LOGISTIC SYSTEM DESIGN OF AN

UNDERGROUND FREIGHT PIPELINE SYSTEM

A Dissertation

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

GAOHAO LUO

Dr. James Noble, Dissertation Supervisor

JULY 2014

The undersigned, appointed by the dean of the Graduate School, have examined the dissertation entitled

LOGISTIC SYSYTEM DESIGN OF

AN UNDERGROUND FREIGHT PIPELINE SYSTEM

presented by Gaohao Luo,

a candidate for the degree of doctor of philosophy,

and hereby certify that, in their opinion, it is worthy of acceptance.

____

Professor James Noble

____

Professor Luis Occeña

____

Professor Mustafa Sir

____

Professor Praveen Edara

Acknowledgement

I would like to express the deepest appreciation to my committee chair Dr. James Noble, for giving me the opportunity to work at the Center for Excellence in Logistics and Distribution (CELDi) which not only provides me the topic of my dissertation but also valuable real world project experience that benefits me in the long term. Without Dr.Noble's abundant encouragement, invaluable support, excellent guidance and persistent help this dissertation would not have been possible.

I owe my gratitude to Dr.Henry Liu who sponsors the project on which this dissertation is based. His suggestions and comments are invaluable and he will forever be remembered.

I am particularly grateful to Dr.Mustafa Sir, for his insightful comments, suggestions. I also would like to express my gratitude to Dr.Luis Occena and Dr. Praveen Edara, for their meticulous comments and suggestions.

I would like to thank my parents for their love and support over the years.

Special thanks go to my beloved wife Lei for always standing by me. It is hard to imagine having made it to this point without her.

Last but not least, I want to thank my son Bryan, for being cute.

Table of Contents

List of Figures and Tables

Academic Abstract

Underground Freight Pipeline (UFP) systems utilize the underground space in metro areas that is otherwise not utilized for freight transportation. Two fundamental logistics issues in the design of a UFP system are network configuration and capsule control. This research develops two capsule control models that minimize total tardiness squared of cargo delivery and associated heuristic algorithms to solve large-scale problems. Two network design models are introduced that minimizes both operational and construction cost of UFP system. The UFP network design Comprehensive Model can only be solved to optimality for small sized problem. To reduce the computational complexity, the UFP network design Two Step Model that is able to generate high quality network design solutions is developed. Then, a case study of a UFP network design in Greater New York area is presented.

Chapter 1 Introduction

Many large metro areas in the world such as New York, Tokyo, Sydney, Beijing, Shanghai, and Seoul are becoming increasing congested due to the amount of freight that flows in and out each day. One means of transportation that has not yet been exploited for freight transportation is the use of underground pipelines. Use of underground capsule pipelines for freight transport has a number of advantages including the following:

1. It utilizes a space in metro areas that is otherwise unutilized.

2. It reduces overcrowding of the city above ground. This helps to reduce traffic jams, accidents, and air pollution generated by vehicles running above ground.

3. Underground transportation is not affected by inclement weather such as snow, ice, hail, rain and floods, provided that the underground system has a good drainage system.

4. Underground transportation is less susceptible to terrorist attacks.

1.1 Historical development

The technology of a pneumatic capsule pipeline (PCP) is a current version of the technology of *"tube transport"* or *"pneumatic tubes"* that began in the late 1960s. (Zandi, 1976; ASCE, 1998) These PCP systems operated with multiple capsules in a continuous stream. Independently, Dr. M. Robert Carstens in the United States and Dr. A.M. Alexandrov in the USSR started the development of PCP technology. The work of both Alexandrov and Carstens was aimed at transporting bulk granular cargo, such as ore or coal, over limited distances (Tubexpress 2011)

Two types of PCP systems were developed in Japan, as shown in Figure 1.1, one used round PCPs of approximately 1 meter diameter, and the other used rectangular PCPs of 1m × 1m cross-section, with each capsule carrying 1 to 2 tons of cargo. These two systems were used successfully for mining, for transporting raw materials to modern steel and cement plants, for construction of tunnels and highways, and for solid waste disposal. Discussion of the implementation of PCP systems in Japan is given in Liu (2003).

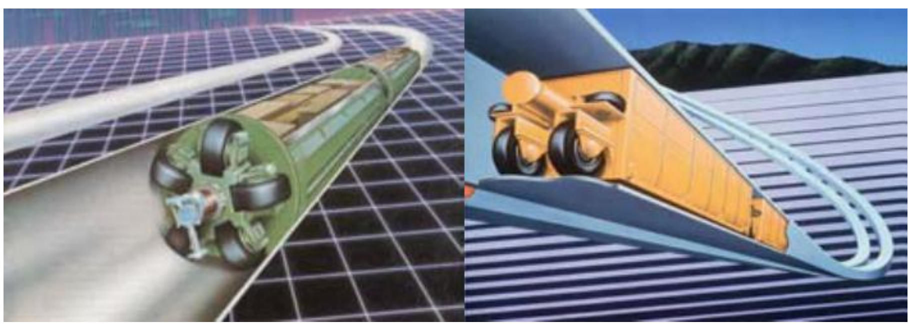

Figure 1.1 Round and Rectangular PCP systems in Japan (Liu 2003)

However, the use of PCP has been limited due to the fact all the current PCP systems are driven by blowers (fans), which block the passage of capsules through the tube. This requires that a complicated switching mechanism be implemented so that capsules can bypass the blowers, of which the impact is that flow is not continuous and throughput is reduced. A greater problem caused by blower technology is that it makes it impractical to use PCP for both long distances and in complex tube systems where the flow must be continuous (i.e. where capsules must be able to enter and leave tubes at intermediate stations, and where the PCP have multiple inlets/outlets or branches). For these reasons, all pneumatic capsule pipeline systems built to date have not achieved commercial success.

1.2 Major technical breakthrough

A major technical breakthrough was made by William Vandersteel of Alpine, New Jersey, President of Tube**X**press Systems, Inc. and its parent, Ampower Corporation, they invented and patented (U.S. Patent No. 4,458,602, 1984) the embodiment of an entirely new concept for motivating a capsule pipeline system: instead of pumping air to propel the capsules, as had been practiced by every system built up to that time, Vandersteel proposed to impart thrust to the capsules directly, in such a way, capsules act like pistons in a long cylinder. Though various means of inducing thrust to the capsules could be considered, the use of linear induction and/or linear synchronous propulsion was proposed in the patent.

Linear Induction Motors (LIMs) in PCPs are used as electromagnetic capsule pumps (Liu 2003 and Lenau 2009). They are not required for the entire length of the PCP tube, instead, they are mounted in strategic locations along the PCP (such as at the inlet and branching points). In a long distance PCP system, each LIM stator is a slab-like structure of 10 to 50 meters long which contains conductors (copper wires) connected to the power grid, or a special generator, as shown in Figure 1.2. The spacing between LIMs in a long

PCP is usually greater than 10 km. The situation is analogous to using booster pumps along a typical long-distance pipeline that contains oil or natural gas.

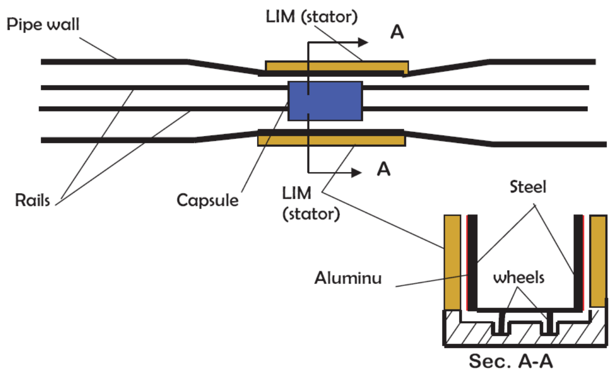

Figure 1.2 LIM capsule pump in PCPs with a square cross-section, Liu and Lenau (2009)

As the capsules are propelled directly by LIM, the new system avoids the restriction imposed by the airlocks and valves with the necessity of stopping the capsules in the airlocks, allowing the system to operate continuously without interruptions or distance limitation. With this fundamental advance, a PCP system equipped with LIM brings significant increase in throughput capacity of the capsule pipeline system. For the first time, it is now practical to consider capsule pipelines for the automated transportation of general commodity freight, in direct competition with surface transport. Therefore, the PCP system studied in this research for freight transport utilizes LIMs.

1.3 System components of LIM-PCP based underground freight transportation

1.3.1 Capsule and Guide Rails

Each capsule used for transporting freight is a 4-wheeled boxcar running on a pair of small railroads inside a rectangular conduit (tube). Liu (2009) proposed that the capsule has a width of 1.475 m, and a height (from boxcar top to the bottom of wheels) of 1.719 m, as shown in Figure 1.3. The nominal track gauge is 1.146 m. The overall length of the capsule is 7.135 m. The wheel base is 6.467 m. The capsule is able to carry four pallets of 1.22 m width, 1.22 m length, and 1.39 m maximum height. Each capsule has an empty weight of 3.35 tons (metric tons), and can carry a maximum payload of 3.18 tons.

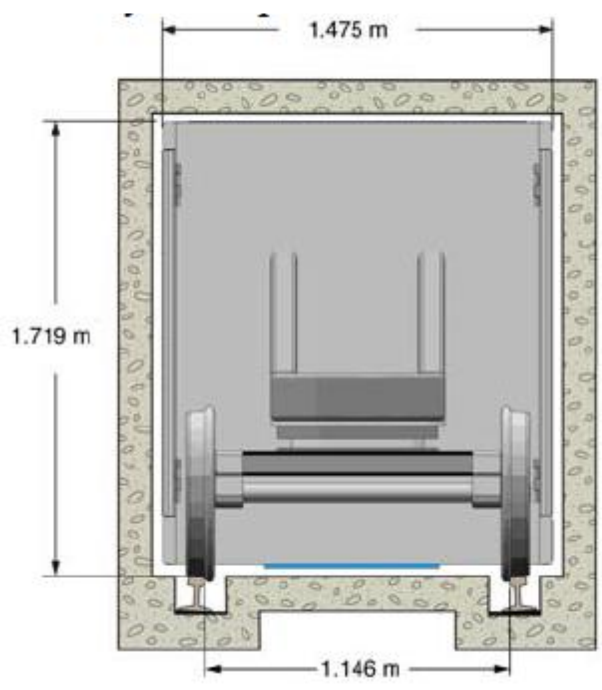

Figure 1.3 Cross-sectional view and key dimensions of capsules, Liu and Lenau (2009)

1.3.2 System general layout and operations

The underground freight transportation system is a horizontal network consisting of stations and tube connecting them, as shown in Figure 1.4. The network consists of a large number of node points (underground substations) and a few inlet/outlet points (main stations located aboveground).

