Service Scheduling and Vehicle Routing Problem to Minimise the Risk of Missing Appointments

By

Chenlu Ji

Doctoral Thesis

Submitted in partial fulfilment of the requirements

for the award of

Doctor of Philosophy of Loughborough University

Jan 2019

Abstract

This research studies a workforce scheduling and vehicle routing problem where technicians drive a vehicle to customer locations to perform service tasks. The service times and travel times are subject to stochastic events. There is an agreed time window for starting each service task. The risk of missing the time window for a task is defined as the probability that the technician assigned to the task arrives at the customer site later than the time window. The problem is to generate a schedule that minimises the maximum of risks and the sum of risks of all the tasks considering the effect of skill levels and task priorities. A new approach is taken to build schedules that minimise the risks of missing appointments as well as the risks of technicians not being able to complete their daily tours on time.

We first analyse the probability distribution of the arrival time to any customer location considering the distributions of activities prior to this arrival. Based on the analysis, an efficient estimation method for calculating the risks is proposed, which is highly accurate and this is verified by comparing the results of the estimation method with a numerical integral method.

We then develop three new workforce scheduling and vehicle routing models that minimise the risks with different considerations such as an identical standard deviation of the duration for all uncertain tasks in the linear risk minimisation model, and task priorities in the priority task risk minimisation model. A simulated annealing algorithm is implemented for solving the models at the start of the day and for re-optimisation during the day.

Computational experiments are carried out to compare the results of the risk minimisation models with those of the traditional travel cost model. The performance is measured using risks and robustness. Simulation is used to

compare the numbers of missed appointments and test the effect of re-optimisation.

The results of the experiments demonstrate that the new models significantly reduce the risks and generate schedules with more contingency time allowances. Simulation results also show that re-optimisation reduces the number of missed appointments significantly. The risk calculation methods and risk minimisation algorithm are applied to a real-world problem in the telecommunication sector.

Keywords: Scheduling; Vehicle Routing with Time Windows; Stochastic Service and Travel Time; Risk Minimisation.

Acknowledgements

I would like to express my sincere gratitude and thanks to my supervisors Prof. Jiyin Liu and Dr. Rupal Mandania for the continuous support through my Ph.D. study and related research. Their patience, encouragement, immense knowledge and professional guidance support me in all the time of research and writing of the thesis. They have been tremendous mentors for me. I have been fortunate to have supervisors who cared so much about my work, and who responded to my questions and queries so promptly.

My sincere thanks also go to Dr. Gilbert Owusu, Dr. Mathias Kern, Dr. Anne Liret and Dr. Raphael Dorne, who provided me an opportunity to join their team as an intern, and gave access to their data and research facilities. Without their precious support it would not be possible to conduct this research.

I would also like to thank the School of Business and Economics, Loughborough University, for providing the funding, support and training that has made this research go smoothly.

Special thanks to my parents and friends who have always been there for me, offered unconditional love, support and encouragement, to fulfil my dream.

Contents

List of Figures

List of Tables

List of Abbreviations

ACA	Ant Colony Algorithm
AoU	Alphorn of Uncertainty
BST	British Summer Time
CCP	Chance-Constrained Programming
CDF	cumulative density function
CLT	Central Limit Theorem
CVRP	Capacitated Vehicle Routing Problem
FLSP	Functional Logistics Service Provider
GA	Genetic Algorithm
GAP	Generalised Assignment Problem
JSSP	Job Shop-Scheduling Problem
Km	kilometre
NN	Neural Networks
PDF	probability density function
RCPSP	Resource-Constrained Project Scheduling Problem
SA	Simulated annealing
SVRP	Stochastic Vehicle Routing Problem
SVRPTW	Stochastic Vehicle Routing Problem with time windows
TSP	Travelling Salesman Problem
TSPSC	Traveling Salesman Problem with Stochastic Customers
VRP	Vehicle Routing Problem
VRPB	Vehicle Routing Problem with Back-hauls
VRPPD	Vehicle Routing Problem with Pickup and Delivery
VRPSC	Vehicle Routing Problem with stochastic customers
VRPSCD	Vehicle Routing Problem with Stochastic Customers and Demands
VRPSD	Vehicle Routing Problem with stochastic demands
VRPTW	Vehicle Routing Problem with Time Windows

Chapter 1

Introduction

1.1 Background and motivation

Globalisation has brought consumers greater choices of goods and services with lower costs. At the same time, it also brings greater challenges for the providers of goods and services to stay profitable in a more competitive environment. Meeting customers' expectation and promised schedule visits is vital for the profit and reputation of service or goods delivery companies. Hence these companies need to continuously view their operation efficiency. Task scheduling is a common problem for most organisations regarding efficiency, *i.e.*, assigning resources including employees to perform tasks to optimise some criteria considering operational constraints. It is crucial to ensure that the available resources match the expected workload.

Many services or goods delivery companies may be influenced by one of the current challenges in logistics optimisation: the high degree of dynamics and uncertainty. We consider the problem where resources (*i.e.*, technicians) deliver a service at customer premises, for example, repairing a telephone line. Every customer is given a time window (*e.g.*, 8.00am to 10am) by the company at which a technician must start the service. If the job is not started by the end of this time window, the appointment is considered as missed and the company must pay for failing the appointment. Two main uncertainties need to be considered when tasks are assigned to technicians: the time which is taken to complete a given task and the travel time. Each technician drives a vehicle to customer locations to perform the tasks assigned to him/her. Thus, the problem carries features of the Vehicle Routing Problem (VRP) which is one of the major problems in the field of logistics and transportation. Dantzig and Ramser (1959) were the first to consider the optimum routing of a fleet of gasoline delivery trucks between a terminal and a large number of service stations supplied by the terminal. It generalises the well-known Travelling Salesman Problem (TSP) which has been studied extensively. VRP has become one of the typical combinatorial optimisation problems, and significant achievements have been made in both theoretical and practical aspects over the last several decades.

Research in VRP in the first stage focused on deterministic cases, it has since been extended to include constraints that align better with real-life applications. These extended versions include VRP with time windows (Fisher *et al*., 1997), the heterogeneous VRP (Golden *et al*., 1984), the open VRP (Li and Tian, 2006; Li *et al*., 2009), the VRP with backhauls (Deif and Bodin, 1984) and the multi-period VRP (Zäpfel and Bögl, 2008). However, these research works are mostly confined to deterministic models, which assume that all related information is known and determined before scheduling. In recent years, the social and economic operating environment has changed a lot. On the one hand, massive uncertain information appears since the number of economic activities increases dramatically. However, the rapid development of computer technology helps to reduce the chaos resulted from the uncertain information. It is urgent for people to utilise this information to create more wealth, which suggests that the strategies and the technical methods for dealing with the indefinite information directly decide the efficiency and profit level of an economic entity.

Furthermore, the growing interest in enhancing customer satisfaction has motivated researchers and businesses in building more customer-oriented models, for instance, taking time windows into consideration as an intrinsic component of workforce scheduling and VRPs (Ehmke *et al*., 2015). Therefore, service provision organisations have to increasingly focus on providing customer satisfaction and reassurance in the delivery process of offering good services or items. Meanwhile, most activities in real-world scheduling problems tend to be uncertain. For instance, at arrival onsite, a technician may realise the task is not matching his skills or tools, or tasks take longer or shorter than expected (Herroelen and Leus, 2005). Hence the environment in which services need to be delivered is inherently dynamic and subject to disruption in the workstack estimates as well as in the execution of jobs by workforce (Lesaint *et al.*, 2000).

Additionally, Real-life scheduling problems, for instance in domains such as workforce scheduling in the utilities or communications sectors, have to deal with large numbers of diverse and multi-skilled resources and usually many different types of work. These problems need to address different job priorities, *e.g.* in terms of the importance of work, whether it is appointed or not, and in which time window the work can be delivered, and varying resource availabilities and capabilities. Some of these characteristics, such as travel or task times, or whether a resource is correctly skilled to complete a task, are only approximately known in advance. Schedules are built based only on average, approximate characteristics without explicitly considering the underlying uncertainties risks service failure. For example, traditional scheduling approaches might well put a high priority task at the end of a technician's schedule if that reduces overall travel. However in real-life, this last task might become impossible to complete if there are any delays in previous tasks, and it might thus be better to move the high priority task to an earlier slot accepting longer travel. In order to meaningfully balance between various objectives, such as travel and job completion, in real-world scenarios, uncertainty considerations should be part of both the problem model and the scheduling approach. In this thesis, we present an approach to illustrate how risk can be incorporated into and considered by the scheduling algorithm.

1.2 Research framework and intended contributions

To begin with, a flow chart in Figure 1.1 is used to demonstrate the research framework. This will be followed by a description for each of the crucial steps.

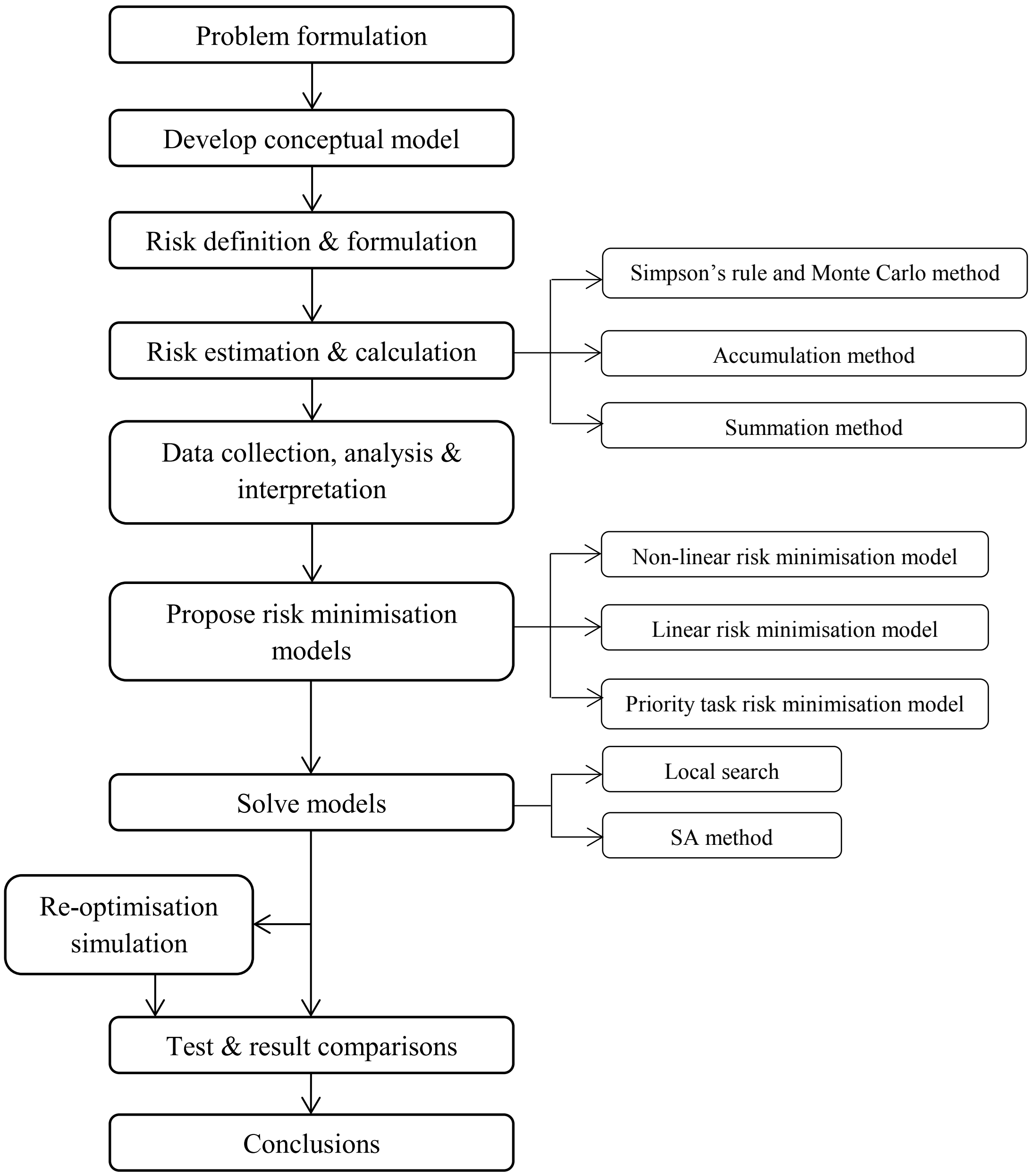


Figure 1.1 Research framework

1.2.1 Research problem

VRPs basically aim to find a way to visit a given set of customer locations using a fixed set of vehicles in such a way that a cost function, often the total distance or total travel time, is minimised. In the most basic version of this problem, each customer must be visited by precisely one vehicle, and each vehicle performs one trip, starting and ending at a depot location. The problem is then to decide for each vehicle which set of customer locations it visits, and in which order it visits them. Typically, a set of operational constraints must be fulfilled. These can limit the number or set of possible locations a given vehicle can visit, the time at which a given location is visited, the order in which a set of locations must be visited and so on.

The research problem in this thesis inherits the basic VRP constraints, but also has its distinctive features because it is about delivering services rather than goods to customer locations. The time spent at each customer location is significant for the technician to perform the service task while the time spent on travel is relatively short; there is no vehicle capacity constraint but there is a maximum daily worktime constraint; each task requires certain skills which can be performed by a subset of all the technicians; tasks may also have different priorities; there is an agreed time window for starting the service for each customer, and a risk of missing the time window exists due to uncertainties in service and travel times.

The aim of this research is to study the workforce scheduling and vehicle routing problem for service delivery from a new perspective, building schedules that minimise the risks of missing appointments as well as the risks of technicians not being able to complete their daily tours on time. A better schedule could help in improving the level of customer satisfaction. Therefore, the company may become more competitive and attract more customers.

1.2.2 Definitions and calculation methods

The risk in the research problem is defined as the probability distribution of the arrival time to any customer location considering the distributions of activities prior to this arrival. Based on the analysis, three methods for calculating the risk of missing the corresponding time window will be explored: a mathematical integral expression, a numerical calculation method and an efficient and highly accurate estimation method. These attribute to the first contribution of this research.

1.2.3 Models

The second contribution is to develop new workforce scheduling and vehicle routing models with the objective of minimising the risks and thus increasing the likelihood of successful service delivery, and to implement an efficient algorithm for solving the models and for re-optimisation. The re-optimisation framework runs the model multiple time during the day, *e.g.*, besides generating initial schedule at the beginning of the day, the schedule may be re-optimised at 12:00 (BST) and 16:00 (BST) based on the operation information gathered from technicians, in order to increase the success of arriving in time at customer sites within their given time window.

1.2.4 Experiments and analysis

The third contribution is to investigate through computational experiments the benefits of the risk models in comparison to the traditional travel cost model. The performance will be measured using risks, robustness and number of missed appointments. Simulation will be used to obtain the number of missed appointments. The effect of re-optimisation will also be tested via simulation. The risk calculation methods and risk minimisation models are applied to a real-world problem in the telecommunication sector.

1.2.5 Conclusions

The experiment results demonstrate that the new models significantly reduce the risks and generate schedules with more contingency time allowances. Simulation results show that re-optimisation dramatically reduces the number of missed appointments. The risk calculation methods and risk minimisation algorithm are applied to a real-world problem in the telecommunication sector.

Based on this research, two conference papers have been published, "Service Scheduling to Minimise the Risk of Missing Appointments" in the Proceedings of the Computing Conference 2017(IEEE technically sponsored), and "Incorporating Risk in Field Services Operational Planning Process" in the SGAI-AI 2018, Artificial Intelligence XXXV in LECT NOTES COMPUT SC.

1.3 An overview of the thesis

The remainder of this thesis is structured as follows.

Chapter 2 introduces the literature on the risks defined in several areas and disciplines, in particular in Vehicle Routing, Scheduling, and corresponding variants problems. Also, heuristic methods are reviewed in terms of the complexity in our problem.

Chapter 3 defines the risks and provides a thorough analysis of the risk, various methods to calculate it, and a comparison of these methods.

Chapter 4 presents several risk models in view of different business cost and valuation.

In Chapter 5, the local search and heuristic search methods used to solve these models are described.

Experiment results are presented in Chapter 6. The experiments compare the risk models with the traditional travel cost model using various performance measures.

Finally, Chapter 7 completes the thesis with the conclusions, summary of contributions, and some suggestions for future research.

Chapter 2

Literature Review

This chapter begins by exploring how risk is defined in different areas. It then defines the vehicle routing problems and describes how they have evolved over the years, followed by a review of scheduling problems related to our research. The main aim of these sections is to identify where our research fits in current literature. Additionally, for the complicated non-linear optimisation model in our problem, various heuristic methods are reviewed.

2.1 Risks

In risk management, the risk is defined as the effect of uncertainty on objectives (IRM, 2018). The deviation from the expected target is regarded as the effect, and the target could be positive or negative. Concerning objectives, they can be in various aspects, such as financial, health and safety, environmental goals, *etc*. The risk may be applied at different levels, for example, strategic, organisation-wide, project, product and process. It is often expressed as a combination of the consequences of an event (including changes in circumstances) and the associated likelihood of occurrence. Potential events and consequences are two critical features of the risk. Moreover, uncertainty is the state of, even partial information related to, understanding or knowledge of an event, its consequence, or likelihood (ISO, 2018).

2.1.1 Risks in different areas and disciplines

Risks exist in many areas. In enterprises, enterprise risks consist of credit risks, interest rate risks, liquidity risks, market risks, and operational risks.

Regarding medical devices, risks are associated with harm to people and damage to property or the environment. In the project management area, the risk refers to the likelihood that a project will fail to meet its objectives. A general project risk management process includes Identify Analyse, Plan Response, Monitor and Control. As for natural disasters, risks may arise from floods, earthquakes for example. In the information technology area, risks are the potential that a given threat will exploit vulnerable points of an asset or group of assets and then it may cause harm to the organisation. With the aim to understand better the risk in our problem, some literature is reviewed on risks in different areas.

2.1.1.1 Job scheduling

Branda *et al*. (2016) studied the fixed interval scheduling problem which is to find a machine assignment for jobs with fixed starting and working times. Risks in this problem are random delays in working times. Stochastic programming and robust colouring formulations were considered to tackle this problem. Moreover, for small simulated instances, CPLEX solver was used. While for larger instances, a Tabu search algorithm shows more efficient.

Jamili (2016) focused on a Job Shop-Scheduling Problem (JSSP) which is to assign ideal jobs to resources at particular times. The JSSP is a robust job shop scheduling problem under disruptions. In the research, he gave the definitions and calculations of buffer times and then utilises the buffer times to solve this problem. The objective for this problem is to minimise the makespan. In addition, two robustness indexes are considered to control the delay: one is $E\left(D_{ij}\right) < \lambda_{ij}$, where the expected delay of operation (i,j) is less than a pre-determined threshold λ_{ij} ; the other is $P\left(D_{ij} > \eta\right) < \gamma$, in which the probability of occurring a delay bigger than η is always smaller than γ.

2.1.1.2 Stochastic vehicle routing problems with time window

Ehmke *et al*. (2015) defined the risk as the probability that individual time window constraints are violated, and the objectives are still based on traditional routing costs. They considered the risk into constraints and defined different service levels. Therefore, they wanted to minimise the costs while ensuring service levels. A chance-constrained programming model was used to solve this problem: restrictions were placed on the probability that individual time window constraints are violated. Li *et al*. (2010) also used a stochastic programming model with recourse in terms of different optimisation criteria.

Andreatta *et al*. (2016) considered a real application related to the optimisation of ground handling operations, where the aircraft needs specific operations before/after departure/landing, and the specific ground service equipment are required by these operations. They defined the risk as a disruption if an equipment cannot serve an aircraft on time (for example because the previous aircraft is late, or the delay cumulated along the service route is sufficiently large, or because of a breakdown or other unforeseen events), so that a customer originally included in a route may be reached after the end of its time window. The objective is to minimise a weighted sum of the total distance travelled and the expected number of disruptions. They also used a recourse action which is to put a penalty in the objective function based on how much the expected arrivals at customers exceed the deadlines.

2.1.1.3 Pickup and Delivery Problems

Ghilas *et al*. (2016) studied the pickup and delivery problem with time windows, scheduled lines and stochastic demands. This problem concerns scheduling a set of vehicles to serve a set of requests. The expected request demands are known in distribution when planning, but are only revealed with certainty upon the vehicles' arrival. A part of the transportation plan can be carried out on limited-capacity scheduled public transportation line services. The risk of this problem arises from uncertain demands. They proposed a scenario-based sample average approximation approach for this problem. Meanwhile, an adaptive large neighbourhood search heuristic embedded into the sample average approximation method was used to generate good-quality solutions.

Elgesem *et al*. (2018) modelled a practical single-ship routing problem with stochastic travel times as a stochastic traveling salesman problem with pickups and deliveries. The risk in their model is related to the probability that the route length is within a threshold and the goal is to maximise the probability. They showed that the uncertainty is determined by the layout of the relevant terminals and their distance to the anchorage. They proposed a simulation method to address the particular sailing pattern and re-evaluation process to handle some error in the approximation.

2.1.1.4 Project management

Pantouvakis and Maravas (2013) discussed the time and cost uncertainty modelling in project management, and they reviewed the uncertainty of projects and programmes. In the goals, the uncertainty tends to lie within the remit of the owner and sponsor and in the methods with the remit of the steward, the project manager, and the contractor. Outside of the project, the general macroeconomic and political environment and the legal framework affects project outcomes. At the level of the project, machine reliability, labour and machine productivity, unpredictable conditions, and flaws in defining project scope are crucial in shaping the outcome.

Bai *et al*. (2016) discussed about managing small projects under uncertainties. The problem is when dealing with uncertain customer arrivals, a contractor has the incentive to accept multiple projects to keep his crew busy in order to string along his customers. The uncertainty is the unknown number of customers while the contractor wants to make good use of his crew. Also, the contractor faces the risk of customer abandonment and customer complaint. To quantify this trade-off, they proposed a queuing model to examine the optimal admission policy in terms of the maximum number of projects that a contractor should accept at any point in time.

In the financial management of projects, Zhang and Elmaghraby (2014) studied the uncertainty where the duration and the cost of activity are modelled as random variables. They proposed a new concept of "alphorn of uncertainty" (AoU) to describe the domain of cumulative cost variation throughout the life of a project, and also applied it to assess the project's financial status over time. Moreover, they concluded that AoU may be a promising method for the financial management of projects under uncertainty. They also revealed that financial status under uncertain conditions is not sensitive to an activity's choice of duration distributions or the form of cost functions. However, the financial status can be greatly affected by payment rules.

2.1.1.5 Supply chain

Considering the logistics service supply chain, the uncertainty lays in the operation time for Functional Logistics Service Providers (FLSPs) in a mass customisation service environment. Liu *et al*. (2015) considered a scheduling model of logistics service supply chain based on the mass customisation service and uncertainty of FLSP's operation time to minimise total scheduling costs, minimise the difference between the scheduled and actual time of each service process, and maximise the average satisfaction of FLSPs.

In the robust environmental closed-loop supply chain, the uncertainty is the cost parameters of the supply chain and demand fluctuations. In the paper from Ma *et al*. (2015), they tried to minimise the economic cost and the second objective function is to minimise the environmental influence.

2.1.1.6 Information systems

In information systems, security risks are caused by various interrelated internal and external factors. Security risk analysis mainly focuses on analysing vulnerabilities and threats to the information resources and deciding what countermeasures to take for reducing risk to an acceptable level. Feng *et al*. (2014) utilised the probability of the threats and the expected loss due to the vulnerability of those threats to measure these risks. A security risk analysis model was also proposed to identify the causal relationships among risk factors and analyse the complexity and uncertainty of vulnerability propagation.

2.1.1.7 Nature disaster

As data analysis develops, the studies on risks caused by natural disasters become more feasible and reliable. Liu *et al*. (2013) measured the risk as the exceedance probability distribution of multi-hazard risk. They addressed the risk assessment in China’s Yangtze River Delta by developing a simple and practicable multi-hazard risk assessment method, which uses information diffusion theory to overcome the difficulty arisen from a lack of historical or spatial data on the natural hazard-induced loss.

2.1.1.8 Risk management

Risk management is an important function in organisations as companies undertake increasingly complex and ambitious projects. Those projects should be executed successfully, in an uncertain and often risky circumstance. There is no unanimous definition of a risk, but it is common to define a risk materialising as the probability or likelihood for an event with negative or unlikely happening consequences (Hopkin, 2012).

A risk matrix is a widely used tool during risk assessment to define the level of risk by considering the category of probability or likelihood against the category of consequence severity. This is a simple mechanism to increase visibility of risks and assist management decision making (Cox Jr., 2009). Some research in the above disciplines also refers the probability as the risk, so the risk is the probability of late arrivals as the basic definition in our problem. Once the distribution of late arrivals can be obtained, the risk would be well defined and calculated.

2.1.2 Differences

The problems in the above areas discussed may have some similarity to our problem or at least may help spark some thoughts. We summarise their similarity and difference with our research. Our research focuses on minimising the risk of missing an appointment with considering the time window for each customer. This is different from the vehicle routing problem with time windows where the objective is based on traditional routing costs. In the pickup and delivery problem they also consider time windows and the demand is stochastic but the times spent at customers' sites are short. Job scheduling problems consider disruptive events and the propagation of delay on tasks, but they generally do not consider travel time or time windows. In project management and supply chain, the interpretation of risk differs from ours. More focus is placed on the outcome of projects at financial or timescale level from personal, organisational or liability risk.

Additionally, differed from analysing the reasons and factors that cause and affect risks profoundly, this research study attributes the risk of missing appointments to the uncertain task durations and travel times. These uncertainties are obtained from the duration and travel time data, therefore, it is not a focus on investigate the reasons and factors that give and influence distributions of such duration and travel time.

In the next section we discussed the origins of VRP and its variants, and also described how features of our research problems relate to some of the existing literature.

2.2 The vehicle routing problems

2.2.1 Variants of the VRP

It has been 60 years since Dantzig and Ramser (1959) proposed a mathematical programming formulation and algorithms for the truck dispatching problem, which is commonly regarded as the first proposal of the VRP. VRP can be treated as a generalisation of the TSP, and over these years, dozens of different versions of the VRP have been introduced, and as the utilisation of optimisation packages have improved, hundreds of models and approaches have been proposed for the VRP.

2.2.1.1 Capacitated and distance-constrained VRP

Capacitated VRP (CVRP) (Fischetti *et al*., 1994) is the fundamental version of the VRP, where all the customers correspond to deliveries, and the demands are deterministic, known in advance, and may not be split. The vehicles are identical and based at a single central depot, and only the capacity restrictions for the vehicles are imposed. The objective is to minimise the total cost to serve all the customers. It could be described as follows.

Let $G = (V, A)$ be a complete graph, where $V = \{0,1,\cdots,n\}$ is the vertex set and A is the arc set. Vertices $i = 1,2,\cdots,n$ correspond to the customers while vertex 0 stands for the depot. Let c_{ij} represent the travel cost from i to j. If the cost matrix c is asymmetric, then the corresponding problem is called asymmetric CVRP. Otherwise, we have $c_{ij} = c_{ji}$, the problem becomes symmetric CVRP. Let d_i to be the known demand of the customer $i(i = 1,2,\cdots,n)$, and the demand of the depot is denoted by $d_0 = 0$. A set of K identical vehicles are available, each with capacity C. Assume that K is not smaller than K_{min}, where K_{min} is the smallest number of vehicles needed to serve all the customers. The capacitated VRP is to find K with minimum cost, and such that

1. Each vehicle route visits the depot;
2. Each customer vertex is visited by exactly one vehicle;
3. The sum of the demands of the vertices visited by the same vehicle does not exceed the vehicle capacity C.

The Distance-Constrained VRP is a VRP where the capacity constraint is replaced by a maximum length (or time) constraint. Let t_{ij} be the length of the arc (i,j). Moreover, a service time s_i can be added to the travel times of the arcs, therefore, t_{ij}=$t'_{ij} + \frac{s_i}{2} + \frac{s_j}{2}$, where t'_{ij} is the original length of the arc(i,j).

Generally, the length matrix is used as the cost which means $c_{ij} = t_{ij}$ for all (i,j). Therefore, the objective of the model is to minimise the total length of the travel routes, as well as the service time if it is considered.

2.2.1.2 VRP with time windows

The VRP with time windows (VRPTW) is an extension of the CVRP (Solomon, 1987), where capacity constraints are imposed and a time interval $[a_i, b_i]$, called a time window, is associated with each customer i. The service must start within the time window. The VRPTW is to find K with minimum cost, and such that

1. Each vehicle visits the depot;
2. Each customer vertex is visited by exactly one vehicle;
3. The sum of the demands of the vertices visited by the same vehicle does not exceed the vehicle capacity C;
4. For each customer i, the service begins within the time window $[a_i, b_i]$, and the vehicle stops for s_i time instants.

2.2.1.3 VRP with back-hauls

The VRP with Back-hauls (VRPB) is an extension of the CVRP (Goetschalckx and Jacobs-Blecha, 1989), where the customer vertices are partitioned into two sets. The first subset L contains n line-haul customers who require a given quantity of product to be delivered. The second subset B, contains m back-haul customers who have a given quantity of inbound product must be picked up. Customers are numbered so that $L = \{1, \cdots, n\}$ and $= \{n+1, \cdots, n+m\}$. In the VRPB, all the line-haul customers must be served before any back-haul customer. The VRPB is to find K with minimum cost, and such that

1. Each vehicle visits the depot;
2. Each customer vertex is visited by exactly one vehicle;

3. The sum of the demands of the line-haul customers visited by the same vehicle does not exceed the vehicle capacity C, and the total pickup quantity of the backhaul customers visited by the same vehicle also does not exceed C;
4. In each vehicle route all the line-haul customers precede the backhaul customers.

2.2.1.4 VRP with pickup and delivery

In the basic version of the VRP with Pickup and Delivery (VRPPD) (Savelsbergh and Sol, 1995), each customer i is associated with two quantities d_i and p_i, representing the demand of homogeneous commodities to be delivered and picked up at customer i respectively. The demand of goods from customer i may be met by the goods picked up from the other customer, and the goods picked up at this customer i also can be delivered to another customer. For each customer i, O_i denotes the vertex that is the origin of the delivery demand, and D_i denotes the vertex that is the destination of the pickup demand. The VRPPD is to find K with minimum cost, and such that

1. Each vehicle visits the depot;

2. Each customer vertex is visited by exactly one vehicle;

3. The current load of the vehicle must be nonnegative and may never exceed the vehicle C;

4. For each customer i, the customer O_i, if different from the depot, must be served by the same vehicle and before customer i;

5. For each customer i, the customer D_i, if is different from the depot, must be served by the same vehicle and after customer i.

2.2.2 The stochastic VRPs

Real life is always filled with much uncertain dynamic information, and with the economic development, this uncertain information generates an increasingly significant impact on business activities. Stochastic programming is a framework for modelling optimisation problems that involve uncertainty, whilst deterministic optimisation problems are formulated with known parameters. However, real-world problems almost invariably include some unknown parameters. Using the historical data, the statistical information about these parameters may be obtained. Thus, stochastic programming models take advantage of the fact that probability distributions governing the data are known or can be estimated, so that the goal here is to find a feasible solution for all such data and maximise the expectation of some function of the decisions and the random variables. More specifically, such models are formulated, solved numerically or analytically, and investigated in order to provide useful information to a decision-maker.

The Stochastic VRP (SVRP) is the VRP with some random elements of the problem. Typical examples are stochastic customer demands and stochastic travel times. Another kind of SVRPs has stochastic customers. In this case, each customer has a probability of being present. Sometimes, more intricate problems combine the stochastic demands with stochastic customers. These variants of the SVRPs are discussed in the following subsections.

The SVRP is different from the deterministic VRP in several fundamental aspects. The concept of a solution is different, several fundamental properties of the deterministic VRP do not hold in the stochastic VRP anymore, and solution methodologies are relatively more complicated (Gendreau *et al.*, 1996). The SVRP is often considered as computationally intractable, due to the fact that it combines the features of stochastic and integer programs.

However, the SVRP represents the real problems better. Thus, heuristics and metaheuristics for solving the SVRP have become popular in recent years. Many computational algorithms have been proposed for tackling this kind of problems.

The SVRP is a branch of the framework in stochastic programming. Commonly stochastic programming is modelled in two phases. In the first phase, a priori solution is generated. In the second phase, the realisations of the random variables become known, and then a recourse or corrective action could be applied to the solution generated in the first phase. The recourse usually encloses a cost or a saving that may have to be considered when designing the first stage solution.

2.2.2.1 The VRP with stochastic demands

Most SVRP studies focus on the VRP with stochastic demands (VRPSD) where customer demands are illustrated by random variables. More specifically, each d_i is replaced by a random variable ξ_i. The first stage solution to this problem would consist of a set of m vehicle routes so that each customer site is visited exactly once. After the first stage solution has been determined, the actual demands are revealed. It may then be impossible to implement the first stage solution as planned since the total demand of a route may exceed the capacity, *i.e.*, route failures may occur. A possible second stage policy would be to follow each route as planned until the vehicle capacity becomes attained or exceeded, return to the depot to unload, and then resume collections at the customer on the planned route where route failure occurred. In this case, the recourse action consists of performing a return trip to the depot.

Tillman (1969) was the first to propose an algorithm for the VRPSD based on the Clarke and Wright saving algorithm (Clarke and Wright, 1964). A second major seminal work is due to Stewart and Golden (1983), in which a chance-constrained model and two recourse models are presented. The chance-constrained model, in which some constraints hold at least with a predefined probability, designs the route sets and minimise transportation costs under stochastic chance constraints. The first of the two recourse models consider the probability of exceeding the vehicle capacity as a penalty, while the second recourse model uses a penalty proportional to the expected demand in excess of the vehicle capacity. Bertsimas (1988) also made a contribution to the study of the VRPSD. He derived several theoretical properties, lower and upper bounds for different strategies, and asymptotic theorems for the problems.

Laporte *et al*. (1989) considered the depot location as a decision variable. Their work deals with the problem with more general demand distributions. Properties and formulations for the recourse version of the problem have been studied by Dror *et al*. (1993), Louveaux and Laporte (1990), Bastian and Rinnooy Kan (1992), Bertsimas (1992) and Dror (1993). Heuristics are described in the works of Dror and Trudeau (1986), Bouzaïene-Ayari *et al*. (1993), and Dror *et al.* (1993).

2.2.2.2 The VRP with stochastic customers

In the VRP with stochastic customers (VRPSC), customers are present with some probability p_i, while customer demands are usually assumed to be deterministic. The VRPSC is an extension of the Traveling Salesman Problem with Stochastic Customers (TSPSC). To tackle this problem, in the first stage the Hamiltonian path through vertices for each vehicle is generated, and the set of present vertices is revealed. Moreover, the vehicle capacity must be considered, and it is necessary for each vehicle to return to the depot. In the second stage solution, absent customers are skipped. Waters (1989) gave three alternatives for the customers who do not want to be visited in a particular period: continue with the same planned routes; bypass the absent customers, and use semi-fixed routes; move to variable routes by using entirely new schedules. Laporte *et al*. (1994) proposed an exact algorithm based on the Integer L-Shaped Method (Laporte and Louveaux, 1993) which is capable of solving instances of size $n < 50$. Few studies focus on the problem with stochastic customers, because as the technology is improving, it becomes easier to know the omitted customers before scheduling.

2.2.2.3 The VRP with stochastic customers and demands

The VRP with Stochastic Customers and Demands (VRPSCD) combines the VRPSC and the VRPSD. Jézéquel (1984), Jaillet (1988), Trudeau and Dror (1992) all worked on the problem in the early stage. The definition proposed by Bertsimas (1992) seems the most interesting. In a first stage, one determines a set of routes starting and ending at the depot and visiting each customer exactly once. The set of customers with zero demand (absent customers) is then gradually revealed, but the actual demand of every remaining customer becomes known only when the vehicle arrives at the customer's location. In the second stage, the first stage routes are followed as planned, with the following two exceptions: (1) any absent customer is skipped; (2) whenever the vehicle capacity becomes exceeded, it returns to the depot to unload and resumes collections starting at the last visited customer. If for any customer the vehicle capacity becomes precisely attained, the vehicle then returns to the depot and resumes collections at the next present customer along its route.

Gendreau *et al*. (1995) were the first to propose an exact algorithm for this challenging problem based on the Integer L-Shape algorithm. In terms of heuristic methods, Gendreau *et al.*, (1996) proposed a Tabu search heuristics for this problem. By comparing with known optimal solutions on problems whose sizes vary from 6 to 46 customers, it was found that this heuristic method generates an optimal solution in 89.45% of cases, with an average deviation of 0.38% from optimality.

2.3 The stochastic VRP with time windows

With increasing customer expectations and as customer-oriented business models develop, the problem with delivery time windows has drawn much more attention in the schedule of delivery routes. In this kind of problem, a time window $[a_i, b_i]$ constraint is associated with each customer. There have been a considerable number of recent papers on how to solve the VRPTW, including Baldacci *et al* (2012), Vidal *et al*. (2013) and Hashimoto *et al*. (2013).

As the uncertainty in scheduling draws more attention, the consideration of time windows and stochastic demands or travel times in vehicle routing problem has become more applicable in recent years. Ong *et al*. (1997) suggested a selection criterion and a sequential heuristic. Jabali *et al*. (2012) studied a variant for the VRP where customer demands are stochastic, and demands are revealed upon arrival at customer locations. A failure occurs when a vehicle reaches a customer and does not have sufficient capacity to collect the realised demand. They formulated the VRPSD as a two-stage stochastic programming model and solved it using an integer L-shaped exact algorithm, in order to minimise the sum of the planned routes cost and the expected recourse cost.

Laporte *et al*. (1992) were the first to consider stochastic service and travel times in the VRP model. They restrict the total duration along a route for each vehicle under a given level instead of considering time windows. They presented general branch and cut algorithms for two models: a chance-constrained model where the objective is to minimise planned route costs while limiting the route duration within a given threshold; a recourse model where the objective is to minimise route costs and expected penalty costs. Lambert *et al*. (1993) modelled the collection of cash and negotiables between banks and branches as a vehicle routing problem with constraints. As well as the deterministic case, they also investigated the case with stochastic travel times and solved the integer mathematical programs with a heuristic method.

Kenyon and Morton (2003) considered SVRP that plan optimal vehicle routes with random travel and service times. Contrary to other researches where the objective is to minimise total travel costs, their models' objective functions depend on the completion time. They presented a branch-and-cut approach for small sample space and a Monte Carlo sampling-based solution procedure when the cardinality of the sample space is large. Mazmanyan and Trietsch (2014) addressed the stochastic TSP involving minimising the due date in the objective with respect to a service level constraint, assuming that the distribution of the tour length is normal or lognormal. They provided effective heuristics to solve the problems.

Some papers discussed the reliability of routes. Cook and Russell (1978) used a simulation to examine the suitability of deterministically-generated routes for the SVRP with uncertain demand and travel times. Their model aimed to minimise total expected travel times. Lecluyse *et al*. (2009) introduced the variability of traffic flows into the VRP problem, and they used a lognormal distribution in their experiments. As more risk-taking behaviour is taken into account, the optimal route shows a slightly longer travel time, but is more reliable. They also proposed a balance between the average travel time and variance in the objective function in terms of different risk tolerance planners.

Lee *et al*. (2012) investigated the VRP where the goal is to minimise travel time while considering uncertain travel time and demand, and customer deadlines (*i.e.*, a late time window). They defined the travel time and demand in uncertainty sets and achieved a robust solution by utilising a Dantzig-Wolfe decomposition approach (Dantzig and Wolfe, 1960) and a dynamic programming algorithm. Agra *et al*. (2013) modelled a realistic VRP in maritime transportation where travel times belong to an uncertainty polytope, hence their research yields a robust optimisation problem. They contributed to reducing the number of extreme points of the uncertainty polytope, and implemented a cutting–plane algorithm to solve the robust instances easily.

A number of authors used a recourse model for the stochastic VRP with time windows (SVRPTW), where arrival times before and after the customer time windows are discouraged through penalties in the objective instead of hard constraints. Taniguchi *et al*. (2001) modelled the problem to minimise the total routing cost, which consists of service and vehicle running cost, travel time between customers, waiting time and delay time for customers. They represented travel times with the lognormal distribution and solved the model with a dynamic traffic simulation approach. Ando and Taniguchi (2006) proposed the time windows-probabilistic model for VRP where the uncertainty of travel times is considered. The fixed cost of used vehicles, operation costs and early arrival and delay penalties at customers are studied in the model, as well as the CO_2, NO_X and Particle Materials emissions. Given a linear two-piecewise function as the penalty, combined with the probability of arrival time, they obtained the penalty distribution of early arrival and delay according to arrival time. They used probe vehicle data to estimate travel times and a Genetic Algorithm (GA) to solve the problem.

Taş, *et al.* (2013) studied a VRP with soft time windows and stochastic travel times, and their model consists of transportation costs, which is not only the distance travelled and a number of vehicles, but also the drivers' total expected overtime, and service costs based on early and late arrivals. They provided a Tabu search algorithm to deal with the problem. Another paper of Taş, *et al*. (2014) investigated a VRP with soft time windows and stochastic travel times where soft time windows allow early and late servicing at customers by incurring some penalty costs and uncertainty of travel times follow Gamma distributions. The objective is the same as the previous research. They applied a column generation procedure to solve it.

Russell and Urban (2008) studied a VRP with soft time windows where travel times are random variables. They assumed shifted Gamma distributed travel time and a variety of penalty structures. They also developed a Tabu search metaheuristic to tackle the problem. Yan *et al*. (2014) studied the SVRPTW for cash transportation and employed the time-space network flow technique to formulate the potential routes of vehicles. Their objective considered the operating costs as well as the unanticipated penalty cost concerning early and late arrival situations. Ehmke and Mattfeld (2011) provided information models to represent the traffic network in terms of time-dependent travel time data sets to improve the efficient and reliable vehicle routing optimisation in urban areas. They used data mining to filter sophisticatedly and aggregate travel data.

