and its battery can be charged when it stays at any station. Note that during peak hours, the load of guide roads could be extremely heavy in order to satisfy the real-time demands and thus need be controlled below a limit to avoid traffic jams. We therefore consider the capacity of any road connecting two adjacent stations at any time. Although the perfect future demands cannot be obtained, we assume that the stochastic information, e.g., the expected customer demands, is available. The stochastic knowledge of the future input can be used for us to make the better arrangement of all the vehicles systematically.

4.4.2 Problem Formulation

Before formulating the dynamic VRP for the PRT, we list all the sets, parameters and decision variables in Table 4.9.

Table 4.9: Summary of Parameters, Sets, and Variables.

Parameters and Sets	
N	The set of all nodes
A	The set of all arcs
T	The set of time periods
K	The set of all vehicles
L	The set of all tracks (guideways that connect any pairs of adjacent stations)
C	The capacity of each vehicle
W_t	The set of time periods within the time window for time t
MC	The maximum charging amount for each vehicle during a time period
EC_k	The electricity capacity for vehicle k
CD_i	The node capacity $\forall i \in N$
CT_l	The track capacity $\forall l \in L$
D_{ij}^t	The demand of request for pickup at node i and dropoff at node j at the beginning of t
s_k	The departing station for vehicle k
t_{ij}	Travel time from i to j
FC_{ij}	Fixed travel cost from i to j
VC_{ij}	Variable travel cost from i to j
d_{ij}	Penalty incurred if customer demand hasn't been met within the time window
c_r	Travel cost incurred if a vehicle takes the route r
Decision Variables	
z_{ij}^{kt}	The binary variable indicating whether vehicle k starts to travel at time t from i to j ; the vehicle stays in the station i during the time period t if $i = j$
x_{ij}^{kt}	The number of customers transported from i to j by vehicle k at time t
e_k^t	The remaining electricity level of vehicle k at the beginning of time period t
s_{ij}^t	The number of customers who haven't been served at time period t
y_{ij}^t	The number of customers who haven't been served within the time window

The electricity consumption of a vehicle when transporting customers consists of fixed and variable parts. Specifically the fixed usage is incurred once the vehicle is assigned to travel from one node to another regardless of how many customers it carries. The variable consumption, however, is dependent on and generally linearly related to the number of customers on the vehicle. As mentioned above, a penalty cost would be applied if any customer is not picked up at his/her origin station within a specific time window. This penalty can be considered as the bad customer experience, potential loss of customers and/or the selection of an alternate transportation tool with less fuel efficiency. The objective function can then be formulated as follows to minimize the total

consumption of electricity plus the cost penalty:

$$\min \sum_{(i,j) \in A} \sum_{k \in K} \sum_{t \in T} (FC_{ij} z_{ij}^{kt} + VC_{ij} x_{ij}^{kt}) + \sum_{(i,j) \in A} \sum_{t \in T} d_{ij} y_{ij}^t \quad (4.40)$$

The electricity level of the vehicles at the current time point should be considered. The vehicle short of electrical power cannot be selected to move customers. While staying at a node, the vehicle can be charged and its accumulative battery level is limited by the maximum electricity capacity of each vehicle and by the charging capacity within a time period at each node. The constraints of electricity for travel are described in (4.41a)-(4.41b).

$$e_k^t \leq e_k^{t-1} - \sum_{(i,j) \in A} \sum_{\tau=t-t_{ij}}^{t-1} \frac{FC_{ij} z_{ij}^{k\tau} + VC_{ij} x_{ij}^{k\tau}}{|i-j|} + MC \cdot \sum_i z_{ii}^{k(t-1)}, \quad \forall k \in K, t \in T \quad (4.41a)$$

$$e_k^t \leq EC_k, \quad \forall k \in K, t \in T \quad (4.41b)$$

All the vehicles in the PRT system are identical and have finite capacity to define the maximum number of customers that can be carried by each vehicle. The corresponding constraints are:

$$x_{ij}^{kt} \leq C z_{ij}^{kt}, \quad \forall (i,j) \in A, k \in K, t \in T \quad (4.42)$$

The following vehicle-flow balance constraints are included to describe that the number of vehicles which arrive at node i at time t should be equal to that of vehicles which leave or stay at the node.

$$\sum_j z_{ij}^{kt} = 1, \quad \forall k \in K, i = s_k, t = 0 \quad (4.43a)$$

$$\sum_i z_{ij}^{kt} - \sum_i z_{ji}^{k\tau} = 0, \quad \forall j \in N, k \in K, t \in T \setminus \{0\}, \tau = t - t_{ji} \geq 0 \quad (4.43b)$$

A timely response need to be made for each customer request to avoid the high penalty cost.

We model the following demand satisfaction constraints where (4.44) defines the customer-flow balance in the system and (4.45) indicates that at the end of the planning time horizon, all the customers have to be picked up and sent out from their origin stations.

$$x_{ij}^{kt} + s_{ij}^t = D_{ij}^t + s_{ij}^{t-1}, \quad \forall (i, j) \in A, t \in T \quad (4.44)$$

$$s_{ij}^t = 0, \quad \forall (i, j) \in A, t = T_E \quad (4.45)$$

The buffer capacity of a station and the capacity of a track restrict the number of vehicles staying at each node and traveling on the track at any time, which are respectively defined by (4.46) and (4.47). CD_i represents the buffer capacity of depot i and CT_l is the capacity of road l . $L(l)$ is the set of arcs which go through road l and $T_l(t)$ is the set of times from which if the vehicle starts then it will go through road l at time t .