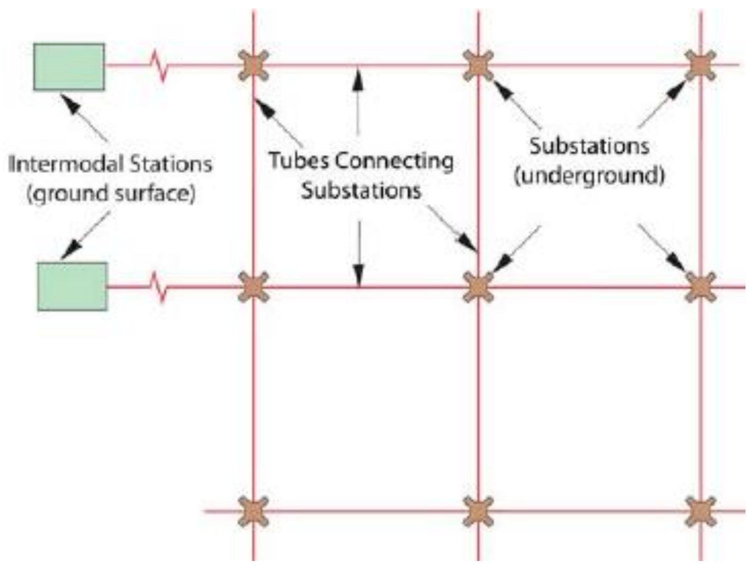

Figure 1.4 Underground substations and aboveground main stations, Liu and Lenau, (2009)

The main stations (aboveground) can be one of two kinds: (1) intermodal transfer stations where the freight is transferred from trucks, trains or waterways to the underground network or vice versa; or (2) terminal stations such as a solid waste disposal or processing plant/unit.

Each substation, as shown in Figure 1.5, is a place where incoming capsules can be stopped and unloaded of their cargo. As soon as a capsule arrives at the target substation, the cargo is unloaded from the capsule by forklifts, as shown in Figure 1.6, and transferred to a small truck for door-to-door delivery of the cargo to neighborhood stores up in the city street level. Upon unloading the cargo from a capsule at a station, the empty capsule may be either loaded with different cargo and then leave the station immediately, or may leave empty if another station needs empty capsules, or may be stored temporarily until needed later.

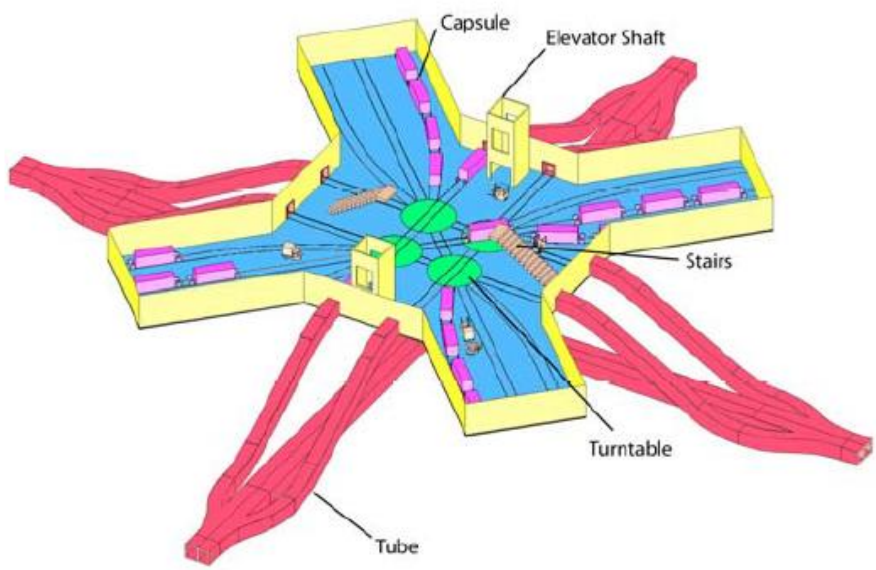

Figure 1.5 Underground station configuration, Liu and Lenau, (2009)

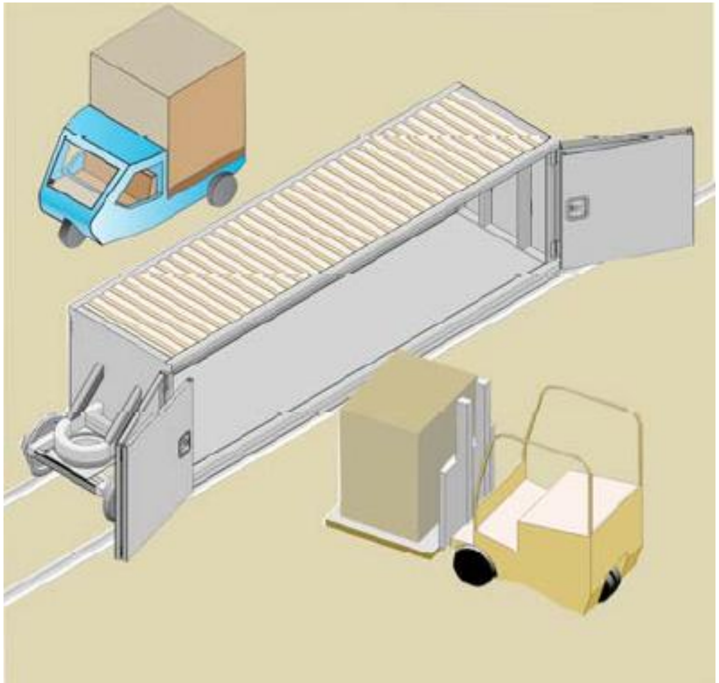

Figure 1.6 Capsule unloading operations, Liu and Lenau, (2009)

1.4 Recent development of UFP system

Weber and Van Zuylen (2000) examine the development of a number of innovations within the transportation industry, both passenger and freight. They evaluate underground freight logistics as being in the phase of between the test and first practical application. During the most recent decade, some developments have evolved further:

CargoCap (Germany): Beckmann (2007) introduces a system resembling a small freight capsule that moves through underground pipes. Each capsule is loaded with two pallets – a standardized form of freight transportation. This system was conceived to free up German roads that were reaching their limits due to bottlenecks and delays. Currently, a half scale test track has been developed to test prototype operations.

Interdepartementale Projectgroep Ondergonds Transport (IPOT, The Netherlands): With 19 million flowers traded daily and over 80% of these destined for oversea customers, the growing congestions in and around Amsterdam has forced the government, in co-operation with business partners such as Schiphol Airport, to consider underground freight transportation. (Van der Heijden et al., 2002) This underground infrastructure will be used for the transportation of goods between the flower auction, the airport and the rail terminal at Hoofdorp. Successful simulation and prototyping of this concept has been developed and the potential of this underground freight transport has been the subject of research by an interdisciplinary research group.

PipeNet (Italy): Researchers from the University of Perugia constructed the first prototype in 2006. This system is based on the idea of transporting small volume freight through pipelines at high speeds and low friction. The capsules for this system is designed to fit one single sized euro-pallet and are propelled by linear electric synchronous motors. Typical goods that could be transported by this system include items that can be found in department stores. And it is envisioned as deliveries to individual homes. Cotana et al. (2008) rates this at an advanced concept stage with number of research works and feasibility studies having been successfully carried out.

Magplane MagPipe System (China): Fang et al. (2011) presented an interesting variation of design of the freight pipeline system: a pre-commercialization demonstration of a new pipeline system transporting coal ores has been built in Baotou, China. In order to assure adequate improvement on the rail stiffness for heavier coal-carrying capsules this new pipeline system use two C-channel rails located at two sides of pipe to replace the current I-type hanging rail system.

As previously presented, the current development and research of the underground freight pipeline system are mainly focusing on the technological and system component design. Yet, to the best of our knowledge, little work had been conducted on the design of the larger and more complex underground freight pipeline system as well as the related logistics design. One reason is that all demonstration systems have been based on rather simple network configurations which have only a limited number of freight carrying capsules and loading/unloading stations. Therefore, this has motivated us to initiate this research to address the underground freight pipeline network design problem and the freight carrying capsule scheduling problem.

This dissertation is structured as follows: Chapter 2 includes a literature review of the single machine scheduling problem, due to its similarity to our capsule scheduling problem, the rapid transit system network design problem, which is the research foundation for the underground freight pipeline network design problem. Chapter 3 presents two mathematical models for the underground pipeline system scheduling problem and the associated solution heuristics. Chapter 4 presents the UFP network design model, it includes the comprehensive UFP network design model and the enhanced two-step network design model, a case study is also given to provide more insights. Chapter 5 summarizes the contribution of this research and proposes future research ideas.

Chapter 2 Literature review

This research presents two mathematical models in Chapter 3 based on a freight pipe line network which consists of a single circle of freight pipeline along with several stations. Cargo of each transportation task departs from its origin station, travels in the pipe line and arrives at its destination. If we consider the network as a virtual machine and the transportations tasks as jobs processed by this virtual machine, we find its similarity to the single machine scheduling problem. In this section, we first give a literature review of the single machine scheduling problem.

To the best of our knowledge, there is no literature available for the network design problem of the underground freight pipeline system. Within this chapter, a review of a similar problem for the rapid transit system network design is given.

2.1 Single machine scheduling problem

Two streams of the research are problems involving a quadratic measure of performance for scheduling a single machine and scheduling a single machine with minimizing total weighted tardiness as the objective. Relatively little literatures can be found on problems involving a quadratic measure of performance for scheduling a single machine.