Some researchers have investigated the problem using a chance-constrained model. Li *et al*. (2010) studied stochastic travel and service times, as well as time windows. They considered both a chance-constrained and a recourse model. In their Chance-Constrained Programming (CCP) model, driver duration and time windows are expected to be feasible with a given confidence level, and a Tabu search based heuristic was applied to solve this model. In terms of the recourse model, the priori set of vehicle routes is determined then the expected correction costs of route failure are obtained after a stochastic simulation realising the travel and service time.

The objective of the model is to minimise the number of vehicles used and the total travel time. Then the recourse models focus on how much the expected arrival at every customer exceed the deadline and regard it as a penalty to put it in the objective function. Branda (2014) applied sample approximation technique to stochastic programming problems with mixed-integer bounded sets of feasible solutions and chance constraints. The VRP problem they studied is with time windows, random demand and random travel times.

The problems most closely related to ours are reviewed next. Chang (2005) who proposed a nonlinear stochastic integer program with recourse formulates the VRP with time windows and uncertain demands, which is similar to the task duration in our problem. The objective of his model was to minimise the total cost of the first-stage solution and the expected recourse cost of the second-stage solution. The total cost of the first-stage problem includes the total travel cost for all routes and the total waiting cost for all customers, while in our model we consider the risk of missing appointments in the objective. When a vehicle capacity is attained or exceeded, recourse actions are needed and recourse costs incurred in order to finish the planned route schedules. He applied the integer L-shaped method to develop a heuristic for this problem. Lei *et al*. (2011) then developed an adaptive large neighbourhood search metaheuristic for this problem. They used the modified Solomon benchmark instances in their experiments, and the computational results show that their approach is practical and superior to others.

Taş (2013) studied a VRP associated with real-life environments where travel time is stochastic with a known probability distribution, and in our research the travel time is assumed to follow normal distributions. His model aimed to balance transportation costs and service costs and he developed a new solution procedure based on Tabu search. Additionally, a VRP where travel times are both time-dependent and stochastic and a VRP where the time window is relaxed with respect to a customised percentage were investigated. He focused on the uncertainty of the travel time whereas we consider both the stochastic task duration and the uncertain travel time.

Jula and Dessouky (2006) investigated a stochastic TSP with time windows where travelling times and service times are stochastic. They developed a methodology to estimate the mean and variance of arrival time at each customer location. They constructed an acceptable route if the probability of arriving at each customer location in the route within their time window is greater than the service level and ran simulations to show the efficiency of the algorithm. In our research, the probability of arriving at the customer site is defined as the risk. Then based on this estimated arrival times, a dynamic programming approximated algorithm is proposed to minimise the expected travel cost along the route.

Chang *et al*. (2005) studied the VRP considering time windows and uncertain travel times which is similar to our problem. And their routing problem is for hazardous materials shipments, so the uncertain attributes include not only the travel time but also risk-related measures. They investigated a new algorithm that propagates means and variances of the uncertain travel times along the TSP routes according to the convolution of a normal distribution. They also argued that even the distribution of the arrival time is not normal, as the paths extended more links, the Central Limit Theorem suggests the arrival time will become normal distributed. One more important assumption they considered was "no waiting" which means only the upper bound of the time window are considered as a constraint. Whereas in the paper of Chang *et al*. (2009) they considered the waiting time that the vehicle arrives before the lower bound of the time window as “wasting resource” in the stochastic TSP with hard time windows. We consider the lower bound of the time window when dealing with the start time and consider the upper bound of the time window when discussing the risk. Chang *et al*. also provided an efficient heuristic for this multi-objective problem.

Ehmke and Campbell (2014) discussed the delivery mechanism in congested metropolitan areas, and they argued that it is valuable to access additional information on travel time variation in order to model the lateness and its propagation properly. Consideration of travel time with time-dependent mechanism outperformed the one of non-time-dependent. They concluded that a simple buffer approach is suitable to reduce lateness for downtown areas while a fixed buffer is not efficient to model the traffic condition in suburban areas. Ehmke *et al.* (2015) investigated the VRP with time windows and carefully considered the start service time and the arrival time distribution changes because of the time windows, for every customer. In our research, we also discussed how the time windows affect the start service time and then the arrival time for following tasks. But they still used a chance-constrained approach, in which the objective remains based on traditional routing costs, while we consider risks in the objective.

In a recent paper by Jaillet *et al*., (2016) considered a series of routing problems with deadlines, which is similar to the upper bound of a time window in our model, and with stochastic travel times. They introduced “lateness index” to measure the deadline violation level of a given criterion, in order to handle risk and ambiguity caused by the uncertain travel times. The aim of their routing optimisation is to minimise the lateness index. They developed an exact algorithm involving Lagrangian relaxation and Benders decomposition to deal with the problem. Damm *et al*. (2016) considered the assignment of a series of service tasks to a group of technicians, the priority values associated with tasks, different skills and working hours for the technicians increase the complexity of the problem, we also consider these features in our problem. They developed a heuristic method and a customised biased random-key genetic algorithm.

Overall, most VRP models focus on routing costs, or some may consider the cost of overall customer service delays, for example, the delay for each worker that he might not finish his work in the whole day, which is related one of the risks in our model. However, only few researches proposed models that focus on the service level at individual customers. Additionally, the job operation time is short to neglect for VRP because most of the jobs are delivery and pick-up works, while the task operation time is a crucial feature in our problem. Hence this is the reason why we investigate risks on the customer level to fill the research gap.

Our problem not only decides vehicle routes *i.e.* as a SVRPTW, but also decides the schedule of customer tasks to vehicles/technicians and hence can be viewed as a scheduling problem especially because the technicians spend a major part of the time in a day on executing the service tasks. A brief literature review on scheduling problem related to our research follows in the next section.

2.4 Scheduling problems

Robust optimisation is widely used in the project scheduling problem with uncertain activity durations, but there are no time windows in this problem. Bruni *et al*. (2011) addressed the Resource-Constrained Project Scheduling Problem (RCPSP) to minimise the project make-span, with uncertain activity durations and deterministic renewable resources. The uncertain duration is represented by independent random variables with known cumulative probability distribution function, which is similar to the assumptions for task duration in our model. Bruni *et al*. (2017) proposed an adaptive robust optimisation model to obtain the resource allocation decisions which minimise the worst-case make-span, under uncertain activity durations subject to interval uncertainty, thus the level of robustness is controlled by a protection factor. They also adopted a tailored decomposition approach to solve the problem.

Wang *et al.* (2018) discussed a job-shop scheduling problem with uncertain working times. They also established a robust optimisation model based on bad scenarios for risk-averse decision makers. They applied a problem-specific neighbourhood structure in a hybrid local search algorithm which combines the Simulated Annealing method and Tabu search. Chakrabortty *et al*. (2017) improved the objective for the RCPSP with stochastic activity durations to minimise both the project make-span and the sum of deviation penalties of all activity uncertainties. They also called the perturbations in the scheduling as risks as we do. They developed six different heuristic approaches based on the robust optimisation concept to seek feasible and high-quality solutions.

Drwal (2018) addressed single machine scheduling problems with the assumption that completion due-dates are not known precisely and only available at time intervals. The objective of their model is to minimise the weighted number of late jobs, as we consider late jobs in our simulations. And their model can be converted to a min-max regret problem to find a robust solution. It is usually applied in the preparation of a schedule for medical staff at a hospital, where the success of treatment depends on whether a patient is treated on time and the due date for the effectiveness of the treatment is uncertain.

Overall, most researchers consider uncertain activity durations in their scheduling models as we do in our model, but they rarely consider time windows and travel times because jobs are usually carried out on machines at the same site, rather than delivery jobs.

To sum up, our problem can be considered as an extension of the VRP which is NP-hard. When the number of customers is large, solving the problem optimally will require impractical computation time. In addition, the characteristics of the risk force us to use a non-linear model if we would like to consider all the complicated factors in our model. However, with the approximation method for risk distributions, we can easily calculate the risks for any given schedule. Therefore, heuristic search methods may be applied to find near-optimal solutions to the problem. The next section reviews heuristic methods used for related VRP variants.

2.5 Heuristic methods for related VRPs

Heuristics for solving VRPs can be broadly classified into two categories: constructive heuristics, including two-phase heuristics, and improvement methods. Constructive heuristics build a feasible solution while keeping eyes close on solution cost, but they do not contain an improvement phase. Two main techniques are used for constructing VRP solutions: merging existing routes by a savings criterion and assigning vertices to vehicle routes by an inserted cost. In terms of two-phase heuristics, they are divided into two classes: cluster-first, route-second methods and route-first, cluster-second methods. In the first case, vertices are first organised into some feasible clusters, and a vehicle route is constructed for each of them. In the second case, firstly a tour is built on all vertices, and then it is segmented into feasible vehicle routes. Lastly, improvement methods focus on upgrading any feasible solution by performing a sequence of edge or vertex exchanges within or between vehicle routes. Metaheuristics are also improvement methods.

2.5.1 Constructive heuristics

2.5.1.1 Clarke and Wright Savings Algorithm

The Clarke and Wright savings algorithm (Clarke and Wright, 1964) is one of the best-known heuristic methods for the VRP and remains widely used in practice nowadays. It is based on the notion of savings. A feasible solution consists of n back and forth routes between the depot and a customer. At any given iteration, two routes $(v_0, \ldots, v_i, v_0)$ and $(v_0, v_j, \ldots, v_0)$ can be merged into a single route $(v_0, \ldots, v_i, v_j, \ldots, v_0)$ whenever this is feasible, thus generating a saving $s_{ij} = c_{i0} + c_{0j} - c_{ij}$. There are two kinds of saving algorithms, a sequential version and a parallel version. In the sequential version, exactly one route is built at a time, excluding routes with only one customer, whereas in the parallel version, more than one route may be built at a time. In the first step of the algorithm, the savings for all pairs of customers are calculated, and all pairs of customer points are sorted in descending order of the savings. Secondly, each pair of points is considered at a time from the top of the sorted list of point pairs. When a pair of points (v_i, v_j) is considered, the two routes that visit v_i and v_j are combined if they are feasible.

Cordeau *et al*. (2002) pointed out that the parallel version is much better in practice. Moreover, the Clarke and Wright algorithm can be widely used because of its simplicity. However, this method seems to be time-consuming since all savings need to be computed, stored and sorted (Laporte *et al*., 1992). When implementing the savings heuristic, computing the maximum saving value should draw much attention. The calculation procedure of the Clarke and Wright savings algorithm is fast and very flexible, but sometimes the solution may be found very far from the optimal solution. Moreover, due to the mess of the un-routed points and the difficulty to the combination of boundary points, the Clarke-Wright algorithm is only suitable for solving small, simple VRP.

2.5.1.2 Sweeping Algorithm

The sweeping algorithm was popularised by Gillett and Miller (1974), which is in the family of cluster-first, route-second methods. This algorithm considers the VRP from the view of planar instances. Initially, feasible clusters are formed by rotating a ray centred at the depot. Then a vehicle route is generated for each cluster by solving a TSP. Some further implementations include a post-optimisation phase in which vertices are exchanged between adjacent clusters, and routes are re-optimised. A simple implementation of this method is as follows. Assume each vertex i is represented by its polar coordinates (θ_i, ρ_i), where θ_i is the angle and ρ_i is the ray length. Assign a value $\theta_i^* = 0$ to an arbitrary vertex i^* and compute the remaining angles from $(0, i^*)$. Rank the vertices in ascending order of their θ_i. In the first step, choose an unused vehicle k. Secondly, starting from the unrouted vertex having the smallest angle, assign vertices to vehicle k as long as its capacity or the maximal route length is not exceeded. In tightly constrained dynamic VRPs, 3-opt may be applied after each insertion. If un-routed vertices remain, go to the first step. Finally, optimise each vehicle route separately by solving the corresponding TSP (exactly or approximately).

One good feature of sweep algorithm is intuitive, but due to the fact that the sweep algorithm for solving VRP is not convergent definitely, the calculation process and results do not show the advantage over other algorithms. In addition, according to Cordeau *et al*. (2002), another limitation of sweep method is its greedy nature which makes it hard to deal with extra constraints and planner structure limits its applicability. Therefore, the application field of sweep algorithm seems quite narrow due to these shortcomings, so few people now use sweep algorithm to solve VRP.

2.5.1.3 Fisher and Jaikunar Algorithm

Another well-known algorithm is the Fisher and Jaikumar algorithm (Fisher and Jaikumar, 1981). Instead of using a geometric method to form the clusters, it solves a Generalised Assignment Problem (GAP). In the GAP, there are some agents and a number of tasks. Any task can be assigned to any agent, so that some cost and profit may vary depending on the agent-task assignment. Furthermore, each agent has a limit and the sum of the costs of tasks assigned to it cannot exceed this limit. It is necessary to find an assignment in which all agents do not exceed their limits and the total profit of the assignment is maximised. Then the Fisher and Jaikumar algorithm can be described as follows. Let there be in total K vehicles in service and denote them by $1,2,3,\cdots,K$ respectively. The number of nodes to be served is denoted by n, and c_{ij} denotes the costs (distance, time *etc.*) involved in traveling from node i to node j. Firstly, K notes are arbitrarily chosen from the set of n nodes which are to be served and they are denoted by $i_1, i_2, \cdots, i_k$, which means each node is joined to one vehicle. Secondly, compute the cost d_{ik} involved in inserting node i into the route cluster of vehicle k as $d_{ik} = \min\{c_{0i} + c_{ij_k} + c_{j_k0}, c_{0j_k} + c_{j_ki} + c_{i0}\} - (c_{0j_k} + c_{j_k0})$. In addition, solve a GAP with costs d_{ij}, customer weights q_i, and vehicle capacity Q. Lastly, solve a TSP for each cluster corresponding to the GAP solution.

The procedure iterates between solving a GAP master problem that assigns vertices to vehicles, and solving a TSP to determine the best vehicle route for each vehicle. The method has the advantage of producing a feasible solution, even if it is not ran to completion. Also, since it repeatedly solves a GAP and a TSP, it can benefit directly from any improvement in algorithms for these two problems (Laporte, 1992).

2.5.1.4 Christofides, Mingozzi, and Toth Algorithm

Christofides, Mingozzi and Toth (1979) proposed a truncated branch-and-bound algorithm. This method could be described as follows.

Step 1 (initialisation). Set $h \coloneqq 1$ and $F_h \coloneqq V \setminus \{0\}$. F_h is the set of unrouted vertices at level h.

Step 2 (route generation). If $F_h = 0$, stop. Otherwise, select an unrouted customer $i \in F_h$ and generate a set R_i of routes containing i and customers in F_h. These routes are gradually generated using a linear combination of two criteria: savings and insertion costs.

Step 3 (route evaluation). Evaluate each route $r \in R_i$ using the function$f(r) = t(S_r \cup \{0\}) + u(F_h \backslash S_r)$, where S_r is the vertex set of route r, $t(S_r \cup \{0\})$ is the length of a good TSP solution on $S_r \cup \{0\}$, and $u(F_h \backslash S_r)$ is the length of a shortest spanning tree over the remaining unrouted customers.

Step 4 (route selection). Determine the route r^* yielding $\min_{r \in R_i}\{f(r)\}$. Set $h \coloneqq h + 1$ and $F_h{:} = F_{h-1} \backslash S_{r^*}$. Go to step 2.

By comparing the computation of several heuristic methods, Laporte and Semet (2002) pointed out that for less computational effort, the truncated branch-and-bound algorithm tends to produce better solutions than the sweep algorithm.

2.5.2 Improvement methods

2.5.2.1 Lin-Kernighan heuristic

In the λ-opt algorithm, during each step, λ edges of the current tour are replaced by other λ edges so that a shorter tour is achieved. That is to say, in each step a shorter tour is obtained by deleting λ edges and putting the resulting paths together in a new way, possibly reversing one or more of them. Nevertheless, the downside of the approach is that λ must be determined in advance. It is complicated to know which λ to use so that the best compromise between the quality of a solution and the running time can be achieved. Lin and Kernighan's algorithm overcomes this drawback by changing the value of λ during the algorithm execution, deciding the value of λ at each iteration.

The algorithm is unique regarding exchanges (or moves) that convert one tour into another. Given a feasible tour, the algorithm aims to reduce the length of the current tour by performing exchanges over and over again, until a tour is obtained where no exchange can yield an improvement. One may repeat the process many times from initial tours generated in some randomised way.

The bottleneck of this method is the searching way for the exchanges. In order to increase efficiency, only exchanges that have a reasonable chance of leading to a reduction of tour length should be considered (Helsgaun, 2000). Therefore, many modified algorithms focus on this point to enhance efficiency.

2.5.2.2 Lagrangian relaxation heuristic

Stewart and Golden, (1984) make use of the Lagrangian relaxation ideas to move the capacity constraints into the objective function, in order to transform the VRP into a modified multiple TSP. More specifically, The objective function $\min \sum_k \sum_{i,j} c_{ij} x_{ijk}$ and the capacity constraints $\sum_{i,j} \mu_i x_{ijk} \leq Q, k = 1, \cdots, m,$ are transformed into $\min(\sum_k \sum_{i,j} c_{ij} x_{ijk} + \sum_k \lambda_k (\sum_{i,j} \mu_i x_{ijk} - Q)$ and $\lambda_k \geq 0, k = 1, \cdots, m.$ After the Lagrangian relaxation procedure, an arc exchange procedure, which is also referred to as a 3-opt procedure, is adopted to solve the relaxed problem. The key to this algorithm is determining the values for the Lagrangian multipliers ($\lambda_k, k = 1, \cdots, m$). If feasibility is the only goal, a very large value for λ_k may meet this need, but the result would be far from optimal. Therefore, firstly λ_k is set of a value that most likely produce an infeasible solution and then the value is incremented at each iteration until the 3-opt procedure generates a feasible solution to the VRP.

Due to the property of the multipliers, the limitation of this algorithm is that the performance is highly related to the selection of the value of Lagrangian multipliers and the initial feasible solution.

2.5.3 Metaheuristics

In recent years, many researchers are inspired by the natural world and apply the features and functions of the natural creatures to solving a practical problem. Consequently, a series of metaheuristics have been proposed for the VRP. Compared with classic heuristics, Metaheuristics perform a much more thorough search of the solution space, allowing inferior and infeasible moves, as well as recombination of solutions to create new ones (Cordeau *et al.*, 2002). By doing this, the development for the VRP is to get out of the local optimal solutions. Although some may notice that the improvements in solution quality are obtained at the expense of running time and algorithm simplicity. There are six main types of metaheuristics that have been applied to the VRP: Simulated Annealing, Deterministic Annealing, Tabu Search, Genetic Algorithms, Ant Systems, and Neural Networks (Gendreau *et al.*, 2002).

2.5.3.1 Tabu Search

By imitating the memory function of human brains, Glover (1989) first proposed and formalised Tabu search. Local (neighbourhood) searches take a potential solution to a problem and check its immediate neighbours to find an improved solution. Local search methods tend to become stuck in suboptimal regions or on plateaus where many solutions are equally fit. However, Tabu search enhances the performance of local search by relaxing its basic rule and making use of memory structures. First, at each step worsening moves can be accepted if no improving move is available, for example, the search is stuck at a strict local minimum. In addition, if a potential solution has been previously visited within a certain short-term period or if it has violated a rule, it is marked as "Tabu" (forbidden) so that the algorithm does not consider that possibility repeatedly. The termination criterion is that the objective function has not been improved for some iterations, or a certain fixed number of iterations is reached. The pseudo-code for Tabu search is shown in Table 2.1.

The speed of searching a solution of Tabu search is fast, and in the searching process, it accepts inferior solutions. Therefore, this method can jump out of local optimal solutions and then turn to search other areas, which can help to find a better solution even a global solution. The limitation of Tabu search is that it strongly depends on the initial solution. A good initial solution can lead to a better solution in the searching space, whereas a bad initial solution may decrease the searching speed. Since in the search process it only has one initial solution and moves the current solution to another one in each iteration, which limits the stability and global searching capability of the algorithm.

Table 2.1 Pseudo-code for Tabu search

```
Start
    Initialise the customer and depot locations,
    Initialise the distances and travel times,
    k ← 0,
    Generate the initial solution s_k,
    Repeat
        Generate a set of feasible solutions A_{k+1},
        Evaluate the solutions in A_{k+1},
        Select the best solution s_{k+1} from A_{k+1},
        Update Tabu list,
        Update penalty coefficient,
        k ← k + 1,
    Until termination condition is met
Finish
```

There have been improvements and changes to the classic Tabu search algorithms, such as Tabu-route heuristic Gendreau *et al.* (1994), the Taillard Tabu search algorithm (Taillard, 1993) and the granular Tabu search algorithm (Toth and Vigo, 2003). Moreover, Rochat and Taillard (1995) introduced a useful and powerful concept, the adaptive memory, which can be used to enhance any Tabu search-based algorithm. Hence, this reactive Tabu search is widely used for solving optimisation problems since then.

2.5.3.2 Simulated Annealing

The Simulated Annealing (SA) approach is a probabilistic method proposed in Kirkpatrick *et al.* (1983) and Černý (1985) to find the global minimum of a cost function that may have several local minima. Burkard and Rendl (1984) were the first to apply the SA method to solve quadratic assignment problems and their computational results indicated that they could obtain the best-known solution with a relatively high probability.

The SA algorithm was originally inspired from the process of annealing in metal work, a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. Both are attributes of the material and its thermodynamic free energy defines them. Heating and cooling the material has an influence on both the temperature and the thermodynamic free energy. While the temperature is decreased by the same amount of cooling, the cooling will decrease the thermodynamic free energy in a bigger or smaller amount, concerning the rate that it happens, with a slower rate producing a bigger decrease.

In simulated annealing, a temperature variable is used to simulate this heating process. It may initially be set high and then allowed to slowly "cool" as the algorithm runs. While this temperature variable is high, the algorithm will be allowed, with a higher probability, to accept solutions that are worse than the current solution. This step gives the algorithm the ability to jump out of any local optima as it explores the solution space. The chance of accepting worse solutions is reduced due to the decline of the temperature, which allows the algorithm to gradually focus in an area of the search space close to the optimum solution. Table 2.2 shows the pseudo-code for SA (Burkard and Rendl, 1984).

Table 2.2 Pseudo-code for Simulated Annealing

```
Start
    k ← 0,
    Initialise T(k),
    Select a current initial solution E_c at random,
    Evaluate point E_c: v(E_c),
    Repeat
        Repeat
            Select a new point E_n in the neighbourhood of E_c,
            Evaluate the new point E_n: v(E_n),
            If v(E_n) < v(E_c) or if r < p(E_n − E_c)
              with
              p(E_n − E_c) = exp[−(E_n − E_c)/T(k)]
              and
              r a random number from a uniform distribution,
            Then E_c ← E_n
        Until termination condition
        T(k + 1) = T_0 / ln k,
        k ← k + 1,
    Until convergence is reached
Finish
```

This notion of slow cooling is what makes the SA algorithm remarkably effective at finding a solution close to the optimal one when dealing with large problems which contain numerous local optima. A fundamental property of metaheuristics is accepting worse solutions because it is beneficial to a more extensive search for the optimal solution. Moreover, according to Gendreau *et* *al.*, (2002), it has been shown that SA asymptotically converges to a global optimum.

2.5.3.3 Genetic Algorithms

The Genetic Algorithm (GA) is a search metaheuristic method and belongs to the larger class of evolutionary algorithms, which generate solutions to optimisation problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Essentially, it is an efficient global search method. It can automatically collect and accumulate knowledge about the search space during the search process, and adaptively control the search process to find an optimum solution.

The evolution is an iterative process, usually starts from a population of randomly generated individuals, and a generation is defined as the population in each iteration. Moreover, the fitness of every individual is closely related to the value of the objective function in the optimisation problem, and is evaluated in each generation. The fitter individuals have more chance to be stochastically selected from the current population as parents whose genomes are recombined and possibly randomly mutated to form a new generation. Then the new generation of candidate solutions is used in the next iteration of the algorithm. Furthermore, the algorithm usually terminates when either a maximum number of generations has been constructed, or an ideal fitness level has been reached for the population. The pseudo-code for a simple GA is illustrated in Table 2.3.

GA is the first method that searches the optimal solution from more than one start point, and it can learn from the natural selection principle to present the complicated scenarios regardless of the field of the issues. Therefore, GA has been widely used in many different areas such as system optimization, machine learning, and project control and so on. However, based on the computational results for VRP, Gendreau *et al.*, (2002) pointed out that the genetic algorithms outperformed some Tabu search developments. In addition, according to Baker and Ayechew (2003), the results of the GA method combined with a neighbourhood search method shows this approach is competitive.

Table 2.3 Pseudo-code for a simple genetic search

```
Start
        k ← 0,
        Initialise P_k,
        Evaluate P_k: v(P_k),
        Repeat
                k ← k + 1,
                Select a new population F_k from P_{k-1},
                Generate a new population P_k by modifying F_k,
                Evaluate P_k: v(P_k),
        Until convergence is reached
Finish
```

2.5.3.4 Ant Colony Algorithms

Ant Colony Algorithms (ACAs) are inspired from an analogy with real ant colonies seeking for food. In the natural world, ants mark the paths they travel by laying an aromatic essence called pheromone. Initially, ants stroll randomly, and return to their colony once they find food while laying down pheromone trails. Then if other ants find such a trail, they are likely not to keep travelling at random, but to instead follow the path, returning and reinforcing it if they eventually find food. Over time, a short path gets marched over more frequently, and thus the pheromone density becomes higher on shorter paths than longer ones. Therefore, when one ant finds a good path from the colony to a food source location, other ants tend to follow that path, and positive feedback eventually leads to all the ants following a single path. This observation led Colorni *et al*. (1991) to propose a new class of metaheuristics for solving combinatorial problems: Artificial ants searching the solution space simulates real ants exploring their environment, objective function values are corresponding with the quality of food sources, and values recorded in an adaptive memory are associated with the pheromone trails.

During each iteration of the algorithm, each ant moves from a state x to state y, regarding a complete intermediate solution. Thus, each ant k computes a set $A_k(x)$ of feasible expansions to its current state during every iteration and moves to one of these in terms of a probability. For ant k, the probability p_{xy}^k of moving from state x to state y is based on the combination of two values: the attractiveness τ_{xy} of the move which indicates the number of artificial pheromones of that move; and the visibility η_{xy} of the move, implying how proficient it has been earlier to make that particular move. The pseudo-code below demonstrates the ACA.

Table 2.4 Pseudo-code for ACA

```
Start
    Initialise the attractiveness τ, and the visibility η,
    Repeat
        For each ant repeat
            Choose the next state to move into,
            Add that move to the Tabu list for each ant,
        Until each ant completed a solution
        Update τ and η for each ant that completed a solution,
        If the local best solution is better than the global solution,
        Then save the local best solution as the global solution,
    Until termination condition is met
Finish
```

ACA can deal well with large size problems and according to Bullnheimer *et al*. (1999), the performance of ACA is competitive with other metaheuristics such as TS and SA, and outperform neural network method. The limitation is the need to choose appropriate parameters which impacts performance a lot. In addition, it is a probabilistic method. Therefore, in every run, the solutions may be different so that one needs to run several times to get a central tendency.

2.5.3.5 Neural Networks

Neural networks (NN) are computational models composed of units that are richly interconnected through weighted connections, like neurons in the human brain: a signal is sent from one unit to another along a connection and is modulated through the associated weight. Although superficially related to their biological counterpart, artificial neural networks exhibit characteristics related to human cognition. In particular, they can learn from experience and induce general concepts from specific examples through an incremental adjustment of their weights. The idea of using NN for combinatorial problems emerged from the work of Hopfield and Tank (1985). They proposed an NN model and utilised the energy function to illustrate that the network is convergent to a steady state. It is verified that as the algorithm proceeds, the value of energy function always decreases until it converges to local optima. Therefore, to apply this model to optimisation problems, the cost function and restrictions are all included in the energy function. Although the algorithm has been improved by several researchers, due to the inflexibility of the model and complexity of the network, the performance of the algorithm is behind other metaheuristics (Gendreau *et al.*, 1994; Rochat and Taillard, 1995).

2.5.3.6 Hybrid Algorithms

An algorithm that incorporates two or more other algorithms that solve the optimisation problem is called a hybrid algorithm. It combines the desired features of the selected algorithms so that the overall algorithm is better than the individual components. For example, Alfa *et al*. (1991) applied a 3-opt based simulation annealing algorithm. Another example is that Potvin *et al*. (1996) proposed a hybrid approach to VRP using NN and GA.

In addition, Baker and Ayechew, (2003) first used a sweep method to obtain an initial population and then applied a hybrid of genetic algorithm with neighbourhood search methods to tackle the VRP, and the results show that this approach is competitive with Tabu search and SA in terms of computing time and solution quality. Therefore, we may conclude that sometimes hybrid algorithms behave better than a single method, and this gives researchers more inspirations to find a proper way to tackle a particular optimisation problem.

2.6 Summary

From the literature review, we can see that there have been many previous studies in different fields concerning about risks of missing targets such as missing deadlines. However, there is little research on the analysis of how to calculate the risks in situations where a series of stochastic events happen, and the distribution of the event start time may be distorted by time window constraints. One part of this research is to fill in the above gap by studying the distributions of the task start times of the service delivery operation and hence the first contribution of this thesis is the method proposed to calculate the distributions of the risk for such a problem. .

While there is a rich literature on VRP and scheduling problems and some of them consider risks, most of them try to minimise the travel cost in terms of distance or time. Risks are usually limited in the constraints if considered. Therefore, another part of this research is to schedule the tasks and route the vehicles to minimise the risks of missing customer appointments.

As risk minimisation models are new, this research also conducts computational experiments to compare the new models and the traditional travel time model with respect to different performance measures.

Chapter 3

Risk Estimation

In this chapter, risk is defined for our problem and several methods are proposed to calculate risks. Then the reliability of each method is tested and compared. Real data are analysed and shown to follow the assumptions of our proposed method *i.e.*, the summation method for risk calculation. Regression models are used to improve the estimation of the summation method and finally factors affecting risk are discussed.

Recent years have seen a growing interest in the development of vehicle routing algorithms to cope with uncertain situations found in real-world applications. For example, an organisation in the communication sector encounters a scheduling problem when they assign their technicians to do different tasks. For customers' convenience, the firm gives customers chances to choose their preferred time windows. However, usually the task duration is not exactly the same as the estimated time and differs for each task. As a consequence one intractable problem is that the technician might miss the next customer if the task of the current customer takes too long time. It is reasonable for the firm to reduce this risk in order to improve their service satisfaction. After reviewing the literature on risks in different subjects, the risk for the real-world problem that is considered in this thesis is defined as below.

3.1 The definition of risks

The field technician scheduling problem considers the assignment of a set of jobs or service tasks to a group of technicians. The tasks are located in different places within a city, with different time windows and processing times. In terms of uncertainty in a problem, it is natural to think of the risk as the probability over a threshold. Therefore, given a schedule, with a sequence of tasks $\{i_1, i_2, i_3, \cdots\}$ allocated to technician k and the start point of technician k is ki_0, which is the depot. Then some notations are as follows.

d_{kij} the travel time of technician k between i and j;

δ_{ki} the uncertain working time of technician k at customer i;

$[a_i, b_i]$ the time window for customer i.

The risk of missing the appointment for task i is the probability of the arrival time AT_{ki} being later than the time window b_i, which is the shaded area of the curves shown in Figure 3.1.

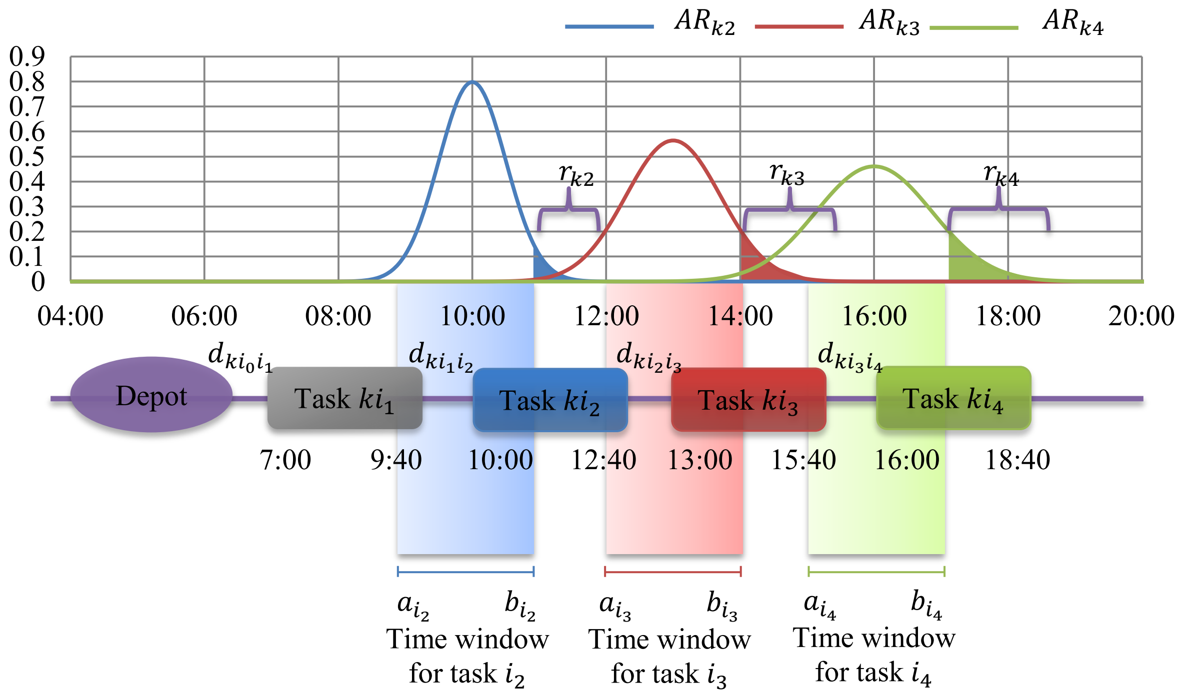


Figure 3.1 Risk definition

3.2 Propagation of risks

Without loss of generality, to simplify the model, we assume that the travel time is certain at this stage. Therefore, for each technician, the arrival time at the 1st task is specified, which is the start time from the depot plus the travel time between the depot and the site of the 1st customer $d_{ki_0i_1}$. Then we can ensure the arrival time of the 1st task within its time window through controlling the start time at the depot s_{k0}. For this task, the arrival time AT_{k1} is also the task start time ST_{k1} and they are constants, *i.e.*,

$$ST_{k1} = AT_{k1} = s_{k0} + d_{ki_0i_1}.$$

Thus, the risk of the 1st task, which demonstrates that the technician arrives at 1st customer after the customer preferred time window, can be as low as zero, which is

$$R_{k1} = \mathrm{P}\left(AT_{k1} > b_{i_1}\right) = 0.$$

First of all, in this problem the task working time is stochastic, so we suppose all the task working times δ_{ki} are independent random variables and follow pre-known distributions, here they are assumed to follow the normal distribution.

Because the 1st task start time is constant and the task working time δ_{ki_1} is a random variable, the finish time FT_{k1} for the 1st task is stochastic and follows a normal distribution of the same variance with the 1st task working time δ_{ki_1}, we have

$$FT_{k1} = ST_{k1} + \delta_{ki_1}.$$

Considering the travel time from the 1st customer to the 2nd customer $d_{ki_1i_2}$ which is deterministic, the arrival time AT_{k2} at the 2nd customer is uncertain and it has the normal distribution shape the same as the finish time FT_{k1}, thus,

$$AT_{k2} = FT_{k1} + d_{ki_1i_2} = ST_{k1} + \delta_{ki_1} + d_{ki_1i_2}.$$

It depends on the time the technician starts s_{k0}, the travel time from the depot to the 1st customer $d_{ki_0i_1}$, the duration of the 1st task δ_{ki_1} and the travel time between the 1st and 2nd customers $d_{ki_1i_2}$, *i.e.*,

$$AT_{k2} = s_{k0} + d_{ki_0i_1} + \delta_{ki_1} + d_{ki_1i_2}.$$

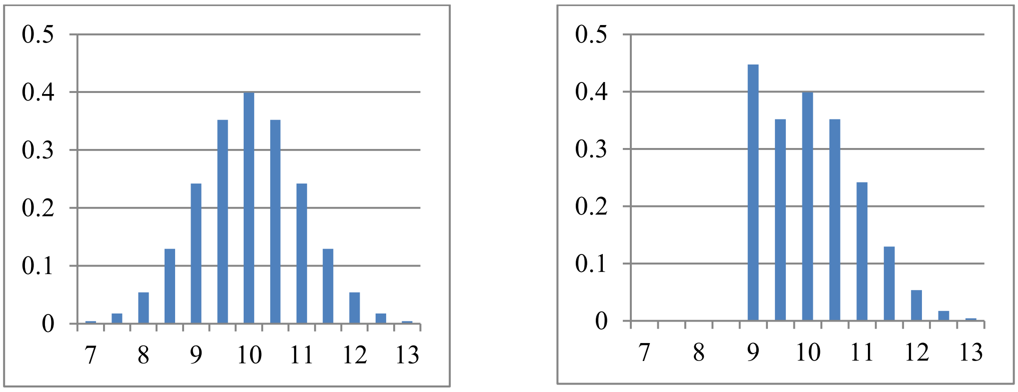
Then the risk for the 2nd task will be the probability of the arrival time AT_{k2} being greater than the upper limit of the time window b_{i_2}, which is

$$\begin{aligned} R_{k2} &= P\left(AT_{k2} > b_{i_2}\right) \\ &= 1 - \mathrm{P}\left(AT_{k2} \le b_{i_2}\right) \\ &= 1 - \int_{AT_{k2} \le b_{i_2}} f_{k1}(AT_{k2})\, \mathrm{d}AT_{ki_2} \\ &= 1 - \int_{-\infty}^{b_{i_2}} f_{k2}(X_{k2})\, \mathrm{d}X_{k2}, \end{aligned}$$

where X_{k2} denotes AT_{k2} and the distribution function of AT_{k2} is denoted as f_{k2}. The arrival time AT_{k2} is a random variable but the technician can also arrive at the customer before the lower limit of the time window a_{i_2}. In this case the technician has to wait until time a_{i_2} to start working. Then the time the technician starts working at the 2nd customer is

$$ST_{k2} = \max\{AT_{k2},\, a_{i_2}\},$$

where AT_{k2} follows a normal distribution and a_{i_2} is a constant. Due to the shift of the earliest time a_{i_2}, the start time ST_{k2} is not normal distributed anymore. More specifically, suppose that the arrival time AT_{k2} follows the distribution in Figure 3.2(a) and the lower bound of the task time window a_{i_2} is 9:00, then the distribution of the task start time ST_{k2} will turn out to be the distribution shown in Figure 3.2(b), the probability at 9:00 will be the sum of the probability of that arrive anytime up to 9:00. From Figure 3.2(b), it can be seen that the distribution of the start time ST_{k2} does not align with a normal distribution.