$$\sum_k z_{ii}^{kt} \leq CD_i, \quad \forall i \in N, t \in T \quad (4.46)$$

$$\sum_k \sum_{(i,j) \in A, (i,j) \supset l} z_{ij}^{k\tau} \leq CT_l, \quad \forall l \in L, t \in T, \tau = t - t_{ii'} \quad (4.47)$$

4.4.3 Numerical Experiments and Computational Results

The formulated model is numerically tested on the PRT system at West Virginia University in Morgantown, for which the network can be seen in Figure 4.14. There are 5 stations in the system, including Medical (MD), Towers (TW), Engineering (ER), Downtown (DT) and Walnut (WN), which are connected by the guideways with the total length of 8.65 miles. 71 vehicles are available in the system to fulfill customer requests in real time. The travel time between any two adjacent stations is 5 min and each vehicle receives its routing command from the computer system every 5 min. The customer requests that were made during the past 5 min are collected to the computer

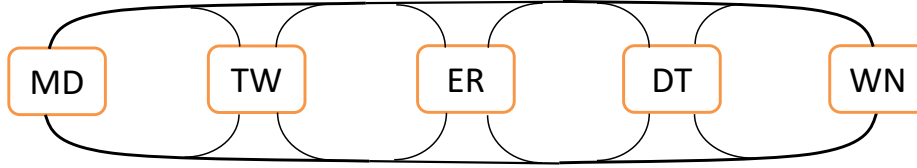


Figure 4.14: The PRT system at WVU

system and the demands in the future one hour are forecasted. The system keeps monitoring the status of each vehicle, including the battery level and the location, for determining whether it can be assigned for particular requests. The routes for all the vehicles within an hour are then determined by solving our model. The solution with respect to the current time period ($t = 0$) will be implemented. The state of the PRT system is then updated. When the exact customer demands are revealed in the future ($t = T \setminus \{0\}$), the routes can be adjusted by re-optimizing the model with the updated information.

The settings of our experiments and some PRT system characteristics are summarized and shown in Table 4.10, and the state of each vehicle at $t = 0$ is shown in Table 4.11. As shown in

Table 4.10: PRT system characteristics.

Time periods	12×5 min
Time window	2×5 min
Stations	5
Arcs	20
Tracks	8
Node capacity	25
Track capacity	5

Table 4.10, there are in total 20 arcs connecting all pairs of stations in the system, and the distance between any two adjacent nodes is 2 miles. The customer demands are randomly generated from poisson distributions as shown in Table 4.12. Specifically the demand corresponding to arc (i, j) represents the number of requests from customers at node i whose destination is j . The routes for all vehicles are obtained after solving the developed model on the generated scenario. The optimal cost (electricity consumption) is 490.2 kWh, which consists of the actual energy use and the penalty cost incurred when any customer are not served within the defined time window. In

Table 4.11: Vehicle info in the PRT system.

Quantity	73
Capacity (passengers)	10
Minimum Battery level (kWh)	5
Maximum Battery level (kWh)	30
fixed cost(kwh/mile)	0.5
variable cost(kwh/mile × passenger)	0.05

Table 4.12: Randomly generated customer demands for one hour.

Arc	Time period											
	1	2	3	4	5	6	7	8	9	10	11	12
1	6	0	6	2	8	12	2	2	8	6	10	4
2	8	2	2	4	6	4	4	2	4	8	2	0
3	2	6	4	2	2	4	2	2	12	0	0	4
4	12	6	4	4	10	6	4	8	6	10	2	6
5	10	4	0	8	6	2	2	0	8	12	4	4
6	6	4	14	0	10	4	6	2	10	0	6	2
7	0	8	4	6	0	4	6	2	4	2	0	4
8	8	2	4	2	2	8	6	4	8	0	2	4
9	2	6	6	2	12	6	2	4	10	4	4	4
10	2	4	0	6	10	6	6	4	6	8	2	4
11	10	6	8	4	4	8	2	8	8	2	6	2
12	12	0	4	6	18	0	2	4	12	4	0	8
13	12	6	2	0	10	4	6	6	6	6	6	4
14	8	6	0	4	6	2	4	4	10	4	2	4
15	12	8	4	4	10	10	2	2	4	8	8	8
16	10	0	6	4	4	4	6	4	2	2	6	6
17	8	10	0	4	8	8	2	6	4	2	4	4
18	2	2	2	2	10	6	0	6	8	4	4	2
19	10	4	6	2	2	6	6	6	12	6	8	2
20	8	2	0	2	2	4	0	4	4	8	8	4

particular, the route for one vehicles is shown as follows,

$DT \rightarrow DT \rightarrow DT \rightarrow * \rightarrow TW \rightarrow TW \rightarrow MD \rightarrow MD \rightarrow * \rightarrow ER \rightarrow ER \rightarrow ER \rightarrow * \rightarrow WN$

where the stations represent the locations of vehicles at the beginning of each time period; * indicates that the vehicle is moving toward its destination.

The numbers of customers who are not immediately served with respect to each arc at each time period are listed in Table 4.13, and those not picked within the time window are shown in Table 4.14, respectively. It can be observed that, although many passengers are not assigned to any vehicles right after their requests are submitted to the system due to the capacity limit of the system, most of them are served within the time window to avoid the penalty.

Table 4.13: Number of customers not immediately served.