2.1.1 Single machine problem with quadratic measure of performance

2.1.1.1 Objective of minimizing the sum of square of the job completion time

The single machine scheduling problem with the objective of minimizing the sum of squares of the job completion times has been studied by Schild and Fredman (1962), Townsend (1978), Bagga and Kalra (1980), Gupta and Sen (1984), and Szwarc et al. (1988).

Schild and Fredman (1962) developed precedence relationships when the objective is a weighted combination of quadratic and linear completion times. It is presented that in the case that there is one machine with one processor where n tasks need to be scheduled, in the linear case once it has been determined that task (i) is to be scheduled before task (j) for minimal loss, then this order remains the same, i.e., irrespective as to where task (i) appears in the schedule, it will always precede task (j). In the non-linear case, on the other hand, the relative order of tasks (i) and (j) may rely on their absolute position in the schedule. In other words, it is possible that, for instance, that task (i) should precede task (j) if (i) and (j) are to be the first two tasks in the schedule, but task (j) should precede task (i) if some other tasks are scheduled before them. This paper establishes a criterion which reduces the total possible number of n! different schedules of n tasks on one processor. An algorithm was proposed to determine an optimal schedule for any type of loss function. Even though this algorithm still requires a considerable amount of calculations, the number of calculations is reduced to $\mathrm{n}(2^{n-1} - 1)$ for an n-job problem.

Szwarc et al. (1988) developed a precedence relation for ordering adjacent jobs for the problem. For each pair of jobs i and j, where $c_i/p_i \geq c_j/p_j$ (c_i is the cost coefficient and p_i is the processing time of job i), a critical value t_{ij} was defined such that if i and j are adjacent in an optimal ordering, then j precedes i if processing begins before t_{ij} and i precedes j if processing begins after t_{ij}. If $t_{ij} \leq 0$, i always precedes j and if $t_{ij} \leq \sum_k p_k$, j always precedes i. It is reported that these precedence relations are used to solve 191 out of 200 test problems of sizes n = 15, 25, 50 and 100 without any enumeration procedures.

Townsend (1978) developed a criterion to order a pair of adjacent jobs in a schedule sequence with the objective of minimize the sum of quadratic completion times and the criteria was used to develop a branch-and-bound algorithm for the problem. Based on Townsend (1978)'s work, Bagga and Kalra (1980) proposed a method which is able to eliminate the creation of certain nodes thus the computational time can be considerably reduced. Gupta and Sen (1984) stated certain conditions which will give a priori precedence relations among some of jobs in an optimal sequence thus curtailing the enumeration tree at the branching stage.

2.1.1.2 Objective of minimizing the sum of weighted/unweighted squared tardiness

There have been relatively few papers that have considered the objective of minimizing the sum of weighted/unweighted squared tardiness values. To the best of our knowledge, the approach of Lagrangian relaxation is used by all the researchers, in which the machine capacity constraints are relaxed to obtain a lower bound and then a certain heuristic is used to create a feasible solution and obtain an upper bound.

Hoitomt et al. (1990) and Luh and Hoitomt (1993) developed procedures for scheduling jobs on parallel machines. Hoitomt et al. (1990) consider the scheduling of jobs with due dates on identical, parallel machines. Each job consists of a small number of operations that must be undertaken in a predefined order. With the constraints of capacity and operational precedence, the objective is to minimize the total weighted quadratic tardiness of the schedule. A Lagrangian relaxation is then proposed to decompose the original problems into several sub-problems for each job by relaxing the coupling capacity constraint. A near-optimal solution is obtained by solving the sub-problems and the dual problems. In addition, the resulting job-interaction information can provide access to the answers of "what-if" scenarios and to help reconfiguring the schedule to incorporate new jobs and other dynamic changes. The proposed scheduling methodology was implemented for a work center at Pratt & Whitney as part of its knowledge-based scheduling system. Luh and Hoitomt (1993) provide a unified frame work for developing scheduling methodologies for large scale, diverse, and interdependent system. The practical solutions of three manufacturing scheduling problems with increasing complexity are examined. The first problem considers scheduling jobs on identical machines. Each job requires only one single operation on the machine. The second problem deals with the scheduling of multiple-operation jobs on identical machines, where one job may need a few operations to be performed in a specific order. The third problem is so-called "job shop problem", jobs may require complicated networks of operations, each of which may be scheduled on a different type of machine. A

Lagrangian relaxation approach was used to decompose each of the scheduling problems into job- or operation-level sub-problems, which results in algorithms that are able to generate near-optimal schedules efficiently. The proposed procedures were demonstrated using data from a Pratt and Whiney plant, including an example with 112 jobs and 44 machines.

Sun et al. (1999) consider the single machine problem with release dates and sequence dependent setup times. They compared their Lagrangian relaxation based heuristic against some simple dispatching rules, a tabu search and simulated annealing algorithms. These heuristics were tested using a variety of data sets most of which consisted of 40 jobs and ranged between 10 and 80 jobs.

Su and Chang (1998) developed procedures for minimizing the sum of squares job lateness on a single machine. Note that: the lateness of a job = Completion time – Due date, which could be negative (earliness). An $O(N^3)$ heuristic algorithm was proposed. It is stated that the computational results indicate that the heuristic not only is efficient but also consistently provides very satisfactory results.

Schaller (2002) proposed a procedure that considered the use of inserted idle time to optimally solve the single machine scheduling problem where the objective is to minimize the sum of squares lateness. It is stated that this procedure reduces the cost of the early completion of a job by the insertion of idle time, and the inclusion of inserted idle time can reduce the sum of squares lateness for a given sequence of jobs and can cause a sequence that is optimal if inserted idle time is not allowed to become suboptimal if inserted idle time is considered.

2.1.2 Scheduling a single machine with minimizing total weighted tardiness as the objective

The exact methods for the single machine with the objective of minimizing total weighted tardiness include dynamic programming method developed by Lawler (1979) and Schrage and Baker (1978) and branch-and-bound algorithms developed by Gelders and Kleindorfer [9,10] and Potts and van Wassenhove [18].

Lawler (1979) pointed out that technical difficulties had been experienced when researchers were trying to reduce the computational complexity which results from the existence of precedence constraints in single machine scheduling problems, thus, a simple computer implementation of the dynamic programming algorithms which overcomes the computational difficulties was presented. This implementation permits sequencing problems with precedence constraints to be solved in O(Kn) time, where K is the number of feasible sets, and in $O(n+k_{max})$ space, where k_m is the number of feasible sets of size m and $k_{max} = max_m\{k_m\}$.

Consider a set of jobs that are partially ordered by precedence constraints. A subset of tasks is called feasible if, for every job in the subset, all predecessors are also in the subset. Schrage and Baker (1978) proposed a method for enumerating all feasible subsets and a method to assign each feasible subset an easily computed label that can be used as a physical address for storing information about the subset. These two methods results in a very lean computer implementation of a dynamic programming algorithm for solving the single machine scheduling problems with precedence constraints. It is also stated that this algorithm were much more efficient than the existing works.

Gelders and Kleindorfer (1974) presented a formal model of the scheduling problem with the objective of minimizing the sum of weighted tardiness and weighted flow-time costs for a given capacity plan. This research generalized sequence theory results to this specific case, analyzes various lower-bounding structures for the problem, and also outlines a preliminary branch-and-bound algorithm. Based on their previous works, Gelders and Kleindorfer (1975) reported some computational experience with the previously proposed algorithm. In order to deal with non-simultaneous job arrivals and the production smoothing problem, extensions and refinements of the algorithm were introduced.

Potts and van Wassenhove (1985) proposed another branch and bound algorithm for the single machine total weighted tardiness problem. It obtains lower bounds by using a Lagrangian relaxation approach to decompose the original problem into sub-problems that the total weighted completion time is the objective. Instead of the sub-gradient optimization technique, the multiplier adjustment method was used, which leads to a fast bound calculation. It is reported that the proposed algorithm performs better over existing methods.

Dominance tests for the problem have also been found to be very effective in reducing the search space for exact methods. Rinnooy Kan et al. (1975) extend the dominance test developed by Emmons (1969) for the unweighted tardiness problem to the weighted tardiness objective. Rachamadugu (1987) developed a dominance test for adjacent jobs and Kanet (2007) developed seven dominance tests for deciding precedence for pairs of jobs.

Among the literatures of single machine scheduling problem, the objective functions are either based on a quadratic measure of performance, i.e. the sum of squared tardiness, or based on a linear sum of performance measure, i.e. the total weighted tardiness. Both objective settings tend to generate a job schedule that increases the utilization and efficiency of the machine. Considering the underground freight pipeline system that this research is based on, the goal of the capsule scheduling problem is to find a transportation schedule that maximizes the overall customer satisfaction. Intuitively, customers are more satisfied if the tardiness level of the transportation tasks is minimized. Yet, the quadratic measure of tardiness is preferred to the linear counterpart. Consider the following example: assume 3 tasks, in case 1, their tardiness is 0, 0, and 9, respectively; in case 2, they have the tardiness of 3, 3, 3. In both cases the 3 tasks have the same total tardiness of 9. However, in case 1 the square of tardiness is 81, whereas, case 2 has a square of total tardiness of 27. Obviously, case 2 is more desirable and reflects our goal of improving the overall customer satisfaction.

2.2 Rapid transit system network design problem

As a response to environmental concerns as well as its superiority in terms of the transportation quantity and quality, many cities have constructed new rapid transit system or have expanded or upgraded old ones. One critical issue when planning rapid transit systems is how to determine the configuration of alignments/lines and stations. Gendreau et al. (1995) examined some 40 rapid transit projects and indicates that the process of planning rapid transit system, as a highly complicated process involving multiple objectives and constraints, uncertainties, large capital expenditures and long term commitments, adapted little or no operations research. Vuchic (2005) notes that most academically oriented network design tools are often inadequate for solving such complex problems. However, it is also pointed out in his research that the use of operations research methods can play a useful role at a technical level when planning rapid transit system.