(a) Arrival time at the 2nd task　　(b) Start time at the 2nd task

Figure 3.2 Arrival time and start time at the 2nd task

Then after working at the 2nd customer with the duration δ_{ki_2} which is a normal random variable, the finish working time at the 2nd customer FT_{k2} will be

$$FT_{k2} = ST_{k2} + \delta_{ki_2}$$

$$= \max\{AT_{k2},\ a_{i_2}\} + \delta_{ki_2}$$

$$= \max\{X_{k2},\ a_{i_2}\} + \delta_{ki_2}.$$

So the arrival time of the 3rd customer AT_{k3} is the sum of the finish time FT_{k2} and the travel time from the 2nd customer site to the 3rd customer $d_{ki_2i_3}$, which is

$$AT_{k3} = FT_{k2} + d_{ki_2i_3}$$

$$= \max\{X_{k2},\ a_{i_2}\} + \delta_{ki_2} + d_{ki_2i_3}$$

$$= \max\{X_{k2},\ a_{i_2}\} + X_{k3},$$

where we suppose

$$X_{k3} = \delta_{ki_2} + d_{ki_2i_3}.$$

It is easy to see that X_{k3} follows a normal distribution of the same variance with the 2nd task duration δ_{ki_2} where the travel time $d_{ki_2i_3}$ is deterministic. Then the risk for the 3rd task which is the probability of the arrival time AT_{k3} being greater than the upper limit of the time window b_{i_3}, that is

$$R_{k3} = \mathrm{P}\left(AT_{ki_3} > b_{i_3}\right)$$

$$= \mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} + X_{k3} > b_{i_3}\right)$$

$$= 1 - \mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} + X_{k3} \le b_{i_3}\right)$$

$$= 1 - \mathrm{P}\left(X_2 + X_{k3} \le b_{i_3},\ a_{i_2} + X_{k3} \le b_{i_3}\right)$$

$$= 1 - \mathrm{P}\left(X_{k2} + X_{k3} \le b_{i_3},\ X_{k3} \le b_{i_3} - a_{i_2}\right)$$

$$= 1 - \iint\limits_{\substack{X_{k2}+X_{k3} \le b_{i_3} \\ X_{k3} \le b_{i_3} - a_{i_2}}} f(X_{k2}, X_{k3}) dX_{k2} dX_{k3}$$

$$= 1 - \iint\limits_{\substack{X_{k2}+X_{k3} \le b_{i_3} \\ X_{k3} \le b_{i_3} - a_{i_2}}} f_{ki_2}(X_{k2}) f_{ki_3}(X_{k3})\, dX_{k2} dX_{k3},$$

where X_{k2}, X_{k3} are independent and

$$\mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} + X_{k3} \le b_{i_3}\right) = \mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} \le b_{i_3} - X_{k3}\right)$$

$$= \mathrm{P}\left(X_{k2} \le b_{i_3} - X_3,\ a_{i_2} \le b_{i_3} - X_{k3}\right).$$

It can be derived from the property for the maximum of several random variables $Y_1, Y_2, \cdots, Y_n$ that (Wackerly *et al.*, 2014)

$$\mathrm{P}(\max\{Y_1, Y_2, \cdots, Y_n\} < y) = \mathrm{P}(Y_1 < y, Y_2 < y, \cdots, Y_n < y).$$

Similarly, because the arrival time AT_{k3} is a random variable, it is possible that the technician arrives at the customer before the lower limit of the time window a_{i_3}. Then the technician has to wait until time a_{i_3} to start working. Thus, the start working time at the 3rd customer is

$$ST_{k3} = \max\{AT_{k3},\ a_{i_3}\}$$

$$= \max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\}.$$

Then after working at the 3rd customer with the duration δ_{ki_3} which is a normal random variable, the finish working time at the 3rd customer FT_{k3} will be

$$FT_{k3} = ST_{k3} + \delta_{ki_3}$$

$$= \max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + \delta_{ki_3}$$

Note, that the finish time is not normally distributed. Then the arrival time of the 4th customer after traveling from the 3rd customer to the 4th customer AT_{ki_4} will be

$$AT_{ki_4} = FT_{k3} + d_{ki_3i_4}$$

$$= \max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + \delta_{ki_3} + d_{ki_3i_4},$$

$$= \max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + X_{k4},$$

where

$$X_{k4} = \delta_{ki_3} + d_{ki_3i_4}.$$

Thus the risk for missing the appointment of the 4th customer is

$$R_{k4} = \mathrm{P}\left(AT_{ki_4} > b_{i_4}\right)$$

$$= \mathrm{P}\left(\max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + X_{k4} > b_4\right)$$

$$= 1 - \mathrm{P}\left(\max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + X_{k4} \le b_{i_4}\right)$$

$$= 1 - \mathrm{P}\left(X_{k2} + X_{k3} + X_{k4} \le b_{i_4}, X_{k3} + X_{k4} \le b_{i_4} - a_{i_2},\ X_{k4} \le b_{i_4} - a_{i_3}\right)$$

$$= 1 - \iiint\limits_{\substack{X_{k2}+X_{k3}+X_{k4} \le b_{i_4} \\ X_{k3}+X_{k4} \le b_{i_4} - a_{i_2} \\ X_{k4} \le b_{i_4} - a_{i_3}}} f_{k2}(X_{k2}) f_{k3}(X_{k3}) f_{k4}(X_{k4})\, dX_{k2} dX_{k3} dX_{k4},$$

due to the fact that X_{k2}, X_{k3}, X_{k4} are independent and

$$\mathrm{P}\left(\max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} + X_{k4} \le b_{i_4}\right)$$

$$= \mathrm{P}\left(\max\{\max\{X_{k2},\ a_{i_2}\} + X_{k3}, a_{i_3}\} \le b_{i_4} - X_{k4}\right)$$

$$= \mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} + X_{k3} \le b_{i_4} - X_{k4},\ a_{i_3} \le b_{i_4} - X_{k4}\right)$$

$$= \mathrm{P}\left(\max\{X_{k2},\ a_{i_2}\} \le b_{i_4} - X_{k4} - X_{k3},\ a_{i_3} \le b_{i_4} - X_{k4}\right)$$

$$= \mathrm{P}\left(X_{k2} \le b_{i_4} - X_{k4} - X_{k3}, a_{i_2} \le b_{i_4} - X_{k4} - X_{k3},\ a_{i_3} \le b_{i_4} - X_{k4}\right)$$

$$= \mathrm{P}\left(X_{k2} + X_{k3} + X_{k4} \le b_{i_4}, X_{k3} + X_{k4} \le b_{i_4} - a_{i_2},\ X_{k4} \le b_{i_4} - a_{i_3}\right).$$

Therefore, by the method of induction in mathematics, the risk of missing the n^{th} task is

$$R_{kn} = 1 - \int \cdots \int_D \prod_{l=2}^{n} f_{ki_l}(X_{kl})\, dX_{k2} dX_{k3} \cdots dX_{kn},$$

where

$$D = \{(X_{k2}, \cdots, X_{kn}): \sum_{l=2}^{n} X_{kl} \leq b_{i_n}, \sum_{l=3}^{n} X_{kl} \leq b_{i_n} - a_{i_2},$$

$$\sum_{l=4}^{n} X_{kl} \leq b_{i_n} - a_{i_3}, \cdots, X_{kn} \leq b_{i_n} - a_{i_{n-1}}\}.$$

In addition, in the analysis above $X_{k2}, \cdots, X_{kn}$ follow the normal distribution since the task durations $\delta_{ki_1}, \cdots, \delta_{ki_{n-1}}$ are normally distributed, and then the probability density functions (PDFs) $f_{ki_2}, \cdots, f_{ki_n}$ are well defined. Furthermore, the mathematical expression for risks also works if the task durations follow any other distributions besides normal distributions, such as Gamma distributions, shifted normal distributions, and so forth.

3.3 Calculations of risks

The risk mathematical expressions described in the previous section are of the multiple integral formats. Multiple integrals of a Riemann integrable function in n variables, $f(x_1, x_2, \dots, x_n)$, over a domain D in n-dimensional space (Larson and Edwards, 2016) can be described as

$$\int \cdots \int_D f(x_1, x_2, \dots, x_n) dx_1 dx_2 \cdots dx_n.$$

More specifically, as is shown in Figure 3.3, the double integral $I = \iint_R f(x, y) dxdy$ is the volume under the surface $f(x, y)$ over the region R at the bottom which is the domain of integration, while the surface is the graph of the two-variable function to be integrated (Larson and Edwards, 2016).

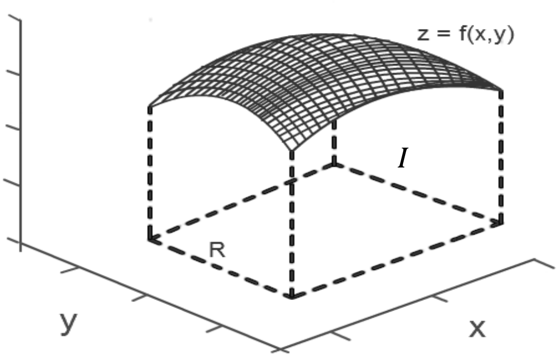


Figure 3.3 Double integral

Numerical integration is the most direct application of function interpolation. In engineering calculations, since many functions of the indefinite integral cannot be expressed by a simple function, for example, only some values at discrete points can be derived, or the function is the solution of a differential equation which does not have an explicit representation. Under these circumstances, the numerical integration is widely used to obtain the solution of an integral in an acceptable amount of time.

3.3.1 The Simpson's rule and Monte Carlo method

Given the polynomial is the most straightforward function class, the Lagrange polynomial interpolation is used widely in the integral calculation. Given $n+1$ points $x_0, x_1, \dots, x_n$, then the Lagrange polynomial interpolation equation (Epperson, 2007) is

$$p_n(x) = \sum_{k=0}^{n} \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{x - x_i}{x_k - x_i} f(x_k).$$

Since $p_n(x)$ estimates the function $f(x)$, $I(p_n)$ can be an estimate of $I(f)$, thus,

$$\begin{cases} I_{n+1}(f) = I(p_n) = \sum_{k=0}^{n} A_k^n f(x_k), \\ A_k^n = \int_a^b \rho(x) \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{x - x_i}{x_k - x_i} dx, \qquad k = 0,1,\dots,n, \end{cases}$$

where $\rho(x)$ is a weight function.

Suppose $\rho(x) \equiv 1$, and

$$x_k = a + kh, \qquad h = \frac{b-a}{n}, k = 0,1,\dots,n.$$

It yields the Newton-Cotes formulas, of which the rectangle rule and the trapezoidal rule are examples. Simpson's rule, which is based on a polynomial of 2nd order, is also a Newton-Cotes formula (Levy, 2010). The core thought of these rules is to use the quadratic interpolation as shown below.

Rectangle rule:

$$\int_a^b f(x)dx \approx (b-a) f\left(\frac{a+b}{2}\right).$$

Trapezoidal rule:

$$\int_a^b f(x)dx \approx (b-a)\frac{f(a)+f(b)}{2}.$$

Simpson's rule:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{b+a}{2}\right) + f(b)\right].$$

More specifically, Figure 3.4 (a) shows the rectangle rule estimation of the integral. The blue line denotes the original function $f(x)$, the integral is the light blue area between $x = a$ and $x = b$, above $y = 0$ and below $y = f(x)$, while the estimation uses the rectangle area within $x = a, y = b, y = 0$ and the orange line $y = f(m)$, where m is the middle point between a and b, *i.e.* $m = \frac{a+b}{2}$. Figure 3.4 (b) demonstrates the trapezoidal rule that uses the trapezoidal area within $x = a, y = b, y = 0$ and the orange line $\frac{y-f(a)}{f(b)-f(a)} = \frac{x-a}{b-a}$ to estimate the integral.

Then the Simpson's rule replaces the integrand $f(x)$ by the quadratic polynomial $P(x)$ where the orange line stands for in Figure 3.4(c), which takes the same values as $f(x)$ at the endpoints a, b and the midpoint $m = \frac{a+b}{2}$. Suppose

$$P(x) = Ax^2 + Bx + C,$$

then we have $P(a) = f(a)$, $P(b) = f(b)$ and $P(m) = f(m)$, thus

$$\int_a^b f(x)\,dx \approx \int_a^b Ax^2 + Bx + C\,dx,$$

$$\int_a^b Ax^2 + Bx + C\,dx = \frac{A}{3}(b^3 - a^3) + \frac{B}{2}(b^2 - a^2) + C(b - a).$$

Note that

$$b^3 - a^3 = (b - a)(b^2 + ab + a^2),$$

$$b^2 - a^2 = (b - a)(b + a).$$

Therefore,

$$\frac{A}{3}(b^3 - a^3) + \frac{B}{2}(b^2 - a^2) + C(b - a)$$

$$= \frac{b-a}{6}[2A(b^2 + ab + a^2) + 3\mathrm{B}(b^2 - a^2) + 6C]$$

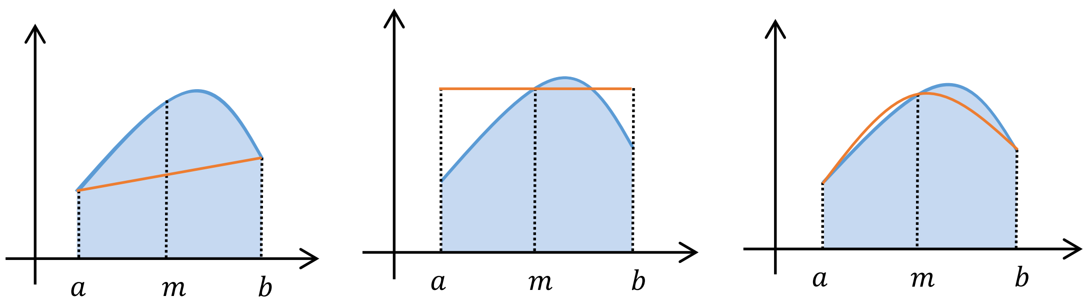
$$= \frac{b-a}{6}\left[(Aa^2 + Ba + C) + (Ab^2 + Bb + C) + 4A\left(\frac{a+b}{2}\right)^2 4B\left(\frac{a+b}{2}\right) + 4C\right],$$

then

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{b+a}{2}\right) + f(b)\right].$$

Another widely used variant of the Simpson's rule is Simpson's 3/8 rule(Jeffreys *et al.*, 1999), it is based upon a cubic interpolation rather than a quadratic interpolation and it is as follows

$$\int_a^b f(x)\,dx \approx \frac{b-a}{8}\left[f(a) + 3f\left(\frac{2a+b}{3}\right) + 3f\left(\frac{a+2b}{3}\right) + f(b)\right].$$



(a) Rectangle rule (b) Trapezoidal rule (c) Simpson's rule

Figure 3.4 Rectangle, trapezoidal and Simpson's rule

As the figures illustrate above, among the three rules Simpson's rule is the most accurate approach for numerical integral. Furthermore, to make the approximation more accurate, one usually divides the interval $[a, b]$ into a certain number n subintervals, applies an approximation rule for each subinterval and adds up these n results. For example, the Simpson's rule for single integral on the interval $[a, b]$ which is split up into n subintervals with n an even number, is given by

$$\int_a^b f(x)\,dx \approx \frac{h}{3}\left[f(x_0) + 2\sum_{j=1}^{n/2-1} f(x_{2j}) + 4\sum_{j=1}^{n/2} f(x_{2j-1}) + f(x_n)\right],$$

where $x_j = a + jh$ for $j = 0,1,\cdots,n-1,n$ with $h = (b-a)/n$.

Additionally, Monte Carlo integration is a powerful method for computing complicated or higher-dimensional integrals (Kalos and Whitlock, 2008). Monte Carlo approach is a sampling method based on probability theory. More precisely, let us draw random numbers in a set R covered the integrand $f(x)$ in the xy-plane (dots in the Figure 3.5), then the integral of the function $f(x)$ is approximately given by the size of the total area R where we sample the dots, $S(R)$, times the fraction of points that fall under the curve $f(x)$ where area D shows $\mathrm{No.}(D)$ over the number of all points $\mathrm{No.}(R)$, *i.e.*

$$\int_a^b f(x)\,dx \approx \frac{\mathrm{No.}(D)}{\mathrm{No.}(R)} \cdot S(R).$$

It is clear that the larger the number of random points is considered, the more accurate the calculation of this integral is. A weakness of Monte Carlo method is its uncertainty, different results may be obtained for the calculation of the same integrand function. However it does save considerable time for the calculation when the integral tends to be in more than five dimensions.

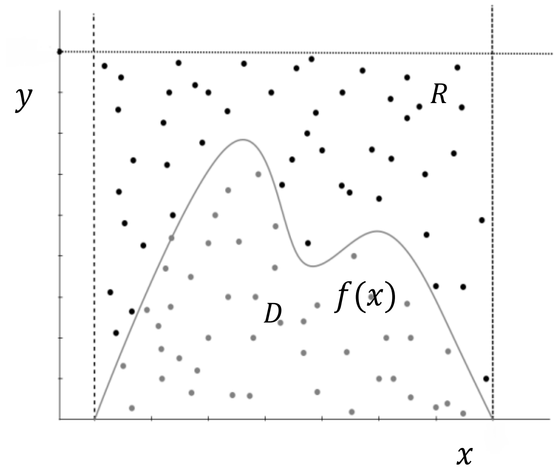


Figure 3.5 Basic idea of Monte Carlo method

In terms of the multiple integrals in the risk mathematical expression, the calculation method for an n-dimensional function is to apply the Simpson's rule on each direction of the multiple integral. While the Monte Carlo integration works as random sampling points in the n-dimensional set and distinguishing the number of points in the integral domain.

Therefore, in order to ensure the efficiency of the calculation, Monte Carlo integration is used for the integrals of more than 5-dimensions, while the Simpson's rule method is used to calculate the integrals for 5-dimensions or lower to guarantee the accuracy.

3.3.2 The accumulation method

Consider that in a schedule, risks of all tasks for each technician will be calculated. Therefore, instead of calculating the complicated integrals in multi-dimensions for each task, one way is to calculate the risk based on the distribution of the previous task finish time, this is where the accumulation method is proposed. The Simpson's rule and Monte Carlo method is based on the distributions of all the previous task durations, whereas the accumulation method uses a discrete approximation to represent all distributions for the calculation. This method is based on the one-dimensional interpolating integral function (Levy, 2010).

More specifically, as we explained before, the arrival time at the 2nd customer is normal distributed which is

$$AT_{k2} = s_{k0} + d_{ki_0i_1} + \delta_{ki_1} + d_{ki_1i_2}.$$

Then the start time considering the time window a_{i_2} is not normal which is

$$ST_{k2} = \max\{AT_{k2},\, a_{i_2}\} = \max\{X_{k2},\, a_{i_2}\}.$$

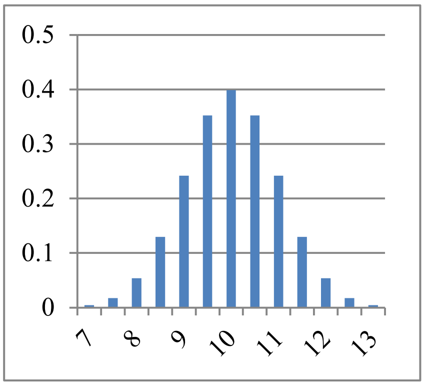
Then we estimate the distribution of $\max\{X_{k2}, a_{i_2}\}$ by transfer it into discrete distribution, which is to set the probability $\mathrm{P}(X_{k2} \leq\, a_{i_2})$ as the probability of $\mathrm{P}(X_{k2} =\, a_{i_2})$, set $\mathrm{P}(a_{i_2} < X_{k2} \leq\, a_{i_2} + h)$ as the probability of $\mathrm{P}(X_{k2} = a_{i_2} + h)$, set $\mathrm{P}(a_{i_2} + h < X_{k2} \leq\, a_{i_2} + 2h)$ as the probability of $\mathrm{P}(X_{k2} = a_{i_2} + 2h)$ and so on, where h is the unit length of the discrete subintervals for X_{k2}. Then the arrival time at the 3rd customer is

$$AT_{k3} = \max\{X_{k2},\, a_{i_2}\} + X_{k3}.$$

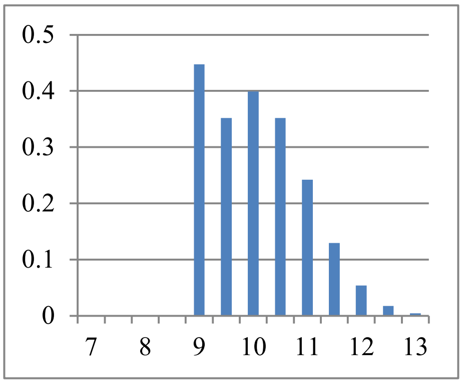
The sum of the 2nd task duration and the travel time between the 2nd and 3rd customers X_{k3} follows a normal distribution, and it also can be approximated into the discrete distribution with the subintervals of length h. Then add the two distributions together to pursue the distribution of AT_{k3} in the way that the probability value of each bar of the ST_{k2} distribution multiples the probability value of each bar of the X_{k3} distribution to get the probability of each value of AT_{k3} where $AT_{k3} = ST_{k2} + X_{k3}$.

Then the risk of missing the 3rd task appointment is that

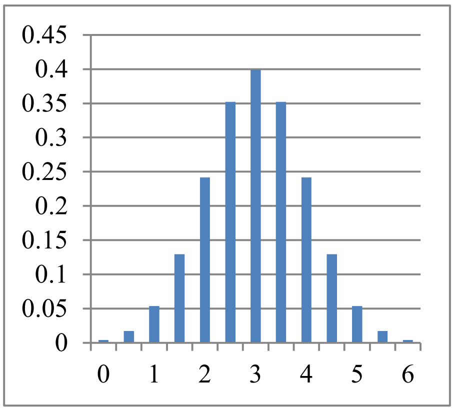
$$
\begin{aligned}
R_{ki_3} &= P\left(AT_{k3} > b_{i_3}\right) \\
&= P\left(\max\{X_{k2}, a_{i_2}\} + X_{k3} > b_{i_3}\right) \\
&= P\left(\max\{X_{k2}, a_{i_2}\} + X_{k3} = b_{i_3}\right) \\
&\quad + P\left(\max\{X_{k2}, a_{i_2}\} + X_{k3} = b_{i_3} + h\right) \\
&\quad + P\left(\max\{X_{k2}, a_{i_2}\} + X_{k3} = b_{i_3} + 2h\right) \\
&\quad + \cdots
\end{aligned}
$$



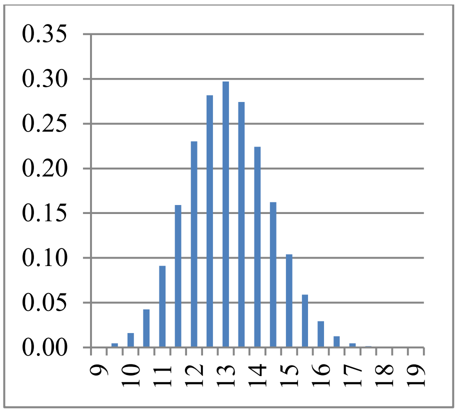
(a) Arrival time at the 2nd customer



(b) Start time at the 2nd customer



(c) Work time distribution



(d) Arrival time at the 3rd customer

Figure 3.6 The accumulation method

As an example, let us assume that a technician has 3 tasks and travel times are ignored to simplify the problem. The technician will start work at 7:00, and the 1st task time window is from 7:00 to 9:00. Because there is no uncertainty before the 1st task, there is no chance to arrive at 1st customer later than 9:00 which means no risk for the 1st task. Suppose the average duration time of the 1st task is 3 hours and this duration time is a random variable with variance of 1. Then after he finishes the 1st task, the distribution of the arrival time at the 2nd customer follows the distribution which is $AT_{k2} \sim N(10,1)$ and is discretely described in Table 3.1 column 2 and Figure 3.6 (a). It implies that the average arrival time is 10:00, and the standard deviation is 1 hour. The 2nd customer time window is from 9:00 to 11:00, and then the risk for this task will be the total probability of the arrival time after 11:00, which is

$$
\begin{aligned}
R_{k2} &= P(AT_{k2} > 11) \\
&= P(AT_{k2} = 11.5) + P(AT_{k2} = 12) + P(AT_{k2} = 12.5) + P(AT_{k2} = 13) \\
&= \mathrm{pdf}(AT_{k2} = 11.5) \cdot 0.5 + \mathrm{pdf}(AT_{k2} = 12) \cdot 0.5 \\
&\qquad + \mathrm{pdf}(AT_{k2} = 12.5) \cdot 0.5 + \mathrm{pdf}(AT_{k2} = 13) \cdot 0.5 \\
&= 0.2237,
\end{aligned}
$$

where $\mathrm{pdf}(\cdot)$ is the PDF value, and 0.5 is the length of subintervals h.

As it is illustrated in Section 3.2, the start work time at the 2nd customer will be of the distribution in Table 3.1 column 3 and Figure 3.6 (b). The PDF values of the arrival time before 9:00 are all added to the one of 9:00 to construct the discrete ST_{k2}, *i.e.*,

$$p(ST_{k2} = 9)$$

$$= p(AT_{k2} = 7) + p(AT_{k2} = 7.5) + p(AT_{k2} = 8)$$
$$+ p(AT_{k2} = 8.5) + p(AT_{k2} = 9)$$

$$= 0.4474.$$

Table 3.1 Arrival and start time discrete distribution at the 2nd customer

Time	PDF (Arrival)	PDF (Start)
7	0.004432	0
7.5	0.017528	0
8	0.053991	0
8.5	0.129518	0
9	0.241971	0.447439
9.5	0.352065	0.352065
10	0.398942	0.398942
10.5	0.352065	0.352065
11	0.241971	0.241971
11.5	0.129518	0.129518
12	0.053991	0.053991
12.5	0.017528	0.017528
13	0.004432	0.004432

Table 3.2 Work time distribution of the 2nd task

Time(Hour)	PDF	Time(Hour)	PDF
0	0.004432	3.5	0.352065
0.5	0.017528	4	0.241971
1	0.053991	4.5	0.129518
1.5	0.129518	5	0.053991
2	0.241971	5.5	0.017528
2.5	0.352065	6	0.004432
3	0.398942		

Then suppose the combination of the 2nd task duration time and travel time between the 2nd and 3rd task follows the discrete distribution in Table 3.2 and Figure 3.6(c), *i.e.*, $X_{k3} \sim N(3,1)$, where the mean of the 2nd task duration time is 3 hours and the standard deviation is 1 hour. The accumulation method is to add each segment of the work time to each segment of the start time, and multiply their probabilities to have the new distribution. Then if the summation of the time is the same, the probabilities will be added up. For a instance, in terms of the arrival time 11:00, the production of the start time of the 2nd task 9:00 and 2nd task duration time 2 hours will contribute the probability, as well as 9:30 and 1.5 hours, 10:00 and 1 hour, 10:30 and 0.5 hour, 11:00 and 0 hour, so the probability regarding the arrival time of 11:00 will be the sum of the probabilities of these five segments, *i.e.*,

$$P(AT_{k3} = 11)$$

$$= P(ST_{k2} = 9) \cdot P(X_{k3} = 2) + P(ST_{k2} = 9.5) \cdot P(X_{k3} = 1.5)$$
$$+P(ST_{k2} = 10) \cdot P(X_{k3} = 1) + P(ST_{k2} = 10.5) \cdot P(X_{k3} = 0.5)$$
$$+P(ST_{k2} = 11) \cdot P(X_{k3} = 0)$$

$$= \mathrm{pdf}(ST_{k2} = 9) \cdot \mathrm{pdf}(X_{k3} = 2) \cdot 0.5^2 + \mathrm{pdf}(ST_{k2} = 9.5) \cdot \mathrm{pdf}(X_{k3} = 1.5) \cdot 0.5^2$$
$$+\mathrm{pdf}(ST_{k2} = 10) \cdot \mathrm{pdf}(X_{k3} = 1) \cdot 0.5^2 + \mathrm{pdf}(ST_{k2} = 10.5) \cdot \mathrm{pdf}(X_{k3} = 0.5)$$
$$\cdot 0.5^2 + \mathrm{pdf}(ST_{k2} = 11) \cdot \mathrm{pdf}(X_{k3} = 0) \cdot 0.5^2$$

$$= 0.447 \times 0.242 + 0.352 \times 0.129 + 0.399 \times 0.054 + 0.352 \times 0.017$$

$$+0.242 \times 0.004$$

$$= 0.04566,$$

then the value of the PDF of the arrival time will be this probability divided by the length of subintervals (*i.e.* 0.5 hour here),

$$\mathrm{pdf}(AT_{k3} = 11)$$

$$= P(AT_{k3} = 11) \div 0.5$$

$$= 0.0913,$$

as is the PDF value regarding the arrival time of 11:00 shown in Table 3.3. Therefore, the accumulation method can calculate the distribution of arrival time at the 3rd task, shown in Table 3.3 and Figure 3.6 (d). Additionally, the length of intervals in this example is 0.5 hour for simplicity of the description, however, in the risk calculation for real scenarios we used 1 minute as the interval length.

Table 3.3 Arrival time distribution of the 3rd task

Time	PDF	Time	PDF
9	0.00099	14.5	0.10411
9.5	0.00470	15	0.05890
10	0.01605	15.5	0.02939
10.5	0.04276	16	0.01279
11	0.09132	16.5	0.00480
11.5	0.15911	17	0.00152
12	0.23008	17.5	0.00039
12.5	0.28149	18	0.00008
13	0.29694	18.5	0.00001
13.5	0.27430	19	0.10411
14	0.22397		

Therefore, in a similar way we can calculate the risks of $R_{k4}, \dots, R_{kn}$. Through this method, we could utilise the distribution and calculation of the previous tasks so that it is more convenient to obtain the risk for a large number of tasks for one technician, since this method is just to loop the same part in terms of programming. We have tested this idea on a single technician, and it works fast and the result is accurate as well. The result for later tasks may not be as accurate as the first several because of the accumulation effect.

3.3.3 The summation method

For large sample data, it is common to assume the work times follow normal distributions or that the number of customers is large enough to justify using the central limit theorem. A property of the normal distribution (Lemons, *et al.*, 2002) is summarised as below.

Property Suppose X and Y are independent random variables that are normally distributed, with means μ_1, μ_2 and standard deviations σ_1, σ_2 then their sum $X + Y$ is also normally distributed with mean $\mu_1 + \mu_2$ and variance $\sigma_1^2 + \sigma_2^2$.

Naturally, the calculation will become more straightforward and faster if we could utilise this property. It has been known that if time windows are ignored, the mean and variance of a job at a given point are just the accumulation of means and variances of the task duration times and travel times of the same technician up to the current task. From the analysis of the risk propagation, in the real world, the start working time ST_{ki} is not normally distributed for all the tasks because of the impact of the earliest time of the time window a_i. However, it is reasonable to approximate the impact of time window constraints without losing the benefits of fast computation. Therefore, different thresholds and similar behaviours are considered and adjusted to follow the normal distributions as below.

- If the arrival time is earlier than the lower limit of the time window with the probability more than 84% then the arrival time distribution is omitted and the lower bound of the time window is used as the start time without the variance;

- If the probability of arrival time after the lower limit of the time window is more than 84% then the normal distribution of the arrival time is used as the distribution of the start time;

- If the absolute difference between the mean of the arrival time and the lower bound of the time window is within a standard deviation of the arrival time, then the adjusted mean and standard deviation of the arrival time is used for the start time.

When the lower limit of the time window is one standard deviation after the mean arrival time, the probability of the arrival time before the lower limit of the time window is 84%, because for any normal distribution $P(X > \mu + \sigma) = 15.86\%$. More specifically, as it is shown in Figure 3.7 (a), the mean of the arrival time $\mu(AT_{ki})$ is before one standard deviation $\sigma(AT_{ki})$ downward movement of the lower limit of the time window a_i, where the standard deviation is the one of the arrival time variable, *i.e.*, $\mu(AT_{ki}) \leq a_i - \sigma(AT_{ki})$. It is the same to say that with the probability of more than 84%, the arrival time is earlier than the given time window. Thus for this scenario, the uncertainty of the arrival time can be ignored because the technician arrives early enough so that the waiting time absorbs the risk of missing the customer's time window. Then the earliest time window a_i is used as the mean of the start time ST_{ki} and no variance is considered, so the variance of the current task working time contributes the uncertainty of the arrival time at the next customer mostly.

Figure 3.7 (b) shows the scenario that the mean of the arrival time $\mu(AT_{ki})$ is after a standard deviation $\sigma(AT_{ki})$ upward move of the lower limit of the time window a_i , *i.e.*, $\mu(AT_{ki}) \geq a_i + \sigma(AT_{ki})$, which means that with the probability of more than 84% the arrival time is after the earliest time of the given time window. Therefore, the lower limit of the time window does not shift the distribution of the arrival time much comparing the start time distribution, especially the right tail of the distribution which combined with the current task duration δ_{ki} attributes most of the risk for the next customer. Then the mean and variance of the arrival time are kept the same for the start time distribution.

In terms of the most complicated scenario as shown in Figure 3.7 (c) and (d), the mean of the arrival time $\mu(AT_{ki})$ is within one standard deviation $\sigma(AT_{ki})$ of the lower limit of the time window, *i.e.*, $a_i - \sigma(AT_{ki}) < \mu(AT_{ki}) < a_i + \sigma(AT_{ki})$, but we still would like to use normal distribution for the start time considering the time window, thus we use an adjusted mean $\mu_a(AT_{ki})$ and an adjusted variance $\sigma_a^2(AT_{ki})$ for the normal distribution of the start time, where the analysis is much based on the data investigation, illustrated in details in Section 3.5. This estimation benefits calculating the risks relatively accurate and saves computing time due to the summation property of the normal distribution.

For example, suppose the task time window $[a_i, b_i]$ is from 11:00 to 13:00 and the arrival time AT_{ki} follows normal distribution with the variance 1 hour, if the mean of the arrival time $\mu(AT_{ki})$ is before 10:00, we use 11:00 as the mean of the start time and no variance, which also can be seen as the normal distribution $N(11,0)$. Then if the mean of the arrival time $\mu(AT_{ki})$ is after 12:00, we use the normal distribution $N(\mu(AT_{ki}), 1)$ as the start working time distribution. Lastly, if $\mu(AT_{ki})$ is between 10:00 and 12:00, the normal distribution with adjusted mean and variance $N(\mu_a(AT_{ki}), \sigma_a^2(AT_{ki}))$ is used as the start time distribution.

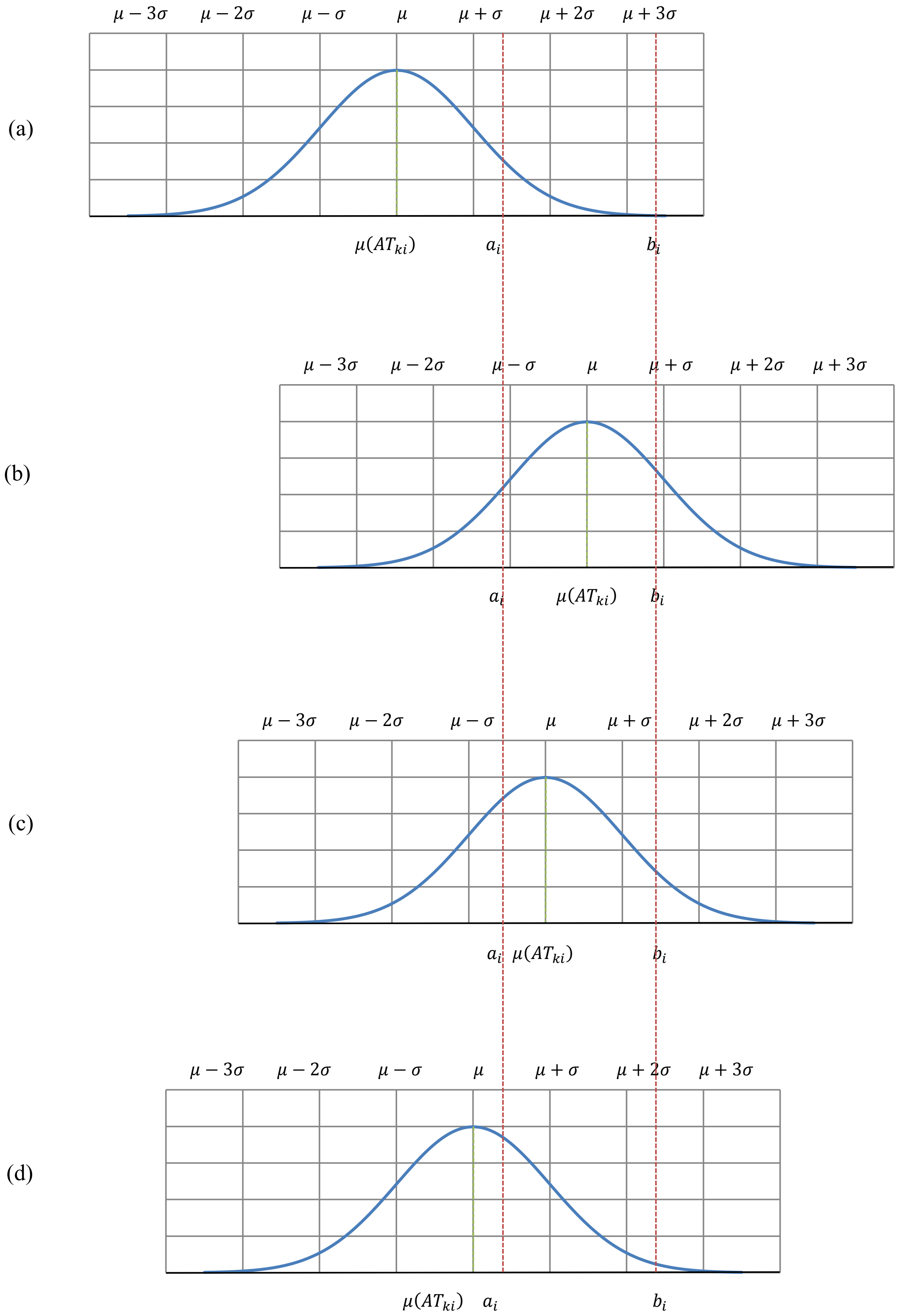


Figure 3.7 Relationship between the mean arrival time and the time window

3.4 Reliability of the risk calculation

As there are several methods to calculate or estimate the risk, the accuracy and reliability of these calculations need to be verified so that these methods can be used in the risk minimisation procedures. Maple is used to calculate the accurate integral for comparison.

3.4.1 Reliability of the Simpson's rule method

To start with, we calculate the risk by utilising the Simpson's rule. Due to the fact that Maple cannot calculate higher integrals, first we only compare the results with the calculation of the single, double and triple integrals by Maple and by using Simpson's rule in Java.

Table 3.4 Example result comparison between Maple & Simpson's rule

Task number	Task duration	Distribution	Time window a_i b_i	Risk by Maple	Risk by Simpson's rule	Error
1st task	X_{k1}	$N(9,1)$	7:00 - 9:00	0	0	0%
2nd task	X_{k2}	$N(2,1)$	9:00 - 11:00	0.02275	0.02272	-0.132%
3rd task	X_{k3}	$N(2,1)$	11:00 - 13:00	0.08693	0.08699	0.069%
4th task	X_{k4}	$N(2,1)$	13:00 - 15:00	0.15316	0.15325	0.059%

For example, suppose the travel times are omitted, all the tasks are identical, and their duration times follow the normal distribution $N(2,1)$, where the mean of the duration is 2 hours and the variance of the random variable is 1 hour. The technician start work at 7:00, then the 1st task finish time is the same as the arrival time for the 2nd task $AT_{k2} = FT_{k1} = X_{k1} \sim N(9,1)$, where the mean of the arrival time is 9:00 and the variance is 1 hour. The details are shown in Table 3.4. The relative error (Golub and Van Loan, 2013) is defined as

$$error = \frac{R(S) - R(M)}{R(M)},$$

where $R(M)$ and $R(S)$ is the risk calculated by Maple and by the Simpson's rule in Java respectively. The risk is as we defined before, which is the probability of the arrival time after the upper bound of the time window. It can be seen in Table 3.4 that the results of the Simpson's rule method are accurate.

However, calculating the risk of 6th task for the example in Table 3.5 takes a long time using the Simpson's method. This is because the five-dimensional matrix used in the Simpson's rule takes more time to calculate. The drawback is not significant when only calculating risks for one technician, but when it comes to more technicians and a large number of tasks, the calculation takes much longer.

3.4.2 Risk approximation of the summation method

In the example in Section 3.3.2, the arrival time AT_{k2} is supposed to follow normal distribution $N(10,1)$, which means the mean of the arrival time is 10:00 and the standard deviation is 1 hour. Moreover, the distribution of the 2nd task duration and travel time between the 2nd and the 3rd tasks X_{k3} is a normal distribution $N(3,1)$, and the curve in Figure 3.8 shows the normal distribution $N(13,2)$, while the bar chart shows the discrete distribution of the arrival time AT_{k3} at the 3rd customer obtained by the accumulation method.

It can be seen that the distribution of the arrival time to the next task can be approximated as a normal distribution even though the start time is affected by a time window earliest time a_i, especially the right tail of the distribution is usually used to calculate the risk, where the discrete distribution and the curve are very close. This is further discussed and analysed in Section 3.5.

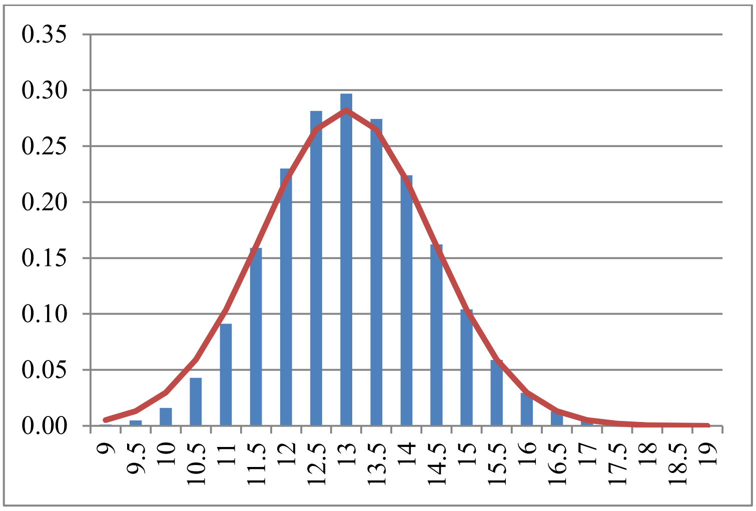


Figure 3.8 Arrival time at the 3rd customer and a normal estimation

3.4.3 Comparison of different methods

This section focuses on comparing the Simpson's rule and the Monte Carlo method, the accumulation method and the summation method in term of running time and quality of results. An example is used to test these approaches. There are seven tasks allocated to a technician and suppose technician working time at each customer follows different normal distributions, of which the means and standard deviations are shown in Table 3.5, as well as the task time windows and travel times. Note that these times are in minutes and the earliest or upper limit of a time window is the number of minutes starts from 0:00. Then the risk calculation results from these three methods and the corresponding running time are illustrated in Table 3.6. The absolute error (Golub and Van Loan, 2013) is sufficient to show the accuracy among methods and is defined as

$$error = R - R(S),$$

where R represents the risk calculated by the accumulation method or the summation method, $R(S)$ gives the risk calculated by the Simpson's rule and Monte Carlo method. We use $R(S)$ as the standard value because in the last section the risk obtained by the Simpson's rule and Monte Carlo method is shown to be close to the integral result calculated by Maple.

From the risk and error results in Table 3.6 and 3.7, it can be seen that the risks given by the summation method are not far away from those from the other two methods, and so it is reasonably accurate for use in a simulator or dynamic scheduling tool as the calculation is fast.

Table 3.5 Test example

Task number	Mean	Standard deviation	Lower limit of the time window	Upper limit of the time window	Travel time
1	60	40	450	632	14
2	31	40	567	717	93
3	46	40	608	758	90
4	31	40	674	824	33
5	40	40	744	894	22
6	25	40	816	966	40
7	30	40	923	1073	45

Table 3.6 Risk result

Risk	Simpson's & M.C.	Accumulation	Summation
Task 1	0	0	0
Task 2	0.00215	0.00089	0.00216
Task 3	0.27833	0.27566	0.27388
Task 4	0.39300	0.38839	0.38087
Task 5	0.32722	0.31616	0.31736
Task 6	0.37398	0.36193	0.36863
Task 7	0.23619	0.23934	0.24702
Maximum	0.39300	0.38839	0.38087
Sum	1.61088	1.58237	1.58992
Running time (ms[a])	3460	28	2

a. Milliseconds

Table 3.7 Error result

Risk	Accumulation	Summation
Task 1	0	0
Task 2	-0.00126	0.00001
Task 3	-0.00267	-0.00445
Task 4	-0.00461	-0.01213
Task 5	-0.01106	-0.00986
Task 6	-0.01205	-0.00535
Task 7	0.00315	0.01083
Maximum	-0.00461	-0.01213
Sum	-0.02851	-0.02096

3.5 Data analysis

In this section, we first discuss the raw data of task duration and how we processed them, then the travel time and the risk in our problem are analysed and discussed. Generally the travel time and the task duration are estimated as variables following normal distributions. Additionally, the scenario when the travel time and task duration follow Gamma distributions are also studied.