Arc	Time period											
	1	2	3	4	5	6	7	8	9	10	11	12
1	6	6	6	0	8	0	2	4	4	0	0	4
2	8	10	2	6	2	0	4	6	0	8	0	0
3	2	8	2	4	6	0	2	4	0	0	0	4
4	12	8	2	6	6	2	6	4	10	10	2	8
5	10	14	4	2	8	0	2	2	0	2	0	4
6	6	10	14	14	4	8	4	6	6	6	2	4
7	0	8	12	0	0	4	0	2	6	0	0	4
8	8	10	4	6	0	0	6	10	8	8	0	4
9	2	8	4	6	8	6	8	2	2	6	0	4
10	2	6	0	6	6	4	10	4	0	8	0	4
11	0	6	4	8	2	0	2	0	0	2	0	2
12	10	0	4	0	18	8	0	4	6	0	0	8
13	12	18	0	0	10	4	10	6	12	8	0	4
14	0	6	6	0	6	8	2	6	6	10	2	6
15	12	10	4	8	8	18	10	2	6	4	2	10
16	10	0	6	10	4	8	6	0	2	4	0	6
17	8	18	8	2	10	8	10	7	1	3	0	4
18	2	4	6	0	10	6	6	2	0	4	0	2
19	10	4	0	2	4	0	6	2	4	2	0	2
20	8	0	0	2	4	0	0	4	4	2	0	4

For solving the proposed PRT problem, however, it often takes considerably long time to achieve the optimal solution, especially when the customer demands are high. Particularly for the instance generated above, we only obtain around 3% gap after solving the model for 5 min with CPLEX and less than 1% after 30 min. No feasible solution has been obtained if the time limit is set to 30 seconds. The route for each vehicle, however, needs to be provided efficiently and effectively, so

Table 4.14: Number of customers not served within the time window.

Arc	Time period											
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	2	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	2	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0

that the PRT system is well scheduled and operated to satisfy the customer demands in real time. To this respect, we developed a greedy heuristic algorithm to get a feasible solution very quickly, which can then be used to warm start the MIP in CPLEX. The algorithm is described as follows,

The greedy algorithm can easily give a feasible and implementable solution for the PRT system. Our result shows that for the generated test instance the total cost of the system solved from this algorithm is 554.2 kWh, 13% higher than the real optimal value. We observed that the number of passengers who are not served within the time window is the same as the optimal solution. However, the optimal schedule uses the vehicles for 117 times to fulfill the demands while the greedy algorithm uses 157 times. The use of more vehicles for meeting the same amount of demands consumes more electricity in the system and thus increase the total cost.

Algorithm 10 PRT Routing Greedy Algorithm

- 1: Let $\Omega^t = \{\omega d^t, \omega l^t, \omega n^t, \omega k^t\}$ be the status of the PRT system at time t , where $\omega d^t, \omega l^t, \omega n^t, \omega k^t$ represents the unsatisfied demands, track load, station load and vehicle status, respectively.
 - 2: Initialize Ω^0 by obtaining the initial status of the system.
 - 3: **for** every time period $t \in T$ **do**
 - 4: Set the status of all arc demands as unassigned.
 - 5: **while** any arc demand is unassigned **do**
 - 6: Find the unassigned arc $(i, j) = \operatorname{argmax}_{(i,j)} VC_{ij}d_{ij}$.
 - 7: Assign the arc demand to a vehicle located in the station i at time t with enough electricity level and set status of the arc (i, j) as assigned.
 - 8: **if** any track is overloaded **then**
 - 9: Cancel the assignment.
 - 10: **end if**
 - 11: **end while**
 - 12: Update the system status $\Omega^t, \forall t \in T$.
 - 13: Check the station load status ωn^t , send vehicles out from i to other stations if the load exceeds the buffer capacity.
 - 14: Cancel vehicle assignments if needed to avoid the track overload.
 - 15: Update the system status $\Omega^t, \forall t \in T$.
 - 16: **end for**
-

CHAPTER 5: CONCLUSIONS

In summary, the stochastic mixed-integer programming models have been developed for various large-scale problems in the area of power systems and transportation. The problems were respectively solved with optimization algorithms and methodologies.

A multistage SMIP was firstly modeled in Chapter 3 to address the capacity expansion planning in power systems. The integration of hourly unit commitment into the strategic-level investment makes the problem multiscale. A nested decomposition algorithm was developed and performed in the context of parallel computation. The algorithm was tested on various instances in IEEE 7-, 57-, and 118- bus systems, respectively, and the computational results showed the efficiency of the method in managing the intractability of the formulated large-scale problem. As an environmentally friendly fuel, natural gas is being extensively used for electricity generation while seeking the optimal profit is challenging for gas plants due to the volatility of the gas and electricity prices. Therefore another multistage SMIP model was developed to solve the optimal dispatch for a natural gas fueled power plant in the seek of profit maximization. The benefit from considering the stochasticity is demonstrated by the results obtained from numerical experiments. Scenario decomposition is enabled to enhance the manageability of the problem by using the Lagrangian relaxation. Both the column generation and subgradient method have been applied in order to determine the best Lagrangian multipliers. Compared to the former in which a master problem includes an extremely large number of constraints in large instances, the latter uses an analytic hill-climbing algorithm to iteratively update the multipliers, making the method easier to implement. A heuristic algorithm was developed to obtain a feasible solution at each iteration and the quality is evaluated by the lower bound achieved from solving the Lagrangian relaxation. Our results demonstrated the better performance of the proposed methods compared to solving the problem directly with the commercial MIP solver, i.e., CPLEX.