The remainder of this sub-section is organized as follows: first, we address the assessment of rapid transit networks. Then we examine some modeling issues for the design of rapid transit networks. Finally, we present the heuristics and algorithms for the location of alignments/lines.

2.2.1 Assessment of rapid transit networks

One main objective of a rapid transit network is to improve population's mobility. A good network design should be able to provide short travel time and relatively direct travel service for the major population. Thus, the population coverage is an important assessment index and the alignments/lines along with the stations need to cover the main transportation corridors. Vuchic (2005) notes that most potential users of rapid transit system live within 5 minutes' walk of a station, and if the walking distance increases to 10 minutes the ridership falls to nearly zero.

Musso and Vuchic (1988) initiated the research work of assessing the quality of a potential or existing network based on the topology of a network $G = (N, E)$, where N is the node set and E is the edge set. The assessment indices in this research includes the number of stations, the total length of the network, the number of lines and the number of multiple stations and five more sophisticated measures are:

C: the number of minimal cycles (not embedding any other cycles) $\mathrm{C} = |E| - |N| + 1$

α: a cycle availability index, defined as the ratio between C and the largest value it could take for a network with $|N|$ nodes and $|E|$ edges. $\alpha = (|E| - |N| + 1)/(2|N| - 5)$

β: a measure of the network complexity. $\beta = |E|/|N|$

γ: a connectivity indicator equal to the ratio of $|E|$ to the maximal number of edges that could exist in a planar network with $|N|$ nodes. $\gamma = |E|/3(|N| - 2)$

δ: a measure of directness of service equal to the number of origin/destination (O/D) paths that can be traveled without transfer.

Besides these measurement indices, Laporte et al. (1994) introduced two other measures: the passenger/network effectiveness index of a network and the passenger/plane effectiveness index.

The passenger/network effectiveness is defined as follows. For each path P between v_i and v_j, define the total passenger cost as:

$$\theta_{ij}(P) = \sum_{v_i,v_j \in P} t_{ij} + r(P)t_f + (s(P) - r(P) - 1)t_s$$

Where t_{ij} the travel is time between v_i and v_j, $r(P)$ is the number of transfers, $s(P)$ is the edge on the path, and t_f is the transfer time and t_s is the stopping time. Then the total passenger cost is $\theta = \sum_{v_i,v_j} \theta_{ij}$ where $\theta_{ij} = \min\{\theta_{ij}(P)\}$. The passenger/network effectiveness index is then:

$$\lambda = \theta / \sum_{v_i,v_j \in E} t_{ij}$$

The second measure of the passenger/plan effectiveness compares passenger travel time on the network to the travel time if travel was made on the street network. It is concluded in this research that among the three network configuration shown in Figure 2.1 the worst network topology is the star while the triangle and the cartwheel are rather effective configurations. Note that these measurements are based only on the network topology without considering passenger volumes and modal competition. Then, Laporte et al. (1997) confirmed the same conclusions in a follow-up study when both traffic volumes and modal competition are considered.

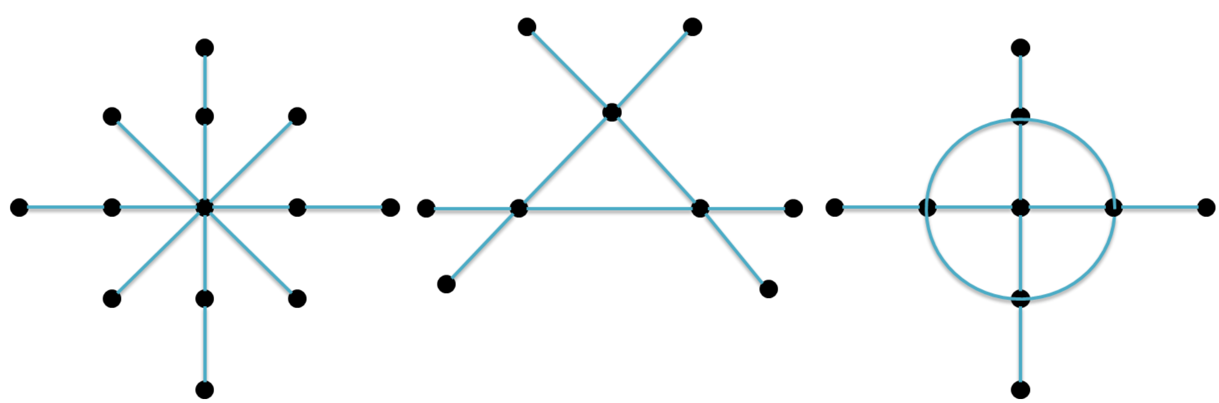

Figure 2.1 Rapid transit system network configurations

2.2.2 Modeling issues of the design of rapid transit system networks

The objective of a rapid transit system is to transport a large number of people efficiently and effectively. The problem of generating a good transit network can be viewed as a network design problem (Ahuja et al. 1995). Three main objective of designing rapid transit system network are to minimize the total construction cost, to maximize the population covered by the line, or to maximize the total O/D traffic captured by the network.

Network design problem with objectives of minimizing construction cost or maximizing population coverage is categorized to the class of Steiner tree problems with profits (STPP). (Costa et al. 2006) Some classical STPP examples include the Prize-collecting STPP, Quota STPP, Fractional STPP etc. In the Prize-collecting STPP, the objective function is a linear combination of the construction cost and the population covered by the network, which is subjected to the constraints of forcing the network to be a tree. In the Quota STPP, the objective function only considers the minimization of the construction cost, while an additional constraint is included to ensure that a minimal population is covered by the network. The objective of the Fractional STPP is to maximize a population to cost ratio in the form of *population coverage/construction cost*. In such a way, the population coverage is maximized and the construction cost is minimized.

STPP models can be used to locate a network, however, these models do not decompose the network into a set of distinct transit lines. By combining the Prize-collecting STPP and a simplified version of model proposed in Laporte et al. (2010), Laporte et al. (2011a) introduce a new model to locate a set L of lines covering a part of G with the objective to minimize a linear combination of the construction cost and the population covered by the transit network.

Laporte et al. (2010) applied the game theory concept to the problem of designing an uncapacited railway transit network in the presence of link failures and a competing mode. The problem was posed as a non-cooperative two-player zero-sum game with perfect information. The saddle points of the associated mixed enlarged game yield robust network designs. Laporte et al. (2011b) proposed a model for the design of a robust rapid transit network which minimizes the effect of disruption on total trip coverage. The network obtained by solving the model provides several alternative routes for some O/D pairs in case of an interruption.

2.2.3 Heuristics and algorithms for the location of alignments and stations

Mathematical models mentioned in the previous subsection are generally hard to solve except for small size problem. Therefore, there are a number of heuristics available in the related literatures for locating a single alignment and locating several alignments.

2.2.3.1 Locating a single alignment

Dufourd et al. (1996) applied tabu search to locate a single alignment. The method uses a random walk in the plane as the initial solution. Iteratively, the algorithm moves one or several stations to neighboring stations while keep its feasibility. Bruno et al. (2002) proposed a heuristic for the problem of constructing an alignment to maximize population coverage, subject to the constraints of a fixed number of stations and the minimal and maximal inter-station spacing. In the first phase, the heuristic starts with a single edge and then iteratively extends an alignment in a greedy fashion by maximizing the population coverage. In the second phase, tabu search is applied to explore the neighborhood of the current solution. The neighbor of a solution is obtained by cutting an edge of the alignment and reconstructing several partial alignments from the break point. It is presented in this research that this heuristic was successfully applied to the city of Milan and easily produced an alignment covering the main population centers.

2.2.3.2 Locating several alignments

The heuristic proposed by Bruno et al. (2002a) was extended by Bruno et al. (2002b) to the case of locating several intersecting alignments. To initialize the algorithm, the user must select a configuration type from a number of possible options, such as cartwheel, star, U shape, etc. Or the user can use customize command to create other networks that combine circles and lines differently. Then the user specifies a corridor on a city map in which the alignment must be located in. With other user defined parameters of the number of stations of each line and the inter-station spacing, the system is able to produce network configurations and compute a number of effectiveness measures such as population coverage, the passenger/network effectiveness and the passenger/plane effectiveness introduced by Laporte et al. (1994).

Laporte et al. (2005) proposed another heuristic for the construction of a single alignment to maximize trip coverage. This heuristic uses the O/D matrix as an input and is subjected to the constraints of the fixed number of stations to be constructed and the interstation spacing. Several constructive heuristics and an improvement procedure were developed and compared. The test results on the Sevilla data show that when the upper bound for interstation distance is greater than 1250m the best results are provided by a simple greedy extension heuristic, while when upper bounds for interstation distance is smaller an insertion heuristic followed by a post-optimization phase yields the best results.

Michaelis and Schöbel (2009) reordered the classic sequence of the public transportation planning steps and proposed a new heuristic approach. This heuristic starts with the design of the vehicle routes, then splits them to lines and finally calculates a timetable. It is stated that the new heuristic has the advantage that costs can be controlled during the whole process while the objective in all three steps is customer-oriented.

2.3 Literature Review Summary

To summarize, this chapter first reviewed the literature of the single machine scheduling problem, as this class of optimization problem shares some of the basic features of the capsule control scheduling problem we are interested in this research. In both problems, a set of tasks are processed. Each task is assigned to a start time, process time and an end time. All tasks are competing for the limited resource, i.e. the processor of the machine in single machine scheduling problem and the underground pipeline system in the capsule control scheduling problem. However, in spite of the similarities, these two problems differ to a great extent.

Instead of the simple machine processor, the definition of “machine capacity” has a more complex meaning in the UFP system. In general, an underground freight pipeline has two types of capacities, one is known as the line-fill capacity, it defines the largest number of capsules are allowed in the pipeline simultaneously. The other one is the limited number of capsules available. Also, we need to keep in mind is that each transportation task has an origin station and a destination station, which also defines its travel time (process time). Therefore, as the transportations tasks are being scheduled, capsules are constantly moving between stations. In other words, the “machine capacity” is always changing.