3.5.1 Raw data collection, cleaning and processing

We gathered real schedule data of 12 months from an industry company which offers web services in the UK. The raw data obtained by SQL query consists of 72113 task operating records that have many features. We choose some features that are most relevant for our problem. An instance of the data is shown in Table 3.8 and the featured are as follows.

Domain: the geographical region where the tasks located;

Task type: the type of the task;

Skill code: the skill required to do the task;

Appointment type: type of the appointment, NOT APPT if the task is not appointed. APPT if the task is appointed.

Technician id: identifier of the technician who has completed the task;

Technician type: a classification of the technicians who have completed the task;

Provision or repair: another feature of the task, repair if the task it is repairing a firm or customer asset, provision if the task is the installation of a new asset;

Estimated task duration: planned task duration in minutes;

Estimated travel: planned travel time in minutes to reach the task from the previous task or start location of the technician;

Actual task time: actual time in minutes spent on the task;

East task coordinate: the east coordinate of the task;

North task coordinate: the north coordinate of the task.

According to our risk assumption, the probability of the arrival time at the customer site, the task duration distribution and the travel time distribution are crucial for our problem. Firstly we investigate the actual task time distribution. Two critical parameters in our model are the mean and variance of the task duration, which we need obtain from processing the historical data. There are 214 task types in the raw data, and some of them only have few records, thus we grouped the tasks with the same domain, task type, skill code, appointed type, provision or repair and estimated task duration together, then we analyse the most representative task groups that have a large number of records, the actual task time of these tasks contributes to obtaining the distribution of the task duration. Meanwhile, the matching between the estimated task duration and the actual is tested.

For example, there are 1291 records for a repair type of task, the records with the actual task time negative or greater than 250 minutes are neglected as outliers, then the histogram as shown in Figure 3.9 is obtained. We found the mean and standard deviation of the actual task duration is 88.13 minutes and

38.80 minutes respectively, while the estimated task duration is 85 minutes. Therefore it can be concluded that the estimated task duration in the database is reliable, and the standard deviation for the actual task duration is not only beneficial for this problem but also useful in further analysis. The curve in Figure 3.9 illustrates the PDF of the normal distribution $N(88.13, 38.80^2)$, which approximates the actual task time distribution well.

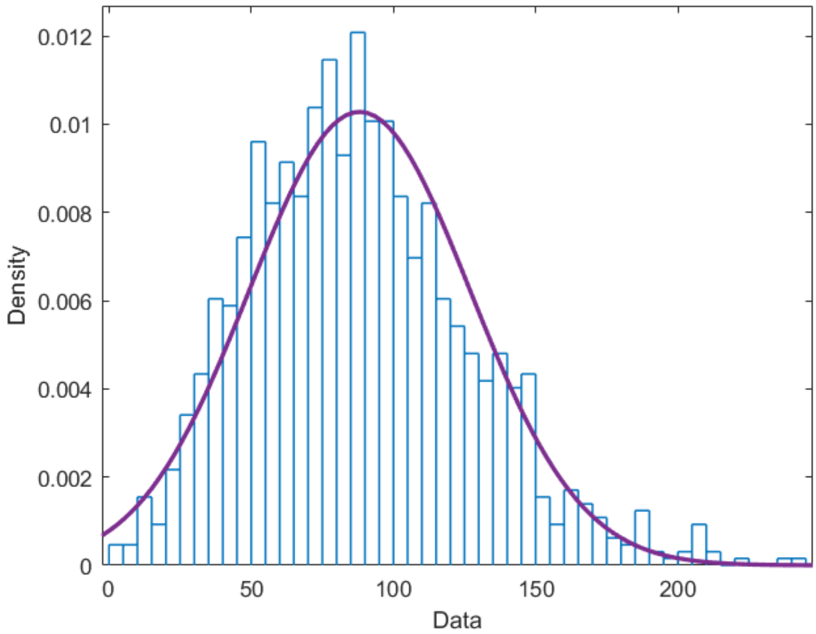


Figure 3.9 Distribution of a repair task with a normal fit

The distribution plots on the actual time spent on the most representative kinds of tasks, reveal that most of the tasks satisfy Gamma distributions while 34.6% of tasks follow normal distributions. Moreover, the Central Limit Theorem (CLT) is stated as below.

Theorem Lindeberg-Lévy CLT (Billingsley, 1995). If the random variables $X_1, \cdots, X_n$ form a random sample of size n from a given distribution with mean μ and variance $\sigma^2 (0 < \sigma^2 < \infty)$, then for each fixed number x,

$$\lim_{n\to\infty} \Pr\left[\frac{\overline{X_n}-\mu}{\sigma/\sqrt{n}} \le x\right] = \Phi\,(x),$$

where Φ denotes the cumulative density function (CDF) of the standard normal distribution.

The CLT is a statistical theory stating that given a sufficiently large sample size from a population with a finite level of variance, the mean of all samples from the same population will be approximately equal to the mean of the population μ. Furthermore, all the samples will follow an approximately normal distribution pattern, with all variances being approximately equal to the variance of the population divided by each sample size σ^2/n. This result helps to justify the use of the normal distribution as a model for many random variables that can be thought of as being made up of many independent parts. Therefore, it is sufficient to assume the actual work time aligns with a normal distribution as a large enough number of records can be gathered to analyse. Hence the summation method can be used to calculate the risks since most of the real data follows the normal distribution.

Moreover, Figure 3.10 shows the distribution for a provision task of 3792 records following a Gamma distribution, and the curve illustrates the Gamma distribution $\Gamma(4.58{,}0.046)$ which fits the actual work time distribution well, where the 4.58 and 0.046 are the shape and scale parameters for a Gamma distribution respectively. The mode of the Gamma distribution is 77.83 minutes, and the mean is 99.89 minutes, while the estimated task duration is 85 minutes which is between the two values. Even though many types of tasks align with Gamma distributions, we may still consider the actual task time as a normal distribution because the task time cannot be negative.

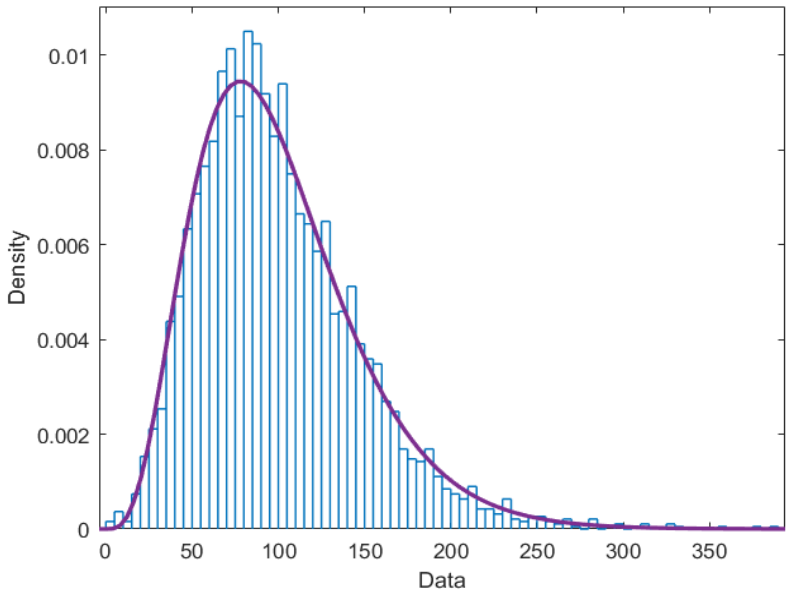


Figure 3.10 Distribution of a provision task with a Gamma fit

Furthermore, the standard deviation for the actual task time is 46.68 minutes. Together with distribution analysis of other tasks we found the standard deviation of the actual task time on average is approximately 40 minutes. Then to simplify the calculation in the schedule, we suppose the standard deviation for all kinds of tasks is 40 minutes.

In order to apply our model to solve the problem in the real world, raw data of tasks and technicians were collected, cleaned and analysed, with the following features.

For tasks:

Task id: identifier of the task;

Skill code: the type of the task;

Task location latitude: the latitude of the task location;

Task location longitude: the longitude of the task location;

Importance score: the priority level of the task;

Estimated task duration: the mean of the normal distribution in minutes for the task;

Task standard deviation: the standard deviation of the normal distribution in minutes for the task;

Earliest time window: the lower limit of the time window;

Latest time window: the upper limit of the time window;

Primary target: the primary service target time;

Secondary target: the secondary service target time.

For technicians:

Technician id: identifier of the technician;

Skill code: a classification of the technician having the skills;

Period start: the time of the technician to start working;

Period end: the time of the technician to end working;

Lunch start time: the time that the technician starts to have lunch;

Lunch duration time: the time for lunch;

Technician location latitude: the latitude of the technician's start location;

Technician location longitude: the longitude of the technician's start location;

Absence time: the time period that the technician offline from working;

Travel speed: the technician average travel speed between any two locations.

Some example data are shown about the task and technician information respectively in Table 3.9 and 3.10.

Table 3.8 Raw data

DOMAIN	TASK TYPE	SKILL CODE	APPOINTMENT TYPE	TECH ID	TECH TYPE	PROV REPAIR	ESTIMATED TASK DURATION	ESTIMATED TRAVEL	ACTUAL TASK TIME
EB	A	RBASIC	NOT_APPT	803180718	Home DL	Repair	60	27	18.35
EB	A	RBASIC	NOT_APPT	801778498	Visiting DL	Repair	60	12	90.03
EB	A461	BTAIL	NOT_APPT	600596705	Visiting DL	Provision	284	28	66.28
EB	A461	BTAIL	NOT_APPT	802966221	Visiting DL	Provision	284	29	60.67
EB	FLTP01	FRM7A	NOT_APPT	801864559	Home DL	Provision	19	27	34.2
MF	FSTP01	FRM9A	NOT_APPT	608327608	Home MWF	Provision	13	16	48.53
EB	FSTP15	FRM9A	NOT_APPT	605510706	Home DL	Provision	20	38	55.78
EB	I1	OBASIC	APPT	600596705	Home DL	Provision	45	17	75
MF	I1	OBASIC	APPT	607102220	Home DL	Provision	45	18	6.13
MF	I1	OBASIC	APPT	606334042	Home DL	Provision	60	13	191.1
EB	I1	OBASIC	APPT	803275117	Home DL	Provision	68	11	87.32
EB	I1	OBASIC	APPT	802035958	Home DL	Provision	68	6	180.9

Table 3.9 Task data

TASK ID	SKILL CODE	LATITUDE	LONGITUDE	IMPORTANCE SCORE	TASK DURATION	TASK SD	EARLIEST TIME	LATEST TIME	PRIMARY TARGET	SECONDARY TARGET
N1-F00319789	TNK64	53.4107694	-2.9958578	55	84	40	05/05/2015 02:00	05/06/2015 00:00	05/05/2015 02:00	05/05/2015 11:59
N1-F00321271	TNK64	53.3811978	-2.8688846	55	84	40	15/05/2015 08:00	05/06/2015 00:00	15/05/2015 08:00	15/05/2015 11:59
N1-M18571693	SYX70	53.8257924	-3.025423	80	40	40	16/05/2015 15:16	05/06/2015 00:00	30/04/2015 04:18	30/04/2015 04:18
N1-M18545981	SYX70	53.8257924	-3.025423	80	40	40	16/05/2015 15:17	05/06/2015 00:00	18/03/2015 04:21	18/03/2015 04:21
N1-M18564454	SYX70	53.8404195	-2.8814023	80	40	40	16/05/2015 15:17	05/06/2015 00:00	20/04/2015 04:21	20/04/2015 04:21
N1-M18476186	TNK64	53.5352439	-2.1719586	55	84	40	16/05/2015 15:18	09/06/2015 00:00	19/03/2015 08:00	19/03/2015 08:00
N1-P18487183	DMSB	53.3864713	-2.5953483	55	45	40	16/05/2015 15:18	05/06/2015 00:00	27/01/2015 04:16	27/01/2015 04:16
N1-P18492162	DMSB	53.4109216	-2.5580813	55	45	40	16/05/2015 15:18	05/06/2015 00:00	03/02/2015 04:16	03/02/2015 04:16
S1-000019487	CXM24	54.3293699	-2.7457504	30	17	40	01/06/2015 13:56	08/06/2015 13:56	08/06/2015 13:56	09/06/2015 09:56

Table 3.10 Technician data

TECH CODE	SKILL CODE	PERIOD START	PERIOD END	LUNCH START	LUNCH DURATION	LATITUDE	LONGITUDE	ABSENCE TIME	SPEED
605479188	"CTF61","CTF6","21CA7","CTF5","FOX01","BT-0008", "CTF3", "CTF2","21CA11","CTF7","FOX02","POP1", "CGGC","CADC","CAD4RM","SYX80","CSXC","21CA17" , "AXE80","CNN2","21CA16","CTF8","CTF9","AGE34", "CTF1","FOX03T","FOX03","21CA6","21CA12", "PWR3G","CGI1","21CA1","CIP2","CGC1","LMDF1","CNN1","CNN3","21CA10","FCP1", "21CA15", "FOX03S", "CTF4", "PWR3","CXM01"	03/06/2015 07:00	03/06/2015 16:30	03/06/2015 11:30	30	53.494092	-2.038277	0	60
700848018	"CPP3","CTF6","21CA3","21CCM1","CTP4","CTA4","BT-0011", "CIP2", "BT-0010", "21CA2","TAN01","AGE99", "BT-0008", "CTF61", "PLN","NN2","21CA10","CTF4", "CTF2","NN1", "21CA7","CIP4","CGC1","21CA17", "CTA3","SYX80","TXD10","CTF5","AXE80","21CA4","CGGC","21CCM2","CGI1","CPP4","CTM3","CTF1","CXM02","CTS4","CXM24","21CQA", "CXM01","CTF3", "SOC01","CIP3"	03/06/2015 07:00	03/06/2015 16:30	03/06/2015 11:30	30	53.52773	-2.193125	0	60

In Table 3.11 the means and standard deviations of the actual task times for several essential task types comparing the estimated task times, the number of records used to calculate the mean and standard deviation are shown. It can be seen that for the task type with large numbers of records, the estimated task time is close to the average actual task time, such as IFTTC8, R1CEWLR and R1LNWLR. Take R1CEWLR for example, the estimated task time is 75 minutes, based on 4598 records of the actual task time for this task type, the mean is 79 minutes. As for the task of fewer records, the estimated task time does not always fit the actual task time well. For example, the mean of the actual task time of the task type R1CAFTTC is 89 minutes whereas the estimated task time is 60 minutes, with 29 minutes difference.

Table 3.11 Task type data

Task type	Estimated (mins[a])	Mean (mins[a])	SD (mins[a])	No. of records
IFTTC8	30	38.23	23.53	3966
P1PR01FJR	30	26.46	18.97	2984
R1PCDTFNK	45	57.41	36.22	1903
R1CAFTTC	60	89.43	47.69	802
R1	75	73.67	47.03	1795
R1CEWLR	75	79.01	45.41	4598
R1DEDTFNK	75	83.20	40.91	3812
I1PR06LLU	85	99.88	46.48	3792
R1CIDTCE	85	88.33	40.41	1379
R1LNWLR	95	100.16	51.35	4050
R1NSYAPP	105	92.81	49.68	832
R1CALLSF2	120	101.21	47.97	1703

[a.] Minutes

The standard deviations are also investigated. For long estimated task times such as R1CEWLR, I1PR06LLU and R1LNWLR, it is approximately 45 minutes, and for short estimated task times like IFTTC8 and P1PR01FJR, it is around 20 minutes. While the average standard deviation obtained by all the 72113 task records is 50.47 minutes. Consequently, 25 minutes is used as the standard deviation for short time tasks which are estimated shorter than 45 minutes, and 45 minutes is used as the standard deviation for long time tasks where the estimated task times are longer than 45 minutes in the risk calculation.

3.5.2 Travel time analysis

Kaparias *et al.* (2008) introduced a measure of travel time based on the log-normal distribution, and this measure was implemented in the dynamic routing algorithm of an intelligent car navigation system. Rahman *et al.* (2018) proposed that there is a significant difference of bus travel time distributions around 8 kilometres (km) of the distance between a real-time bus location and a bus stop: the bus travel time distribution converges from a rightly skewed distribution to a more symmetrical distribution from a shorter to a longer distance to the bus stop, lognormal distributions are the best models for the scenario when the distance is less than 8 km and normal distributions well approximate the travel time for the distance to the bus stop more than 8 km.

Arezoumandi (2011), Pu (2011), Rakha *et al*. (2006) and Richardson and Taylor (1978) concluded the travel time follows a lognormal distribution for private car commuter journeys. Uno *et al*. (2009) analysed some real travel time data from the Global Positioning System (GPS) and proposed that the travel times for buses conform lognormal distributions, and they also estimated the travel time distributions of arbitrary routes by statistically summing up directly observed composite travel time distributions. Russell and Urban (2008), Taş *et al*. (2014) supposed that the travel times approximately follow Gamma distributions, while Chang *et al*. (2009), Hall (1986), Kulkarni (1986) Wellman *et al*. (1995) used normal distributions for travel times.

In terms of our research problem, the travel time is assumed to be normal distributed if it is considered as uncertain. The standard deviation is obtained via historical travel data, which is approximately 20 minutes, and the mean of the distribution is the travel time used in the model with deterministic travel times. Since the latitudes and longitudes for technicians and customers are available, the distance between any two locations is easy to calculate via the following method.

If laA denotes the latitude of location A in radian and loA denotes the longitude of location A in radian. While laB denotes the latitude of location B in radian and loB denotes the longitude of location B in radian. R denotes the radius of the earth, which is 6378.137 kilometres (Moritz, 1980). Then the distance between A and B (Kells *et al.*, 1940) is

$$D = R \cdot \arccos(\sin laA \sin laB + \cos laA \cos laB \cos(loA - loB)).$$

The distance D is the straight-line distance between A and B. Then comparing the real road distance with the straight-line distance between two locations via Google Map, and we found that the real road distance is approximate 1.3 times of the straight-line distance. Some examples are shown in Table 3.12. Given the latitudes and longitudes of location A and B, the straight-line distance between A and B can be obtained by the above formula, also the real road distance is gathered from Google Map data. Then the multiplier is calculated by

$$\text{multiplier} = \frac{\text{real road distance}}{\text{straight-line distance}}.$$

The multiplier is approximately 1.3. Therefore, the travel distance is

$$TD = 1.3 \cdot D = 1.3 \cdot \ R \cdot \arccos(\sin laA \sin laB + \cos laA \cos laB \cos(loA - loB)).$$

Then suppose the travel speed is 40 km/h so that we can calculate the deterministic travel time. Furthermore, if one prefers more accurate travel time, he may obtain the instance travel time via Google Map API at any specific time.

Table 3.12 Distance examples

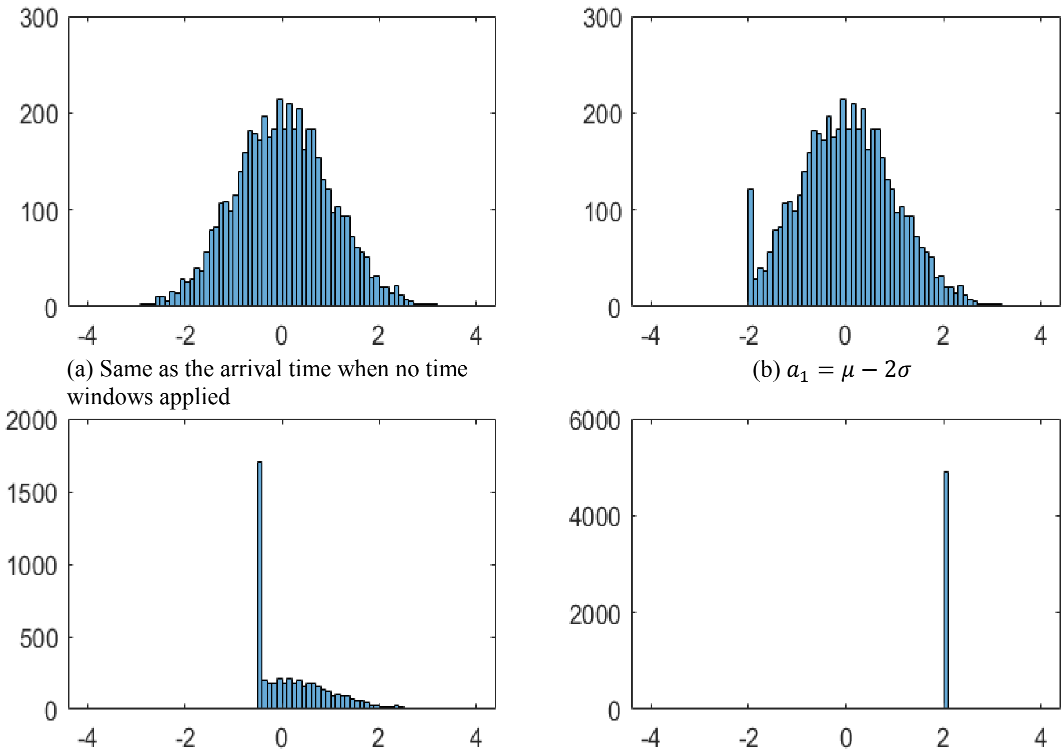
A Latitude	A Longitude	B Latitude	B Longitude	Straight-line distance (km)	Real road distance (km)	Multiplier
53.4107694	-2.9958578	53.3811978	-2.8688846	9.06	10.4	1.148
53.8404195	-2.8814023	53.8257924	-3.025423	9.61	11.4	1.186
53.8404195	-2.8814023	53.3864713	-2.5953483	53.88	73.3	1.360
53.494092	-2.038277	53.4109216	-2.5580813	35.59	52.0	1.461

3.5.3 Exploratory risk data analysis

Initially, the scenario where task duration and travel time follow normal distributions is investigated. Given a technician, the arrival time at his 1st task, AT_1 is a random variable that follows a normal distribution due to the stochastic travel time. Because a time window is associated with each task, the distribution for the start-operating time is not the same shape as the arrival time considering the lower limit of the time window. From section 3.2, the start time of the 1st task, ST_1 is a random variable that $ST_1 = \max\{AT_1, a_1\}$, where a_1 is a constant representing the lower limit of the time window. Then after doing the 1st task and travel from the 1st to the 2nd customer, the technician arrives at the 2nd task and the arrival time is a variable such that $AT_2 = \max\{AT_1, a_1\} + \delta_1 + d_{12}$, where $\delta_1$is the stochastic task duration and d_{12} is the uncertain travel time. Since the task operating time and the travel time are both normal distributed, $\delta_1 + d_{12}$ can be regarded as one random variable of the task and travel time, TT_1. The parameters for this normal distribution can be obtained using the property for the sum of normally distributed random variables.

The risk of the 2nd task is defined as the probability, $P(AT_2 > b_2)$, of the arrival time being later than b_2, the upper bound of the time window for the 2nd task. It is easy to see that $AT_2 = \max\{AT_1, a_1\} + TT_1$ is not normally distributed because the start time of the 1st task, $ST_1 = \max\{AT_1, a_1\}$ does not follow a normal distribution. Therefore, technically a normal distribution cannot be used to calculate the probability $P(AT_2 > b_2)$ directly, but as we mentioned in Section 3.4.2 the risk is defined in the area at the right tail of the arrival time distribution which seems to fit the normal distribution well. Hence the normal distribution is used to estimate the skewed start time distribution. The relationship between the risks calculated by the accumulation method and the summation method are investigated next. The accumulation method shows the risk (risk by accumulation) in terms of the shifted start time by the way that we discretise the start time distribution ST_1 and the task and travel time distribution TT_1, then calculate the discrete distribution for the arrival time AT_2. While in the summation method the time window effect on the risk (risk by summation) is discussed regarding three cases. The risk is defined previously as $P(AT_2 > b_2) = P\{\max\{AT_1, a_1\} + TT_1 > b_2\}$, hence we propose that the time window boundaries a_1 and b_2 will affect the power of the estimate. We use the risk by accumulation as the benchmark to investigate the behaviour of the risk estimation by the summation method because the risk by accumulation is close to the real risk.

It is straightforward to see that the closer the average arrival time μ is to the 1st task lower limit of the time window, a_1, the more the start time distribution changes. If the lower limit $a_1 = \mu - 2\sigma$, the effect of the time window is small as is shown in Figure 3.11 (b) comparing the original arrival time distribution shown in Figure 3.11 (a). Whilst if the time window is closer to the average arrival time, such as $a_1 = \mu - 0.5\sigma$, the effect of the time window to the start time distribution is shown in Figure 3.11 (c), the shape of the start time is quite different from the arrival time. As for the scenario when the mean of the arrival time μ is much earlier than the time window a_1, *e.g.*, $a_1 = \mu + 2\sigma$, the start time is at time a_1 with a high probability so that the variance of the arrival time can be omitted.



(a) Same as the arrival time when no time windows applied

(b) $a_1 = \mu - 2\sigma$

(c) $a_1 = \mu - 0.5\sigma$

(d) $a_1 = \mu + 2\sigma$

Figure 3.11 Time window effect

The 1st task and travel time TT_1 follow the same normal distribution $N(\mu, \sigma^2)$ and let $a_1 = p\sigma$, $b_2 = q\sigma_2$, where $\sigma_2 = \sqrt{2}\sigma$ is the standard deviation for the estimated normal distribution of the arrival time at the 2nd task which is $N(2\mu, 2\sigma^2)$. Without loss of generality, assume the normal distribution is standard, *i.e.*, $\mu = 0,\ \sigma = 1$, thus $a_1 = p, b_2 = q \cdot \sqrt{2}$. Then the relationship between the risk by accumulation and risk by summation in terms of p and q will be discussed as follows. 5000 pairs of p and q are randomly chosen between 0 and 3 in Java to construct samples by using the Random function in the package java.util.Random, and the risk by accumulation for the 2nd task is calculated by the accumulation method, while the risk by summation is calculated by $P(AT_2 > q)$ where $AT_2 \sim N(0,2)$ in Java using the package jsc.distributions.Normal.

Furthermore, we assume that the mean of the arrival time AT_2 is earlier than the upper limit of the time window at the 2nd customer b_2, and the difference between AT_2 and b_2 is at most 3 standard deviations of the arrival time AT_2, which is to say b_2 is at the right tail of the distribution of AT_2, under this assumption the risks are comparable and hence this assumption is used for all scenarios. If the mean of the arrival time is after b_2, then there is at least 50% risk to miss the appointment, which will be regarded as high risk in the scheduling process.

Now first we discuss the overall scenario where the mean of the arrival time AT_1 varies before or after the lower limit of the time window a_1 within 3 standard deviations. To analyse the data, firstly the risk by accumulation is plotted against the risk by summation which is demonstrated in Figure 3.12, the two risks show a significant linear relationship. The correlation between these two risks is 0.997, which is significant at the 0.01 level (2-tailed) as is shown in Table 3.13. The results provide evidence that the risk by summation can be used instead of the risk by accumulation. This will save a lot of time in a scheduling engine.

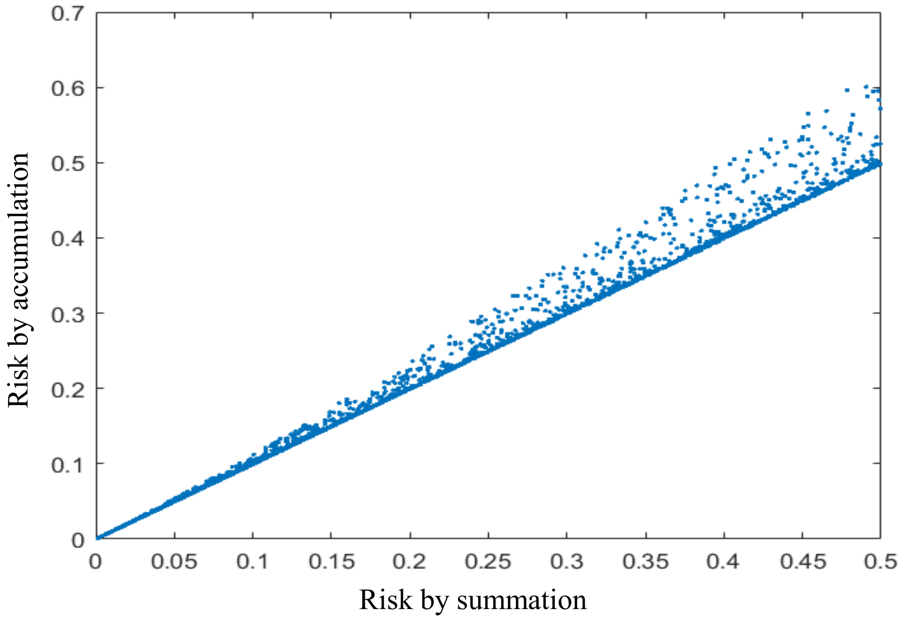


Figure 3.12 Risk by accumulation and risk by summation

Table 3.13 Risk correlations

		Real	Estimated
Risk by accumulation	Pearson Correlation	1	.997**
	Sig. (2-tailed)		.000
	N	5000	5000
Risk by summation	Pearson Correlation	.997**	1
	Sig. (2-tailed)	.000	
	N	5000	5000

**. Correlation is significant at the 0.01 level (2-tailed).

In statistical modelling, regression analysis is widely used for estimating the relationships among variables. Linear regression includes the method of least squares for modelling and analysing the relationship between a dependent variable and one or more independent variables. More specifically, regression analysis helps one understand how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed.

Table 3.14 Regression Model I Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-.001	.000		-5.443	.000	-.002	-.001
	Risk by summation	1.033	.001	.997	954.455	.000	1.031	1.035

a. Dependent Variable: real

Table 3.15 Regression Model I Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.997[a]	.995	.995	.01111	2.006

a. Predictors: (Constant), estimate
b. Dependent Variable: real

Based on the high correlation between the risk by accumulation and the risk by summation, a regression model I can be obtained as shown in Tables 3.14 and 3.15. The regression equation is as follows,

$$\text{regressional risk by accumulation } = -0.0012 + 1.0033 \cdot \text{risk by summation},$$

and the p-values for these coefficients are 0.000, 0.000. Figure 3.13, show the residual values, the differences between the conditionally-imputed values for risk by accumulation and the fitted values, *i.e.*,

$$\text{residual} = \text{risk by accumulation} \ - \text{regressional risk by accumulation}.$$

In Figure 3.13, most residual values fall between -0.02 and 0.02 which are relatively small. The adjusted R-square is 0.9945 for this model and it can be interpreted that the risk by summation can predict the risk by accumulation well.

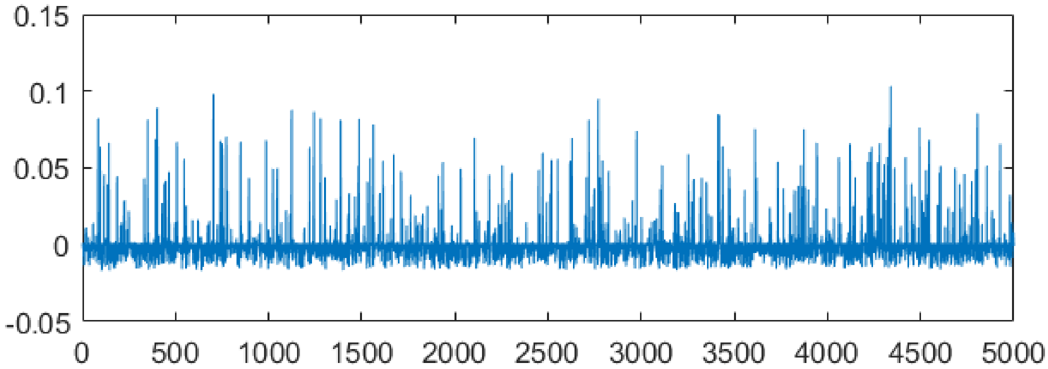


Figure 3.13 Risk Residual of regression model I

Moreover, in Figure 3.12, increasing numbers of extreme values can be observed as risks increases, these outliers force us to consider the behaviour of the risk residual, which is the value of the risk by accumulation minus the risk by summation, then the risk residuals are plotted according to risks by summation in Figure 3.14. As shown in the figure, most residuals are pretty symmetrically distributed around 0 when the risks are relatively small *i.e.*, less than 20%. Then as the risk increases the residual increases but even when the risk hits 50% the residual is not more than 0.1.

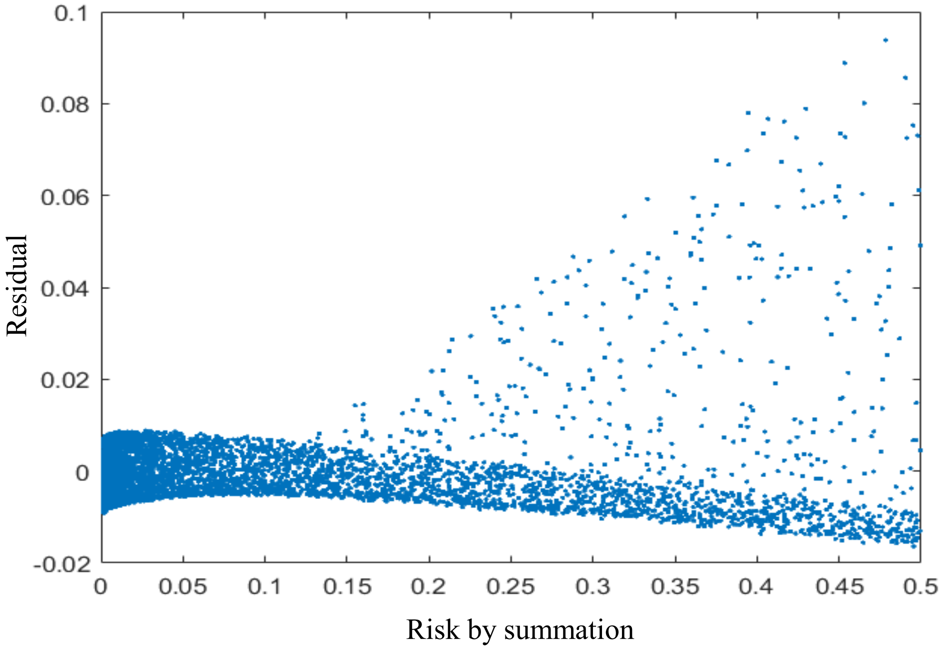


Figure 3.14 Risk Residual

However, if we discover a more accurate risk by summation model, and p and q get closer to μ which is 0 here, the difference between two risks is more significant. As discussed before, p is the low limit of the time window expressed as the number of standard deviation from the mean arrival time at the first customer, q is the upper limit of the next task time window represented as the number of standard deviation from the mean arrival time at the second customer. Then different scenarios may appear with regard to different p, q values. Therefore, to approximate the risk by accumulation more accurately in terms of the risk by summation, the time-window boundaries p and q can be included.

Overall, the regression model II (as shown in Tables 3.16 and 3.17) obtained by the 5000 sample values is that regressional risk by accumulation

$$= 0.0011 + 0.0055p + 0.0027q + 1.048 \cdot \text{risk by summation},$$

and the p-values for these coefficients are 0.2477, 0.0000, 0.0000 and 0, which means the intercept term is not significant to be non-zero. And the residual for this regression is shown in Figure 3.15, the adjusted R-square is 0.9956, which may mean that this model is just a little better than Model I but as two more explanatory factors are introduced to calculate the risk, this estimate becomes much more complicated.

Table 3.16 Regression Model II Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.001	.001		1.156	.248	-.001	.003
	P	.006	.000	.032	33.794	.000	.005	.006
	Q	.003	.000	.016	6.451	.000	.002	.004
	Risk by summation	1.048	.002	1.012	420.032	.000	1.043	1.053

a. Dependent Variable: real

Table 3.17 Regression Model II Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.998[a]	.996	.996	.00999	1.990

a. Predictors: (Constant), estimate, p, q

b. Dependent Variable: real

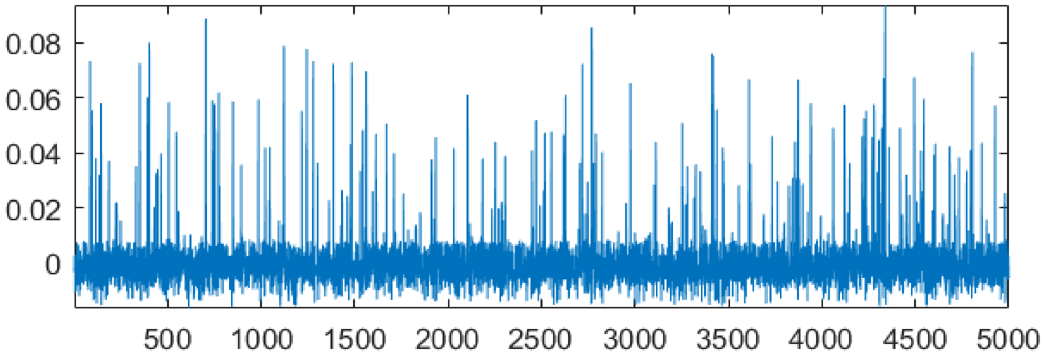


Figure 3.15 Risk Residual of regression model II

If we consider the scenario that $p \in [-3, -1)$ and $q \in (0,3]$, as an example shown in Figure 3.11 (b). Based on the analysis of 5000 new random samples in this scenario, we only treat the risk by summation as the explanatory variable (the independent variable), and then the regression model III shows that

$$\text{regressional risk by accumulation} = -0.0002 + 1.0011 \cdot \text{risk by summation},$$

where p-values for the coefficients are 5.9×10^{-36} and 0. The residuals are plotted in Figure 3.16, which shows that the absolute values of residuals are mostly less than 0.5%. Moreover, the R-square for this model is 1.0, which means the risk by summation explains the risk by accumulation well. Therefore, for the scenario that the earliest time window $a_1 \in [\mu - 3\sigma, \mu - \sigma)$, it is reasonable to use the normal distribution estimation model I for the arrival time AT_2 that

$$\mu(AT_2) = \mu(ST_1) + \mu(TT_1)$$

$$\sigma^2(AT_2) = \sigma^2(ST_1) + \sigma^2(TT_1).$$

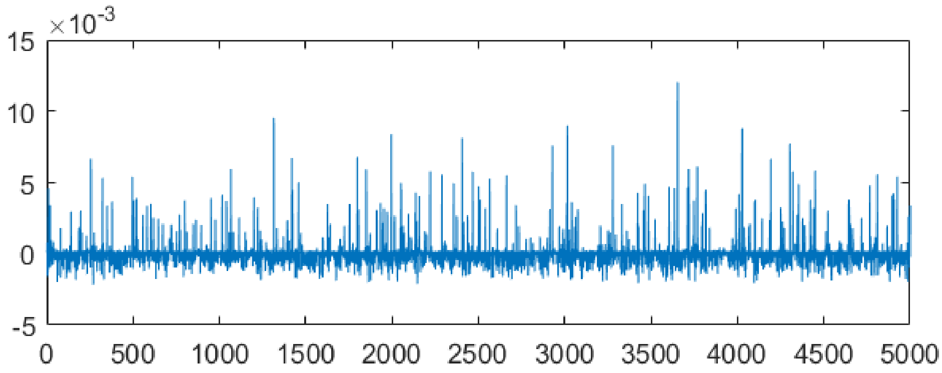


Figure 3.16 Risk residuals of regression model III

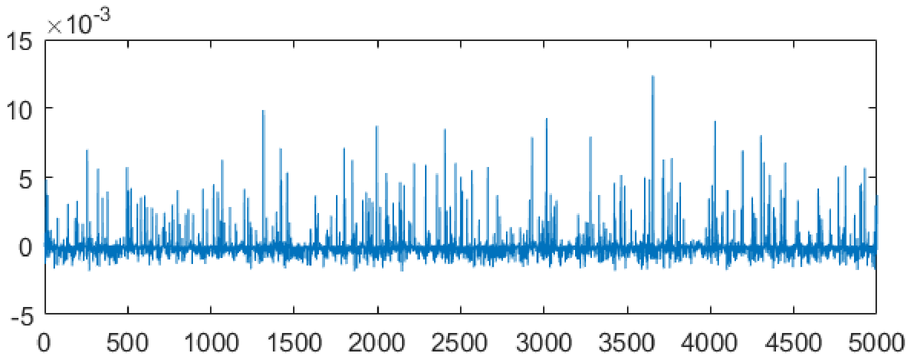


Figure 3.17 Errors of estimation model I for p∈[-3,-1)

And consider the approximation absolute error (Golub and Van Loan, 2013) that

$$\text{error} = \text{risk by accumulation} - \text{risk by summation}.$$

The errors show values as small as 0.5% in Figure 3.17, which are similar to the residuals of regression model III, because the regression model is close to the estimation that

$$\text{regressinal risk by accumulation} = \text{risk by summation}.$$

Then we consider the scenario that $p \in (1,3]$ and $q \in (0,3]$, which is shown in the example of Figure 3.11 (d), the variance of the arrival time is omitted because of the long waiting time. Therefore the risk by summation is calculated by the summation method where the lower bound of the time window, a_1 and the mean of the task and travel time TT_1 are added as the mean of the arrival time AT_2, and the variance of AT_2 is same as the variance of TT_1. *i.e.*, the estimation model II for the arrival time AT_2 is

$$\mu(AT_2) = a_1 + \mu(TT_1)$$

$$\sigma^2(AT_2) = \sigma^2(TT_1).$$

Based on the risk by summation from the above model and the risk by accumulation, the regression model IV is given that

$$\text{regressional risk by accumulation} = 0.0020 + 1.0153 \cdot \text{risk by summation},$$

where p-values for the coefficients are 5.4×10^{-74} and 0. The residuals are mostly within 2% and the R-square for this model is 0.9985, which means the risk by summation can explain the risk by accumulation well. Thus, for the scenario that the lower bound of the time window, $a_1 \in (\mu + \sigma, \mu + 3\sigma]$, it is also reasonable to use the estimation that

$$\text{regressinal risk by accumulation} = \text{risk by summation}.$$

and the errors according to the risks by accumulation are shown in Figure 3.18.