Four different routing problems were investigated in Chapter 4. Specifically two respective stochastic traveling salesman problems with discretely-distributed and continuously-distributed arc costs were modeled. In the context of stochastic arc costs, the deterministic traveling salesman problem's optimal solutions would be ineffective because the selected route might be exposed to a greater risk where the actual cost can exceed the resource limit in extreme scenarios. We therefore presented stochastic variations that incorporate risk management and developed algorithms for solving the optimal routes. Another SMIP models of reliable routing problems under uncertain topology are then derived in which the risk is controlled with chance constraints. Various cutting plane methods were developed and applied to solve the problems and their performances were compared. we also demonstrated that the results of the tradeoff between confidence level and optimal cost can be obtained from solving the models and such sensitivity analysis can be of help to practitioners seeking the best way to deal with the uncertainty in the routing problems. The last problem that was studied in the dissertation is a dynamic vehicle routing problem with soft time window for a PRT system. A fleet of vehicles are available in the system and provide each customer non-stop transportation service in real time, subject to the capacity limits of the system. We simulated a PRT system based on the one at West Virginia University and generated customer demand data for testing our model. The computational result showed that an efficient solution can be obtained and used to well schedule all the vehicles such that most customer requests can be satisfied within the defined time window.

LIST OF REFERENCES

- [1] Administration, U. S. Energy Information. 2009. The National Energy Modeling System: An Overview. <http://www.eia.gov/forecasts/aeo/er/index.cfm>.
- [2] Agency, United States Environmental Protection. 2009. EPA modeling applications using the Integrated Planning Model (IPM). <http://www.epa.gov/airmarkt/progsregs/epa-ipm/index.html>.
- [3] Ahmed, S., A. J. King, G. Parija. 2003. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization* **26** 3 – 24.
- [4] Ahmed, S., D.J. Papageorgiou. 2013. Probabilistic set covering with correlations. *Annals of Operations Research* **61**(2) 438C452.
- [5] Ahuja, R., T. L. Magnanti, J. B. Orlin. 1993. *Network Flows: Theory, Algorithms and Applications*. Printice Hall, Englewood Cliffs, NJ.
- [6] Alexander, G. J., A. M. Baptista. 2004. A comparison of VAR and CVaR constraints on portfolio selection with the mean-variance model. *Management Science* **50**(9) 1261–1273.
- [7] Applegate, D.L., R.E. Bixby, V. Chvatal, Cook W.J. 2006. *The traveling salesman problem: A computational study*. Princeton University Press, Princeton, New Jersey.
- [8] Arulselvan, A., C.W. Commander, P. Pardalos. 2007. A hybrid genetic algorithm for the target visitation problem .
- [9] Balas, E. 1979. Disjunctive programming. *Annals of Discrete Mathematics* **5** 3–51.
- [10] Baldick, R. 1995. The generalized unit commitment problem. *Power Systems, IEEE Transactions on* **10**(1) 465 –475. doi:10.1109/59.373972.

- [11] Bard, J.F., J.L. Miller. 1989. Probabilistic shortest path problems with budgetary constraints. *Computers & Operations Research* **16**(2) 145–159.
- [12] Barth, R., H. Brand, P. Meibom, C. Weber. 2006. A stochastic unit commitment model for the evaluation of the impacts of the integration of large amounts of wind power. *Probabilistic Methods Applied to Power Systems, 2006. PMAFS 2006. International Conference on*. Stockholm, Sweden, 1–8.
- [13] Bean, J. C., J. L. Hagle, R. L. Smith. 1992. Capacity expansion under stochastic demands. *Operations Research* **40**(Supplement 2) S210 – S216.
- [14] Benders, J. F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4** 238–252.
- [15] Benders, J.F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4**(1) 238–252.
- [16] Beraldi, P., M.E. Bruni. 2010. An exact approach for solving integer problems under probabilistic constraints with random technology matrix. *Annals of Operations Research* **177** 127–137.
- [17] Beraldi, P., A. Ruszczyński. 2002. The probabilistic set-covering problem. *Operations Research* **50**(6) 956–967.
- [18] Berman, O. 1989. The traveling salesman location problem on stochastic networks. *Transportation Science* **23**(1) 54–57.
- [19] Berman, O., D. Simchi-Levi. 1989. The traveling salesman location problem on stochastic networks. *Transportation Science* **23** 54–57.
- [20] Bertsimas, D.J. 1992. A vehicle routing problem with stochastic demand. *Operations Research* **40** 574–585.

- [21] Birge, John R. 1985. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research* **33**(5) 989–1007.
- [22] Birge, John R., François V. Louveaux. 1988. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research* **34**(3) 384 – 392.
- [23] Blair, C., R. Jeroslow. 1982. The value function of an integer program. *Mathematical Programming* **23** 237–273.
- [24] Bloom, J. A. 1983. Solving an electricity generating capacity expansion planning problem by generalized Benders’ decomposition. *Operations Research* **31**(1) 84 – 100.
- [25] Boginski, V.L., C.W. Commander, T. Turko. 2009. Polynomial-time identification of robust network flows under uncertain arc failures. *Optimization Letters* **3** 461–473.
- [26] Boyles, S.D., S.T. Waller. 2010. A mean-variance model for the minimum cost flow problem with stochastic arc costs. *Networks* **56**(3) 215–227.
- [27] Briant, Olivier, Claude Lemaréchal, Ph Meurdesoif, Sophie Michel, Nancy Perrot, François Vanderbeck. 2008. Comparison of bundle and classical column generation. *Mathematical programming* **113**(2) 299–344.
- [28] Campbell, A.M., B.W. Thomas. 2008. Probabilistic traveling salesman problem with deadlines. *Transportation Science* **42**(1) 1 – 21.
- [29] Canto, Salvador Perez. 2008. Application of benders decomposition to power plant preventive maintenance scheduling. *European journal of operational research* **184**(2) 759–777.
- [30] Carøe, C. C., J. Tind. 1998. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming* **83** 451–464.