To address these challenging issues, we envisioned that it is necessary to develop a customized capsule control scheduling model for the UFP system and the model is presented in Chapter 3.

The second sub-section of this chapter reviewed the rapid transit system network design. There are three main criteria used in mathematical models for rapid transit network design: (1) the total construction cost; (2) the total population covered by the network; and (3) the total traffic captured by the network. The problem can either be modeled as a multi-objective optimization problem or to be modeled with the objective of minimizing cost subjected to a population or traffic coverage constraint or with the objective of maximizing coverage subjected to a budget constraint. However, if we change the scenario to freight transportation instead of passenger movement, some adjustments to the objective function are needed and additional issues need to be considered.

In the setting of designing a network configuration for an underground freight pipeline system, the overall construction cost remains a critical issue, yet the population coverage is no longer in consideration.

In addition to the construction cost, the operational cost is also considered in the UFP system network design. Since the operational cost is calculated based on the total travel distance of freight cargo, the UFP network design model need to make sure that freight travels the shortest path.

Therefore, with the modified objective function and additional modeling issues, we extend the current network design model of the rapid transit system and present the UFP network design model in Chapter 4.

Chapter 3 The Capsule Control Scheduling Problem

This chapter presents two mathematical models of the capsule control/scheduling problem. The objective functions of both models are to minimize the Total Tardiness Square (TTS) of the transportation tasks. The first model does not consider the empty capsule routing while the second model does. As both models have quadratic objective functions, two problem specific heuristic algorithms are introduced to solve large size problems.

3.1 Capsule Control Model (without empty capsule routing)

A mathematical model based on a freight pipe line network is illustrated in Figure 3.1. For ease of illustration, we name this model as Model 1. The network consists of a single circle of freight pipeline along which there are several stations. Each station is considered as a terminal with both inlet and outlet functions. It is assumed that the network is unidirectional and capsules can only travel clockwise in the freight pipeline. The freight pipeline also has a capacity constraint which restricts the number of capsules entering it at any single time.

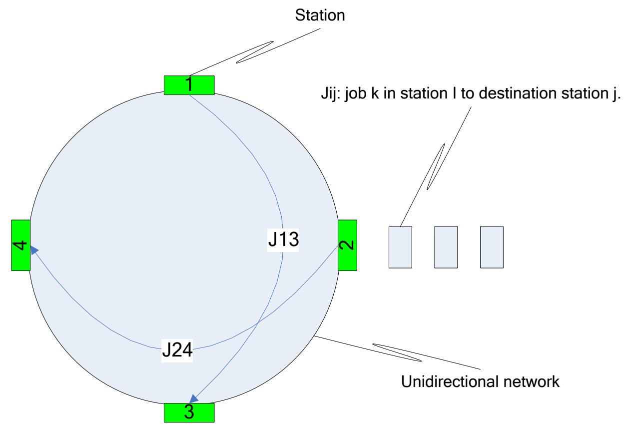

Figure 3.1 A unidirectional freight pipe line network with four stations

During a specified time period at each station there can be several transportation tasks that are released, each with its own release time and destination to which it needs to be shipped. Each transportation task requires at least one, and potentially several capsules, and is assigned a due date. At the beginning of the time period, each station is assigned a specified certain number of capsules and they are used to ship the cargo out of the station.

The models objective is to minimize the total tardiness square (TTS). The square of total tardiness instead of the total tardiness is used as it better captures customer value with respect to tardiness. Consider the following example: assume 3 tasks, in case 1, their tardiness is 0, 0, and 9, respectively; in case 2, they have the tardiness of 3, 3, 3. In both cases the 3 tasks have the same total tardiness of 9. However, in case 1 the square of tardiness is 81, whereas, case 2 has a square of total tardiness of 27. Obviously, case 2 is more desirable and reflects minimizing the square of tardiness.

3.1.1 Model Notation

The following notation and definitions are used to describe the mathematical model of the control problem.

- Sets:

  K: set of terminals

  N: set of tasks

  N_k^{out}: set of outgoing tasks from terminal $k \epsilon K$

  N_k^{in}: set of incoming tasks to terminal $k \epsilon K$

- Parameters:

  r_j: the release time of task j

  d_j: due date of task j

  t_j: travel time of task j

  c_j: required number of capsules for task j

  L: line fill constraint (maximum number of capsules in the freight pipe line at any time)

  C: total number of capsules

- Decision Variables:

n_k: number of capsules at terminal $k \epsilon K$

s_j: start time of task j

x_{ij}: = 1 if task j is in transit when task i starts

0 otherwise

y_{ij}^k: = 1 if task j at terminal k starts earlier than task i at terminal k

0 otherwise

z_{ij}^k: = 1 if task j arrives at terminal k before task i starts at terminal k

0 otherwise

β_{ij}: = 1 if $s_i - s_j \leq 0$

0 otherwise, $\forall\ i, j \ \epsilon\ N$

γ_{ij}: = 1 if $s_i - s_j \geq t_j$

0 otherwise, $\forall\ i, j \ \epsilon\ N$

w_j: Tardiness of task j

3.1.2 Model Formulation

Minimize $\sum_{j \epsilon N} {w_j}^2$ (1)

Subject to

$$s_j \geq r_j, \forall\, j\, \epsilon\, N \quad (2)$$

$$s_i - s_j \geq -M\beta_{ij}\ , \forall\, i, j\, \epsilon\, N \quad (3)$$

$$s_i - s_j \leq -M\left(1 - \beta_{ij}\right) - \varepsilon\ , \forall i, j\, \epsilon\, N \quad (4)$$

$$s_i - s_j \leq t_j + M\gamma_{ij}, \forall\, i, j\, \epsilon\, N \quad (5)$$

$$s_i - s_j \geq t_j - M(1 - \gamma_{ij}),\ \forall i, j\, \epsilon\, N \quad (6)$$

$$x_{ij} + \beta_{ij} + \gamma_{ij} = 1, \forall i, j\, \epsilon\, N \quad (7)$$

$$c_i + \sum_{i \neq j} c_j x_{ij} \leq L, \forall i, j\, \epsilon\, N \quad (8)$$

$$s_i - s_j \leq My_{ij}^k - \varepsilon, \forall k \epsilon K, \forall i, j\, \epsilon\, N_k^{out} \quad (9)$$

$$s_i - s_j - t_j \leq Mz_{ij}^k - \varepsilon, \forall k \epsilon K, \forall i, j\, \epsilon\, N_k^{in} \quad (10)$$

$$n_k - \sum_{i \neq j} c_j y_{ij}^k + \sum_{i \neq j} c_j z_{ij}^k \geq c_i \quad (11)$$

$$s_j + t_j - d_j \leq w_j\ , \forall j\, \epsilon\, N \quad (12)$$

$$\sum_{k \epsilon K} n_k \leq C \quad (13)$$

Constraint (2) ensures that the start time of each task is not earlier than its release time. Constraints (3) to (8) are line fill constraints that ensure that the number of capsules in the freight pipe line does not exceed the maximum capacity. Constraints (9) to (11) are capsule balance constraints that balance the number of capsules leaving and entering a station. Constraint (12) specifies the tardiness of each task. Constraint (13) ensures that the total number of capsules assigned to stations does not exceed the total number of capsules available in the system.

3.1.3 A heuristic algorithm

The mathematical Model 1 presented above has the complexity of $O(KN^2)$ for both the variables and constraints where K is the number of stations and N is the number of transportations tasks. Since the model size increases exponentially as the number of transportations tasks increases, it can be solved for small problem instances to optimality using commercial solvers. Therefore, the optimal solution is not always available within a reasonable time. This fact motivated the development of a heuristic scheduling algorithm.

The heuristic developed implements a 'non-delay' concept which always keeps the freight pipe line busy. According to this concept, when one capsule finishes its delivery task and leaves the pipe line, the heuristic must decide which task among the on-hand WIP has the top priority to be processed next.

Chou, et al. (2005) proposed a heuristic algorithm to minimize total weight tardiness on a single machine with release times which is similar to this problem which seeks to find a schedule that minimizes the total tardiness square. In order to modify this algorithm the freight pipe line must be considered as a machine by which the transportation tasks are processed However, since the freight pipe line only allows a certain number of capsules in it at any time, this virtual "machine" has multiple processors.

The heuristic developed has two phases. Phase I decides which capsules have arrived by means of the rolling of a timer which indicates the next available time of the pipeline and places capsules in a queue. The first two capsules in the queue are selected (i.e. capsules *a* and *b)* in order to determine the net increase in Total Tardiness Square (TTS) generated by sequencing the two capsules in *(a, b)* or *(b, a)* order. The preceding capsule in the order that has a smaller net increase of TTS is sequenced and enters the next iteration of comparison. This process is repeated until all open capacity of the freight pipe line is occupied. Phase II improves the result of phase I by using the capsule sequence generated in phase I as the initial solution and tries to improve the value of objective function by pair-wise sequence interchanges.

3.1.3.1 Phase I: the initialization step

Timer[N]: an array tracking the next available time point for scheduling the next job, note that N is the pipeline's linefill capacity

ASJS: already scheduled job set.

NSJS: not scheduled job set.

CJS: candidate job set containing jobs available to be scheduled

BJS: the job which has the top priority in CJS to be next scheduled

IJS: the initial job set

AVA: an variable keeps tracking current available linefill capacity

Step 0: Initial step,

Let timer[N]={0,…,0} (N is the pipeline's capacity), ASJS={Ø}, NSJS=IJS, CJS={Ø}, AVA=N

Step 1: Confirm the machine starting time for the next scheduled jobs

Let min_release_time = $min_{i \in NSJS}\{r_j\}$

If timer[0] < min_release_time, then let timer[0] = min_release_time

Step 2:

Pick up jobs from NSJS, whose release times are smaller than or equal to timer[0] and add them to CJS, $CJS = \{Job_i, i \in NSJS \; and \; r_j \leq timer[0]\}$

Step 3: Judge and select the BJS from CJS

If the candidate job set, CJS has m jobs ($m \leq AVA$), then go to step 3.1;

Otherwise ($m > AVA$), go to step 3.2

Step 3.1: Declare the m jobs in CJS be the BJS and eliminate them from CJS, go to step 4.