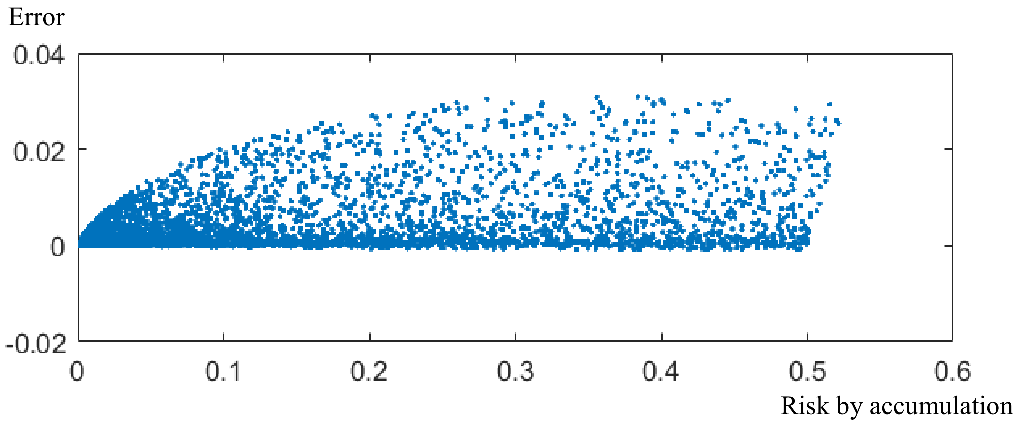


Figure 3.18 Errors of estimation model II for p∈(1,3] according the risk by accumulation

From the figure we can see that the errors are not as small as the ones in the scenario $p \in [-3,-1)$ because of ignoring the start time variance, however, for this scenario the technician arrives much earlier than the time window, therefore the risk is usually relatively small and when the risks are smaller than 10%, the errors are less than 2%. Moreover, if we consider the relative error (Golub & Van Loan, 2013) which is

$$\text{relative error} = \frac{\text{risk by accumulation} - \text{risk by summation}}{\text{risk by accumulation}},$$

then the relative errors are less than 10% of the risk regarding to the large values in absolute errors. Thus the conclusion is that the estimation model II can be used for the scenario when the lower bound of the time window$a_1 \in (\mu + \sigma, \mu + 3\sigma]$.

Then as for the scenario $p \in [-1,1]$ and $q \in (0,3]$, we introduce a normal distribution estimation for the start time. As we analyse before, the term $\max\{AT_{k2}, a_{i_2}\}$ makes the risk not aligning with a normal distribution. It is the maximum value of a random variable and a constant number, then if we expand the constant number to a random variable, a research (Nadarajah and

Kotz, 2008) gave the exact distribution of the maximum and minimum of two Gaussian random variables. Suppose $X_1$and X_2 are Gaussian random variables, then the mean and variance of $X = \max\{X_1, X_2\}$ are

$$E(X) = \mu_1 \Phi\left(\frac{\mu_1 - \mu_2}{\theta}\right) + \mu_2 \Phi\left(\frac{\mu_2 - \mu_1}{\theta}\right) + \theta \phi\left(\frac{\mu_1 - \mu_2}{\theta}\right),$$

$$E(X^2) = (\sigma_1^2 + \mu_1^2)\Phi\left(\frac{\mu_1 - \mu_2}{\theta}\right) + (\sigma_2^2 + \mu_2^2)\Phi\left(\frac{\mu_2 - \mu_1}{\theta}\right) + (\mu_1 + \mu_2)\theta\phi\left(\frac{\mu_1 - \mu_2}{\theta}\right),$$

where $\theta = \sqrt{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}$, ρ is the correlation between $X_1$and X_2, $\Phi(\cdot)$ and $\phi(\cdot)$ are the PDF and CDF of the standard normal distribution respectively. Nadarajah and Kotz (2008) also stated that if the standard deviations σ_1, σ_2 of the two Gaussian random variables are identical, the Gaussian random variable with the mean $E(X)$ and variance $E(X^2) - E^2(X)$ can well approximate the distribution of $X = \max\{X_1, X_2\}$. As more different between σ_1 and σ_2, the estimate gets worse.

In terms of our risk model, a constant which is the lower limit of the time window is used replacing one of the Gaussian random variables and the correlation $\rho = 0$, therefore the parameters of the normal distribution estimation for $ST_1 = \max\{AT_1, a_1\}$ are

$$E(ST_1) = \mu_1 \Phi\left(\frac{\mu_1 - a_1}{\sigma_1}\right) + a_1 \Phi\left(\frac{a_1 - \mu_1}{\sigma_1}\right) + \sigma_1 \phi\left(\frac{\mu_1 - a_1}{\sigma_1}\right), \tag{3.1}$$

$$E(ST_1^2) = (\sigma_1^2 + \mu_1^2)\Phi\left(\frac{\mu_1 - a_1}{\sigma_1}\right) + a_1^2 \Phi\left(\frac{a_1 - \mu_1}{\sigma_1}\right) + (\mu_1 + a_1)\sigma_1\phi\left(\frac{\mu_1 - a_1}{\sigma_1}\right), \tag{3.2}$$

where μ_1 and σ_1 are the mean and standard deviation of the arrival time AT_1 (Ehmke *et al.*, 2015). Then the risk by summation of the 2$^{\text{nd}}$ task is obtained via the ST_1 estimation and the normal distribution of the task and travel time

TT_1, thus the normal distribution estimation model III for the arrival time AT_2 is

$$\mu(AT_2) = E(ST_1) + \mu(TT_1) \tag{3.3}$$

$$\sigma^2(AT_2) = E(ST_1^2) - E^2(ST_1) + \sigma^2(TT_1). \tag{3.4}$$

Comparing with the risk by the accumulation method, a regression model IV is

$$\text{regressional risk by accumulation} = 0.0033 + 0.9702 \cdot \text{risk by summation},$$

with p-values of parameters are both 0 and R-square is 0.9997. Therefore, we also can conclude that the estimation that

$$\text{risk by accumulation} = \text{risk by summation}$$

works for the estimation model III. The errors for this estimation are at most 1% as shown in Figure 3.19.

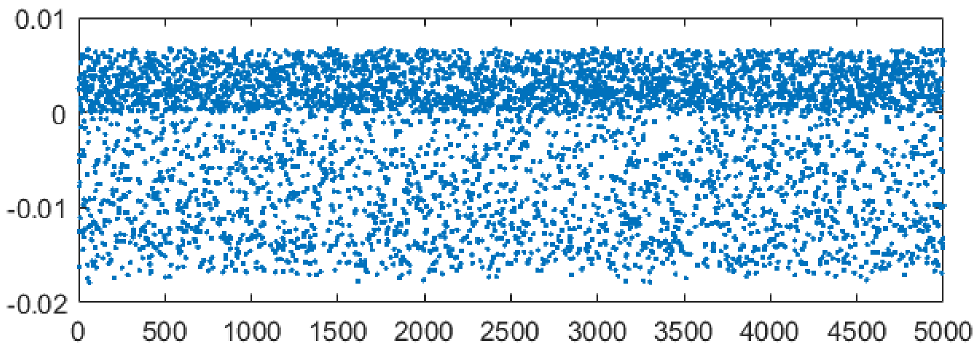


Figure 3.19 Error of estimation model III for p∈[-1,1]

Overall, it can be concluded that for different relations between the arrival time AT_1 and the lower limit of the time window, a_1, the estimation model I, II and III can approximate the distribution of the start time ST_1 as a normal distribution well.

3.5.4 Gamma distribution estimation and addition

From analysing the data from the communication provider we found that some task duration time follows Gamma distributions. Also, it is widely known that the sum of two independent normally distributed random variables is normal, where the mean is the sum of the two means, and its variance is the sum of the two variances. It would be very useful for real-world problems if we could find a similar property for Gamma distributions.

Consider random variables X_i $i = 1, \dots, n,$ having Gamma distributions with shape parameter $\alpha_i > 0$ and scale parameter $\beta_i > 0,$ and denote

$$X_i \sim \Gamma(\alpha_i, \beta_i).$$

Then the PDF of X_i shows (Papoulis and Pillai, 2002)

$$p(x_i) = \frac{x_i^{\alpha_i - 1}}{\beta_i^{\alpha_i} \Gamma(\alpha_i)} e^{-x_i/\beta_i}, \tag{3.5}$$

and the mean and variance are

$$\mu_i = E(X_i) = \alpha_i \beta_i,$$

$$\sigma_i^2 = V(X_i) = \alpha_i {\beta_i}^2.$$

First there is a property of the summation (Papoulis and Pillai, 2002), if all X_i are independent and their distributions have the same scale parameter, *i.e.*, $\beta_i = \beta$ for all i, then

$$Y = \sum_{i=1}^{N} X_i \sim \Gamma\left(\sum_{i=1}^{N} \alpha_i, \beta\right).$$

Second, as for the cases where X_i, $i = 1, \dots, N$ are independent but have different scale parameters, Mathai (1982) has given a number of expressions for the density of $Y = X_1 + X_2 + \cdots + X_n$ in terms of different techniques, different assumptions of α_i or β_i. Mathai and Saxena (1978) expressed the density according to a confluent hypergeometric function in $n - 1$ variables where α_i and β_i are distinct. Then Moschopoulos (1985) diversified Mathai's method and led to a single Gamma series for the density and distribution function of Y. The exact density of Y is given by the following theorem.

Theorem if $\{X_i\}, i = 1, \dots, n$ are independently distributed as (3.5), then the density of $Y = X_1 + X_2 + \cdots + X_n$ can be expressed as

$$g(y) = C \sum_{k=0}^{\infty} \frac{\delta_k y^{\rho+k-1} e^{-\frac{y}{\beta_1}}}{\Gamma(\rho + k)\beta_1^{\rho+k}}, \qquad y > 0,$$

and 0 elsewhere, where $\rho = \sum_{i=1}^{n} \alpha_i$, $C = \prod_{i=1}^{n} (\beta_1/\beta_i)^{\alpha_i}$, and

$\delta_{k+1} = \frac{1}{k+1} \sum_{i=1}^{k+1} i \gamma_i \delta_{k+1-i}, k = 0,1,2, \cdots.$

Furthermore, as Stewart *et al.* (2006) said if the convolution is an end itself, these methods are satisfactory, if it is just a step in a continuing argument, a simple closed expression to approximate the convolution would be much more useful than a numerical method, even at the cost of a small approximation error. Consequently, they proposed an approximation of Y as a Gamma distribution. Suppose $Y = X_1 + X_2 + \cdots + X_n$ with independent $X_i \sim \Gamma(\alpha_i, \beta_i)$; the mean and variance are

$$\mu = E(Y) = \sum_{i=1}^{n} \alpha_i \beta_i,$$

$$\sigma^2 = V(Y) = \sum_{i=1}^{n} \alpha_i {\beta_i}^2.$$

Then the approximate distribution for Y is

$$\Gamma(\alpha, \beta), \text{with } \alpha = \frac{\mu^2}{\sigma^2}, \beta = \frac{\sigma^2}{\mu}. \tag{3.6}$$

This approximation fits the convolution well if the number of scale parameters β_i is less than 10 and if the shape parameters α_i are greater than 1. The researchers also compare the skewness and kurtosis, the PDF and CDF and the 95% percentage points between the approximate and exact distributions to illustrate the accuracy of the approximation. Note that this approximation aligns with our problem because there are always less than ten tasks for each technician and the shape parameters are on average 3.5 according to the analysis of the task duration historical data.

Table 3.18 Sum of Gamma distributions

	Task 1	Task 2	Task 3	Sum
Alpha	3.5000	3.5000	3.5000	9.0000
Beta	0.0700	0.0350	0.0233	0.0300
Mean	50	100	150	300
Mode	36	71	107	267

An example is given to show the approximation fits the data. Let us suppose there are three tasks and their parameters and moments are as shown in Table 3.18 and the approximation Gamma distribution parameters are as stated in the Sum column, where the approximation is obtained by (3.6). The task durations of Tasks 1, 2 and 3 are randomly generated and added together, 5000 samples are generated to constitute the real task duration distribution of their sum, which is demonstrated as the blue wave. The estimated Gamma distribution obtained by the above approximation is shown as the red curve in Figure 3.20.

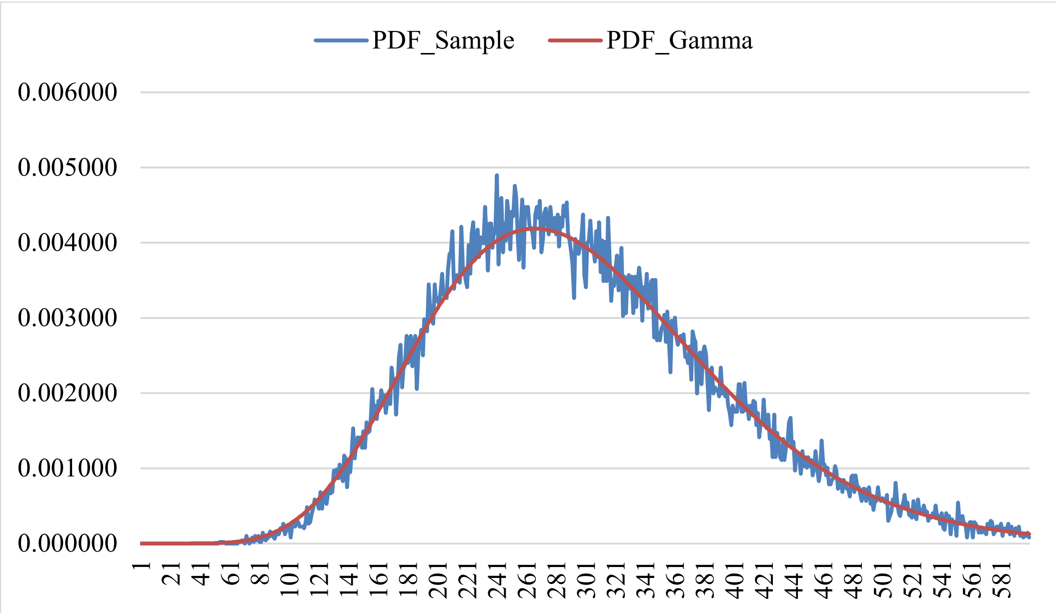


Figure 3.20 Approximation of a Gamma distribution

Then from Figure 3.20, it is sufficient to say that the approximation Gamma distribution fits the real Gamma distribution summation well. Therefore, it is reasonable to use this approximation to calculate the risks if the task duration is considered as a Gamma distribution.

The above estimation of the summation works well when the technician can start the task immediately when he/she arrives at the customer's site, *i.e.*, the beginning points of the two Gamma distributions are at the same time points, which means the origin point of the arrival time is after the lower bound of the task time window. However, in real-life circumstances, the lower bound of a time window influences and shifts the task duration distribution a period behind the start time point of the Gamma distribution for the arrival time, as is the blue curve shown in Figure 3.20. Therefore, the distribution of the start time cannot be estimated by the previous method directly. However, it can be seen that the distribution of the start time (*i.e.*, the variable which is the summation of the arrival time and task duration) follows a Gamma distribution with the same shift from the origin point as the task start time.

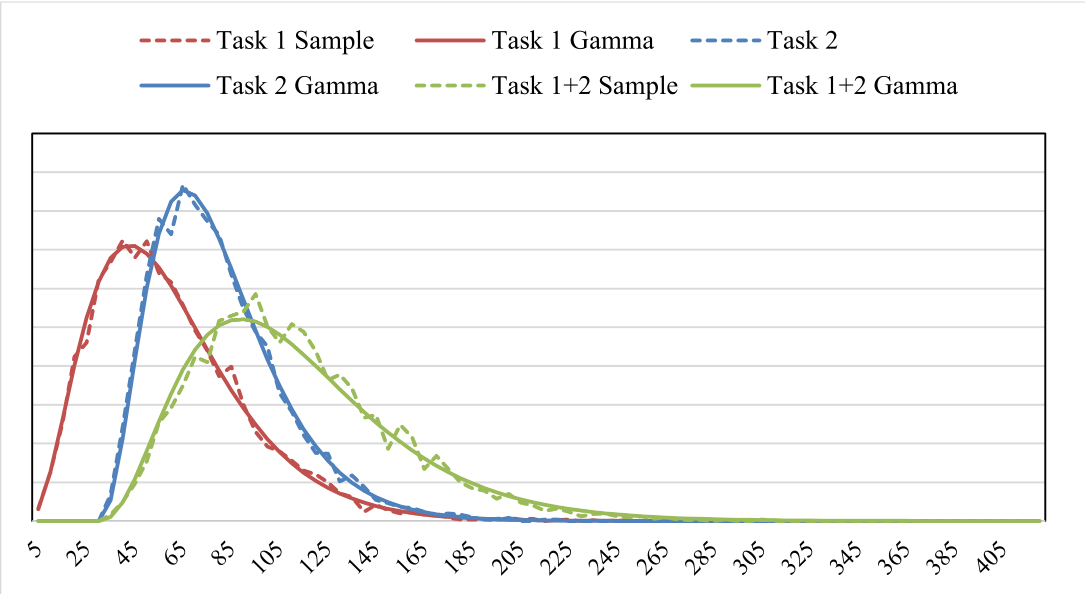


Figure 3.21 Sum of Gamma distributions for two tasks operation start times

Table 3.19 Sum of Gamma distributions with shift (minutes)

	Task 1	Task 2	Sum
alpha	3.500	3.500	3.689
beta	0.058	0.058	0.040
mean	60	50	92
shift	0	30	30

More specifically, suppose the red curve in Figure 3.21 demonstrates the Gamma distribution of Task 1 and the corresponding parameters are shown in Table 3.19, and without loss of generation, the uncertainty of the travel times can be neglected for simplification, and suppose the origin point for the Gamma distribution of the arrival time is also the origin point for the distribution of the operation time for Task 2. So the distribution of Task 1 stands for the distribution of the arrival time, and the blue curve shows the Gamma distribution of Task 2 operation time. Because of the lower bound of the time window, there is a shift (suppose it is 30 minutes) regarding the Task 2 operation time, which means the arrival time is 30 minutes earlier, then the technician has to wait for 30 minutes to start Task 2.

For example, a sample gives that the technician works on Task 1 for 25 minutes and Task 2 starts from 30 minutes, and then he spends 40 minutes on Task 2, so the arrival time at Task 3 is 30 + 40 = 70 minutes. Whereas, if a sample gives that the technician stays at Task 1 for 45 minutes and Task 2 for 40 minutes, then the arrival time at Task 3 is 45 + 40 = 85 minutes. Consequently, by taking 5000 pairs of samples, we may have a population of the summation for Task 1 and 2 operation times, where its distribution is shown as the green dash curve in Figure 20. Then from those sample figures one may obtain the parameters α and β if the arrival time is supposed to be a Gamma distribution. The solid green curve shows the PDF of the Gamma distribution in terms of the α and β obtained from sample figures. We notice that the solid green curve fits the sample distribution well, which helps us assessing that the arrival time at Task 3 can be regarded as a Gamma distribution with the shift as the same as the lower bound of the time window of Task 2.

This estimating approach is based on real data and fast computation-wise. In the heuristic searching process, we use this sampling method to calculate the parameters for the Gamma distribution with shift as an estimation of the arrival time distribution, and use the estimation equations (3.6) in the scenario that the origin point of the distribution for the arrival time is later than the lower bound of the time window for the task operation time, in order to save calculation time.

Gamma distributions are sometimes the more accurate choice to model distributions occurring in real-life scenarios, *e.g.*, for certain travel and task times. In this section, we outlined how this type of distribution, instead of the Normal distribution, can be used in our approach to risk modelling. In the next chapter we will describe the metaheuristic used to find the optimal schedule.

3.6 Factors affecting risks

Based on the risk definition, several characteristics of the risks can be observed.

Proposition 2.1 Risk increases as it propagates over time along the task schedule.

As mention previously for each technician the risk increases as it propagates. To illustrate this, suppose the means of the arrival time distribution are all equal to 0, all having the same distribution type, *i.e.*, normal. Figure 3.22 shows that when the means of the arrival times are all 1 hour before their corresponding time windows, the risks, which are represented as the areas greater than 1, get larger from the 2^{nd} task to the 4^{th} task. This is because the variance increases, as each task has its own uncertainty and these uncertainties accumulate, there is more uncertainty of arrival times at later tasks.

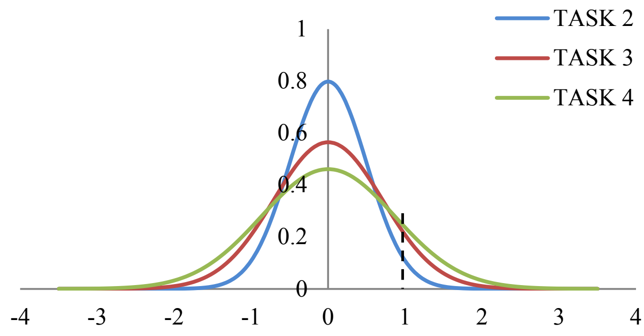


Figure 3.22 Risk comparisons

Proof 2.1 Base 1: Let the mean arrival time for technician k at task i_1 be $\mu\left(AT_{ki_1}\right)$ and the mean travel time $\mu\left(d_{ki_0i_1}\right)$ from the depot to first task for any engineer k, we have omitted the k for simplicity of notation, but this hold for any engineer's schedule. Then $\mu\left(AR_{ki_1}\right) = \mu(d_{ki_0i_1})$. The standard deviation of the arrival task i_1 is the variation of the travel from the depot to the first task, $\sigma\left(AR_{ki_1}\right) = \sigma(d_{ki_0i_1})$. Then the mean and standard deviation of the arrival time for the second task are $\mu\left(AR_{ki_2}\right) = \mu\left(\delta_{ki_1}\right) + \mu(d_{ki_1i_2})$ and $\sigma\left(AR_{ki_2}\right) = \sigma\left(\delta_{ki_1}\right) + \sigma(d_{ki_1i_2})$.

Suppose the latest time to start the task as always b minutes after the mean start time of the task, and $\sigma\left(d_{ki_1i_2}\right) > 0$, by definition $\sigma\left(AR_{ki_2}\right) > \sigma\left(AR_{ki_1}\right)$ and $AR_{ki_1} \sim N\left(\mu\left(AR_{ki_1}\right), \sigma\left(AR_{ki_1}\right)\right), AR_{ki_2} \sim N\left(\mu\left(AR_{ki_2}\right), \sigma\left(AR_{ki_2}\right)\right)$ then

$$\int_{\mu\left(AR_{ki_2}\right)+b}^{\infty} f_{ki_2}\left(AR_{ki_2}\right) dAR_{ki_2} > \int_{\mu\left(AR_{ki_1}\right)+b}^{\infty} f_{ki_1}\left(AR_{ki_1}\right) dAR_{ki_1}.$$

Induction step: $\forall$ task n in any one schedule, will have a higher risk of failure than its previous task $n-1$ if $\sigma\left(d_{ki_{n-1}i_n}\right) > 0$, and then

$$\int_{\mu(AR_{ki_n})+b}^{\infty} f_{ki_n}(AR_{ki_n})dAR_{ki_n} > \int_{\mu(AR_{ki_{n-1}})+b}^{\infty} f_{ki_{n-1}}(AR_{ki_{n-1}})dAR_{ki_{n-1}}$$

holds as

$$\sigma(AR_{ki_n}) = \sigma(\delta_{ki_{n-1}}) + \sigma(d_{ki_{n-1}i_n}) > \sigma(\delta_{ki_{n-2}}) + \sigma(d_{ki_{n-2}i_{n-1}}) = \sigma(AR_{ki_{n-1}}).$$

Proposition 2.2 The wider the time window, the lower the risk.

Proof 2.2 For any task n if the upper bound of the time window $b1 < b2$ another upper bound then

$$\int_{\mu(AR_{ki_n})+b1}^{\infty} f_{ki_n}(AR_{ki_n})dAR_{ki_n} > \int_{\mu(AR_{ki_n})+b2}^{\infty} f_{ki_n}(AR_{ki_n})dAR_{ki_n}$$

holds, hence the narrower the time window the higher the risk.

To illustrate the effect on risks of having different time windows, we let the widths of time windows change whilst everything else remains the same. It is clear to see that the risk decreases as the time window gets wider. This is as expected because a larger time window gives the technician more opportunity to arrive at the customer in time. For example, Figure 3.23 shows that the arrival times are mostly around 12:00, and the risk if the time window is from 11:00 to 13:00 is higher than the one if the time window is from 10:30 to 13:30. Therefore, one way to reduce risks is to expand the time window to a reasonable extent, but a large time window may not be convenient for customers.

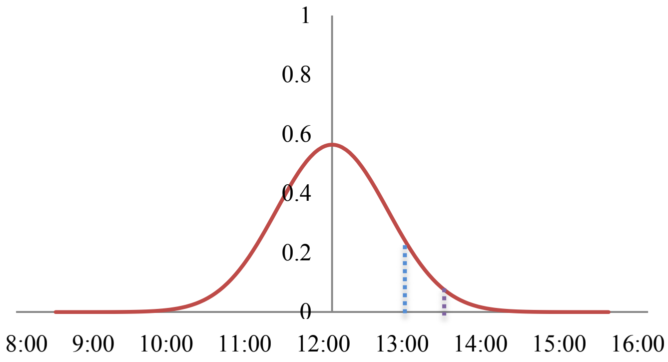


Figure 3.23 Time window width effect

Proposition 2.3 The less uncertainty in the task and travel times, the lower the risk.

Proof 2.3 Let $\sigma_1(AR_{ki_n}) = \sigma_1(\delta_{ki_{n-1}}) + \sigma_1(d_{ki_{n-1}i_n})$ and $\sigma_2(AR_{ki_n}) = \sigma_2(\delta_{ki_{n-1}}) + \sigma_2(d_{ki_{n-1}i_n})$ denote two different standard deviations for any task n. If $\sigma_1(AR_{ki_n}) < \sigma_2(AR_{ki_n})$ and $AR_{ki_n} \sim N\left(\mu(AR_{ki_n}), \sigma_1(AR_{ki_n})\right)$, $AR'_{ki_n} \sim N\left(\mu(AR_{ki_n}), \sigma_2(AR_{ki_n})\right)$. Then

$$\int_{\mu(AR'_{ki_n})+b}^{\infty} f_{ki_n}(AR'_{ki_n}) dAR'_{ki_n} > \int_{\mu(AR_{ki_n})+b}^{\infty} f_{i_n}(AR_{ki_n}) dAR_{ki_n},$$

hence less uncertainty leads to lower risk.

The variations of task and travel times are another factor affecting the risks in our model, suppose all else equal, if the standard deviation is larger, the distribution of the task duration, as well as the arrival time, is flatter, the risk is higher. As it is shown in Figure 3.24, the green shaded area which represents the risk with larger standard deviation is bigger than the blue one which stands for the risk with relatively small standard deviation. It models the real-world fact that with high task duration uncertainty the risk tends to be higher.

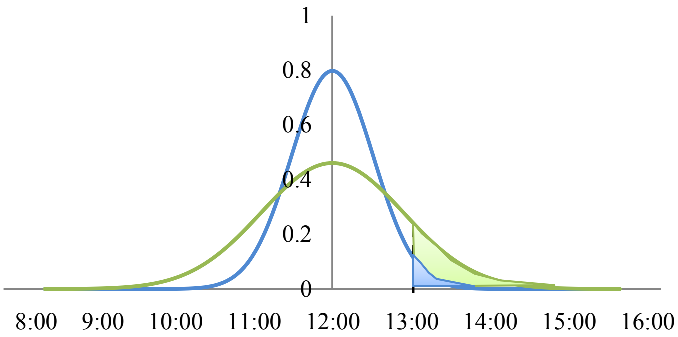


Figure 3.24 Task uncertainty effect

Additionally, a good estimation of the task duration is beneficial to reduce risks. One way to better estimate the task duration value may be to identify samples of tasks and group these task samples according to more specific characters such as skills, locations, types, *etc.* The other approach could be behavioural, knowledge or human management process oriented, for instance, to improve technician abilities and skills, so that it reduces the uncertainty upon work effort completion.

In a nutshell, we give the risk expression for our problem in this chapter. The Simpson's rule and Monte Carlo method, accumulation method and summation method are proposed to calculate risks, followed by the tests on the reliability of each method. Furthermore, real data are analysed in Section 3.5 and shown to follow the assumptions of our proposed method *i.e.*, the summation method is suitable for risk calculation. Task duration is shown to follow the normal distribution and a method for calculating travel time through coordinates is proposed.

Additionally, regression models are used to demonstrate the effectiveness of the estimation of the summation method. And finally, factors affecting risk are discussed on the parameter basis. Based on the knowledge of risks in this problem, risk minimisation mathematical models are proposed in the following chapter.

Chapter 4

Risk Minimisation Models

In this chapter, mathematical models to minimise the risk of missing an appointment and the risk of engineers' working beyond their working days are formulated; both non-linear and linear models are considered. Additionally, how the model changes if the tasks have different priorities and if the risk is defined as the task needs be completed within the time window rather than just starting, is also examined.

In the real life, most service activities tend to be uncertain, for instance, new tasks arrive, tasks do not match engineers or tools, tasks take longer than expected (Herroelen and Leus, 2005). Therefore, this research concerns a service scheduling problem to minimise the risk of missing appointments, where the risk arises from these uncertainties. Moreover, it becomes increasingly important for firms to focus on customer satisfaction, rather than only on offering good quality goods and services. A better schedule can help them improving the level of customer satisfaction and consequently becoming more competitive and attracting more customers.

This research focuses on the stochastic vehicle routing problem, in which technicians drive to customer sites to provide services. In the problem we assume that service times and travel times are stochastic, and a time window is required to start the service for each customer.

As we can see from the literature review on risks, most previous relevant research uses a chance-constrained approach to the problem. Some consider the probability of route duration exceeding the threshold of the driver's workload while others set restrictions on the probability of individual time window constraints being violated (Li *et al.*, 2010). However, their objectives are still some forms of traditional routing costs whilst we try to minimise the risks.

4.1 A non-linear risk minimisation model

To model the problem, we first introduce some notion. There are K available vehicles each used by a technician. The locations and the road network can be expressed on a complete graph $G = (V_0, A)$, where $V_0 = \{0, \dots, N\}$ is a set of vertices which denote locations and $A = \{(i,j): i,j \in V_0, i \neq j\}$ is a set of arcs. Vertex 0 represents the depot. The set of customers is $V = V_0 \backslash \{0\} = \{1, \dots, N\}$. Each customer $i \in V$ has a time window $[a_i, b_i]$ for starting the service. If the technician arrives at customer i before a_i, it is necessary for him/her to wait until a_i. Further notations are listed below.

Parameters

- M a large number;
- C the maximum number of customers that can be served by each technician;
- b_0 the maximum daily work time for each technician;
- $\mathcal{K}$ the set of available technicians $\mathcal{K} = \{1, \dots, K\}$;
- $\mathcal{K}_i$ the set of technicians who can perform the task for customer i, a subset of $\mathcal{K}$;
- d_{kij} the travel time of technician k between locations i and j;

δ_{ki} the uncertain work time of technician k at the customer i, a random parameter with known and independent probability density; for modelling convenience, we define $\delta_{k0} = 0$.

$ST_{k0} = 0$, the start time of technician k at the depot;

Variables

x_{kij} a binary variable equal to 1 if technician k travels directly through arc (i, j) and 0 otherwise;

AT_{ki} the arrival time of technician k at customer i;

AT_{k0} the arrival time of technician k at the depot after completing all his/her tasks of the day;

ST_{ki} the service start time of technician k at customer i;

R_{ki} the risk of technician k arriving late at customer i;

R_{k0} the risk of technician k not finishing work within the maximum work time.

R_{max} the maximum among all risks in the schedule.

The model for the problem can then be formulated below:

$$\text{minimise} \;\; w \cdot R_{\max} + \sum_{k \in \mathcal{K}} \sum_{i \in V_0} R_{ki} \tag{4.1}$$

Subject to:

$$\sum_{j \in V} x_{k0j} = \sum_{i \in V} x_{ki0} \leq 1, \;\; \forall\, k \in \mathcal{K} \tag{4.2}$$

$$\sum_{j\in V_0} x_{kij} = \sum_{j\in V_0} x_{kji}, \qquad \forall\, i \in V, \forall k \in \mathcal{K} \tag{4.3}$$

$$\sum_{k\in\mathcal{K}} \sum_{j\in V, j\neq i} x_{kij} = 1, \qquad \forall\, i \in V \tag{4.4}$$

$$\sum_{i\in V} \sum_{j\in V} x_{kij} \le C + 1, \quad \forall\, k \in \mathcal{K} \tag{4.5}$$

$$ST_{ki} + \delta_{ki} + d_{kij} - M\left(1 - x_{kij}\right) \le AT_{kj}, \forall\, i, j \in V_0, i \neq j, \forall k \in \mathcal{K} \tag{4.6}$$

$$AT_{ki} \le ST_{ki}, \qquad \forall\, i \in V, \forall k \in \mathcal{K} \tag{4.7}$$

$$a_i \le ST_{ki}, \qquad \forall\, i \in V, \forall k \in \mathcal{K} \tag{4.8}$$

$$R_{ki} = \mathrm{P}(AT_{ki} > b_i) \sum_{j\in V_0} x_{kji}, \quad \forall\, i \in V_0, \forall k \in \mathcal{K} \tag{4.9}$$

$$R_{\max} \ge R_{ki}, \qquad \forall\, i \in V_0, \forall k \in \mathcal{K} \tag{4.10}$$

$$x_{kij} = 0, \ \ \forall\, i, j \notin V, \forall k \notin \mathcal{K}_i \tag{4.11}$$

$$x_{kij} \in \{0, 1\}, \ \ \forall\, i, j \in V_0, \forall k \in \mathcal{K} \tag{4.12}$$

$$AT_{ki}, ST_{ki}, R_{ki}, R_{Tk}, R_{max} \ge 0, \ \ \forall\, i, j \in V_0, \forall k \in \mathcal{K} \tag{4.13}$$

The objective (4.1) is to minimise the weighted sum of the maximum risk of a schedule and the total risk for all tasks. The weight w can be chosen sufficiently large to ensure the maximum risk is minimised first. Constraints (4.2) and (4.3) ensure that each technician starts from and finishes at the depot and goes along a connected tour. Constraints (4.4) indicate that each customer is served by one technician. Constraints (4.5) guarantee that each technician serves no more than C customers. Constraints (4.6)-(4.8) ensure the time relationship if technician k serves customer i and then customer j, and he should start each task in the time window. Equations (4.9) define the task risks and the risk of technician k working longer than b_0. Constraints (4.10) calculate the maximum risk used in the objective function. Constraints (4.11) ensure that each task is assigned to a technician with the required skills. Constraints (4.12) and (4.13) are binary and non-negativity constraints.

Due to the nature of probabilities that are used to express the risks, this model is not linear. The complexity of risks makes the model difficult to solve directly by using existing methods. Nevertheless, the model describes the problem clearly, showing the constraints and the way of calculating the objective. These can be helpful when implementing heuristic solution methods.

In order to obtain a linear risk model, we make some assumption about some problem parameters.

4.2 A linear risk minimisation model

In this section, we propose a linear model under the following two assumptions.

(1) Each task duration follows a normal distribution and their variances are the same, all equal to σ^2. The travel times are deterministic.
(2) The objective is only to minimise the maximum risk.

Under these assumptions, the arrival time to a task can be considered as approximately normally distributed. The risk of a task is linked to the corresponding standard normal z-score. If X follows a normal distribution, the standard form of X has a standardised normal distribution (Kalbfleisch, 2012), i.e.,

$$X \sim N(\mu, \sigma^2) \ \Rightarrow Z = \frac{X - \mu}{\sigma} \sim N(0,1).$$

Then minimising the maximum risk can be achieved by maximising the minimum z-score for all tasks. This is illustrated by Figure 4.1. On the standard normal distribution of arrival times, after normalising, we can see that $R_1 = P(Z > 1)$ is bigger than $R_2 = P(Z > 1.5)$. Because the value of the risk is not necessary in the risk minimisation model as long as the order of these values in every schedule can be obtained, the model can be linear since calculating risks is replaced by comparing the z-score.

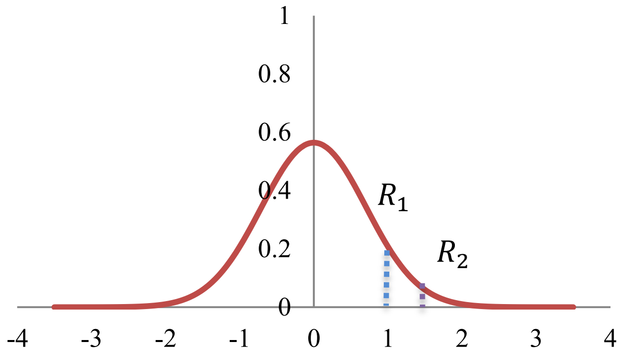


Figure 4.1 Z-score demonstration

Most notations in Section 4.1 will be used for this model as well. The following list re-defines and introduces new notations.

Parameters

I the set of tasks for each technician numbered in the visiting order, $I = \{1, \dots, C\}$ and C is the maximum number of customers that may be served by each technician;

d_{klj} the travel time of technician k between locations l and j;

μ_{kj} the mean of the task duration that technician k spends at customer j;

σ^2 the variance of the task duration time, supposed to be identical for all tasks;

Variables

y_{kij} a binary variable equal to 1 if technician k serves customer j as his/her i^{th} task and 0 otherwise;

AT_{ki} the arrival time of technician k at his/her i^{th} task;

ST_{ki} the start time of technician k serving his/her i^{th} task;

Z_{ki} the standard z score of the risk probability for technician k's i^{th} task;

Z_{k0} the standard z score of the risk probability for technician k work beyond the maximum work time of the day.

Z the lower bound of the standard score of the risk probability for all tasks.

The objective for this model is to minimise the maximum risk in the schedule and the formulation is linear as shown below.

$$\text{maximise } Z \tag{4.14}$$

Subject to:

$$\textstyle\sum_k \sum_i y_{kij} = 1,\ \forall j \in V \tag{4.15}$$

$$\textstyle\sum_j y_{kij} \leq 1,\ \forall k \in \mathcal{K}, \forall i \in I \tag{4.16}$$

$$\textstyle\sum_j y_{kij} \leq \sum_j y_{ki-1j},\ \ \forall i \geq 2, \forall k \in \mathcal{K} \tag{4.17}$$

$$ST_{ki-1} + \sum_{l}[(\mu_{kl} + d_{klj})y_{ki-1l}] + M(y_{kij} - 1) \le AT_{ki},$$
$$\forall k \in \mathcal{K}, \forall i \in I\backslash\{1\}, j \in V \quad (4.18)$$

$$AT_{k1} \ge ST_{k0} + \sum_j d_{k0j}y_{k1j}, \ \forall k \in \mathcal{K} \quad (4.19)$$

$$AT_{ki} \le ST_{ki}, \ \forall k \in \mathcal{K}, \forall i \in I \quad (4.20)$$

$$\sum_j a_j y_{kij} \le ST_{ki}, \ \forall k \in \mathcal{K}, \forall i \in I \quad (4.21)$$

$$AT_{ki} \le \sum_j b_j y_{kij}, \ \forall k \in \mathcal{K}, \forall i \in I \quad (4.22)$$

$$\sqrt{i-1} \cdot \sigma \cdot Z_{ki} \le \sum_j b_j y_{kij} - AT_{ki} + M(1 - \sum_j y_{kij}), \ \forall k \in \mathcal{K}, \forall i \in I \quad (4.23)$$

$$\sqrt{i} \cdot \sigma \cdot Z_{k0} \le b_0 - AT_{ki} - \mu_i - \sum_j[(\mu_{ki} + d_{kj0})y_{kij}] + M(1 - \sum_j y_{kij}),$$
$$\forall k \in \mathcal{K}, \forall i \in I \quad (4.24)$$

$$Z \le Z_{ki}, \ \forall k \in \mathcal{K}, i \in I \quad (4.25)$$

$$y_{kij} = 0, \ \forall\, i \in I, j \in V, \forall k \notin \mathcal{K}_i \quad (4.26)$$

$$y_{kij} \in \{0, 1\}, \ \forall\, i \in I, j \in V, \forall k \in \mathcal{K} \quad (4.27)$$

$$AT_{ki}, ST_{ki}, Z_{ki}, Z \ge 0, \ \forall\, i \in I, j \in V, \ \forall k \in \mathcal{K} \quad (4.28)$$

Constraints (4.15) indicate that each customer is served by one technician. Constraints (4.16) and (4.17) make sure that each position in the task list of each technician can have no more than one task and that the tasks are in consecutive positions from the start of the list. Constraints (4.18) show that for each technician the arrival time of the current task cannot be earlier than the completion time of previous task plus the travel time from there to the current task. Constraints (4.19) are the special case for the first task of each technician. Constraints (4.20) and (4.21) ensure that the start time is after both the arrival time and the lower limit of the time window. Constraints (4.22) require that the expected arrival time to a task is before the upper limit of its time window. Constraints (4.23) calculate the z-score corresponding to the task risks, each risk is defined as the probability of the standard normalized value of the arrival time being greater than Z_{ki}. Constraints (4.24) calculate the z-score corresponding to risks of the technicians work beyond the maximum work time of the day. Z denotes the minimum of all Z_{ki} as is shown in constraints (4.25). Constraints (4.26) ensure that the technician assigned to each task has the required skills. Constraints (4.27) and (4.28) are binary and non-negativity constraints.