- [31] Carøe, Claus C., Jørgen Tind. 1998. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming* **83**(1-3) 451–464. doi: 10.1007/BF02680570.
- [32] Carraway, R.L., T.L. Morin, H. Moskowitz. 1989. Generalized dynamic programming for stochastic combinatorial optimization. *Operations Research* **37**(5) 819–829.
- [33] Dantzig, G., Fulkerson R., S. Johnson. 1954. Solution of a large-scale traveling-salesman problem. *Operations Research* **2** 393–410.
- [34] Dantzig, G. B., J. H. Ramser. 1959. The truck dispatching problem. *Management Science* **6**(1) 80–91.
- [35] Dantzig, George B., Philip Wolfe. 1960. Decomposition principle for linear programs. *Operations Research* **8**(1) pp. 101–111.
- [36] Dantzig, George B, Philip Wolfe. 1960. Decomposition principle for linear programs. *Operations research* **8**(1) 101–111.
- [37] Davis, M. H. A., M. A. H. Dempster, S. P. Sethi, D. Vermes. 1987. Optimal capacity expansion under uncertainty. *Advances in Applied Probability* **19**(1) pp. 156–176. URL <http://www.jstor.org/stable/1427378>.
- [38] Davis, M. H. A., M. A. H. Dempster, S. P. Sethi, D. Vermes. 1987. Optimal capacity expansion under uncertainty. *Advances in Applied Probability* **19**(1) 156 – 176.
- [39] Dellinger, Daniel, Thomas Pajor, Renato F. Werneck. 2014. Round-based public transit routing. *Transportation Science* doi:10.1287/trsc.2014.0534.
- [40] Dentcheva, Darinka, Werner Römisch. 2004. Duality gaps in nonconvex stochastic optimization. *Mathematical Programming* **101**(3) 515–535.

- [41] Desrochers, Martin, Gilbert Laporte. 1991. Improvements and extensions to the miller-tucker-zemlin subtour elimination constraints. *Operations Research Letters* **10**(1) 27 – 36.
- [42] Desrosiers, Jacques, Francois Soumis, Martin Desrochers. 1984. Routing with time windows by column generation. *Networks* **14**(4) 545–565.
- [43] Dijkstra, E.W. 1959. A note on two problems in connexion with graphs. *Numerische Mathematik* **1**(1) 269–271.
- [44] du Merle, Olivier, Daniel Villeneuve, Jacques Desrosiers, Pierre Hansen. 1999. Stabilized column generation. *Discrete Mathematics* **194**(1) 229–237.
- [45] Duffie, D., J. Pan. 1997. An overview of value at risk. *The Journal of derivatives* **4**(3) 7–49.
- [46] Evans, G. W., T. L. Morin. 1986. Hybrid dynamic programming/branch-and-bound strategies for electric power generation planning. *IIE transactions* **18**(2) 138 – 147.
- [47] Fan, N., Q.P. Zheng, P.M. Pardalos. 2012. Robust optimization of graph partitioning involving interval uncertainty. *Theoretical Computer Science* **447** 53–61.
- [48] Fan, Y. Y., R. E. Kalaba, J. E. Moore II. 2005. Shortest paths in stochastic networks with correlated link costs. *Computers and Mathematics with Applications* **49** 1549–1564.
- [49] Fischetti, Matteo, Michele Monaci. 2012. Cutting plane versus compact formulations for uncertain (integer) linear programs. *Mathematical Programming Computation* **4**(3) 239–273.
- [50] Fisher, Marshall L. 2004. The lagrangian relaxation method for solving integer programming problems. *Management science* **50**(12_supplement) 1861–1871.
- [51] Flood, M.M. 1956. The traveling-salesman problem. *Operations Research* **4**(1) 61–75.

- [52] Ford, L. R. Jr., D. R. Fulkerson. 1958. A suggested computation for maximal multi-commodity network flows. *Management Science* **5**(1) pp. 97–101.
- [53] Geoffrion, A. M. 1972. Generalized Benders decomposition. *Journal of Optimization Theory and Applications* **10**(4) 237–260.
- [54] Gilmore, P. C., R. E. Gomory. 1961. A linear programming approach to the cutting-stock problem. *Operations Research* **9**(6) pp. 849–859.
- [55] Gilmore, Paul C, Ralph E Gomory. 1961. A linear programming approach to the cutting-stock problem. *Operations research* **9**(6) 849–859.
- [56] Glockner, G.D., G.L. Nemhauser, C.A. Tovey. 2001. Dynamic network flow with uncertain arc capacities: Decomposition algorithm and computational results. *Computational Optimization and Applications* **18**(3) 233–250.
- [57] Goldstein, G. Energy planning and the development of carbon mitigation strategies. <http://www.etsap.org/reports/markal-irg.pdf>.
- [58] Gondzio, Jacek, Robert Sarkissian. 1996. Column generation with a primal-dual method. *Relatorio tecnico, University of Geneva* **102**.
- [59] Guan, Y., S. Ahmed, G.L. Nemhauser. 2009. Cutting planes for multi-stage stochastic integer programs. *Operations Research* **57** 287–298.
- [60] Haarbrucker, Gido, Daniel Kuhn. 2009. Valuation of electricity swing options by multistage stochastic programming. *Automatica* **45**(4) 889 – 899.
- [61] Held, Michael, Philip Wolfe, Harlan P Crowder. 1974. Validation of subgradient optimization. *Mathematical programming* **6**(1) 62–88.