Step 3.2: let the first job in CJS be the defender job (j_α)

Step 3.3: let the second job in CJS be the challenger (j_β)

Step 3.4: use the performance index(Δ) to determine the winner

If the processing sequence of j_α and j_β is (j_α, j_β), the starting time of j_β is $\min\{timer[0] + p_\alpha, timer[1]\}\,(s_\beta)$, then

$\Delta_{\alpha\beta}$= [max $(timer[0] + p_\alpha - d_\alpha)]^2$ + [max $(s_\beta + p_\beta - d_\beta, 0)]^2$

If the processing sequence of j_α and j_β is (j_β, j_α), the starting time of j_β is $\min\{timer[0] + p_\beta, timer[1]\}\,(s_\beta)$, then

$\Delta_{\beta\alpha}$= [max $(timer[0] + p_\beta - d_\beta)]^2$ + [max $(s_\alpha + p_\alpha - d_\alpha, 0)]^2$

We can say the challenger defeats the defender if $\Delta_{\beta\alpha} < \Delta_{\alpha\beta}$ and the defender job will be replaced.

Step 3.5: determine whether we have found out the most appropriate next scheduled jobs BJS. If there are other jobs in CJS then continue executing step 3.3.

If there was not any other jobs in CJS then the final defender j_α will be in BJS, let $\text{AVA}_{\text{temp}} = \text{AVA}$, and $\text{AVA}_{\text{temp}} = \text{AVA}_{\text{temp}} - 1$;

Step 3.6: if $\text{AVA}_{\text{temp}} = 0$, go to step 4;

Otherwise, go back to step 3.2;

Step 4: add the most appropriate jobs, BJS, to the set of jobs already scheduled ASJS

CJS={Ø}, $\text{ASJS} = \text{ASJS} \cup \text{BSJ}$, NSJS=NSJS-BJS;

Update timer[N] (in increasing order)

Update AVA: check timer[N], see how many elements in timer[N] are equal to timer[0], say this number is b, which means b jobs can be scheduled to start at timer[0], then AVA=b;

Step 5: check if the stop condition is reached

If NSJS $\neq \{\emptyset\}$, then go to step 1 and continue executing the algorithm;

Otherwise, the solution is obtained;

3.1.3.2 Phase II: the improvement phase (pairwise interchange)

Following the initial schedule generated from phase I, a pairwise interchange procedure is applied for further improvement in the second phase. For N jobs, the pairwise interchange will generate N(N-1)/2 new sequence by interchanging all pairs of jobs, not just the adjacent pairs.

In the process of comparison, suppose the original sequence is S and its TTS (Total Tardiness Square) value is TTS(S). Every time, when we make an interchange and obtain a new order ($S^{'}$), we re-calculate the TTS value (TTS(S')) for it. If TTS(S) < TTS(S'), we retain S as the current best solution and proceed to the next interchange and make the TTS value comparison again. If TTS(S) > TTS(S'), we abandon S and make S' as the new best sequence. The iterations continue until all interchanges are finished and no better solution has been found. The very last sequence with the smallest TTS will be the final result.

According to the non-delay concept, phase I provides the initial solution in which the jobs sequenced depending on the minimum increase of TTS in the interchange of pair jobs order. Phase II adjusts the job sequence to decrease the value of objective function. Because of the consideration of minimum increase in TTS when we assign a job to be scheduled next in phase I, the result should be a fairly good one. Based on this outcome, phase II takes some steps for further improvement. Inheriting the result of phase I, it could save considerable swapping iterations and time in contrasting with randomly generated initial solution.

3.1.4 Results

A seven-task problem is used to test both the mathematical model and the developed heuristic. Table 3.1 shows the relevant information of the illustrative problem. An optimal solution for total tardiness square of 73 was obtained by solving the mathematical model using AMPL®. A detailed solution is given via a Gantt chart as shown in Figure 3.2. The heuristic results in the same solution as the optimal solution as shown in Figure 3.3. As a basis for comparison, three simple scheduling heuristics used: Shortest Processing Time (SPT), Earliest Release Time (ERT) and Earliest Due Date (EDD) to obtain solution to the example problem. The results of all five methods are given in Table 3.2.

Table 3.1 Example problem data

Tasks	Release time	Travel time	Due date
Task1	0	2	1
Task2	0	2	2
Task3	0	4	4
Task4	0	3	5
Task5	0	5	4
Task6	1	2	5
Task7	2	6	9

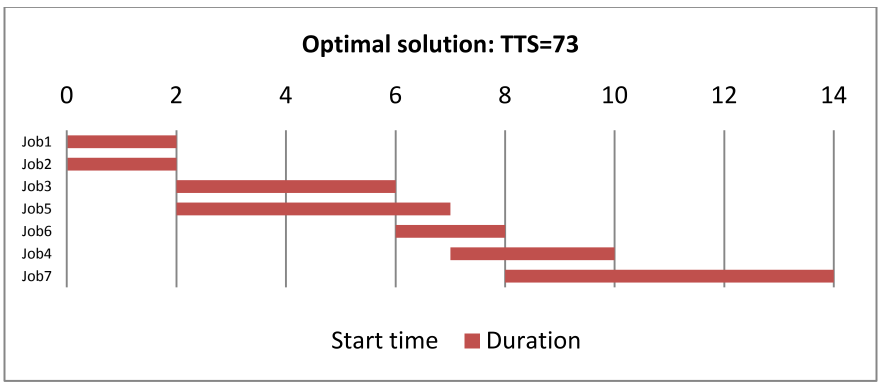

Figure 3.2 Optimal solution

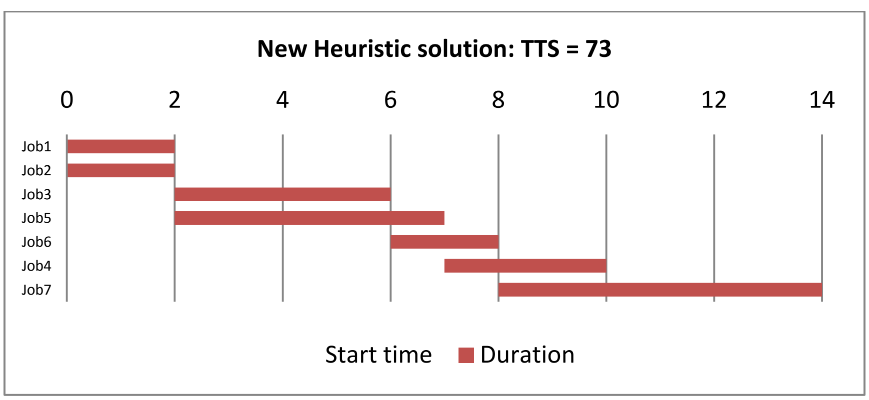

Figure 3.3 Solution obtained from the new heuristic

As shown in Table 3.2, both the mathematical model solved using AMPL® and the developed heuristic produce the optimal solution compared to the three simple heuristic that have significantly worse performance. It should be noted that it took 221453 MIP simplex iterations for AMPL® to solve the problem optimally, whereas only 31 pair wise comparisons were required for the heuristic to obtain the good solution (optimal in this case).

Table 3.2 Solutions obtained for all five methods

Test methods	Optimal	New Heuristic	SPT	ERD	EDD
Solution (TTS)	73	73	78	75	79

3.2 Capsule Control Model (with empty capsule routing)

Even though the mathematical model proposed in the last section is able to handle the scheduling problem to minimize the total tardiness square, there is an assumption which does not capture the real world resource limits. It is previously assumed that when a transportation task in a station is ready to be scheduled, there are always enough capsules ready in that station to be loaded, which is not always true. The model proposed below captures this issue by limiting a fixed number of capsules to the overall underground freight pipeline system thus it is possible that the free capsules are dispatched to stations where they are needed.

3.2.1 Model Notation

The following notation and definitions are used to describe the mathematical model of the control problem with empty capsule routing, for ease of illustration, we name it as Model 2.

- Sets:

  K: set of terminals

  T: set of time periods

  J: set of jobs

- Parameters:

  l_{mn}: travel time from terminal *m* to *n*

  r_i: release time of job *i*

  d_i: due date of job *i*

  α_{mni}: = 1, if job *i's* origin is terminal *m* and destination is terminal *n*

  = 0, otherwise

  L: line-fill capacity

  C: total number of capsules

- Decision variables:

s_i: start time of job i

R_{mtv}: number of capsules that are inbound to station m at time v and will arrive at station m at time t ($v \in T,\ v \leq t \leq v + l_{max}$). Note that When v = t, R_{mtv} denotes the number of capsules station m holds at time t

x^e_{mnv}: number of capsules start moving empty from station m to n at time v

x^l_{mnv}: number of capsules start moving loaded from station m to n at time v

w_i: tardiness of job i

$a^{t-v}_{l_{mn}}$: = 1 $if\ l_{mn} = t - v$

= 0 otherwise

Note that $a^{t-v}_{l_{mn}}$ plays a role as a "stop watch": a capsule traveling from m to n starting at time v arrives at its destination at time t if $t = v + l_{mn}$

$b^{t-v}_{l_{mn}}$: = 1 $if\ l_{mn} \leq t - v$

= 0 otherwise

$c^{t-v}_{l_{mn}}$: = 1 $if\ l_{mn} \geq t - v$

= 0 otherwise

y_{iv}: = 1 $if\ s_i = v$ (if job *i's* start time is *v*)

= 0 otherwise

3.2.2 Model Formulation

Minimize $\sum_{i \in J} w_i^2$ (1)