One of the assumptions for this model is that the travel times are deterministic. This can be relaxed to a less restrictive assumption that the travel times are normally distributed with mean of d_{kij} and standard variance of σ. In this case, the model needs only minor change, changing $\sqrt{i-1}$ in constraints (4.23) to $\sqrt{2i-1}$ and $\sqrt{i}$ in constraints (4.24) to $\sqrt{2i+1}$.

More specifically, as for the start time of a technician's ith task, there are $i-1$ tasks with stochastic duration time of the same standard deviation σ, hence $\sqrt{i-1}\sigma$ was the standard deviation of the start time distribution for technician k's ith task in constraints (4.23), *i.e.*,

$$\sqrt{i-1} \cdot \sigma \cdot Z_{ki} = X_{ki} - \mu_{ki},$$

where X_{ki} is a value of the random variable for the arrival time and μ_{ki} is the mean of the normal distribution for the arrival time. Here the upper bound of the time window b is used as X_{ki} to obtain the z-value which represents the risk in the model. In addition, constraints (4.24) consider the finish work time for the final task of technicians, so the ith task also contributes the uncertainty in the finish time if the technician has i task in total. This explains that $\sqrt{i} \cdot \sigma$ is the standard deviation for the finish work time.

As for the scenario considering uncertain travel time with the same standard deviation σ as the task duration, there will be i travels, i.e., from the depot to the 1st task, the 1st to the 2nd task, until the $(i-1)$th to the ith task. Therefore, the standard deviation turn from $\sqrt{i-1}\cdot\sigma$ to $\sqrt{2i-1}\cdot\sigma$ in terms of the arrival time, and turn from $\sqrt{i}\cdot\sigma$ to $\sqrt{2i}\cdot\sigma$ regarding to the finish time of the final task. The modified model for this case will still be linear.

4.3 Priority task risk minimisation model

In real-world problems, tasks usually have different importance or priorities according to the business objectives. Therefore, tasks with different priorities need be considered differently. If a technician fails to start a high priority task then the penalty or cost would be higher. For instance, an emergency task may have high priority, a task for an important customer may have high priority, the task with great influence may have high priority, *etc.* In risk management, risk usually has two dimensions: probability and impact. When assessing the significance of any given risk, it is necessary to consider both dimensions (Hillson and Hulett, 2004) and the risk is the product of the two dimensions (Dumbravă and Vladut-Severian, 2013), *i.e.*

$$Risk = Probability \times Impact.$$

Therefore, to consider different task priorities, a priority task risk model is formulated by modifying the nonlinear model in section 4.1. Each task is given an importance score and the priority task risk is defined as the previously used probability multiplied by the corresponding task importance score, *i.e.*

$$PR_{ki} = R_{ki} \cdot IMP_i,$$

where R_{ki} is the probability of technician k missing the time window of task i, PR_{ki} is the priority task risk for technician k missing the time window of task i, and IMP_i is the importance score of task i. Therefore, the priority task risk model is identical to the non-linear risk model except that the priority task risks PR_{ki} are minimised instead of the task risks R_{ki}.

Then the model for problem considering task priorities is formulated below:

$$\text{minimise} \quad w \cdot R_{\max} + \sum_{k \in K} \sum_{i \in V_0} IMP_i \cdot R_{ki} \tag{4.29}$$

Subject to: (4.2) – (4.9), (4.11) – (4.13), and

$$R_{\max} \geq IMP_i \cdot R_{ki}, \ \forall \, i \in V_0, \forall k \in \mathcal{K} \tag{4.30}$$

The objective (4.29) is to minimise weighted sum of the maximum priority risk and the total priority risk in the schedule. Constraints (4.30) are a modified version of constraints (4.10), calculating the maximum priority risk. Other constraints are the same as those of the model in section 4.1.

In addition, as it is mentioned before, the risk increases as it propagates along a technician's task list. Thus, in the optimal solution of this priority risk model, the tasks with high priority will be scheduled in the early places to keep a low-risk level for the whole schedule.

4.4 Other risks

In the previous models, we focus on the arrival-in-slot risk, where the arrival time is used to define the risk. There are four factors contributing to this kind of risks.

- uncertain task duration risk
- uncertain travel time risk
- task priority affecting the weight of the risk
- roster time risk of the technician working over their roster time

In the scheduling, these risk factors may be considered individually or combined according to the optimisation purpose.

In some situations, instead of arrival-in-slot risk, the finish-in-slot risk may be considered. In such cases, it is required that the task starts and finishes in a time window. The risk is defined as the probability that the finish time falls after the upper limit of the time window, *i.e.*,

$$R_{ki} = \mathrm{P}(FT_{ki} > b_i),$$

Where FT_{ki} is the finish time of Task i by technician k, and can be calculated as $FT_{ki} = ST_{ki} + \sum_{j \in V_0} \delta_{ki} x_{kji}$. Therefore, from the definition, it is clear that all previous analysis can be used to model the problem of minimizing the finish-in-slot risk. Again, most notations in Section 4.1 will be used for this model as well. The following lists re-defined and new notations:

$[a_i, b_i]$	Time window for starting and finish the task at customer i, $i \in V_0$,
R_{ki}	The probability of technician k finishing late than the upper limit of the time window at customer i;

The model for the problem is formulated below:

$$\text{mininise} \quad w \cdot R_{\max} + \sum_{k \in K} \sum_{i \in V_0} R_{ki} \tag{4.31}$$

Subject to: (4.2) – (4.8), (4.10) – (4.13), and

$$R_{ki} = \mathrm{P}\big(ST_{ki} + \textstyle\sum_{j \in V_0} \delta_{ki} x_{kji} > b_i\big) \sum_{j \in V} x_{kij}\,, \ \forall\, i \in V, \forall k \in \mathcal{K} \tag{4.32}$$

$$R_{ki} = \mathrm{P}(AT_{k0} > b_0) \textstyle\sum_{j \in V_0} x_{kj0}\,, \quad \forall k \in \mathcal{K}. \tag{4.33}$$

The objective is still to minimise the weighted sum of the maximum risk and the total risk in the schedule. Constraints (4.32) and (4.33) calculate the task risks and the risks of technicians finishing work beyond the maximum work time of the day, respectively. Other constraints are the same as those in the model in section 4.1.

In addition, another scenario frequently happens is that technicians may be unable to finish a task, for example, the technician finds out his/her skill level do not match the task when he/she arrives at the customer. Some other instances are the technician needs more tools or the task needs further work. In these cases, there is a risk for the task being interrupted. For such scenario, a second visit by a technician on the same day is usually arranged and the schedule cannot anticipate this for now. For further study on the interrupted risk, it is necessary to have more data about the tasks and technicians to anticipate this scenario.

Overall, risk minimisation models were demonstrated in this chapter. The non-linear risk minimisation model was given for general cases, while the linear risk minimisation model is simplified by adding some constraints. Moreover, considering different types of tasks, the risk was redefined in the priority task risk minimisation model, and also some other risks were mentioned here. These models are the fundamentals of the following chapters, since Chapter 5 shows our methods to solve these risk minimisation models, and Chapter 6 illustrates the model experiments and corresponding results.

Chapter 5

Heuristic Solutions

The risk models in Chapter 4 clearly present the problem in different settings. However, it is difficult to apply them in practice. This is because the assumptions for the linear model may not be satisfied in most practical problems while the other models are highly nonlinear due to the complicated probability expressions. In addition, our problem can be considered as an extension of the VRP which is NP-hard. When the number of customers is large, solving the problem optimally will require impractical computation time.

On the other hand, with the approximation method for risk distributions, we can easily calculate the risks for any given schedule. Therefore, heuristic search methods may be applied to find near-optimal solutions to the problem. Considering the definition of risks, the sequence of tasks performed by a technician in an optimal schedule would not be significantly different from the order of their time windows. Thus, a neighbourhood-based search method with the guidance of the time windows would be effective.

We chose to use local optimisation and SA method to search for a task schedule with minimum risk. This is firstly because we are minimising risk rather than distance. For problems of minimising distance, such as TSP or VRP, a new move can be obtained by switching any pair of these customers, which may make a better solution. However, the objective for our problem is to minimise the risk, i.e., the probability of arriving a customer after the upper bound of the time window. Hence, for any technician, there is no reason to assign to him/her a customer with a later time window before a customer with an early time window, because in that way one task would have almost no risk whereas the other would have high risk, and this would not be a good solution.

A good schedule tends to be that tasks are arranged according to their time windows for every technician. Therefore, in the scenario of tight time windows, another good solution may be just a local move from the current one.

Secondly, population based search methods such as GA starts from diverse and different initial solutions, and then the new solutions generate can be completely different by the crossover and mutation. So they search for solutions in a large search space, not necessarily in the neighbourhood anymore. However, in this research, the solution will not be far away from a schedule in which tasks are assigned to technicians according to the upper bound of task time windows. Hence, the efficiency will not be high if we use GA because it will waste time on a lot of useless solutions.

Thirdly, apart from these population-based methods we may have the local search, SA and Tabu search to choose. Tabu search may be a competitor because it also can jump out the local optima. Future work could consider using Tabu search on the tests and experiments. This chapter presents the SA algorithm for solving our problem and the implementation details.

5.1 SA for our problem

SA mimics the process of metal annealing. Since its introduction, it has been successfully applied to solve many combinatorial optimisation problems including VRP. Starting from an initial solution, SA generates a neighbouring solution; if the new solution is better, it is accepted as the new current solution, and otherwise it may still be accepted with a probability. The ability of accepting worse solutions with probability helps to avoid the search being trapped to local optima. The probability is set high at the beginning and is gradually reduced as the search continues. This allows the algorithm to gradually focus in an area of the search space close to the optimum solution.

5.1.1 Initial schedule generation

Before carrying out the searching process, it is important to start from a good initial schedule, especially for heuristic methods. According to the character of this problem where tasks have time windows, one method to generate a solution is to sort all tasks according to the upper limit of their time windows as the non-scheduled task list, then for the first technician, assign the first comparable task in the task list to him/her, delete the task from the non-scheduled task list, then do the same assignment for the next technician. Once all technicians get their first task, start the assignment to the first technician again until all tasks are scheduled. The pseudo-code is shown in Table 5.1. Consequently, the initial schedule constructed by this method is better than the schedule built by assigning tasks to technicians randomly.

Table 5.1 The pseudo-code for generating an initial schedule

```
Start
    Sort non-scheduled tasks in ascending order of the upper limit of their time
    windows;
    K ← total number of technicians
    T ← total number of unscheduled tasks
    Repeat
        For technician k = 1 to K
            For task t = 1 to T
                If technician meets task's skill
                    Assign the t^th task in the list to technician k;
                    Remove task t from the list;
                    T ← T − 1;
                    Jump out the loop for the task list;
                End if
            End for
        End for
    Until T = 0
End
```

5.1.2 Objective value calculation

The maximum risk and the average risk of tasks in the schedule are considered in the scheduling objective. As we mentioned in the last chapter, we use the following calculation as the objective value,

$$f = R_{\max} + \frac{1}{n} \cdot \sum_{k \in \mathcal{K}} \sum_{i \in V_0} R_{ki},$$

where n is the total number of tasks in the schedule, $R_{\max}$ is the maximum risk and $R_{ki}$is the specific risk for technician k while doing task i. Once the schedule is changed, the objective function should be recalculated.

The risks are calculated by the summation estimation method and the pseudo-code for its implement function is shown in Table 5.2.

Furthermore, if the priority risk model is considered, the objective becomes

$$f = R_{\max} \cdot IMP_x + \frac{1}{n} \cdot \sum_{k \in \mathcal{K}} \sum_{i \in V_0} R_{ki} \cdot IMP_i,$$

where IMP_x is the importance score of the task which has the maximum risk value, and IMP_i is the importance score of the task i. Then if one would like to consider the traditional model with minimising the average travel time, the objective would be

$$f = \frac{1}{n} \cdot \sum_{k \in \mathcal{K}} \sum_{i,j \in V_0} d_{kij},$$

where $d_{kij}$is the travel time between adjacent tasks i and j carried out by technician k.

Table 5.2 The pseudo-code for the risk calculation function

```
Start
   K ← total number of technicians
   For technician k = 1 to K
        Mean of the process time: Mean ← techinian k rostered start time;
        Variance of the process time: Var ← 0;
        T_k ← total number of scheduled tasks for technician k
        For task t ← 1 to T_k
             If t = 1
                 Mean ← Mean + mean of the travel time from the depot to the
                 1st task;
                 Var ← Var + variance of the travel time from the depot to the
                 1st task;
             Else
                 Mean ← Mean + mean of the duration time of task t − 1 +
                 mean of the travel time from task t − 1 to task t;
                 Var ← Var + variance of the duration time of task t − 1 +
                 variance of the travel time from task t − 1 to task t;
             End if
             If Mean < lower bound of the time window of task t - Var
                 Mean = lower bound of the time window;
                 Var = 0;
             Else if Mean > lower bound of the time window task t + Var
                 No change for Mean and Var;
             Else
                 Use estimated Mean and Var by equations 3.1-3.4
             End if
             Calculate the risk of this task
        End for
        Calculate the risk of this technician working overtime
   End for
   Calculate the objective value of the schedule
End
```

5.1.3 Operators to generate neighbour solutions

There are two operators used to generate neighbour solutions in the searching process: the swap operator and the insert operator. Given a certain task, the swap operator swaps the task with another task, as long as the skill codes are matched with technicians' and their time windows overlap. The swap operator function is shown in Table 5.3.

Table 5.3 The pseudo-code for the swap operator

```
Start
    successful ← false
    Repeat
        Randomly select technicians k_1 and k_2;
        If both scheduled task lists of k_1 and k_2 are not empty
            Randomly select task t_1 from the scheduled task list of k_1;
            Randomly select task t_2 from the scheduled task list of k_2;
            If k_1 meets t_2's skill requirement, k_2 meets t_1's skill requirement
            and time windows of t_1, t_2 are overlapping
                Withdraw t_1 from task list of k_1 and assign it to k_2;
                Withdraw t_2 from task list of k_2 and assign it to k_1;
                Update route information for k_1 and k_2;
                successful ← true;
            End if
        End if
    Until successful is true;
End
```

The insert operator withdraws a task from a technician and allocate it to another technician. As is shown in Table 5.4, it takes out a task and assigns it to another technician who has the matched skill and places it in an appropriate position in the technician's task sequence according to the upper limits of task time windows. To be more specific, given a task m of technician p and task n of technician q, the swap operator exchanges the task m of p and task n of q. The insert operator withdraws the task m from technician p and assigns it to technician q. For example, a technician has 3 tasks and their time windows are 8:00 to 10:00, 10:00 to 12:00 and 15:00 to 17:00, while the time window of the task which will be assigned to this technician is 13:00 to 15:00, then it will be the 3rd task for the technician and the task with time window 15:00-17:00 will become his/her 4th task.

Table 5.4 The pseudo-code for the insert operator

```
Start
    successful ← false
    Repeat
        Randomly select technicians k1 and k2;
        If the scheduled task list of k1 is not empty
            Randomly select task t1 from the scheduled task list of k1;
            If k2 meets t1's skill requirement
                Withdraw t1 from k1 and add it to the scheduled task list of k2;
                Sort the new scheduled task list of k2 according the time order
                of the upper limit of the time windows;
                Update route information for k1 and k2;
                successful ← true;
            End if
        End if
    Until successful is true;
End
```

5.1.4 Local search

In general, local search is widely used for solving computationally hard optimisation problems. Local search is usually used on problems that can be formulated as looking for a solution maximising a criterion among loads of candidate solutions. It yields high-quality solutions by iteratively applying small modifications (local moves) to a solution in the hope of finding a better one, until a solution deemed optimal is found or a time bound is elapsed.

Initially, the above two operators are used separately in a local searching process. Instead of randomly selecting technicians and tasks in the two operators, the process starts from the first task of the first technician and searches until the last task from the last technician in the schedule. Then return to the first technician and keep searching until no improvement is found in neighbourhood solutions.

Table 5.5 The pseudo-code for the swap searching process

```
Start
   Initialisation: Generate an initial schedule ω as described in section 5.1.1 or
   using the schedule obtained by the last searching process;
   Calculate objective f(ω);
   Set the best solution as this solution;
   Repeat
        better ←false;
        K ← total number of technicians;
        For technician k1 = 1 to K
                For technician k2 = 1 to K
              T1 ← number of scheduled tasks for technician k1;
              T2 ← number of scheduled tasks for technician k2;
                 For task t1 = 1 to T1
                    For task t2 = 1 to T2
                       If task t1 and task t2 swap successful (skills feasible)
                         A new schedule ω' is generated in the neighbourhood
                         of ω by the swap operator, calculate objective f(ω');
                         If ω' is better than ω based on the two-objective
                         checking
                            Accept ω' as the current solution ω and update the
                            best solution with ω';
                            update T1 and T2;
                            better ←true;
                         Else
                            Stay with the current solution ω;
                         End if
                       End if
                       t2 ← t2 + 1;
                    End for
                    t1 ← t1 + 1;
                 End for
                 k2 ← k2 + 1;
           End for
           k1 ← k1 + 1;
        End for
   Until better is false;
End
```

Furthermore, we found that the search result is better if we use the maximum risk and the average risk as primary and secondary objectives in the search rather than using their weighted sum as the combined objective. Hence in the local search, we accept a better schedule if its maximum risk is smaller or the maximum risk is the same but the average risk is smaller. Details of the swap searching process are shown in Table 5.5. The insert searching process is similar, and is shown in Table 5.6. Additionally, the swap and insert searching processes are repeatedly used to obtain a better final schedule.

Table 5.6 The pseudo-code for the insert searching process

```
Start
   Initialisation: Generate an initial schedule ω as described in section 5.1.1 or
   using the schedule obtained by the last searching process;
   Calculate objective f(ω);
   Set the best solution as this solution;
   Repeat
        better ←false;
        K ← total number of technicians;
        For technician k_1 = 1 to K
           For technician k_2 = 1 to K
              T_1 ← number of scheduled tasks for technician k_1;
              T_2 ← number of scheduled tasks for technician k_2;
              For task t_1 = 1 to T_1
                 If insert task t_1 to technician k_2 successful (skills feasible)
                    A new schedule ω' is generated in the neighbourhood of ω
                    by the swap operator, calculate objective f(ω');
                    If ω' is better than ω based on the two-objective checking
                       Accept ω' as the current solution ω and update the best
                       solution with ω';
                       update T_1 and T_2;
                       better ←true;
                    Else
                       Stay with the current solution ω;
                    End if
                 End if
              t_1 ← t_1 + 1;
              End for
           k_2 ← k_2 + 1;
           End for
        k_1 ← k_1 + 1;
        End for
   Until better is false;
End
```

5.1.5 The SA procedure and implementation

Even though local search has been shown to be the most popular class of approximate algorithms because of its simple concept and easy application, a drawback is that it is usually trapped in local optima. Thus we use the SA heuristic method to escape from the local optima and obtain a better and near-optimal result.

The SA search process is controlled through two loops. The outer loop controls the temperature decrease. We adopt the commonly used geometric decrease, which allows the temperature decrement gets smaller and smaller. At each temperature, the inner loop searches for a fixed number of iterations. With the above components, the SA procedure for our problem can be summarised in Table 5.7.

In the SA method, the temperature decreases as the following way (Burkard and Rendl, 1984)

$$\text{new temperature} = \text{old temperature} \cdot \alpha,$$

where $\alpha = 0.005$ in our implementation. Then let Ω be the solution space, $f: \Omega \to \Re$ be an objective function defined on the solution space. The goal is to find a global minimum, $\omega^* \epsilon \Omega$ such that $f(\omega) \geq f(\omega^*)$ for all $\omega \epsilon \Omega$. The acceptance probability for a candidate solution ω' is that (Burkard and Rendl, 1984)

$$P = \begin{cases} \exp\left[-\dfrac{f(\omega') - f(\omega)}{T}\right] & \text{if } f(\omega') - f(\omega) > 0 \\ 1 & \text{if } f(\omega') - f(\omega) \leq 0. \end{cases}$$

Generally, for a traditional vehicle routing problem, the cost is the total travel distance, so that the acceptance probability usually is that (Osman, 1993)

$$P = \begin{cases} \exp\left[-\frac{D_{new} - D_{current}}{T}\right] & \text{if } D_{new} - D_{current} > 0 \\ 1 & \text{if } D_{new} - D_{current} \le 0. \end{cases}$$

where D_{new} and $D_{current}$ are the total travel distance for the new schedule and the current schedule respectively, and T is the current temperature. The distance is usually the same scale as the temperature. However, the objective in our problem is about the risk, which is a probability and is naturally between 0 and 1. Thus, the scale is much smaller compared with the temperature. Therefore, a factor β is introduced to let the acceptance probability make sense. Therefore, the acceptance probability for a worse solution in our SA method is

$$P = \exp\left\{-[(MaxRisk_{new} + AverageRisk_{new}) - (MaxRisk_{current} + AverageRisk_{current})] \cdot \frac{\beta}{T}\right\}.$$

Generally we use

$$\beta = 200 \cdot T_0,$$

where T_0 is the initial temperature. One may use a dynamic β in terms of temperature changes to enhance the SA search. From the definition of the probability, it is easy to see that the probability is high when the temperature is high, in order to accept a worse solution by a high chance at the beginning of the searching to jump out the local optima. Then as the temperature gets lower, the probability to accept a worse schedule gets smaller so that a stable solution can arrive.

Table 5.7 The SA search procedure

Start

Initialisation: Generate an initial schedule ω as described in section 5.1.1, calculate the objective $f(\omega)$, set the best solution as this solution, set initial temperature T, terminating temperature T_e, temperature changing factor $\alpha < 1$, max number of iterations at each temperature $Iter_{max}$.

Repeat

Set $Iter \leftarrow 0$

Repeat

$Iter \leftarrow Iter + 1$;

Generate a new schedule ω' in the neighbourhood of ω, by swap or insert operator selected randomly, calculate objective $f(\omega')$;

If $f(\omega') < f(\omega)$

Accept ω' as ω and update the best solution if appropriate;

Else

Draw a random number r from a uniform distribution [0, 1];

If $r < e^{\frac{f(\omega)-f(\omega')}{T}\cdot\beta}$

Accept ω' as ω;

End if

End if

Until $Iter \geq Iter_{max}$

$T = \alpha T$;

Until $T \leq T_e$

End

Additionally, if one would like to use this SA algorithm on the traditional model where travel time is minimised, there are only two modifications needed. One is to replace the risk objective function with the total travel time objective function as we mentioned in Section 5.1.2. Another is to give a huge travel time cost penalty to the solution if any of the time window constraints or the technician work time constraint is violated.

5.2 Applying the algorithm and simulations

The SA algorithm can be applied to generate a schedule once a day at the beginning of the day and the resulting schedule is used for operation of the whole day.

It can also be applied multiple times a day. That is, a schedule is generated with the SA at the beginning of the day and is followed by the technicians in the operation, then it is re-optimised later in the day by rerunning the SA with updated information from the operation.

While the objective is to minimise the risks in making the schedule, the actual performance of the schedule, or the methods used for making the schedule, can be observed in practice in terms of the number of task time windows missed for the day. Simulation can also be used to evaluate and compare the performance of different models.

Simulation aims at mimicking the operations of a real-world process or system over time. Workforce Scheduling Simulation reproduces the operation in the way it is perceived from the workforce scheduling system side, and to observe the behaviour of a work allocation system over time. To study such behaviour, a simulation model is developed, and the key characteristics, behaviours and functions of the physical or abstract system or process are illustrated in the model. The applications of simulation are vast, such as manufacturing (Benedettini and Tjahjono, 2009), engineering and project management (Shi and Vickers, 2016), military applications (Robinson, 2002), business process (Robinson, 2002), transportation modes and traffic (Mualla *et al.*, 2018), healthcare (Almagooshi, 2015), *etc*.

Simulation benefits the study of and the experimentation with the internal interactions of a complex system or a subsystem within a complex system.

Moreover, the knowledge obtained during the simulation could be valuable to improve the system under investigation. Most importantly, simulation can be used to verify analytic solutions, and animation shows a system in some simulated operations so that the plan can be visualized. Therefore, we introduce simulation into our models.

Furthermore, Monte Carlo simulation is a computerised mathematical technique that allows people to account for risk in quantitative analysis and decision making. It demonstrates to the decision-maker a range of possible outcomes and the probabilities they will occur for any choice of action. It involves a range of values, from a probability distribution, for any factor which has inherent uncertainty to perform risk analysis by building models of possible results. It then calculates results repeatedly, each time using a different set of random values from the probability components. Depending on the uncertainties and the specified value ranges for them, this simulation usually involves massive recalculations in the process. Monte Carlo simulation gives distributions of possible outcome values, and probability distributions are a much more realistic way of describing uncertainty in variables of a risk analysis.

5.2.1 Optimisation once for a day

The SA search algorithm can be applied to solve any optimisation model to make a schedule for the day. This can be the risk model, the priority risk model, or the travel time model. Then simulation can be used to simulate the operation following the schedule and observe the number of missed tasks as the evaluation for the schedule. The implementation of the simulation process is outlined below.

Recall that the analysis in the risk models is based on that the task duration and travel times are assumed to be uncertain and follow normal distributions, however in reality, the task operation time or travel time becomes determined once it has happened. Therefore, to investigate the performance of a schedule in real-world scenarios, one best way is to execute the resulting schedule in the real-world, or use simulation to test the model. We realise the task duration based on the task corresponding distribution, *i.e.*, the value of the real task duration time is randomly sampled from the assumed task duration distribution which is a normal distribution with known mean and variance. In addition, to be more reliable, we give a lower bound of 5 minutes and an upper bound of three standard deviations plus the mean. So if a realisation value is outside the range, another sample value should be generated.

Then we realise the resulting schedule simulation in the way that we generate a report object at the end of the day, which contains the information such as the report time, the reporting technician, the finished tasks, the real operation time for tasks, *etc.* The details will be discussed in the next section, because we use the reports from different time points in the re-optimisation simulation. For this single optimisation simulation, it is simplified to only one report at the end of the day.

5.2.2 Re-optimisation

In most research and real-world operations, like the way described in the last subsection, the scheduling process often searches once every day at the beginning of the day, then the resulting schedule is delivered to technicians to perform and is not amended in the day. However, as time goes on, the service times of the completed tasks and the travel times of the completed trips are realised and become known, which may or may not be the same as the estimated values used in the initial schedule. As a result, the risks for the later tasks are changed and the schedule will not be optimal for the rest of the day. With convenient modern communication tools, the updated information in the operation can be reported by the technicians back to the scheduler, and the schedule can be re-optimised during the day. Consequently, the performance of the schedule and the quality of the services may be improved. Such a re-optimisation framework can be implemented in practice, and can be simulated as well.

Briefly speaking, the re-optimisation uses the SA algorithm to search a resulting schedule at the beginning of the day. Then the engine constructs a schedule of the unreached tasks (the remaining non-started tasks in the earlier schedule) at the report time, using the SA algorithm. The information of unreached tasks is included in the reports returned by technicians. The re-optimisation procedure is shown in Table 5.8.

Table 5.8 The re-optimisation procedure

Start

    $T_0 \leftarrow$ the initial optimise time;

    $T_i \leftarrow$the report time for the ith report;

    $N \leftarrow$ total number of optimisations;

    Generate an initial schedule $\omega(T_0)$ by SA;

    For $i = 1$ to $N - 1$

        Process $\omega(T_{i-1})$ and construct the report $R(T_i)$;

        Generate a new schedule $\omega(T_i)$ by SA for the unreached tasks in $\omega(T_{i-1})$ at T_i;

    End for

    Construct the final report $R(T_N)$;

End

The reports returned to the searching engine contain the information at the time it is reported, the id of the technician who sends the record, completed tasks, failed tasks, the status (the technician is doing a task, travelling or waiting), the location, and the statistics of the estimated remaining time if the technician is doing a task. The method to construct a report for a given technician is illustrated in Table 5.9.

Table 5.9 The report construction procedure

Start

T ←the report time

σ ← the standard deviation for the task being performed when the technician gives the report;

$time$ ← the technician start roster time;

If the technician scheduled task list is not empty

N ← total scheduled task number;

For $i = 1$ to N

t_i ← the ith task;

$time$ ← $time$ + travel time from the last position to t_i's location;

$isTaskRemaining$ ← false, this value will change to true if there is any remaining task at T;

If $time \geq T$

$isTravelling$ ←true, the technician is traveling at T;

Set the remaining time as $time - T$, standard deviation as σ and location as t_i's location;

$isTaskRemaining$ ←true;

Jump out the for loop;

End if

If $time <$ the lower bound of t_i's time window

$time$ = the lower bound of t_i's time window;

End if

If $time \geq T$

$isWaiting$ ← true, the technician is waiting at t_i;

Set both the remaining time and standard deviation as 0 and location as t_i's location;

$isTaskRemaining$ ← true;

Jump out the for loop;

End if

If $late$ ←true if $time \geq$ the upper bound of t_i's time window;

Add t_i to the missed task list;

Else

$time \leftarrow time$ + realised duration time of t_i;

End if

If $time \leq T$

Add t_i to the past task list (missed tasks are also in this list);

If $time = T$

Set both the remaining time mean and standard deviation as 0 and location as t_i's location;

$isTaskRemaining$ ←true;

Jump out the for loop;

End if

Else

$isDoingTask$ ←true;

Add t_i to the doing task list;

Set the remaining time mean as $time - T$, standard deviation as σ and location as t_i's location;

$isTaskRemaining$ ← true;

Jump out the for loop;

End if

End for

If $isTaskRemaining$ = false

Set the remaining time as 0 and location as t_T's location;

End if

End if

End

Then apply this report construction for every technician in a schedule $\omega(T_{i-1})$, we can get a series of reports $R(T_i)$ to generate an initial schedule for the unreached tasks at the reported time T_i. The variables in the class Report are shown in Table 5.10 to demonstrate the information we have in the report. It can be implemented in the following way: Remove the past tasks which is in the reports, from the scheduled task list in $\omega(T_{i-1})$, and update the start time mean and variance, and the start locations; then apply the SA on this schedule to generate the solution $\omega(T_i)$ for technicians to execute from T_i.

Table 5.10 Variables in class Report

Variable	Type	Explanation
Id	String	a specific id for the report record
reportTime	Date	the time the report returned
Resource	Resource	the technician who sends the report
doneTasks	ArrayList<Task>	the tasks the technician have done
taskRealTimes	ArrayList<Long>	the task real duration time of the completed tasks
lateTasks	ArrayList<Task>	the list of tasks that the technician arrives late
isDoingTask	boolean	denote whether the technician is doing a task at the report time
isTraveling	boolean	denote whether the technician is travelling at the report time
processTask	Task	record the task if the technician is doing a task
estimateRemainingTime	Long	the estimated mean remaining time of the task duration that the technician is doing
remainingTimeSd	Long	The estimated standard deviation of the remaining time of the operating task
locationLatitude	Double	The latitude of the technician location at the report time
locationLongitude	Double	The longitude of the technician location at the report time

From the technician report at certain time points, we may know the latest task execution so that an action may be taken in the next optimisation schedule. For examples, suppose Technician 1 has a task which has exceeded regular operation time significantly, hence the risks for the remaining tasks in the technician's task list get higher. Meanwhile, Technician 2 may finish his/her task earlier than expected and the time windows of his/her remaining tasks are at later time slots. In this case, it is an ideal solution to assign Technician 2 to do the remaining tasks of Technician 1, and let Technician 1 to do the tasks of Technician 2. Consequently, re-optimisation during the day could theoretically improve the schedule performance as compared to only scheduling once at the start of the day.

Additionally, as we mentioned before, the report construction could be used on the singular estimation by using one reported time such as 7:00pm in the evening. Generally, at that time, all tasks would be executed and the total number of missed tasks could be gathered. Also, one may investigate the behaviours of different models or the improvement that the re-optimisation gives to the task scheduling system in terms of the number of missed tasks, as well as the time cost of the re-optimisation.

To summarise, in this chapter we have presented the details of the local search and SA methods for our problem, the ways of applying SA such as re-optimisation, and the way of evaluating the resulting schedules using simulation. The next chapter will report the experiments that evaluate and compare different models solved by the SA method, using several different measurements.

Chapter 6

Computational Experiments

This chapter investigates the performance of the risk minimisation models presented in Chapter 4 by comparing their results with that from a traditional travel time minimisation model. The searching algorithms developed and described in Chapter 5 have been applied to solve the models. The investigation evaluates the model performance using a number of different measures.

All the computational experiments were carried out using Eclipse Java integrated development environment on a personal computer with a 64-bit windows 7 system, Intel Core i5-4210U 1.7GHz-2.4GHz CPU, 8G RAM. Since the data and models are unique to this problem, Java applications give more flexibility and can be personalised for different scenarios.

6.1 Input data and the initial schedule

Our testbeds consist of information from the company's database. This includes the task information *i.e.*, the task ID, the mean and variance of the task duration time, the upper and lower limits of the time window, the task location coordinates, the task required skill, and the task importance level. The technician information includes the technician ID, start and end rostered times, the technician start location, as well as the technician skill capability. The input data are of JSON format because JSON is a lightweight data format which is easy to parse and generate by computers, and widely used as the data format in communication sectors.

As we explained in Section 3.5.2, the travel distance is the straight-line distance between any two locations defined by their coordinates, combined with a correction factor 1.3, which means the real travel distance is 30% longer than the straight-line distance. Then the travel time is calculated by the average speed of 40 km/h.

First, the testbed of the experiments in Section 6.2 composes of 50 technicians and 300 tasks. Using this testbed in Section 6.2.3, different lengths of time windows are considered *i.e.*, 3 hours, a half day or a whole day. The local search method is used to generate a solution in Section 6.2 and the SA method is also tested in Section 6.2.1. Second, in the testbeds for the experiments in Section 6.3, 6.4 and 6.5, there are 17 technicians and 100 tasks. The SA method is used for tests in these sections. Third, for the experiments in a prototyping tool of the scheduler from a communication company, 593 tasks and 157 technicians constitute the testbed in Section 6.6. Metaheuristic methods are used here. Additionally, the testbed of 12 resources and 100 tasks is used in the experiments in Section 6.7 to demonstrate the advantage of multi-optimisations. The SA method executes the search procedure during the simulation.

It is widely acknowledged that to start from a good initial solution benefits a heuristic method. According to the nature of the task time windows, a relative good initial schedule can be obtained in the way as we explained in Section 5.1.1. More specifically, for the case of 50 technicians and 300 tasks, all tasks are sorted according to the upper bound of their time windows and compose a list, then loop from the 1^{st} technician, suppose the 1^{st} task is compatible with him/her, then assign it to him/her, and delete the task from the list. Then for the 2^{nd} technician, now the previous 2^{nd} task becomes the 1^{st} task in the list, suppose the 2^{nd} technician is unable to do this task, then check the compatibility with the 2^{nd} task (*i.e.*, the 3^{rd} task in the previous list), and suppose he/her meet this task skill requirement so assign the 2nd task to him/her and delete this task in the list. In a similar way, for every technician, always check from the 1st task in the list until finding a compatible task. Then if the assignment is carried out for all technicians once, start from the 1st technician again until all the tasks are allocated.

6.2 Basic experiments

In this section, the testbed is composed of 50 technicians and 300 tasks with all the information available for each task and technician as described above.

6.2.1 Experiment on the risk model

The local search used here is first the insert process, then the swap process, and repeat both processes for four times. In the end, the resulting schedule shows the allocation of tasks to technicians, the risk for each task, the maximum risk and the sum of the risks for all tasks.

Table 6.1 Risk comparison of the initial schedule and the resulting schedule

Risk	Initial schedule	Local Search	SA method
Maximum risk	1.0	0.05062	0.04978
Average risk	0.25979	0.00662	0.00490
Running time (ms[a])	-	32663	576931

[a]. Milliseconds

In this basic experiment, a specific time window is set for each task, and the same priority (or no priority) is assigned to all tasks. Table 6.2 shows the initial schedule and the resulting schedules after the local and SA searching process. From the table, it is clear that the searching algorithm greatly improves the initial schedule. In the resulting schedule, the maximum risk decreased significantly, and the average risk is small. The SA method shows a better solution while taking longer time to find it. Hence, we may conclude that the searching methods are effective.

6.2.2 Experiment on the risk model with priorities

It is more realistic that tasks appear to have different importance or priority according to the business objectives. If a technician fails to start a high priority task, then the penalty should be higher. In this experiment, an importance score is assigned to each task to represent the priority of the task. For the same testbed as of the Section 6.2.1, 300 tasks are considered, 50 of them are with high priority, 100 of them are with medium priority, and the others are with no priority.

Consequently, it is beneficial for the cost model to take into consideration the task priority. In order to reach this objective, the risk penalty is multiplied by the task importance score. In other words, the higher the importance score, and the higher the risk cost should be.

Table 6.2 Average position for tasks of different importance scores[a]

Task importance	Risk model	Priority risk model
High	3.47	3.22
Medium	3.57	3.48
Low	3.58	3.73

[a.] The average task position in its corresponding technician's tour in the corresponding model

The tasks are still with specific time windows. The high priority tasks tend to be scheduled early to avoid the high penalty cost. Table 6.2 shows the average positions of tasks with different priorities in the task sequences of the corresponding technicians. The task position gives the position in the task list of a route for that task. For example, a task with a high importance score is at position 1 demonstrates that this task is assigned to its technician as his 1st task.

From Table 6.2 it can be seen that the high priority tasks are moved in the resulting schedule, from an average value of 3.47 to 3.22. As a result of minimising the penalty for high priority tasks, the risk of those with low priorities will increase, it can be seen from the table that the average task position for the tasks with a low priority from value 3.58 to 3.73. Additionally, the differences seem not significant because tasks of the same priority have all kinds of time windows in the data and the length of the time window is short as 2 hours or 3 hours, there's rare possibility to adjust positions for all tasks while the objective of reducing the possibility of missing tasks should be satisfied first. However, the difference will be significant when we consider wider time windows which we will discuss later.

Table 6.3 Searching results (value of objective functions and running time)

Risk	Risk model	Priority risk model
Maximum	0.05062	0.09302
Average	0.00662	0.00817
Running time (ms[a])	32663	30551

[a] Milliseconds

Moreover, as shown in Table 6.3, the average risk is a bit higher in the priority risk model than in the case where all tasks have the same priority, this is to decrease the risk for high priority tasks. And from Table 6.4 we can see that minimising the priority-weighted risk will decrease three quarters of the risk for high priority tasks and a half of the risk for medium priority tasks on average.

Table 6.4 Average task risk[a]

Task importance	Risk model	Priority risk model
High	0.00684	0.00146
Medium	0.00666	0.00389
Low	0.00652	0.01325

[a.] The average task risk of the schedule in the corresponding model according to different task priorities

6.2.3 Risks of tasks under different time windows

It is very common in practice that the service company give only two types of time windows for customers to choose. In this case, the time window for any task is either in the morning or in the afternoon. Using these two common time windows, from Table 6.5 one can see that the results for the tasks with and without priorities have a similar relationship to the situation where tasks having specific time windows. The results of the positions of the morning and afternoon task are shown in Table 6.6. As shown in the table, the important tasks are all moved forward during the searching process when the importance scores for tasks are considered.

Table 6.5 Searching result for tasks with morning and afternoon time windows

Risk	Risk model	Priority risk model
Maximum	0.08637	0.12461
Average	0.00608	0.00804
Running time (ms[a])	30306	36244

[a.] Milliseconds

Comparing the results for the tasks with specific time windows and those for the tasks with two larger time windows (morning and afternoon), it can be seen that the wider time window makes the risk smaller, which verifies the statement about the effect of the size of a time window on the risk as we discussed in Section 3.6.

Table 6.6 Task position for tasks with different priorities and time windows[a]

Importance & time window (TW)	Risk model	Priority risk model
High importance	3.58	2.98
Morning TW	2.27	1.53
Afternoon TW	5.11	4.59
Medium importance	3.61	3.22
Morning TW	2.05	1.68
Afternoon TW	5.17	4.75
Low importance	3.53	3.96
Morning TW	2.02	2.51
Afternoon TW	5.00	5.39

[a] The average task position in its corresponding technician's tour of the corresponding model

If we consider the time windows as the whole day which is from 8 am to 6 pm (BST). Due to the wide time windows, the risks are all relatively small. Thus, the task importance scores have to be large enough to force the searching engine to schedule the important tasks in early positions. Table 6.7 gives the task positions in terms of different task priority weights for risks. The result in the small priority column comes from the risk model with the priority values of 1, 100 and 10000 as the low, medium and high importance score, respectively. The figures from the large priority column are obtained assuming the priority values of 1, 10^5 and 10^{10}. As we consider the large time windows, we also conducted a test which set an additional target time for high importance tasks *i.e.*, 12 am, so that the important tasks can be scheduled as early as possible in the day. The priority values are 1, 100 and 10000 here respectively. Table 6.8 shows the average risk of the tasks with the low, medium and high importance scores.

Table 6.7 Task position comparison

Model	High importance	Medium importance	Low importance
Identical priority[a]	3.58	3.63	3.53
Small priority value[b]	2.35	2.68	4.57
Large priority value[c]	1.90	2.39	4.91
Additional target[d]	1.11	3.07	4.73

[a] The resulting schedule by minimising risks
[b] The resulting schedule by minimising small-priority-weighted risks
[c] The resulting schedule by minimising large-priority-weighted risks
[d] The resulting schedule by minimising medium-priority-weighted risks with a midday target time for high priority tasks

Table 6.8 Risk comparison

Model	High importance	Medium importance	Low importance	Total average
Identical priority[a]	1.10×10^{-4}	1.07×10^{-4}	9.89×10^{-5}	1.03×10^{-4}
Small priority value[b]	0	1.1×10^{-7}	3.06×10^{-4}	1.53×10^{-4}
Large priority value[c]	0	0	3.73×10^{-4}	1.87×10^{-4}
Additional target[d]	0	1.8×10^{-7}	4.1×10^{-4}	2.1×10^{-4}

[a] The resulting schedule by minimising risks
[b] The resulting schedule by minimising small-priority-weighted risks
[c] The resulting schedule by minimising large-priority-weighted risks
[d] The resulting schedule by minimising medium-priority-weighted risks with a midday target time for high priority tasks

It can be concluded that applying the task priority to risks can schedule high importance tasks in earlier positions and the average risk of missing the high importance tasks decreased significantly, no matter whether the priority values are small or large. It is reasonable that the tasks with high priorities are conducted first as high risks are prioritised. Furthermore, if we use a huge importance score such as 10^{10}, the resulting schedule shows that the tasks of high importance are executed as technicians' 1st or 2nd task because the figure gives the average position is 1.90. If high importance tasks are given a midday target time, the average task position for high priority tasks is 1.11, which means high priority tasks will be executed first in the resulting schedule.