- [62] Hobbs, B. F. 1995. Optimization methods for electric utility resource planning. *European Journal of Operational Research* **83**(1) 1 – 20.
- [63] Hobbs, B. F., M. H. Rothkopf, R. P. O’Neil, H. Chao. 2001. *The Next Generation of Electric Power Unit Commitment Models*. Kluwer Academic Publishers, Norwell, MA, USA.
- [64] ISO, California. 2010. 72-hour residual unit commitment (RUC). <http://www.caiso.com/27ae/27aebe3060d40.html>.
- [65] Jaillet, P. 1988. A priori solution of a traveling salesman problem in which a random subset of the customers are visited. *Operations Research* 929–936.
- [66] Kao, E.P.C. 1978. A preference order dynamic program for a stochastic traveling salesman problem. *Operations Research* **26**(6) 1033–1045.
- [67] Kelley, J.E.J. 1960. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics* 703–712.
- [68] Kenyon, A.S., D.P. Morton. 2003. Stochastic vehicle routing with random travel times. *Transportation Science* **37**(1) 69–82.
- [69] Kovacevic, Raimund M., Georg Ch. Pflug. 2014. Electricity swing option pricing by stochastic bilevel optimization: A survey and new approaches. *European Journal of Operational Research* **237**(2) 389 – 403.
- [70] Krauth, W., M. Mézard. 1989. The cavity method and the traveling-salesman problem. *Europhysics Letters* **8**(3) 213–218.
- [71] Lamont, A. 1997. User’s guide to the META*Net economic modeling system, version 2.0. Technical report, Lawrence Livermore National Laboratory.

- [72] Laporte, G. 2010. A concise guide to the traveling salesman problem. *Journal of the Operational Research Society* **61** 35–40.
- [73] Laporte, G., F. V. Louveaux. 1993. The integer L-shaped methods for stochastic integer programs with complete recourse. *Operations Research Letters* **13** 133–142.
- [74] Laporte, G., F. V. Louveaux, H. Mercure. 1994. A Priori optimization of the probabilistic traveling salesman problem. *Operations Research* **42**(3) 543–549.
- [75] Laporte, G., F.V. Louveaux. 1993. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters* 133–142.
- [76] Laporte, G., F.V. Louveaux, H. Ercure. 1994. A priori optimization of the probabilistic traveling salesman problem. *Operations Research* **42**(3) 543–549.
- [77] Laporte, Gilbert. 1992. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research* **59**(3) 345–358.
- [78] Laporte, Gilbert, Francois V. Louveaux. 1993. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters* **13**(3) 133 – 142.
- [79] Lee, C. W., S. K. K. Ng, J. Zhong, F. F. Wu. 2006. Transmission expansion planning from past to future. *IEEE PES Power Systems Conference and Exposition*. 257 – 265.
- [80] Lees-Miller, John D. 2016. Minimising average passenger waiting time in personal rapid transit systems. *Annals of Operations Research* **236**(2) 405–424.
- [81] Leipälä, T. 1978. On the solutions of stochastic traveling salesman problems. *Eur. J. Oper. Res.* **2** 291–297.
- [82] Levy, D., K. Sundar, S. Rathinam. 2014. Heuristics for routing heterogeneous unmanned vehicles with fuel constraints. *Mathematical Problems in Engineering* **2014**.

- [83] Li, S., D. Tirupati. 1994. Dynamic capacity expansion problem with multiple products: Technology selection and timing of capacity additions. *Operations Research* **42**(5) 958 – 976.
- [84] Louveaux, F. V. 1986. Multistage stochastic programs with block-separable recourse. *Stochastic Programming 84 Part II, Mathematical Programming Studies*, vol. 28. Springer Berlin Heidelberg, 48–62.
- [85] Louveaux, François V. 1986. *Multistage stochastic programs with block-separable recourse*. Springer.
- [86] Lübbecke, Marco E. 2010. Column generation. *Wiley Encyclopedia of Operations Research and Management Science* .
- [87] Lübbecke, Marco E, Jacques Desrosiers. 2005. Selected topics in column generation. *Operations Research* **53**(6) 1007–1023.
- [88] Luss, H. 1982. Operations research and capacity expansion problems: A survey. *Operations Research* **30**(5).
- [89] Mäkelä, Marko. 2002. Survey of bundle methods for nonsmooth optimization. *Optimization Methods and Software* **17**(1) 1–29.
- [90] Malcolm, S. A., S. A. Zenios. 1994. Robust optimization for power systems capacity expansion under uncertainty. *The Journal of the Operational Research Society* **45**(9) 1040 – 1049.
- [91] Marlow, D.O., P. Kilby, G.N. Mercer. 2007. The travelling salesman problem in maritime surveillance techniques, algorithms and analysis. *Proceedings of the International Congress on Modelling and Simulation* 684–690.

- [92] Marsten, RE, WW Hogan, Jacob Watson Blankenship. 1975. The boxstep method for large-scale optimization. *Operations Research* **23**(3) 389–405.
- [93] Massé, P., R. Gibrat. 1957. Application of linear programming to investments in the electric power industry. *Management Science* **3**(2) 149–166.
- [94] Matai, R., S. Singh, M.L. Mittal. 2010. *Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches*, chap. 1. Traveling Salesman Problem, Theory and Applications.
- [95] Mendoza, J. E., J. G. Villegas. 2013. A multi-space sampling heuristic for the vehicle routing problem with stochastic demands. *Optimization Letters* **7** 1503–1516.
- [96] Mirchandani, P. B. 1976. Shortest distance and reliability of probabilistic networks. *Computers & Operations Research* **3**(4) 347–355.
- [97] Mo, B., J. Hegge, I. Wangensteen. 1991. Stochastic generation expansion planning by means of stochastic dynamic programming. *IEEE Transactions on Power Systems* **6**(2) 662 – 668.
- [98] Muckstadt, John A, Sherri A Koenig. 1977. An application of lagrangian relaxation to scheduling in power-generation systems. *Operations research* **25**(3) 387–403.
- [99] Mukerji, R., H.M. Merrill, B.W. Erickson, J.H. Parker, R.E. Friedman. 1991. Power plant maintenance scheduling: optimizing economics and reliability. *Power Systems, IEEE Transactions on* **6**(2) 476–483.
- [100] Mulvey, John M., Andrzej Ruszczyński. 1995. A new scenario decomposition method for large-scale stochastic optimization. *Operations Research* **43**(3) pp. 477–490.
- [101] Mumford, C.L. 2013. New heuristic and evolutionary operators for the multi-objective urban transit routing problem. *Evolutionary Computation (CEC), 2013 IEEE Congress on*. 939–946.