Therefore, in the scenario of the whole day time windows for tasks, the decision maker may use different values of priorities or additional target time to balance between different targets in scheduling.

6.3 Comparisons among models

The testbed consists of 17 technicians and 100 tasks, in section 6.3.1 we assume the task duration is normally distributed, and in section 6.3.2 the task duration follows a Gamma distribution in the experiments.

6.3.1 Comparison between risk models and the traditional VRP model

VRP manages the design of a set of routes with the minimal cost that meet the demands for services and goods of a set of customers with different geographic locations, while satisfying a number of operational constraints at the same time. Because the searching goals of the risk models differ from the traditional travel cost, it is meaningful to compare the experimental results of the traditional cost with the ones from the risk perspective. The travel time model is proposed to model the VRP with the traditional travel cost, the risks are considered neither in the constraints nor in the objective function, whereas the time windows constraints are strictly valid.

In our experiments, the risk in the risk models is a result of the uncertain task duration and the fluctuating travel time. Several factors are considered during the scheduling process, the time windows, the mean of the estimated duration time, the variance of the duration time, the required skill levels and the task importance scores are different among tasks. Each technician additionally has a unique start depot and a skill capability level. The travel time variance is distinct in the morning compared to the afternoon. The distributions of uncertain factors in the model are all supposed as normal distributions.

Hence there are three scheduling models, the travel time model is for a traditional scheduling problem in which the objective is to minimise the average travel time; while the risk model and priority risk model aim at minimising the combination of maximum risk and the average risk of all tasks with the consideration of both the stochastic task time and the uncertain travel time. In the risk model, all tasks are treated as having the same importance while for the priority risk model each task has one of the two different priorities. To see the effect of considering risks in the scheduling, as well as the effect of considering task priorities, we suppose that the priorities are distinguished as two levels: high and low.

Figure 6.1 shows the average risks of high and low priority tasks for the three models. When comparing the travel time model with the risk model it can be seen that the average risk for all tasks for the travel time model is much higher than that obtained when minimising the task and travel risk. This is reasonable because we did not consider risk when minimising the total travel time during scheduling. Although Figure 6.1 shows that the average risk for the travel time model is not too high in value, this is because the time window constraints can limit the risks to some extent in the travel time model.

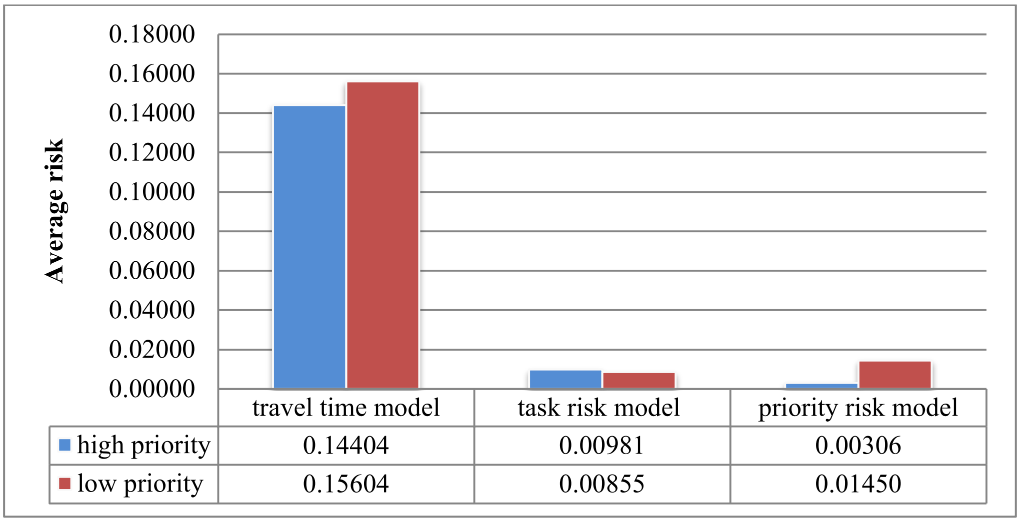


Figure 6.1 Risks for different models

Furthermore, the task importance or priority is a key factor in business operations. If a technician fails to start a top priority task in its time window, the penalty cost will be higher. Therefore, the priority task risk is introduced in the scheduling where the priority risk of a given task is defined as the risk of the task multiplied by an adjusted task importance score, so that it lets high priority tasks take precedence over low risks. The importance of a task is given one of two scores, 1 and 100, and a pre-determined adjusted factor is used to adjust the priority task risks. Thus, the priority risk is given by

$$\text{priority risk} = \text{task risk} \times \frac{\text{task importance score}}{\text{factor}}.$$

According to the data from the telecommunication organisation, the importance scores of their tasks vary between 25 and 300, so here we use 1 and 100 to simplify the model. Also the adjusted factor can be different values in terms of the value of the priority tasks. In this test, the factor value used is 30.

The priority risk is defined and only used as an objective in the priority risk model solution process. For a fair comparison, the risks of missing appointments (*i.e.*, the task risk) in the schedule result are calculated in the same way as for the other two models and shown in the figures. As we can see in Figure 6.1, the risk of high priority tasks is smaller at a cost of the increased risk for low priority tasks.

Table 6.9 Travel time for different models

Model	Travel time model	Risk model	Priority risk model
Average travel time (minutes)	10.74	18.49	19.28

A comparison of the average travel time is shown in Table 6.9. The average travel time is the total travel time spent by all technicians divided by the total number of tasks. As expected, the travel time model results in the smallest travel time among the three models; but the risk model and priority risk model also show relatively short travel time. An explanation could be that by minimising the risks, there is a side effect of minimising the travel time simultaneously. Specifically, the risk is considered as the probability of the arrival time after the upper limit of the time window, and the mean of the arrival time is associated with estimated durations of all previous tasks and travel times for each technician. Therefore, during scheduling, when we try to minimise risks we also minimise the travel time simultaneously.

6.3.2 Comparison assuming Gamma distributed task duration

In this section, the same experiments are conducted as the last section. The difference is that the task duration and travel time here both follow a Gamma distribution. The dataset is the same as the previous test, but the shape parameters are the same, which are 3.5 here, for all tasks according to the data analysis from the communication organisation. Then given the task duration mean from the data, we may obtain the scale parameter for each task.

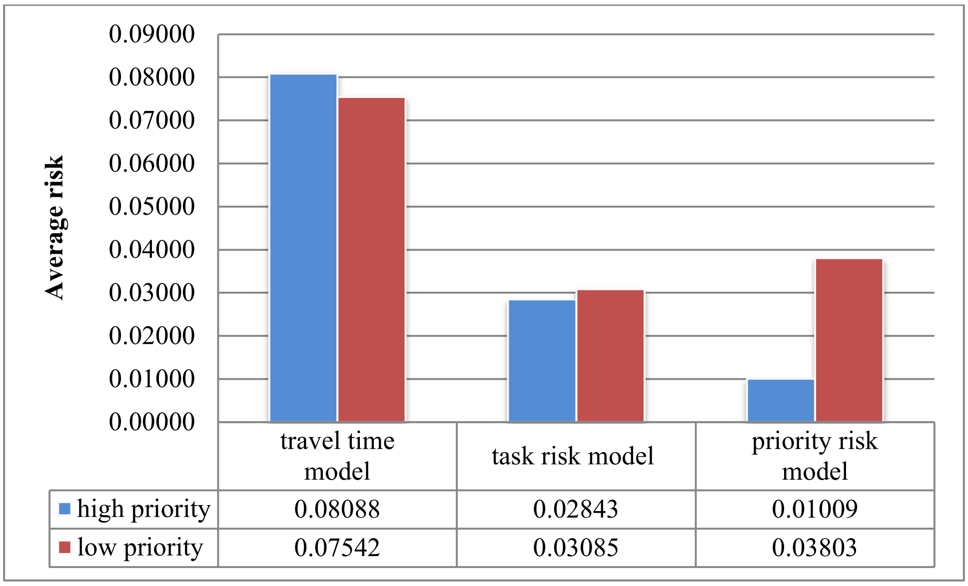


Figure 6.2 Risks for different models

Figure 6.2 shows that the average risks of high and low priority tasks respectively for the three models. From the comparing of the travel time model and the risk model, the average risk for all tasks in the case of the travel time model is higher than twice of that obtained from minimising task risk and travel risk together. This is expected as we did not consider risk when minimising the total travel time during scheduling. Figure 6.2 also demonstrates that the average risk in the result of the travel time model is not very high in value, and this is because of time window constraints in the travel time model (instead of being driven by cost objectives minimisation in the other two models).

Table 6.10 Travel time for different models

Model	Travel time model	Risk model	Priority risk model
Average travel time (minutes)	13.14	31.64	32.31

For the priority risk model, Figure 6.2 shows the risk of high priority tasks is smaller at a cost of the increased risk for low priority tasks. As expected, the travel time model results in the shortest travel time among the three models, which is shown in Table 6.10; but the risk model and the priority risk model do not show extremely long travel time values.

6.4 Schedule structure with different priority tasks

From the definition of risks, a conclusion can be drawn that the risk increases as it propagates due to the increase in the variance of the arrival time along the task list of each technician. The position of the task in the planned tour of visits is important information to verify the robustness of the plan during the day against disturbances. Inheriting the experiments on the three models in Section 6.3.1 and from Figure 6.1 in that section, we notice that the risks for high priority tasks becomes smaller on average in the priority risk model compared with that in the risk model. Then after analysing the structure of the task priority at each position in the task list for every technician, we can find that high priority tasks are completed at the early position in the tour of visits under the time window conditions. It models the real-world fact that technicians prefer to do the important task first to make sure its completion achievable.

In our application case, the average number of tasks for each technician is 5.9, which is derived from 100 tasks divided by 17 technicians. Figure 6.3 illustrates the number of high priority tasks for all technicians as the vertical axis (figures on lines) according to each ordinal position in the task list as the horizontal axis. For instance, on average 6.5 technicians have a high priority task as his/her 1st task in the scheduling result obtained by the priority risk model, whereas 5.5 technicians by the risk model and 5.4 technicians by the travel time model.

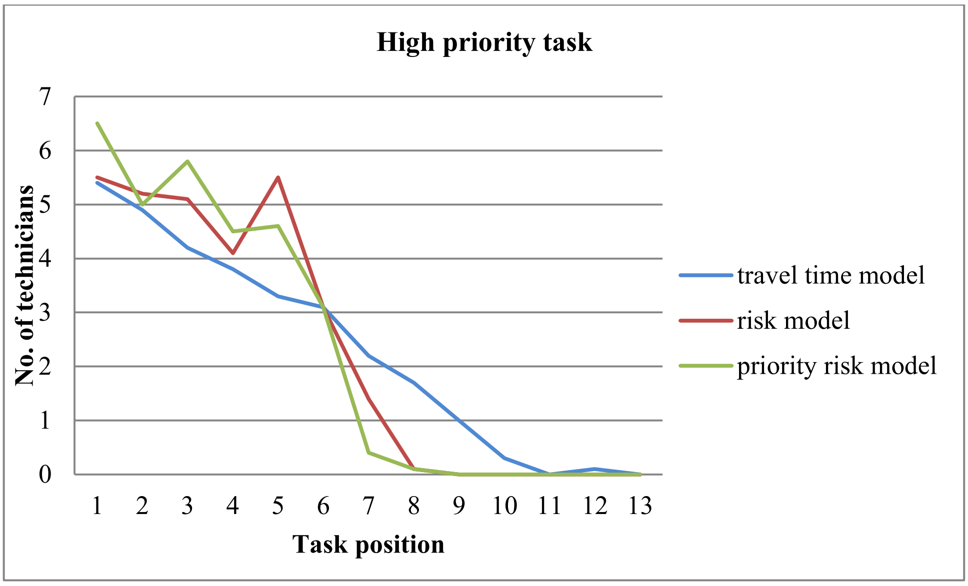


Figure 6.3 High priority task position composition

Moreover, the high priority tasks at position 1 and 3 are more in the priority risk model than those in the risk model, which means that the high priority tasks are executed earlier both in the morning and in the afternoon. More specifically, even though considering importance score in the risk calculation can stimulate the scheduler to assign the top priority tasks as early as possible, the nature of the task time windows limits the ordinal positions of all important tasks to be in the 1^{st} place. Namely, if the task time window is in the afternoon, the engine would not assign the task as the 1^{st} task of any technician.

Additionally, with more top priority tasks moving to the early positions, there are less important tasks assigned at the late time in a day, so that the number of the top priority tasks drop at the 7^{th} task position for the priority risk model while the other two models still have some tasks assigned at later positions.

Furthermore, the number of the important tasks in the schedule obtained by the risk model which has no consideration of task importance during scheduling shows evenly distributed regarding different ordinal task positions in technician scheduled task lists. For example, there are on average 5.5, 5.2, 5.1 and 5.5 top-priority tasks as technicians' 1^{st}, 2^{nd}, 3^{rd} and 5^{th} task as is shown in Figure 6.3.

Additionally, we can notice that the graph also indicates that technicians may have at most 8 tasks in both risk models. Whereas in the case of the travel time model, some technicians may have more than 8 tasks which is a really tight schedule hence the risks for missing these appointments can be much high.

On the contrary, there is a cost for valuing top priority tasks: the less important or ordinary tasks are pushed to late positions as is shown in Figure 6.4. Note that the number of technicians at each task position for low priority tasks looks twice as that for high priority tasks, because there are 30 high importance tasks and 70 low importance tasks. Consequently, the sum of the number of technicians on all positions will be 30 for high importance tasks, and 70 for low importance tasks.

The ordinary tasks in the priority risk model are assigned on average at a later position than those in the risk model. In the meantime, the risks of ordinary tasks in the priority risk model are consequently higher than those in the risk model without priorities, so that the risks of the tasks with a high importance score could be much smaller. Additionally, the difference of the improvement for the top priority tasks or the cost for the low importance tasks between the risk model and the priority risk model depends on the factor when calculating the priority risk in the model. The larger the factor is, the less the effect of the importance score is and the smaller the difference between the two models becomes, and vice versa. In terms of our testbed with a factor of 30, after considering the tasks with different priority levels in the scheduling, the risks for top priority tasks is three times lower than those in the risk model, and the risks for ordinary tasks is three times higher than those in the risk model.

Last but not least, we may observe that the number of tasks in the travel time model and the risk model did not fluctuate as much as that in the priority risk model. It makes sense due to the fact that the scheduler treats all the tasks with no difference in the travel time model and the risk model, while the tasks have different weights according to the task significance in the priority risk model.

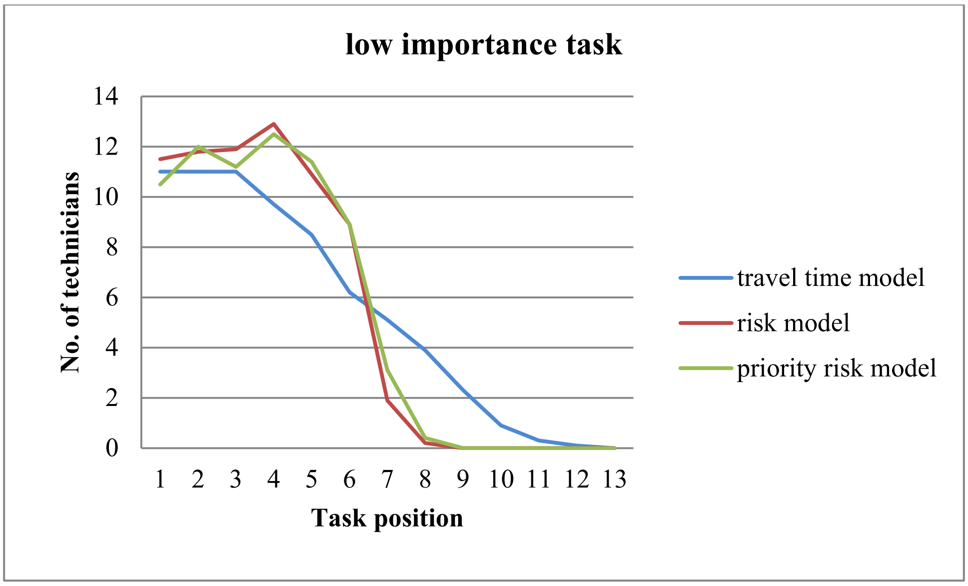


Figure 6.4 Low priority task position composition

6.5 Productivity of technicians

Productivity is widely used to describe various measures of the efficiency of operations. A productivity measure is traditionally expressed as the ratio of output to inputs used in a production process, *i.e.*, output per unit of input. Productivity is a crucial factor in the operations performance of firms and organisations. Moreover, productivity growth also helps businesses to be more profitable. There are many different definitions of productivity and the choice among them depends on the purpose of productivity measurement and data availability.

In the case of the application in the telecommunication sector, the productivity for a technician to deliver field engineering services in a real scheduling problem is introduced in the view to study the technicians' behaviour, it can be defined as

$$\text{technician's productivity} = \frac{\text{work hours}}{\text{roster hours}}.$$

Note that the work hours are the summation of the means of the scheduled tasks, the means of the travel time between any two locations.

On the account of the same experiment, Figure 6.5 shows the relative frequency of the number of technicians (vertical axis) according to the value of productivity (horizontal axis). For example, there are around 63.5% of technicians whose productivity is between 60% and 70% when building the start-of-day service visits plan by the risk model.

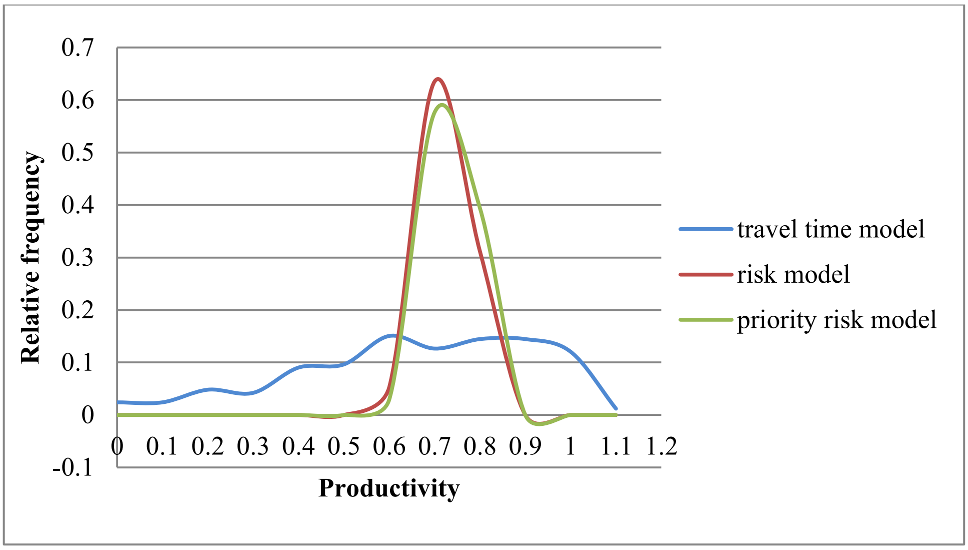


Figure 6.5 Productivity distribution

Moreover, the shape of the distribution for the productivity obtained by the priority risk model is similar to the one from the risk model. In other words, the variances of the productivity for the two risk models are much smaller compared with the variance for the travel time model from Table 6.11, *i.e.*, a large number of technicians have the productivity of the average value, which is around 70%.

Table 6.11 Productivity for different models

Productivity	Travel time model	Risk model	Priority risk model
Average	61.5%	67.5%	68.3%
Standard deviation	23.7%	4.6%	4.6%

The productivity of the technicians in the schedule obtained by the travel time model aligns with an even-like distribution as is shown in Figure 6.5, which means the technician workload for the schedule from the travel time model fluctuates much among technicians. Meanwhile, the technicians in the schedules from the risk model and priority risk model have an even workload, that is to say, few technicians have extremely few or many tasks scheduled to him/her, hence a robust schedule is obtained by the two risk models.

6.6 Contingency

As the problem derives from an industry case, there are lots of measurements to evaluate the performance of the models and the resulting schedules in terms of economic benefits for organisations, such as the contingency are introduced in this section. Contingency refers to costs that will probably occur based on past experience, but with some uncertainty regarding the amount. The contingency allowance is designed to cover the costs which are not known exactly at the time of the estimate but which will occur on a statistical basis (Jelen and Black, 1983).

Normally, while approximating the cost for a project in business, product or other item or investment, there is always uncertainty as to the precise content of all items in the estimate, how work will be performed, what work conditions will be like when the project is executed and so on. These uncertainties are risks to the project. The estimated costs of the risks are referred to by cost estimators as cost contingency in business risk management.

As to the risk defined in our models, the contingency for a task can be given as the time gap between the start time at the task and the upper bound of the task time window. More specifically, given a task in a schedule, the contingency is calculated as the difference between the mean of the start time distribution and the upper limit of the task and is expressed in minutes. As is shown in the following tables, contingency is usually analysed as an average value of a group of tasks.

The collaborated organisation has applied the risk component in their scheduling engine, and the results demonstrated in this section come from experiments in a prototyping tool of the scheduler as is shown in Figure 6.6.

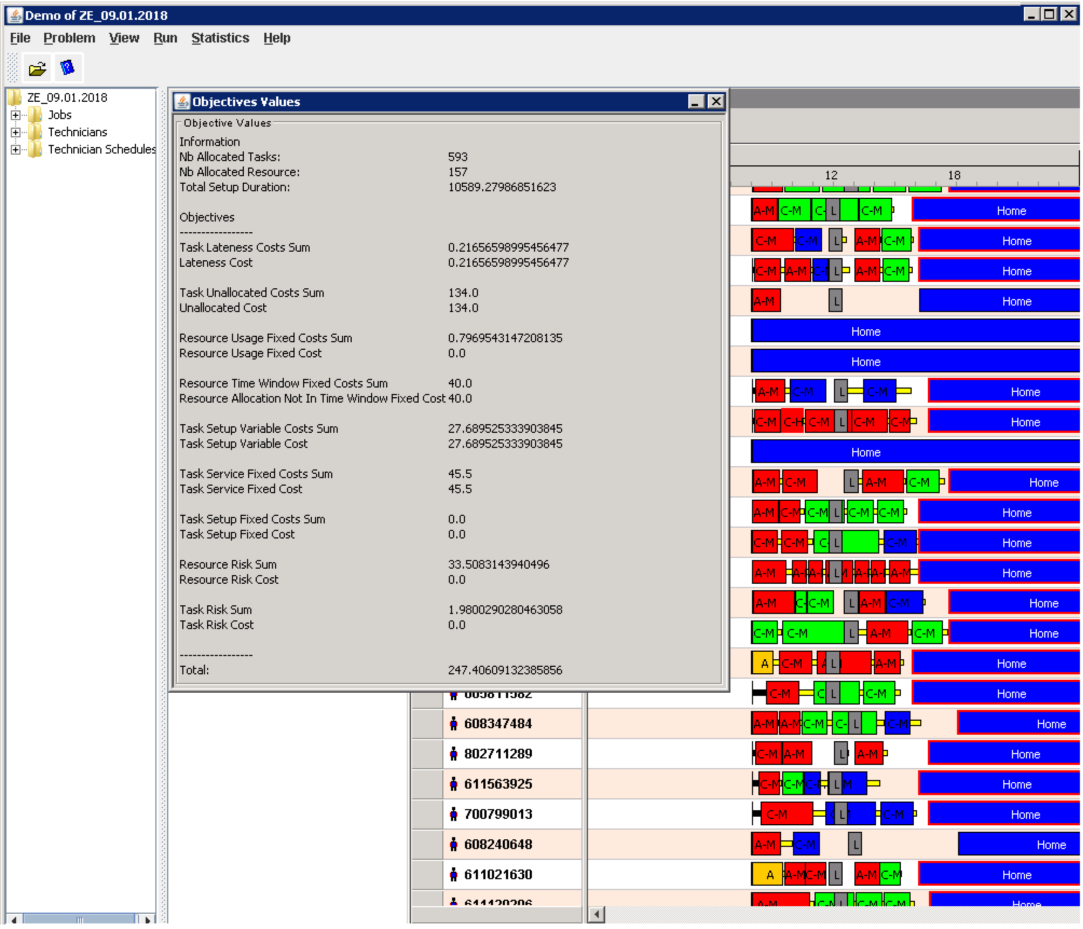


Figure 6.6 Scheduler demonstration visualizer

Gamma distributions model the uncertainty of the tasks in this test. Furthermore, the scheduling engine of the tool considers many factors in the optimisation objectives which are inherently present into the complicated real-world use cases, for instance the costs include the task unallocated cost, task setup variable cost (cost for setting up the relevant resource between two sequential tasks), task service fixed cost (cost for the relevant resource to perform the task at the relevant time position), *etc.*, where the task unallocated cost represents the cost that a task is not able to be allocated, and the variable cost and the fixed cost are associated with the travel time costs in the case of human resources (the field engineering team). The risk cost is the summation of the task related risk likelihood multiplied the task risk impact for all tasks. Then if the resulting schedule is searched by an SA method without considering risks in the objectives, the task risk cost term is zero in the objectives values box (risk impact is set to zero). Whereas, if we consider risks in the objectives, the value of the task risk cost is not zero anymore.

The complication in the real-world model also gives massive information about the tasks in a schedule, so we extract mainly the useful statistics from the database as several tasks of a schedule are stated in the Table 6.12, where the schedule is on 9th Jan 2018. TASK ID gives the unique task identity number and RESOURCE CREW ID gives the technician identity code, while POSITION shows the ordinal position for the task in its assigned technician job list.

The EARLIEST TIME and LATEST TIME give the lower and upper bounds of the task time window respectively, and as we can see the tasks have extremely wide time windows for this communication service organisation, for instance, a task has the time window from 08:00 on 9th Jan 2018 to 00:00 on 6th Feb 2018. Thus, an extra time called PRIMARY TARGET is proposed as the upper limit for the calculation of the risks. The columns MEAN DURATION and TRAVEL show the mean of the task operation time and the travel time in minutes respectively.

Furthermore, tasks in the test are composed of two types as are distinguished in the column TASK TYPE, and for task type 1 the risk is defined as the not-to-attend-by risk where the arrival time at the task is used to calculate the risk. At the same time, for task type 2 the risk is called the not-to-complete-by risk when the task completion time is applied to obtain the risk. In fact, the distribution of the task completion time results from combining the distribution of the arrival time at the task and the one of the task duration time. Therefore, the column NOT TO ATTEND BY shows the value of the not-to-attend risk and NOT TO COMPLETE BY gives the value of the not-to-complete risk. So that if the task is of type 1, it only has the not-to-attend risk and if the task is of type 2, the not-to-complete risk is considered as the risk when scheduling.

Moreover, the importance score as is shown in the column IMPORTANCE SCORE is more complicated than the one defined in previous tests, the values vary from 25 to 300, and the values of the RISK IMPACT are calculated based on the task importance score and the features of the group that the task belongs to in the system. Then the column RISK WEIGHTED COST gives the value of the risk cost which is defined as

$$\text{risk cost} = \sum_{\text{all tasks}} \text{risk likelihood} \times \text{risk impact},$$

where the risk impact depends on the task importance score, and the risk likelihood is either the not-to-attend risk or the not-to-complete risk which depends on the task type in the system.

Additionally, the start time at a task is uncertain in our models and follows a normal-like distribution, so the column ESTIMATED START TIME demonstrates the mean of the start time distribution in a time format. Then the contingency can be derived in the way that

$$\text{contingency} = \text{primary target} - \text{estimated start time},$$

and Contingency is expressed as of minute units in our analysis.

Table 6.12 Part of a result schedule from the database of the demo

RESOURCE CREW ID	TASK ID	TASK TYPE	NOT TO ATTEND BY	NOT TO COMPLETE BY	IMPORTANCE SCORE	POSITION	EARLIEST TIME	LATEST TIME	RISK IMPACT	RISK WEIGHTED COST	ESTIMATED START TIME	PRIMARY TARGET	MEAN DURATION	TRAVEL
600138240	AV-SS0JUR24	1	0		175	1	09/01/2018 08:00	06/02/2018 00:00	0.58333	0	09/01/2018 08:06	09/01/2018 13:00	105	6
600138240	AV-SS0JUB64	2		0	125	2	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 10:01	08/01/2018 17:00	95	10
600138240	AV-SS0JVA88	2		0	125	3	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 11:52	10/01/2018 13:00	95	16
600138240	AV-SS0JVA83	2		0	125	5	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 14:16	10/01/2018 13:00	95	9
600172886	AV-SS0JUL77	1	0		200	1	09/01/2018 08:00	06/02/2018 00:00	0.66667	0	09/01/2018 08:04	09/01/2018 13:00	120	4
600172886	AV-SS0JRW48	2		0	125	2	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 10:08	04/01/2018 13:00	30	4
600172886	AV-SS0JVD10	2		0	125	3	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 10:42	08/01/2018 16:00	75	4
600172886	AV-SS0JUT39	1	0		175	5	09/01/2018 13:00	06/02/2018 00:00	0.58333	0	09/01/2018 13:14	09/01/2018 17:00	75	4
600172886	AV-NGW34106	2		0	125	6	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 14:34	18/01/2018 21:00	105	5
600563165	AV-SS0JUR46	2		0	125	1	09/01/2018 06:00	06/02/2018 00:00	0.41667	0	09/01/2018 08:00	09/01/2018 18:30	95	8

We found that apart from the given task type, the tasks can be divided into three groups based on the task primary target time. The tasks with the target time before 09 Jan 2018 belong to failed tasks, and the tasks with the target time after that day are defined as future tasks, while the tasks with the target time within the day are called present tasks. We noticed that the tasks of type 1 are all present tasks whereas only tasks of type 2 have time windows on a different day other than 09 Jan 2018. Therefore, type 1 tasks and type 2 tasks with the time window within 09 Jan all belong to the group of present tasks. Additionally, the task with a task importance score more than 200 is regarded as a task with high importance, and the importance score between 125 and 200 defines a task with medium importance, so that a task with a score less than 125 is a task with low importance.

Table 6.13 Average Contingency (minutes) for present tasks (with risk cost)

Average Contingency	**Task importance**		
Start time	High	Medium	Low
09/01/2018 08:00	303	347	626
09/01/2018 08:00 - 09:00	294	362	624
09/01/2018 09:00 - 10:00	205	324	543
09/01/2018 10:00 - 11:00	158	307	469
09/01/2018 12:00 - 13:00	240	240	363
09/01/2018 13:00 - 14:00	211	218	241
09/01/2018 14:00 - 15:00	151	157	294
09/01/2018 15:00 - 16:00	No tasks	311	No tasks
Total	262	309	515

Table 6.14 Number of tasks as to different Contingency (with risk cost)

No. of tasks	Task importance		
Average contingency (minutes)	High	Medium	Low
0 – 50	0	0	0
50 – 100	0	0	1
100 – 150	8	2	0
150 – 200	8	13	4
200 – 250	18	42	7
250 – 300	33	45	5
300 – 350	0	1	4
350 – 400	0	1	10
400 – 450	0	2	13
450 – 500	0	3	6
500 – 550	0	6	10
550 – 600	1	12	19
600 – 650	1	1	11
650 – 700	0	1	11
700 – 750	1	0	17
750 – 800	0	0	2

There are two resulting schedules, one is obtained by considering risks in objectives and the other is without considering risks in objectives. Table 6.13 shows the average Contingency for each group of tasks where the start time is within the same time slot and at the same importance level. Table 6.14

demonstrates the frequency for different average Contingency of tasks in terms of different importance levels. These two tables show statistics of the schedule searched with risks in objectives in the model, whereas Table 6.15 and 6.16 shows those of the schedule obtained without risks in objectives. Moreover, we also acquired Contingency for failed and future tasks, but as we could imagine Contingency is negative for the failed tasks and is extremely large for future tasks. Therefore, it is more valuable to discuss Contingency of present tasks as the four tables show.

Table 6.15 Average Contingency (minutes) for present tasks (no risk cost)

Average Contingency	**Task importance**		
Start time	High	Medium	Low
09/01/2018 08:00	287	322	647
09/01/2018 08:00 – 09:00	300	396	634
09/01/2018 09:00 – 10:00	120	298	555
09/01/2018 10:00 – 11:00	143	318	454
09/01/2018 11:00 – 12:00	89	245	380
09/01/2018 12:00 – 13:00	240	240	326
09/01/2018 13:00 – 14:00	209	221	295
09/01/2018 14:00 – 15:00	143	157	250
09/01/2018 15:00 – 16:00	115	311	179
Total	245	303	507

Table 6.16 Number of tasks as to different Contingency (no risk cost)

No. of tasks	Task importance		
Average contingency (minutes)	High	Medium	Low
0 – 50	2	0	0
50 – 100	1	0	1
100 – 150	2	2	2
150 – 200	11	3	3
200 – 250	3	7	6
250 – 300	17	49	6
300 – 350	33	42	5
350 – 400	0	1	3
400 – 450	0	1	8
450 – 500	0	4	10
500 – 550	0	2	5
550 – 600	0	5	8
600 – 650	1	10	20
650 – 700	1	2	9
700 – 750	0	0	10
750 – 800	1	1	22

What is more, we compared some statistics between the two schedules. In Table 6.17 and Figure 6.7, Contingency of high importance tasks between two schedules is put together to investigate the improvement. As we can see, including risks in the objectives increases Contingency a lot for top priority tasks, especially for those tasks scheduled at the time close to noon in the morning or in the late afternoon. The more the Contingency, more flexibility one technician may have to work on the task, so that more chance to be successful to arrive at or finish the task in time.

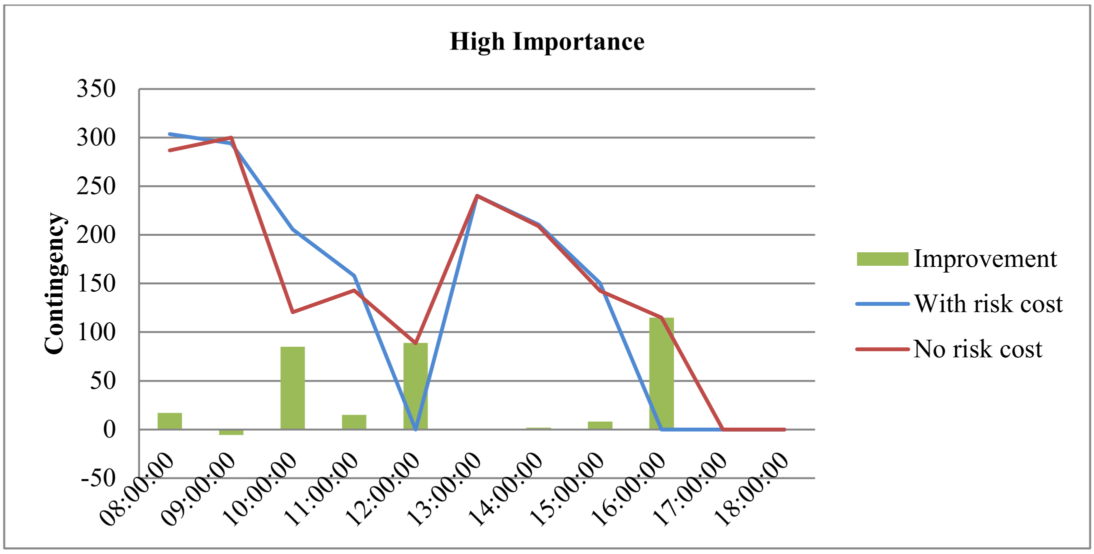


Figure 6.7 Contingency for high importance tasks

Table 6.17 Contingency of high importance tasks

	Contingency (minutes)		
start time	No risk cost	With risk cost	Improvement
08:00:00	287	303	17
08:00:00 – 09:00:00	300	294	-6
09:00:00 – 10:00:00	120	205	85
10:00:00 – 11:00:00	143	158	15
11:00:00 – 12:00:00	89	No task	–
12:00:00 – 13:00:00	240	240	0
13:00:00 – 14:00:00	209	211	2
14:00:00 – 15:00:00	143	151	8
15:00:00 – 16:00:00	115	No Task	–
Total	245	262	17

Meanwhile, Table 6.18 and Figure 6.8 give Contingency of low importance tasks between the two schedules. As we may observe, the average Contingency for the less important tasks scheduled in the early morning or the early afternoon decreases a little in order to make room for top priority tasks. However, the average Contingency for the tasks scheduled in the late morning or late afternoon also increases which is similar to the improvement for top priority tasks. Therefore, a conclusion can be reached that considering risks in the objectives will increase Contingency on average for all the tasks with different importance scores, and improve the schedule to carry out tasks with more success.

Table 6.18 Contingency of low importance tasks

	Contingency (minutes)		
start time	No risk cost	With risk cost	Improvement
08:00:00	647	626	-21
08:00:00 – 09:00:00	634	624	-9
09:00:00 – 10:00:00	555	543	-12
10:00:00 – 11:00:00	454	469	14
11:00:00 – 12:00:00	380	428	48
12:00:00 – 13:00:00	326	363	37
13:00:00 – 14:00:00	295	241	-54
14:00:00 – 15:00:00	250	294	44
15:00:00 – 16:00:00	179	No task	–
16:00:00 – 17:00:00	127	No task	–
Total	507	515	8

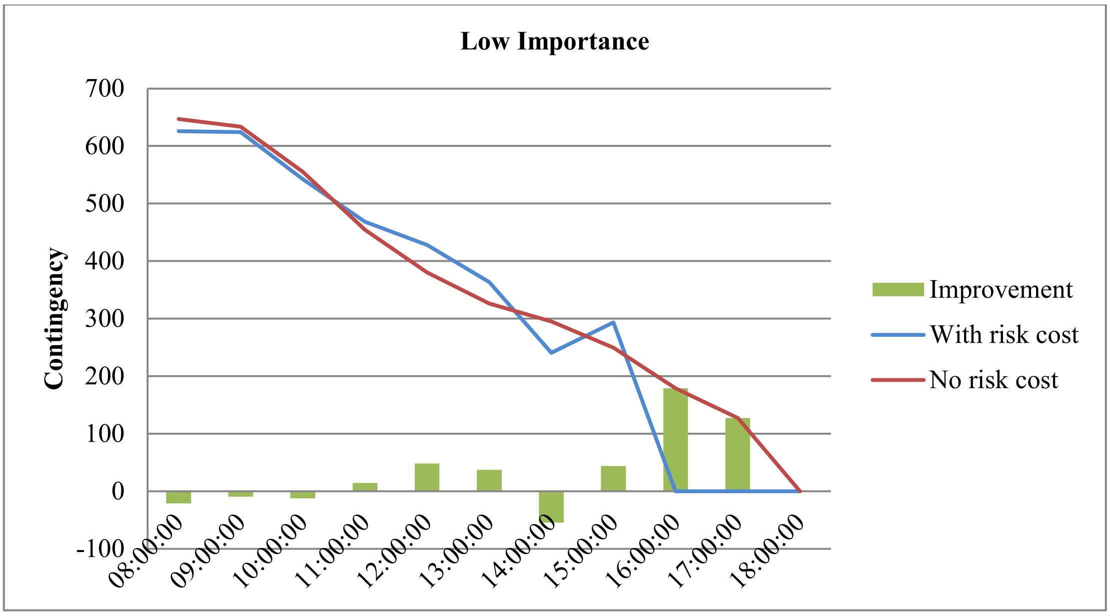


Figure 6.8 Contingency for low importance tasks

In addition, Table 6.19 shows the not-to-attend-by risks between two schedules while Table 6.20 gives the not-to-complete-by risks. Obviously, both kinds of risks decrease and the number of risk-free tasks increases when the engine use risks in the objective functions. Moreover, the number of missed tasks, which represents the potential number of tasks that technicians may fail to make or finish in time, is defined as

$$\text{No. of missed tasks} = \text{average risk} \times \text{No. of tasks}.$$

In fact, the number of missed tasks is the unweighted total risk of all tasks. Thus from the table, we can see that when risks are considered in objectives, the total risks of two types both drop a lot compared to the resulting schedule with no risks considered to the schedule.

Table 6.19 Not-to-attend-by risks

No. of tasks	NOT_TO_ATTEND_BY		
Risk	No risk cost	With risk cost	Decreased
> 10%	6	0	6
5% - 10%	7	3	4
0% - 5%	15	7	8
0%	192	210	-18
Missed task	1.91	0.29	1.62

Table 6.20 Not-to-complete-by risks

No. of tasks	NOT_TO_COMPLETE_BY		
Risk	No risk cost	With risk cost	Decreased
> 10%	3	0	3
5% - 10%	1	0	1
0% - 5%	2	2	0
0%	368	372	-4
Missed task	2.23	0.04	2.18

6.7 Multi-optimisations

As is illustrated in Chapter 5.2, simulation is beneficial to verify analytic solutions, and the operation plan can be monitored at different time positions in the day, which is the reason we introduce simulation into our models.

6.7.1 Simulation comparison between the travel time and the risk models

It has been discussed in Chapter 5.2.1 that the realisation of a schedule is straightforward to generate a duration time for each task, and then the number of the failed tasks at the end of the day is counted to assess the behaviour of the schedule. With the aim to have some failed tasks in a given schedule, the test sample we used in the simulation is of 100 tasks and 12 technicians. In other words, one technician on average has 8.3 tasks, which can be regarded as a tightly scheduled task list, so that some tasks may be missed. Here if the technician arrives at a customer site after the upper bound of the task time window, this task will be regarded as a failed task or a missed task.

Meanwhile, the average risk of tasks in such schedule is high, even as much as 15% chance to miss the appointment for some tasks. Otherwise, suppose all task risks are smaller than 1% in a schedule as we obtained before, it usually occurs that there are no failed tasks when we realise the schedule in the simulation. Thus, it is hard to assess the improvement of a schedule that includes the risks in the optimisation.

In addition, because the task duration following a normal distribution or a Gamma distribution does not affect the task behaviour significantly, it is assumed that all the task durations are normally distributed in the simulation tests. Also, without loss of generality, we assume all the tasks require the same skills and all the technicians have the ability to deal with these tasks.

Table 6.21 Number of failed tasks in simulations

Simulation	Travel time optimisation once in a day	Risk optimisation once in a day
Total number of failed tasks in 30 tests	362	248
Average number of failed tasks	12.07	8.27

Simulation is run to compare the risk models with the traditional travel time model using the number of missed appointments. The figure drops from on average 12.07 to 8.27 as is shown in Table 6.21. The missed tasks are not many because the time window constraints are considered in the travel time model.

6.7.2 Results from the re-optimisation simulation

A simulation of the single optimisation realises a schedule which implements optimisation once at the beginning of the day, then the number of failed tasks can be acquired in the end. Whereas, a simulation of the multiple optimisations shows the effect of the re-optimisation in the way that after the initial optimisation, the resulting schedule is followed in the operation until the second optimisation time point, then the number of failed tasks up to this time point can be obtained, and based on the process status we optimise the schedule for the remaining tasks, then the new schedule is carried out until the next optimisation time and follows the same procedures. Finally, the total number of the failed tasks accumulated in the process during the day is assessed.

Table 6.22 Number of failed tasks in simulations

Simulation	Single optimisation in a day	Multi-optimisation in a day
Total number of failed tasks in 30 tests	248	132
Average number of failed tasks per test	8.27	4.40

The simulation for 100 tasks and 12 technicians runs 30 tests and results are shown in Table 6.22, and the optimisation time points are at 8:00, 11:00, 14:00

and 17:00 (BST). From the table we can see that the number of failed tasks in the simulation of the multiple optimisations drops significantly compared to the one of the single optimisation, from on average 8.27 failed tasks to 4.4 tasks. Thus, the simulation verifies our thoughts that by introducing re-optimisation, the risk of missing appointments can be decreased further, and the re-optimisation improves the schedule dynamically.

Table 6.23 Schedule at 8:00 (BST)

Resource id	Task id	Lower limit of the time window	Upper limit of the time window	Risk
7	40	08:00:00	11:00:00	0.00000
7	3	08:00:00	11:00:00	0.000007
7	38	08:00:00	11:00:00	0.078459
7	56	10:00:00	13:00:00	0.044039
7	53	10:00:00	13:00:00	0.384512
7	75	12:00:00	15:00:00	0.187905
7	94	14:00:00	17:00:00	0.138337
7	27	14:00:00	17:00:00	0.297542
7	95	14:00:00	17:00:00	0.461294
8	8	08:00:00	11:00:00	0.000000
8	41	08:00:00	11:00:00	0.037560
8	60	10:00:00	13:00:00	0.025286
8	65	10:00:00	13:00:00	0.203328
8	78	12:00:00	15:00:00	0.102347
8	72	12:00:00	15:00:00	0.376922
8	91	14:00:00	17:00:00	0.230421
8	84	14:00:00	17:00:00	0.430325

More specifically, a part of the schedule result at 8:00 (BST) is shown in Table 6.23, we can see that Task 65 is scheduled initially to Technician 8, and it has a high probability of being missed in the initial schedule and it actually fails in the simulation result if there is no re-optimisation action. Then from a part of the report gathered at 11:00 (BST) in Table 6.24, it can be seen that Technician 8 spent longer than expected on his 1st and the task he/she is doing, especially at the 2nd task site, he was going to spend 100 minutes to finish the job while the expected working time is 65 minutes.

Table 6.24 Report at 11:00 (BST)

Resource id	Task id	Task status	Task start time	Travel time	Real duration time	Mean of the estimate duration
7	40	Finish	08:04:23	4	53	45
7	3	Finish	08:57:40	0	48	50
7	38	Finish	09:48:57	3	56	60
8	8	Finish	08:16:26	16	80	65
8	41	Doing	09:54:49	18	100	65

However, we re-optimise at 11:00, and the new schedule at 11:00 shown in Table 6.25 illustrates that Task 65 is assigned to Technician 7 as his/her 1st task from 11:00, this ensures that Task 65 is successfully carried out. And in the initial schedule, Technician 8 would go to Customer 60 before visit 65, which would increase the possibility of missing Task 65. Therefore, the results conclude that multi-optimisation at different time points in a day can improve the schedule with the up-to-date task operating time.

Table 6.25 Schedule at 11:00 (BST)

Resource id	Task id	Lower limit of the time window	Upper limit of the time window	Risk
7	65	10:00:00	13:00:00	0.000000
7	60	10:00:00	13:00:00	0.269380
7	22	12:00:00	15:00:00	0.140170
7	19	12:00:00	15:00:00	0.386894
7	91	14:00:00	17:00:00	0.220154
7	84	14:00:00	17:00:00	0.448647
8	54	10:00:00	13:00:00	0.000000
8	82	12:00:00	15:00:00	0.000000
8	75	12:00:00	15:00:00	0.158527
8	94	14:00:00	17:00:00	0.117360
8	28	14:00:00	17:00:00	0.355559

Chapter 7

Conclusions and Future Research

This chapter concludes the thesis by summarising the contributions and suggesting opportunities for further research and development.

7.1 Conclusions and summaries

This research has analysed the risks observed in service-delivery operations and considered the risks in the operational planning process from a new perspective. The research makes three contributions to the new workforce scheduling and vehicle routing problem with time windows and stochastic durations.

- Calculating the risks considering their propagation from one task to the next, and proposing efficient methods for estimating the risks;
- Developing new workforce scheduling and vehicle routing models with an objective of minimising the risks, and implement a simulated annealing algorithm for solving the models and for re-optimisation;
- Conducting computational experiments to evaluate the performance of the new models and compare the results with those from the traditional travel time model, also applying the risk calculation methods and risk minimisation models to a real-world problem in the telecommunication sector.

First, after reviewing risks in different areas and discussing variants of VRPs in Chapter 2, the creative definition, expression and estimation for risks are given in Chapter 3, which are derived from the analysis of the real data from a telecommunication organisation.

Then the Simpson's rule and Monte Carlo method and a new developed accumulation method are used to calculate the multi-integral expression of risks for all kinds of distributions of task duration times. Thereafter, a summation method is proposed to calculate risk which works well when the task duration time is normally distributed. The start time of each task does not align exactly with a normal distribution due to the effect of the lower limit of the time window, but the summation method supposes the start time is approximately normal distributed so that the summation property for normal distributions is used to obtain the arrival time distribution of the next customer, and the property can benefit calculating the risks relatively accurate and saving computing time.

Therefore, in order to have the approximate normal distribution for the start time, a stratified estimation method analyses three kinds of relations between the arrival time and the lower limit of the time window and constructs three estimation models to approximate the distribution of the start time as a normal distribution. Afterwards, the summation method can utilise this estimation to obtain the risks in models.

Second, increasing customer satisfaction is always a popular topic for managers and researchers with the aim to build a more customer-oriented business. It is particularly true when planning geographically distributed services in the fields. Therefore, it is important for service providing organisations to consider visiting time windows, the stochastic service time and travel time in the workforce scheduling and VRPs.

The risk models proposed in Chapter 4 are another contribution where the risks are incorporated into the set of objectives to be minimised during the optimisation process, whereas previous relevant researches use a chance-constrained approach to the problem. For instance, some of the previous works consider the probability of route duration exceeding the threshold of the driver's workload, while the others set extra restrictions on the probability of individual time window constraints to be violated. Furthermore, the objectives of the problem are related to traditional routing costs in these approaches.

Additionally, it is valuable to consider tasks possessing different importance or priority according to the business objectives. Therefore, the priority task risk model is proposed to deal with this case, and the priority risk would be the task risk multiplies the corresponding task importance score.

To solve the models, a simulated annealing algorithm is implemented in chapter 5, utilising the swap and insert operators for generating neighbouring solutions and the risk estimation method for calculating the objective value of solutions. The algorithm is used to produce a whole-day schedule at the start of the day and also to re-optimise the schedule at certain time points considering new information.

The third contribution is investigating the effect of the risk models and re-optimisation on the service delivery performance through computational experiments. A travel time minimisation model is used as a benchmark for comparison. The results are presented in chapter 6.

In terms of risks of missing appointments, the average risk of the schedule drops from 40.97% in the travel time minimisation model to 11.11% in the risk minimisation model. Furthermore, if we consider the priority when minimising risks, the average risk for high priority tasks decreases to 1.78% at a cost of increasing the risk of 4.82% for low priority tasks.

The comparison is also done in terms of contingency. The contingency increases on average for both high and low priority tasks, and especially significantly for high priority tasks if we consider the risk cost in the objective comparing with the test of excluding the risk cost. Moreover, there is no task scheduled in the late morning and afternoon when scheduling considering the risk cost.

Simulation is run to compare the risk minimisation models with the traditional travel time model on the number of missed appointments. The missed appointments drop from on average 12.07 tasks to 8.27 tasks. In addition, the missed tasks are not numerous because the time window constraints are considered in the travel time model.

Simulation is also used to compare the approach of optimising once at the beginning of the day with the approach that re-optimises the schedule during the day. The results show that re-optimisation can reduce the number of missed appointments by half of the number in the original schedule execution.

Additionally, the risk component has been incorporated into the scheduling engine of the collaborating organisation in the telecommunication sector, as part of the objective function.

7.2 Future research

For further investigation, the research undertaken in this thesis could be extended so that more practical findings can be obtained.

To start with, in the thesis, the effect of the lower bound of the time window is regarded primary and significant to the start time for a given task, and further to the arrival time for the task following the given task. While trying to minimise the risks of arriving task sites later than the upper limits of their time windows, the tasks are assumed to be carried out even though the technician arrives late. For some service systems, one may argue that with the probability of the arrival time later than the upper bound of the time window, the technician would not be able to execute the task and if this happens the service time for this task should not be included in calculating the risk for the next task. Therefore, it is meaningful and useful to investigate the risk propagation under such strictly hard time windows.

Furthermore, the task duration which follows either a normal distribution or a Gamma distribution is studied in the thesis. Further research could investigate the combination of Gamma and normal distributions for the task duration, which may better reflect the realistic task duration. This investigation might be useful to some other areas involving a combination of Gamma and normal distributions.

In the re-optimisation part of this research, experiments are carried out for the situation where information is updated and the model is run at fixed time points. It would be interesting and beneficial to apply the risk minimisation model in an event-driven framework, where the model will dynamically update the risks according to external events such as the task progression or completion, task rejection, task actual delays.

References

Agra, A., Christiansen, M., Figueiredo, R., Hvattum, L. M., Poss, M., & Requejo, C. (2013). The robust vehicle routing problem with time windows. *Computers and Operations Research*, *40*(3), 856–866. http://doi.org/10.1016/j.cor.2012.10.002

Alfa, A. S., Heragu, S. S., & Chen, M. (1991). A 3-OPT based simulated annealing algorithm for vehicle routing problems. *Computers and Industrial Engineering*, *21*(1–4), 635–639. http://doi.org/10.1016/0360-8352(91)90165-3

Almagooshi, S. (2015). Simulation Modelling in Healthcare: Challenges and Trends. *Procedia Manufacturing*, *3*(Ahfe), 301–307. http://doi.org/10.1016/j.promfg.2015.07.155

Ando, N., & Taniguchi, E. (2006). Travel time reliability in vehicle routing and scheduling with time windows. *Networks and Spatial Economics*, *6*(3–4), 293–311. http://doi.org/10.1007/s11067-006-9285-8

Andreatta, G., Casula, M., De Francesco, C., & De Giovanni, L. (2016). A branch-and-price based heuristic for the stochastic vehicle routing problem with hard time windows. *Electronic Notes in Discrete Mathematics*, *52*, 325–332. http://doi.org/10.1016/j.endm.2016.03.043

Arezoumandi, M. (2011). Estimation of travel time reliability for freeways using mean and standard deviation of travel time. *Journal of Transportation Systems Engineering and Information Technology*, *11*(6), 74–84. http://doi.org/10.1016/S1570-6672(10)60149-3

Bai, J., So, K. C., & Tang, C. (2016). A queueing model for managing small projects under uncertainties. *European Journal of Operational Research*, *253*(3), 777–790. http://doi.org/10.1016/j.ejor.2016.02.052

Baker, B. M., & Ayechew, M. A. (2003). A genetic algorithm for the vehicle routing problem. *Computers and Operations Research*, *30*, 787–800. http://doi.org/https://doi.org/10.1016/S0305-0548(02)00051-5

Baldacci, R., Mingozzi, A., & Roberti, R. (2012). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research*, *218*(1), 1–6. http://doi.org/10.1016/j.ejor.2011.07.037

Bastian, C., & Rinnooy Kan, A. H. G. (1992). The stochastic vehicle routing problem revisited. *European Journal of Operational Research*, *56*(3), 407–412. http://doi.org/10.1016/0377-2217(92)90323-2

Benedettini, O., & Tjahjono, B. (2009). Towards an improved tool to facilitate simulation modelling of complex manufacturing systems. *International Journal of Advanced Manufacturing Technology*, *43*(1–2), 191–199. http://doi.org/10.1007/s00170-008-1686-z

Bertsimas, D. (1988). Probabilistic combinatorial optimization problems. http://doi.org/10.1016/j.foodchem.2015.03.003

Bertsimas, D. J. (1992). A vehicle routing problem with stochastic demand. *Operations Research*, *40*(3), 574–585.

Billingsley, P. (1995). *Probability and Measure*. *Soccer and Society* (3rd ed., Vol. 19). New York: John Wiley & Sons, Inc. http://doi.org/10.1080/14660970.2016.1171215

Bouzaïene-Ayari, B., Dror, M., & Laporte, G. (1993). Vehicle routing with stochastic demands and split deliveries. *Foundations of Computing and Decision Sciences*, *18*, 63–69.

Branda, M. (2014). Sample approximation technique for mixed-integer stochastic programming problems with expected value constraints. *Optimization Letters*, *8*(3), 861–875. http://doi.org/10.1007/s11590-013-0642-5

Branda, M., Novotný, J., & Olstad, A. (2016). Fixed interval scheduling under uncertainty - A tabu search algorithm for an extended robust coloring formulation. *Computers and Industrial Engineering*, *93*, 45–54. http://doi.org/10.1016/j.cie.2015.12.021

Bruni, M. E., Beraldi, P., Guerriero, F., & Pinto, E. (2011). A heuristic approach for resource constrained project scheduling with uncertain activity durations. *Computers and Operations Research*, *38*(9), 1305–1318. http://doi.org/10.1016/j.cor.2010.12.004

Bruni, M. E., Di Puglia Pugliese, L., Beraldi, P., & Guerriero, F. (2017). An adjustable robust optimization model for the resource-constrained project scheduling problem with uncertain activity durations. *Omega (United Kingdom)*, *71*, 66–84. http://doi.org/10.1016/j.omega.2016.09.009

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). Applying the ANT System to the Vehicle Routing Problem. In S. Voß, S. Martello, I. H. Osman, & C. Roucairol (Eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization* (pp. 285–296). Boston, MA: Springer US. http://doi.org/10.1007/978-1-4615-5775-3_20

Burkard, R. E., & Rendl, F. (1984). A thermodynamically motivated simulation procedure for combinatorial optimization problems. *European Journal of Operational Research*, *17*(2), 169–174. http://doi.org/10.1016/0377-2217(84)90231-5

Černý, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, *45*(1), 41–51. http://doi.org/10.1007/BF00940812

Chakrabortty, R. K., Sarker, R. A., & Essam, D. L. (2017). Resource constrained project scheduling with uncertain activity durations. *Computers and Industrial Engineering*, *112*, 537–550. http://doi.org/10.1016/j.cie.2016.12.040

Chang, M. S. (2005). A vehicle routing problem with time windows and stochastic demands. *Journal of the Chinese Institute of Engineers, Transactions of the Chinese Institute of Engineers,Series A/Chung-Kuo Kung Ch'eng Hsuch K'an*, *28*(5), 783–794. http://doi.org/10.1080/02533839.2005.9671048

Chang, T.-S., Nozick, L. K., & Turnquist, M. A. (2005). Multiobjective Path Finding in Stochastic Dynamic Networks, with Application to Routing Hazardous Materials Shipments. *Transportation Science*, *39*(3), 383–399. http://doi.org/10.2307/25769258

Chang, T. S., Wan, Y. wah, & OOI, W. T. (2009). A stochastic dynamic traveling salesman problem with hard time windows. *European Journal of Operational Research*, *198*(3), 748–759. http://doi.org/10.1016/j.ejor.2008.10.012

Christofides, N., Mingozzi, A., & Toth, P. (1979). The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, & C. Sandi (Eds.), *Combinatorial Optimization* (pp. 315–338). Chichester: Wiley.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, *12*(4), 568–581.

Colorni, A., Dorigo, M., & Maniezzo, V. (1991). Distributed optimization by ant colonies. In F. Varela & P. Bourgine (Eds.), *Proceedings of the european conference on artificial life* (pp. 134–142). Amsterdam: Elsevier.

Cook, T. M., & Russell, R. (1978). A Simulation and Statistical Analysis of Stochastic Vehicle Routing with Timing Constraints. *Decision Sciences*, *9*, 673–687. http://doi.org/10.1111/j.1540-5915.1978.tb00753.x

Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J. Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, *53*(5), 512–522. http://doi.org/10.1057/palgrave.jors.2601319

Cox Jr., L. A. (2009). *Risk Analysis of Complex and Uncertain Systems*. Denver: Springer. Retrieved from https://books.google.co.uk/books?id=qu8QdRIcX5QC

Damm, R. B., Resende, M. G. C., & Ronconi, D. P. (2016). Computers & Operations Research A biased random key genetic algorithm for the fi eld technician scheduling problem. *Computers and Operation Research*, *75*, 49–63. http://doi.org/10.1016/j.cor.2016.05.003

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*(1), 80–91. http://doi.org/10.1287/mnsc.1090.1000

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management Science*, *6*(1), 80–91. http://doi.org/10.1287/mnsc.6.1.80

Dantzig, G. B., & Wolfe, P. (1960). Decomposition Principle for Linear Programs. *Operations Research*, *8*(1), 101–111.

Deif, I., & Bodin, L. (1984). Extension of the Clarke and Wright Algorithm for solving the vehicle routing problem with backhauls. *Proceedings of the Babson*

Conference on Software Uses in Transportation and Logistics Management, 75–96.

Dror, M. (1993). Modeling vehicle routing with uncertain demands as a stochastic program: Properties of the corresponding solution. *European Journal of Operational Research*, *64*(3), 432–441. http://doi.org/10.1016/0377-2217(93)90132-7

Dror, M., Laporte, G., & Louveaux, F. V. (1993). Vehicle routing with stochastic demands and restricted failures. *ZOR Zeitschrift Für Operations Research Methods and Models of Operations Research*, *37*(3), 273–283. http://doi.org/10.1007/BF01415995

Dror, M., & Trudeau, P. (1986). Stochastic vehicle routing with modified savings algorithm. *European Journal of Operational Research*, *23*(2), 228–235. http://doi.org/10.1016/0377-2217(86)90242-0

Drwal, M. (2018). Robust scheduling to minimize the weighted number of late jobs with interval due-date uncertainty. *Computers and Operations Research*, *91*, 13–20. http://doi.org/10.1016/j.cor.2017.10.010

Dumbravă, V., & Vladut-Severian, I. (2013). Using Probability – Impact Matrix in Analysis and Risk Assessment Projects. *Journal of Knowledge Management, Economics and Information Technology*, *42*(December), 76–96. Retrieved from http://www.scientificpapers.org/wp-content/files/07_Dumbrava_Iacob-USING_PROBABILITY__IMPACT_MATRIX_IN__ANALYSIS_AND_RISK_ASSESSMENT_PROJECTS.pdf

Ehmke, J. F., & Campbell, A. M. (2014). Customer acceptance mechanisms for home deliveries in metropolitan areas. *European Journal of Operational Research*, *233*(1), 193–207. http://doi.org/10.1016/j.ejor.2013.08.028

Ehmke, J. F., Campbell, A. M., & Urban, T. L. (2015). Ensuring service levels in routing problems with time windows and stochastic travel times. *European Journal of Operational Research*, *240*(2), 539–550. http://doi.org/10.1016/j.ejor.2014.06.045

Ehmke, J. F., & Mattfeld, D. C. (2011). Integration of information and optimization models for vehicle routing in urban areas. *Procedia - Social and Behavioral Sciences*, *20*, 110–119. http://doi.org/10.1016/j.sbspro.2011.08.016

Elgesem, A. S., Skogen, E. S., Wang, X., & Fagerholt, K. (2018). A traveling salesman problem with pickups and deliveries and stochastic travel times: An application from chemical shipping. *European Journal of Operational Research*, *0*, 1–16. http://doi.org/10.1016/j.ejor.2018.02.023

Epperson, J. F. (2007). *An Introduction to Numerical Methods and Analysis*. John Wiley & Sons. Retrieved from https://books.google.co.uk/books?id=Mp8-z5mHptcC

Feng, N., Wang, H. J., & Li, M. (2014). A security risk analysis model for information systems: Causal relationships of risk factors and vulnerability propagation analysis. *Information Sciences*. http://doi.org/10.1016/j.ins.2013.02.036

Fischetti, M., Toth, P., & Vigo, D. (1994). A branch-and-bound algorithm for the capacitated vehicle routing problem on directed graphs. *Operations Research*, *42*(5), 846–859. http://doi.org/10.1287/opre.42.5.846

Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, *11*(2), 109–124. http://doi.org/10.1002/net.3230110205

Fisher, M. L., Jörnsten, K. O., & Madsen, O. B. G. (1997). Vehicle routing with time windows: Two optimization algorithms. *Operations Research*, *45*(3), 488–492. http://doi.org/10.1287/opre.45.3.488

Gendreau, M., Hertz, A., & Laporte, G. (1994). A Tabu search heuristic for the vehicle routing problem. *Management Science*, *40*(10), 1276–1290. http://doi.org/10.1287/mnsc.40.10.1276

Gendreau, M., Laporte, G., & Potvin, J.-Y. (2002). Metaheuristics for the capacitated VRP. In P. Toth & D. Vigo (Eds.), *The Vehicle Routing Problem* (pp. 129–154). SIAM.

Gendreau, M., Laporte, G., & Seguin, R. (1996). A Tabu search heuristic for the vehicle routing problem with stochastic demands and customers. *Operations Research*, *44*(3), 469–477.

Gendreau, M., Laporte, G., & Séguin, R. (1995). An exact algorithm for the vehicle routing problem with stochastic demands and customers. *Transportation Science*, *29*(2), 143–155. http://doi.org/10.1287/trsc.29.2.143

Ghilas, V., Demir, E., & Woensel, T. Van. (2016). A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. *Transportation Research Part B: Methodological*, *91*, 34–51. http://doi.org/10.1016/j.trb.2016.04.015

Gillett, B. E., & Miller, L. R. (1974). A Heuristic Algorithm for the Vehicle-Dispatch Problem. *Operations Research*, *22*(2), 340–349. http://doi.org/10.1287/opre.22.2.340

Glover, F. (1989). Tabu Search-Part I. *ORSA Journal on Computing*, *1*(3), 190–206. http://doi.org/10.1002/jbm.820231004

Goetschalckx, M., & Jacobs-Blecha, C. (1989). The vehicle routing problem with backhauls. *European Journal of Operational Research*, *42*(1), 39–51. http://doi.org/10.1016/0377-2217(89)90057-X

Golden, B., Assad, A., Levey, L., & Gheysens, F. (1984). The fleet size and mix vehicle routing problem. *Computers and Operation Research*, *11*, 49–66. http://doi.org/10.1057/palgrave.jors.2600763

Golub, G. H., & Van Loan, C. F. (2013). *Matrix Computations*. (Baltimore, Ed.) (4th ed.). Johns Hopkins University Press. Retrieved from https://books.google.co.uk/books?id=X5YfsuCWpxMC

Hall, R. W. (1986). The fastest path through a network with random time-dependent travel times. *Transportation Science*, *20*(3), 182–188.

Hashimoto, H., Yagiura, M., Imahori, S., & Ibaraki, T. (2013). Recent progress of local search in handling the time window constraints of the vehicle routing problem. *Annals of Operations Research*, *204*(1), 171–187. http://doi.org/10.1007/s10479-012-1264-5

Helsgaun, K. (2000). Effective implementation of the Lin-Kernighan traveling salesman heuristic. *European Journal of Operational Research*, *126*(1), 106–130. http://doi.org/10.1016/S0377-2217(99)00284-2

Herroelen, W., & Leus, R. (2005). Project scheduling under uncertainty: Survey and research potentials. *European Journal of Operational Research*, *165*(2), 289–306. http://doi.org/10.1016/j.ejor.2004.04.002

Hillson, D. a., & Hulett, D. T. (2004). Assessing risk probability: alternative approaches. *2004 PMI Global Congress Proceedings*, (Collins 1979), 1–7. Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Assessing+Risk+Probability:+Alternative+Approaches#1

Hopfield, J. J., & Tank, D. W. (1985). "Neural''' computation of decisions in optimization problems." *Biological Cybernetics*, *52*(3), 141–152. http://doi.org/10.1007/BF00339943

Hopkin, P. (2012). *Fundamentals of Risk Management: Understanding, Evaluating and Implementing Effective Risk Management* (2nd ed.). London: Kogan Page. Retrieved from https://books.google.co.uk/books?id=XyfmyNbC5a8C

IRM. (2018). *A risk practitioners guide to ISO 31000 : 2018*. Institute of Risk Management.

ISO. (2018). *Risk management – principles and guidelines*. ISO 31000:2018.

Jabali, O., Rei, W., Gendreau, M., & Laporte, G. (2012). *New Valid Inequalities for the Multi- Vehicle Routing Problem with Stochastic Demands*. *CIRRELT*.

Jaillet, P. (1988). A priori solution of a traveling salesman problem in which a random subset of the customers are visited. *Operations Research*, *36*(6), 929–936.

Jaillet, P., Qi, J., & Sim, M. (2016). Routing Optimization Under Uncertainty. *Operations Research*, *64*(1), 186–200. http://doi.org/10.1287/opre.2015.1462

Jamili, A. (2016). Robust job shop scheduling problem: Mathematical models, exact and heuristic algorithms. *Expert Systems with Applications*, *55*, 341–350. http://doi.org/10.1016/j.eswa.2016.01.054

Jeffreys, H., Jeffreys, B., & Swirles, B. (1999). *Methods of Mathematical Physics*. Cambridge University Press. Retrieved from https://books.google.co.uk/books?id=Qs-xdYBQ_5wC

Jelen, F. C., & Black, J. H. (1983). *Cost and optimization engineering* (2nd ed.). New York: McGraw-Hill Book Company.

Jézéquel, A. (1984). *Probabilistic vehicle routing problem*. Massachusetts institute of technology.

Jula, H., & Dessouky, M. (2006). Truck route planning in nonstationary stochastic networks with time windows at customer locations. *IEEE Transactions on Intelligent Transportation Systems*, *7*(1), 51–62. http://doi.org/10.1109/TITS.2006.869596

Kalbfleisch, J. G. (2012). *Probability and Statistical Inference: Volume 1: Probability* (2nd ed.). New York: Springer Science & Business Media. Retrieved from https://books.google.co.uk/books?id=DenpBwAAQBAJ

Kalos, M. H., & Whitlock, P. A. (2008). *Monte Carlo Methods* (2nd ed.). Weinheim: Wiley-VCH. Retrieved from https://books.google.co.uk/books?id=5z-AI0pbNsYC

Kaparias, I., Bell, M., & Belzner, H. (2008). A New Measure of Travel Time Reliability for In-Vehicle Navigation Systems. *Journal of Intelligent Transportation Systems*, *12*(4), 202–211. http://doi.org/10.1080/15472450802448237

Kells, L. M., Kern, W. F., & Bland, J. R. (1940). *Plane and spherical trigonometry* (2nd ed.). McGraw Hill Book Company, Inc.

Kenyon, A. S., & Morton, D. P. (2003). Stochastic Vehicle Routing with Random Travel Times. *Transportation Science*, *37*(1), 69–82. http://doi.org/10.1287/trsc.37.1.69.12820

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671–680. http://doi.org/10.1126/science.220.4598.671

Kulkarni, V. G. (1986). Shortest paths in networks with exponentially distributed arc lengths. *Networks*, *16*(3), 255–274. http://doi.org/10.1002/net.3230160303

Lambert, V., Laporte, G., Louveaux, F., & Sebag, A. (1993). Designing collection routes through bank branches. *Computers & Operations Research*, *20*(7), 783–791. http://doi.org/10.1145/2745234.2746797

Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, *59*(2), 231–247. http://doi.org/10.1016/0377-2217(92)90138-Y

Laporte, G., Louveaux, F., & Mercure, H. (1989). Models and exact solutions for a class of stochastic location-routing problems. *European Journal of Operational*

Research, *39*(1), 71–78. http://doi.org/10.1016/0377-2217(89)90354-8

Laporte, G., Louveaux, F., & Mercure, H. (1992). The Vehicle Routing Problem with Stochastic Travel Times. *Transportation Science*, *26*(3), 161–170. http://doi.org/10.1287/trsc.26.3.161

Laporte, G., Louveaux, F., & Mercure, H. (1992). The Vehicle Routing Problem with Stochastic Travel Times. *Transportation Science*, *26*(3), 161–170. http://doi.org/10.1287/trsc.26.3.161

Laporte, G., & Louveaux, F. V. (1993). The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, *13*, 133–142.

Laporte, G., Louveaux, F. V., & Mercure, H. (1994). A Priori Optimization of the Probabilistic Traveling Salesman Problem. *Operations Research*, *42*(3), 543–549.

Laporte, G., & Semet, F. (2002). Classical Heuristics for the Capacitated VRP. In P. Toth & D. Vigo (Eds.), *The Vehicle Routing Problem* (pp. 109–128). Philadelphia: Society for Industrial and Applied Mathematics.

Larson, R., & Edwards, B. H. (2016). *Multivariable Calculus* (11th ed.). Boston: Cengage Learning. Retrieved from https://books.google.co.uk/books?id=uPC5DQAAQBAJ

Lecluyse, C., van Woensel, T., & Peremans, H. (2009). Vehicle routing with stochastic time-dependent travel times. *4OR*, *7*(4), 363–377. http://doi.org/10.1007/s10288-009-0097-9

Lee, C., Lee, K., & Park, S. (2012). Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *Journal of the Operational Research Society*, *63*(9), 1294–1306. http://doi.org/10.1057/jors.2011.136

Lei, H., Laporte, G., & Guo, B. (2011). The capacitated vehicle routing problem with stochastic demands and time windows. *Computers and Operations Research*, *38*(12), 1775–1783. http://doi.org/10.1016/j.cor.2011.02.007

Lemons, D. S., Langevin, P., & Gythiel, A. (2002). *An Introduction to Stochastic Processes in Physics*. Baltimore: Johns Hopkins University Press. Retrieved from https://books.google.co.uk/books?id=Uw6YDkd_CXcC

Lesaint, D., Voudouris, C., & Azarmi, N. (2000). Dynamic workforce scheduling for British Telecommunications plc. *Interfaces*, *30*(1), 45–56. Retrieved from http://dx.doi.org/10.1287/inte.30.1.45.11615

Levy, D. (2010). *Introduction to numerical analysis*. Department of Mathematics and Center for Scientific Computation and Mathematical Modeling, University of Maryland.

Li, X., & Tian, P. (2006). An ant colony system for the open vehicle routing problem, 356–363. http://doi.org/10.1007/11839088_33

Li, X., Tian, P., & Leung, S. C. H. (2010). Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm. *International Journal of Production Economics*, *125*(1), 137–145. http://doi.org/10.1016/j.ijpe.2010.01.013

Li, X. Y., Tian, P., & Leung, S. C. H. (2009). An ant colony optimization metaheuristic hybridized with tabu search for open vehicle routing problems. *Journal of the Operational Research Society*, *60*(7), 1012–1025. http://doi.org/10.1057/palgrave.jors.2602644

Liu, B., Siu, Y. L., Mitchell, G., & Xu, W. (2013). Exceedance probability of multiple natural hazards: Risk assessment in China's Yangtze River Delta. *Natural Hazards*, *69*(3), 2039–2055. http://doi.org/10.1007/s11069-013-0794-8

Liu, W., Wang, Q., Mao, Q., Wang, S., & Zhu, D. (2015). A scheduling model of logistics service supply chain based on the mass customization service and uncertainty of FLSP's operation time. *Transportation Research Part E: Logistics and Transportation Review*, *83*, 189–215. http://doi.org/10.1016/j.tre.2015.09.003

Louveaux, F., & Laporte, G. (1990). Formulation and bounds for the stochastic capacitated vehicle routing problem with uncertain supplies. In J. Gabszewicz, J. F. Richard, & L. A. Wolsey (Eds.), *Economic Decision-Making : games, econometrics and optimization* (pp. 443–455). North-Holland.

Ma, R., Yao, L., Jin, M., Ren, P., & Lv, Z. (2016). Robust environmental closed-loop supply chain design under uncertainty. *Chaos, Solitons and Fractals*, *89*, 195–202. http://doi.org/10.1016/j.chaos.2015.10.028

Mathai, A. M. (1982). Storage capacity of a dam with gamma type inputs. *Annals of the Institute of Statistical Mathematics*, *34*(3), 591–597. http://doi.org/10.1007/BF02481056

Mathai, A. M., & Saxena, R. K. (1978). The H-Function with applications in statistics and other disciplines.

Mazmanyan, L., & Trietsch, D. (2014). Stochastic travelling salesperson and shortest route models with safety time. *International Journal of Planning and Scheduling*, *2*(1), 53–76. http://doi.org/10.1504/IJPS.2014.066707

Moritz, H. (1980). Geodetic reference system 1980. *Bulletin Géodésique*, *58*(3), 388–398. http://doi.org/10.1007/BF02519014

Moschopoulos, P. G. (1985). The distribution of the sum of independent gamma random variables. *Annals of the Institute of Statistical Mathematics*, *37*(3), 541–544. http://doi.org/10.1007/BF02481123

Mualla, Y., Bai, W., Galland, S., & Nicolle, C. (2018). Comparison of Agent-based Simulation Frameworks for Unmanned Aerial Transportation Applications. *Procedia Computer Science*, *130*, 791–796. http://doi.org/10.1016/j.procs.2018.04.137

Nadarajah, S., & Kotz, S. (2008). Exact distribution of the max/min of two Gaussian random variables. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *16*(2), 210–212. http://doi.org/10.1109/TVLSI.2007.912191

Ong, H. L., Ang, B. W., Goh, T. N., & Deng, C. C. (1997). A vehicle routing and scheduling problem with time windows and stochastic demand constraints. *Asia-Pacific Journal of Operational Research*, *14*.

Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, *41*, 421–451. Retrieved from http://hdl.handle.net/10044/1/7596

Pantouvakis, J.-P., & Maravas, A. (2013). Guidelines for modelling time and cost uncertainty in project and programme management. *Procedia - Social and Behavioral Sciences*, *74*, 203–211. http://doi.org/10.1016/j.sbspro.2013.03.045

Papoulis, A., & Pillai, S. U. (2002). *Probability, Random Variables, and Stochastic Processes* (4th ed.). New York: McGraw-Hill Higher Education.

Potvin, J.-Y., Dubé, D., & Robillard, C. (1996). A hybrid approach to vehicle routing using neural networks and genetic algorithms. *Applied Intelligence*, *6*(3), 241–252. http://doi.org/10.1007/BF00126629

Pu, W. (2011). Analytic relationships between travel time reliability measures. *Transportation Research Record*, *2254*(1), 122–130. http://doi.org/10.3141/2254-13

Rahman, M. M., Wirasinghe, S. C., & Kattan, L. (2018). Analysis of bus travel time distributions for varying horizons and real-time applications. *Transportation Research Part C: Emerging Technologies*, *86*(July 2016), 453–466. http://doi.org/10.1016/j.trc.2017.11.023

Rakha, H. A., EL-Shawarby, I., Arafeh, M., & Dion, F. (2006). Estimating path travel-time reliability. *2006 IEEE Intelligent Transportation Systems Conference*, 236–241. http://doi.org/10.1109/ITSC.2006.1706748

Richardson, A. J., & Taylor, M. A. P. (1978). Travel time variability on commuter journeys. *High Speed Ground Transportation Journal*, *6*, 77–79.

Robinson, S. (2002). Modes of simulation practice: Approaches to business and military simulation. *Simulation Modelling Practice and Theory*, *10*(8), 513–523. http://doi.org/10.1016/S1569-190X(02)00117-X

Rochat, Y., & Taillard, É. D. (1995). Probabilistic diversification and intensification in local search for vehicle routing. *Journal of Heuristics*, *1*(October), 147–167. http://doi.org/10.1007/BF02430370

Russell, R. A., & Urban, T. L. (2008). Vehicle routing with soft time windows and Erlang travel times. *Journal of the Operational Research Society*, *59*(9), 1220–1228. http://doi.org/10.1057/palgrave.jors.2602465

Savelsbergh, M. W. P., & Sol, M. (1995). The general pickup and delivery problem.

Transportation Science, *29*(1), 17–29. http://doi.org/10.1287/trsc.1030.0071

Shi, Z., & Vickers, C. E. (2016). Molecular Cloning Designer Simulator (MCDS): All-in-one molecular cloning and genetic engineering design, simulation and management software for complex synthetic biology and metabolic engineering projects. *Metabolic Engineering Communications*, *3*, 173–186. http://doi.org/10.1016/j.meteno.2016.05.003

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, *35*(2), 254–265. http://doi.org/10.1287/opre.35.2.254

Stewart, T., Strijbosch, L. W. G., Moors, J. J. A., & van Batenburg, P. (2006). A simple approximation to the convolution of Gamma distributions. *Ssrn*. http://doi.org/10.2139/ssrn.900109

Stewart, W. R., & Golden, B. L. (1983). Stochastic vehicle routing: A comprehensive approach. *European Journal of Operational Research*, *14*(4), 371–385. http://doi.org/10.1016/0377-2217(83)90237-0

Stewart, W. R., & Golden, B. L. (1984). A Lagrangean relaxation heuristic for vehicle routing. *European Journal of Operational Research*, *15*(1), 84–88. http://doi.org/10.1016/0377-2217(84)90050-X

Taillard, É. (1993). Parallel iterative search methods for vehicle routing problems. *Networks*, *23*, 661–673.

Taniguchi, E., Thompson, R. G., Yamada, T., Duin, R. Van, & Ronchail, G. (2001). *City Logistics*. http://doi.org/10.1108/9780585473840

Taş, D. (2013). *Time and Reliability in Vehicle Routing Problems*. Technische Universiteit Eindhoven. http://doi.org/10.6100/IR757863

Taş, D., Dellaert, N., Van Woensel, T., & De Kok, T. (2013). Vehicle routing problem with stochastic travel times including soft time windows and service costs. *Computers and Operations Research*, *40*(1), 214–224. http://doi.org/10.1016/j.cor.2012.06.008

Taş, D., Gendreau, M., Dellaert, N., Van Woensel, T., & De Kok, A. G. (2014). Vehicle routing with soft time windows and stochastic travel times: A column generation and branch-and-price solution approach. *European Journal of Operational Research*, *236*(3), 789–799. http://doi.org/10.1016/j.ejor.2013.05.024

Tillman, F. (1969). The multiple terminal delivery problem with probabilistic demands. *Trans*, *3*(3), 192–204. http://doi.org/10.1287/trsc.4.2.232

Toth, P., & Vigo, D. (2003). The granular Tabu search and its application to the vehicle-routing problem. *INFORMS Journal on Computing*, *15*(4), 333–346. http://doi.org/10.1287/ijoc.15.4.333.24890

Trudeau, P., & Dror, M. (1992). Stochastic inventory routing: Route design with stockouts and route failures. *Transportation Science*, *26*(3), 171–184.

Uno, N., Kurauchi, F., Tamura, H., & Iida, Y. (2009). Using bus probe data for Aanalysis of travel time variability. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *13*(1), 2–15. http://doi.org/10.1080/15472450802644439

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers and Operations Research*, *40*(1), 475–489. http://doi.org/10.1016/j.cor.2012.07.018

Wackerly, D., Mendenhall, W., & Scheaffer, R. L. (2014). *Mathematical Statistics with Applications* (7th ed.). Belmont: Cengage Learning. Retrieved from https://books.google.co.uk/books?id=lTgGAAAAQBAJ

Wang, B., Wang, X., Lan, F., & Pan, Q. (2018). A hybrid local-search algorithm for robust job-shop scheduling under scenarios. *Applied Soft Computing Journal*, *62*, 259–271. http://doi.org/10.1016/j.asoc.2017.10.020

Waters, C. D. J. (1989). Vehicle-scheduling problems with uncertainty and omitted customers. *The Journal of the Operational Research Society*, *40*(12), 1099–1108.

Wellman, M. P., Ford, M., & Larson, K. (1995). Path Planning Under Time-dependent Uncertainty. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, (August 1995), 532–539.

Yan, S., Wang, S. S., & Chang, Y. H. (2014). Cash transportation vehicle routing and scheduling under stochastic travel times. *Engineering Optimization*, *46*(3), 289–307. http://doi.org/10.1080/0305215X.2013.768240

Zäpfel, G., & Bögl, M. (2008). Multi-period vehicle routing and crew scheduling with outsourcing options. *International Journal of Production Economics*, *113*(2), 980–996. http://doi.org/10.1016/j.ijpe.2007.11.011

Zhang, J., & Elmaghraby, S. E. (2014). The relevance of the “alphorn of uncertainty” to the financial management of projects under uncertainty. *European Journal of Operational Research*, *238*(1), 65–76. http://doi.org/10.1016/j.ejor.2014.03.048