- [102] Nie, Y., X. Wu. 2009. Shortest path problem considering on-time arrival probability. *Transportation Research Part B* **43** 597–613.
- [103] Nikovski, D., W. Zhang. 2010. Factored Markov decision process models for stochastic unit commitment. *Innovative Technologies for an Efficient and Reliable Electricity Supply (CITRES), 2010 IEEE Conference on*. Waltham, MA, USA, 28–35.
- [104] Ntaimo, L. 2010. Disjunctive Decomposition for Two-Stage Stochastic Mixed-Binary Programs with Random Recourse. *Operations Research* **58**(1) 229–243.
- [105] Ntaimo, L., S. Sen. 2008. Branch-and-cut algorithm for two-stage stochastic mixed-binary programs with continuous first-stage variables. *International Journal of Computational Science and Engineering* **3**(6) 232–241.
- [106] Oberlin, P., S. Rathinam, S. Darbha. 2010. Today’s traveling salesman problem: heterogeneous, multiple depot, multiple uav routing problem. *IEEE Robotics & Automation Magazine* **17** 70–77.
- [107] of Energy, U. S. Department. 2009. Recovery Act – Resource assessment and interconnection-level transmission analysis and planning. Funding Opportunity Number: DE-FOA0000068.
- [108] Orman, A.J., H.P. Williams. 2007. *A Survey of Different Integer Programming Formulations of the Travelling Salesman Problem*, vol. 9. Optimisation, Econometric and Financial Analysis, Springer Berlin Heidelberg, 91–104.
- [109] Pereira, M.V.F., L.M.V.G. Pinto. 1991. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming* **52**(1-3) 359–375.
- [110] Pflug, Georg C, Nikola Broussev. 2009. Electricity swing options: Behavioral models and pricing. *European journal of operational research* **197**(3) 1041–1050.

- [111] Polychronopoulos, G. H., J. N. Tsitiklis. 1996. Stochastic shortest path problems with recourse. *Networks* **27** 133–143.
- [112] Rajagopalan, S., M. R. Singh, T. E. Morton. 1998. Capacity expansion and replacement in growing markets with uncertain technological breakthroughs. *Management Science* **44** 12 – 30.
- [113] Rockafellar, R.T., S. Uryasev. 2000. Optimization of conditional Value-at-Risk. *Journal of Risk* **2**(3) 21–42.
- [114] Rockafellar, R.T., S. Uryasev. 2000. Optimization of conditional value-at-risk. *In Journal of Risk* **2**(3) 21–41.
- [115] Ruiz, P. A., C. R. Philbrick, E. Zak, K. W. Cheung, P. W. Sauer. 2009. Uncertainty management in the unit commitment problem. *IEEE Transactions on Power Systems* **24**(2) 642–651.
- [116] Ryan, J.L., T.G. Bailey, J.T. Moore, W.B. Carlton. 1998. Reactive tabu search in unmanned aerial reconnaissance simulations. *Proceedings of the 1998 Winter Simulation Conference* **1** 873–879.
- [117] Sarykalin S., Uryasev S., Serraino G. 2008. Value-at-risk vs. conditional value-at-risk in risk management and optimization.tutorials in operations research. *INFORMS* 270–294.
- [118] Savelsbergh, Martin. 1997. A branch-and-price algorithm for the generalized assignment problem. *Operations Research* **45**(6) pp. 831–841.
- [119] Schrijver, A. 2005. *On the history of combinatorial optimization till 1960*, vol. 12. Elsevier.
- [120] Schultz, R., S. Tiedemann. 2006. Conditional value-at-risk in stochastic programs with mixed-integer recourse. *Mathematical Programming* **105**(2) 365–386.

- [121] Sen, S., J. L. Hige. 2005. The c^3 theorem and a d^2 algorithm for large scale stochastic optimization: set convexification. *Mathematical Programming* **104** 1–20.
- [122] Sen, Suvrajeet, Lihua Yu, Talat Genc. 2006. A stochastic programming approach to power portfolio optimization. *Operations Research* **54**(1) 55–72.
- [123] Shahidehpour, M., H. Yamin, Z. Li. 2002. *Market Operations in Electric Power Systems*. John Wiley and Sons.
- [124] Sheble, G.B., G.N. Fahd. 1994. Unit commitment literature synopsis. *Power Systems, IEEE Transactions on* **9**(1) 128 –135. doi:10.1109/59.317549.
- [125] Sherali, H. D., W. P. Adams. 1994. A hierarchy of relaxations and convex hull characterizations for mixed integer zero-one programming problems. *Discrete Applied Mathematics* **52** 83–106.
- [126] Sherali, H. D., B. M. P. Fraticelli. 2002. A modification of benders’ decomposition algorithm for discrete subproblems: an approach for stochastic programs with integer recourse. *Journal of Global Optimization* **22** 319 – 342.
- [127] Sherali, H. D., A. L. Soyster, F. H. Murphy, S. Sen. 1984. Intertemporal allocation of capital costs in electric utility capacity expansion planning under uncertainty. *Management Science* **30**(1) 1 – 19.
- [128] Sherali, H. D., X. Zhu. 2006. On Solving discrete two-stage stochastic programs having mixed-integer first and second stage variables. *Mathematical Programming* **108** 597–616.
- [129] Shiina, T., J. Birge. 2003. Multistage stochastic programming model for electric power capacity expansion problem. *Japan Journal of Industrial and Applied Mathematics* **20** 379–397.

- [130] Short, W., N. Blair, P. Sullivan, T. Mai. 2009. ReEDS model documentation: Base case data and model description. National Renewable Energy Laboratory. http://www.nrel.gov/analysis/reeds/pdfs/reeds_full_report.pdf.
- [131] Sigal, C. Elliott, A. Alan B. Pritsker, James J. Solberg. 1980. The stochastic shortest route problem. *Operations Research* **28**(5) 1122–1129.
- [132] Silva, EL, M Morozowski, LGS Fonseca, GC Oliveira, ACG Melo, J Mello. 1995. Transmission constrained maintenance scheduling of generating units: a stochastic programming approach. *Power Systems, IEEE Transactions on* **10**(2) 695–701.
- [133] Singh, Kavinesh J., Andy B. Philpott, R. Kevin Wood. 2009. Dantzig-wolfe decomposition for solving multistage stochastic capacity-planning problems. *Operations Research* **57**(5) 1271–1286.
- [134] Sniedovich, M. 1981. Analysis of a preference order traveling salesman problem. *Operations Research* **29**(6) 1234–1237.
- [135] Takriti, Samer, John R Birge, Erik Long. 1996. A stochastic model for the unit commitment problem. *Power Systems, IEEE Transactions on* **11**(3) 1497–1508.
- [136] Toriello, A., W.B. Haskell, M. Poremba. 2013. A dynamic traveling salesman problem with stochastic arc costs URL http://www.optimization-online.org/DB_HTML/2012/09/3593.html.
- [137] Tuohy, A., P. Meibom, E. Denny, M. O’Malley. 2009. Unit commitment for systems with significant wind penetration. *IEEE Transactions on Power Systems* **24**(2) 592–601.
- [138] Van Slyke, R., R. J. Wets. 1969. L-shaped linear program with application to optimal control and stochastic linear programming. *SIAM Journal on Applied Mathematics* **17**(4) 638–663.

- [139] Van Slyke, Richard M, Roger Wets. 1969. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics* **17**(4) 638–663.
- [140] Verweij, B., S. Ahmed, A.J. Kleywegt, G. Nemhauser, A. Shapiro. 2003. The sample average approximation method applied to stochastic routing problems: a computational study. *Computational Optimization and Applications* **24**(2) 289–333.
- [141] Wallace, S. W., S.-E. Fleten. 2003. Stochastic programming models in energy. A. Ruszczyński, A. Shapiro, eds., *Stochastic Programming*, vol. 10. Elsevier, 637 – 677.
- [142] Wang, J., A. Botterud, V. Miranda, C. Monteiro, G. Sheble. 2009. Impact of wind power forecasting on unit commitment and dispatch. *8th Int. Workshop on Large-Scale Integration of Wind Power into Power Systems*. Bremen, Germany.
- [143] Wang, J., M. Shahidehpour, Z. Li. 2008. Security-constrained unit commitment with volatile wind power generation. *IEEE Transactions on Power Systems* **23**(3) 1319 – 1327.
- [144] Warren, P. B., A. George, A. Lamont, J. Stewart. 2010. SMART: A stochastic multi-scale model for the analysis of energy resources, technology and policy. Working paper, <http://www.castlelab.princeton.edu/Papers/SMARTAugust112010.pdf>.
- [145] Watanabe, I., N. Yamaguchi, T. Shiina, I. Kurihara. 2004. Agent-based simulation model of electricity market with stochastic unit commitment. *Probabilistic Methods Applied to Power Systems, 2004 International Conference on*. 403 – 408.
- [146] Xiao, J., B. S. Hodge, A. L. Liu, J. F. Pekny, G. V. Reklaitis. 2011. Long-term planning of wind farm siting in the electricity grid. To appear in *Proceeds of the 21st European Symposium on Computer Aided Process Engineering – ESCAPE 21*.

- [147] Yadlapalli, S., S. Rathinam, S. Darbha. 2012. 3-approximation algorithm for a two depot, heterogeneous traveling salesman problem. *Optimization Letters* **6** 141–152.
- [148] Zheng, Q. P., P. M. Pardalos. 2010. Stochastic and risk management models and solution algorithm for natural gas transmission network expansion and lng terminal location planning. *Journal of Optimization Theory and Applications* **147** 337–357.
- [149] Zheng, Q. P., S. Shen, Y. Shi. 2014. Loss-constrained minimum cost flow under arc failure uncertainty with applications in risk-aware kidney exchange Submitted.
- [150] Zheng, Q. P., J. Wang, P. M. Pardalos, Y. Guan. A new decomposition approach to stochastic security-constrained unit commitment problems in electrical power generation. *Submitted to Annals of Operations Research* .
- [151] Zheng, Qipeng, Andrew Liu. 2011. Transmission and generation capacity expansion with unit commitment - a multiscale stochastic model. INFORMS Annual Meeting, Charlotte, NC.
- [152] Zheng, Qipeng P., Siqian Shen, Yuhui Shi. 2015. Loss-constrained minimum cost flow under arc failure uncertainty with applications in risk-aware kidney exchange. *IIE Transactions* **47(9)** 961–977.
- [153] Zhou, Weiteng, Baoming Han, Dewei Li, Kai Lu, Qingjie Zheng. 2015. Dynamic schedule-based assignment model for beijing urban rail transit network with capacity constraints. *Transportation Research Board 94th Annual Meeting*. 15-2554.