Subject to

$$s_i \geq r_i \ \ \forall\, i \in J \quad (2)$$

$$\sum_{v \in T} y_{iv} = 1, \forall\, i \in J \quad (3)$$

$$s_i = \sum_{v \in T} y_{iv} v,\ \forall\, i \in J \quad (4)$$

$$x_{mnv}^{l} = \sum_{i \in J} \alpha_{mni} y_{iv}\,, \forall\, v \in T, \forall\, m, n \in K \quad (5)$$

$$l_{mn} - (t - v) \geq -M b_{l_{mn}}^{t-v},\ \ \forall m, n \in K, v \in T, v + 1 \leq t \leq v + l_{max} \quad (6)$$

$$l_{mn} - (t - v) \leq M\left(1 - b_{l_{mn}}^{t-v}\right) - \varepsilon,\ \ \forall m, n \in K, v \in T, v + 1 \leq t \leq v + l_{max} \quad (7)$$

$$l_{mn} - (t - v) \geq -M\left(1 - c_{l_{mn}}^{t-v}\right),\ \ \forall m, n \in K, v \in T, v + 1 \leq t \leq v + l_{max} \quad (8)$$

$$l_{mn} - (t - v) \leq M c_{l_{mn}}^{t-v} - \varepsilon,\ \ \forall m, n \in K, v \in T, v + 1 \leq t \leq v + l_{max} \quad (9)$$

$$a_{l_{mn}}^{t-v} = \left(b_{l_{mn}}^{t-v} + c_{l_{mn}}^{t-v}\right) - 1 \quad (10)$$

$$\sum_{n \in K}\left(x_{mnt}^{e} + x_{mnt}^{l}\right) \leq R_{mtt}, \forall\, t \in T, \forall m \in K \quad (11)$$

$$\sum_{m \in K}\left(a_{l_{mn}}^{t-v} x_{mnv}^{e} + a_{l_{mn}}^{t-v} x_{mnv}^{l}\right) = R_{nvt},\ \ \forall n \in K, \forall v \in T, v + 1 \leq t \leq v + l_{max} \quad (12)$$

$$\sum_{m \in K} \sum_{n \in K}\left(x_{mnt}^{e} + x_{mnt}^{l}\right) + \sum_{m \in K} \sum_{n \in K} \sum_{v<t}\left(\left(1 - b_{l_{mn}}^{t-v}\right) x_{mnv}^{e} + \left(1 - b_{l_{mn}}^{t-v}\right) x_{mnv}^{l}\right) \leq$$

$$L, \forall\, t \in T \quad (13)$$

$$\sum_{m \in K} R_{mtt} + \sum_{m \in K} \sum_{n \in K} \sum_{v<t}\left(\left(1 - b_{l_{mn}}^{t-v}\right) x_{mnv}^{e} + \left(1 - b_{l_{mn}}^{t-v}\right) x_{mnv}^{l}\right) \leq C, \forall\, t \in T \quad (14)$$

$$R_{ntt} = \sum_{v<t} R_{nvt} + R_{n(t-1)(t-1)} - \sum_{m \in K}\left(x_{nm(t-1)}^{e} + x_{nm(t-1)}^{l}\right), \forall\, t \in T, \forall n \in K \quad (15)$$

$$s_i + \sum_{m \in K} \sum_{n \in K} l_{mn} \alpha_{mni} - d_i \leq w_i, \forall i \in J \quad (16)$$

Constraint (2) ensures that the start time of job *i* is not earlier than its release time. Constraint (3) ensures that one transportation job can only be scheduled for once. Constraint (4) calculates the start time of job *i*. Constraint (5) calculates the number of jobs start at time *v*. Constraint (6) and (7) ensure the correct value of variable $b_{l_{mn}}^{t-v}$. Similarly, constraint (8) and (9) ensure the correct value of variable $c_{l_{mn}}^{t-v}$. Constraint (10) makes sure variable $a_{l_{mn}}^{t-v}$ has its correct value. Constraint (11) specifies that the number of capsules going out from a station at time t is not more than the number of capsules the station holds at time t. Constraint (12) calculates the number of capsules at each time t at each station. Constraint (13) is the line fill constraint that ensures that the number of capsules in the freight pipe line does not exceed the maximum capacity. Note that $\sum_{m\in K}\sum_{n\in K}(x_{mnt}^{e}+x_{mnt}^{l})$ denotes the number of capsules start moving at time *t*, $\sum_{m\in K}\sum_{n\in K}\sum_{v<t}((1-b_{l_{mn}}^{t-v})x_{mnv}^{e}+(1-b_{l_{mn}}^{t-v})x_{mnv}^{l})$ denotes the number of capsules that are still moving (previously departed) at time *t*, then the sum of both is the total number of capsules moving at time t. Constraint (14) is the capsule resource capacity constraint that ensures that there are always a fixed number of capsules either moving in the tube or hold by stations at any given time. Note that $\sum_{m\in K}R_{mtt}$ denotes the number of capsules each station holds at time *t*, $\sum_{m\in K}\sum_{n\in K}\sum_{v<t}((1-b_{l_{mn}}^{t-v})x_{mnv}^{e}+(1-b_{l_{mn}}^{t-v})x_{mnv}^{l})$ denotes the number of capsules moving at time *t*, then the sum of both terms is the total number of capsules in the system. Constraint (15) calculates the number of capsules each terminal holds at each time *t*. Note that $\sum_{v<t}R_{nvt}$ denotes number of capsules that are arriving at station n at time *t*, $R_{n(t-1)(t-1)}$ denotes the number of capsules that station already have at time *t*-1, and $\sum_{m\in K}\left(x_{nm(t-1)}^{e}+x_{nm(t-1)}^{l}\right)$ denotes

the number of capsules that are leaving station n. Finally, Constraint (16) specifies the tardiness of each task.

3.2.3 Illustration problem

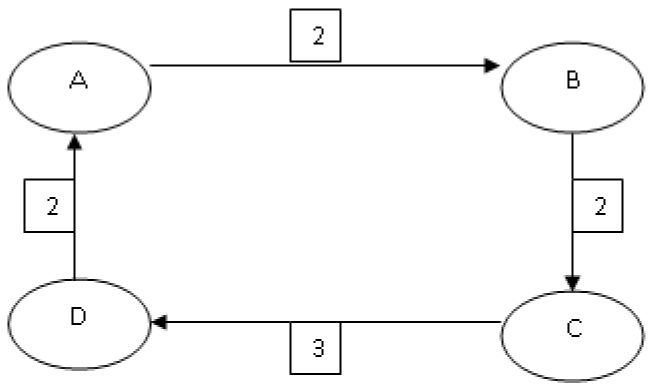

Figure 3.4 Network configurations for illustration problem

As previously stated, Model 2 takes into account the issue that in a real world situation the capsule resource is limited, and integrate the idle empty capsule routing into the model. To illustrate this unique ability, we propose a simple 2-job problem based on the network configuration shown above:

There are 4 stations, individually named A, B, C and D, the numbers shown in the Figure 3.4 denotes the travel time between stations. There are 2 transportation tasks and the related data is shown in Table 3.3.

Table 3.3 Transportation tasks related data

Task	Release Time	Due date	Origin	Destination	Travel Time
1	1	2	A	B	2
2	1	4	C	D	3

We assume that there is only one capsule in the system, which make it impossible to start both two transportation tasks simultaneously without routing and sharing the only capsule. Therefore, this problem cannot be solved in Model 1.

The solution of this simple problem is obvious: task 1 uses the only capsule traveling from station A to station B, then the capsule needs to move empty to station C where task 2 can ride on it and move from station C to station D.

The result from actually solving Model 2 is shown in Table 3.4. Task 1 starts at station A at time 1, arrives at station B at time 3 (travel time = 2), capsule moves empty at time 3 from station B and arrives at station C at time 5, Job 2 starts from station C at time 5. We notice that Model 2 gives the result we expect.

Table 3.4 Solution of illustration problem

Start Time:	**1**
	5
Total tardiness squared:	17

3.2.4 Analysis

There are many factors that can affect the overall performance of Underground Freight Pipeline System, such as average due date, due date slack time of the transportation tasks and resource limit of the system (number of capsules). In this section several tests are made to illustrate how different factors affect the TTS (Total Tardiness Square).

3.2.4.1 Average due date scenarios

Due date is the time by which a transportation task need to be finished so cargo of that specific task should be transported to its destination. We tested two sets of problems, one with 12 transportation tasks and the other with 15 transportation tasks, for each set of problem, the average due date is varied while other parameters are kept constant. In this way, we are able to show how the average due date affects the system's performance in terms of Total Tardiness Square. Table 3.5 is the 12-task problem data and the result is given in Figure 3.5.

Table 3.5 Data: 12-task problem

Due date	set 1	set 2	set 3	set 4	set 5
Job 1	2	1	1	1	2
Job 2	3	2	1	1	2
Job 3	5	4	2	2	2
Job 4	6	5	4	3	2
Job 5	5	4	3	2	2
Job 6	6	5	4	3	3
Job 7	10	9	8	6	4
Job 8	10	9	8	7	5
Job 9	16	15	14	12	9
Job 10	17	16	15	12	10
Job 11	16	15	14	13	9
Job 12	11	10	9	9	9
average	**8.92**	**7.92**	**6.92**	**5.92**	**4.92**
TTS	**24**	**53**	**92**	**133**	**199**

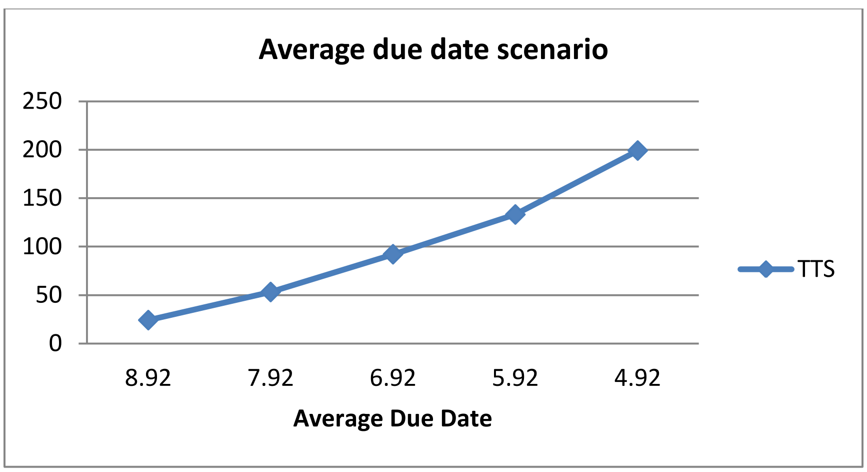

Figure 3.5 Average due date scenario: a 12-task problem

We observe that as the average due date decreases, the Total Tardiness Square increases in a linear fashion. Similarly, the 15 tasks problem set also shows the linear relationship between the average due date and Total Tardiness Square:

Table 3.6 Data: 15-task problem

Due date	set 1	set 2	set 3	set 4	set 5	set 6	set 7
Job 1	4	3	2	2	1	1	1
Job 2	5	4	3	3	2	1	1
Job 3	7	6	5	5	3	2	1
Job 4	9	8	9	5	4	3	2
Job 5	7	6	5	5	4	3	2
Job 6	6	5	5	5	4	3	2
Job 7	9	8	7	5	5	4	2
Job 8	8	7	6	5	5	4	3
Job 9	10	9	8	7	5	4	3
Job 10	11	10	9	7	6	5	4
Job 11	15	14	13	8	7	6	5
Job 12	10	9	8	9	8	6	5
Job 13	13	12	11	9	8	7	6
Job 14	12	10	8	9	8	7	6
Job 15	13	12	10	10	9	8	6
average	**9.27**	**8.20**	**7.27**	**6.27**	**5.27**	**4.27**	**3.27**
TTS	**91**	**155**	**215**	**412**	**534**	**697**	**880**

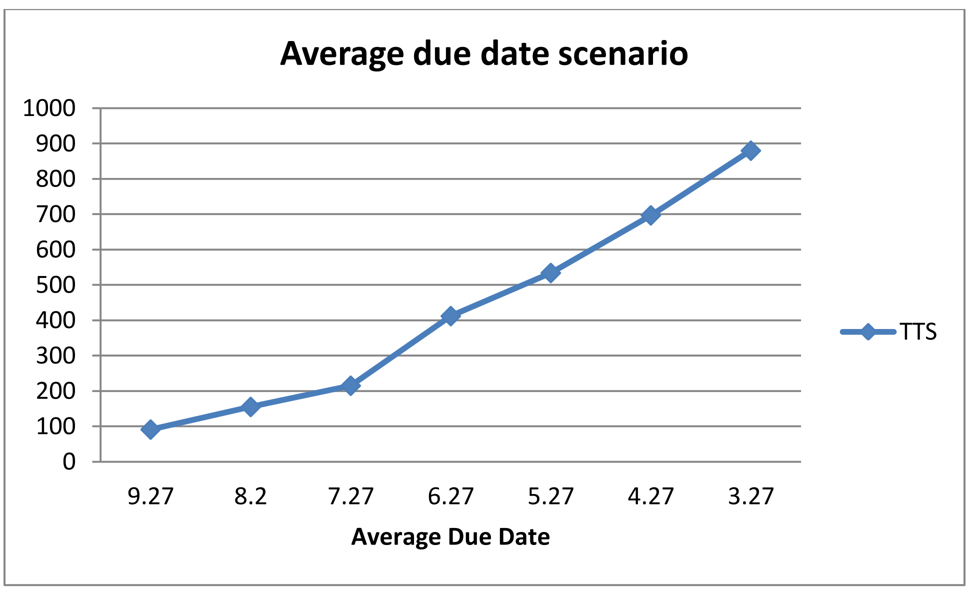

Figure 3.6 Average due date scenario: a 15-task problem

3.2.4.2 Due date slack scenario

Besides the average due date, the due date slack is another way to measure how timely critical a transportation task is. Note that Due Date Slack = Due Date – (Release Time + Travel Time), for instance, if a transportation task's Release Time = 1, Travel Time = 3, Due Date = 8, then its Due Date Slack = 8 – (1 + 3) = 4.

In this subsection, we tested a problem set with 12 transportation tasks, the ratio of the Due date slack and Travel time (Due Date Slack/Travel Time) is varied while other parameters are kept unchanged. As the result shown below, a slightly change of the Due date slack and Travel time ratio results in a tremendous change in TTS.

Table 3.7 Data: Due date slack scenario

SL/PT:	TTS:
0.5	534
1	264
2	21
3	0

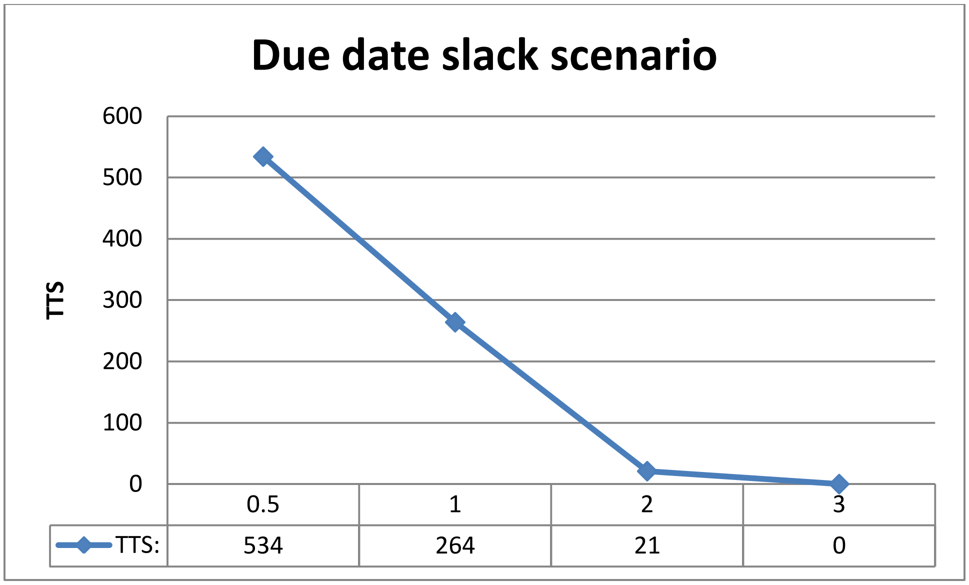

Figure 3.7 Due date slack scenario

3.2.4.3 Resource limit scenario

One of many resource limits of an underground freight pipeline system is the number of capsules. In general, if more capsules are available, more transportation tasks can start on time, and the overall system performs better. Three sets of problems are tested, individually with 10, 12 or 15 transportation tasks, the result are shown below:

Table 3.8 Result of Resource limit scenario: 10-task problem

number of capsules	*2*	*3*	*4*	*5*	*10*
Start time of job 1	3	3	1	1	1
Start time of job 2	3	3	1	1	1
Start time of job 3	1	1	5	3	1
Start time of job 4	10	1	1	1	1
Start time of job 5	1	1	1	1	3
Start time of job 6	5	5	3	3	3
Start time of job 7	8	8	6	6	6
Start time of job 8	12	7	7	7	7
Start time of job 9	16	13	7	8	5
Start time of job 10	17	12	10	7	8
TTS	**237**	**59**	**19**	**15**	**15**

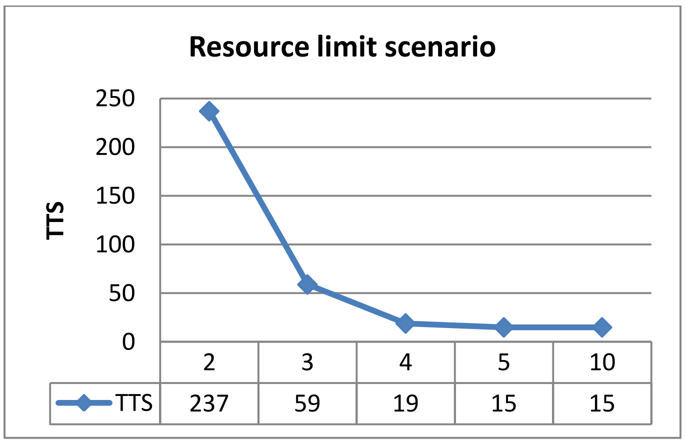

Figure 3.8 Resource limit scenario: 10-task problem

Table 3.9 Resource limit scenario: 12-task problem

number of capsules	2	3	4	5	12
Start time of job 1	1	1	1	1	1
Start time of job 2	5	3	1	1	1
Start time of job 3	3	1	1	3	3
Start time of job 4	1	1	1	1	1
Start time of job 5	10	8	6	1	1
Start time of job 6	12	3	3	3	3
Start time of job 7	12	8	6	5	6
Start time of job 8	7	7	7	7	7
Start time of job 9	23	14	10	13	12
Start time of job 10	21	12	10	10	10
Start time of job 11	19	17	12	11	8
Start time of job 12	10	10	8	9	9
TTS:	**528**	**91**	**24**	**15**	**15**

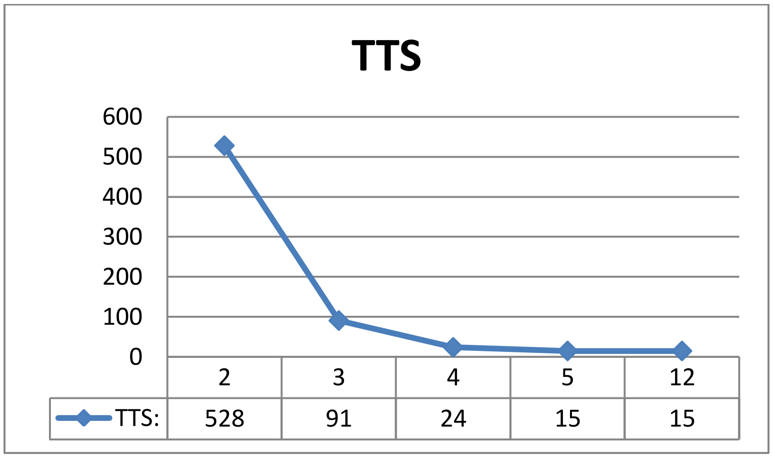

Figure 3.9 Resource limit scenario: 12-task problem

Table 3.10 Resource limit scenario: 15-task problem

Number of capsules	3	4	5	6	7	8	15
Start time of job 1	1	1	1	1	1	1	1
Start time of job 2	3	1	1	1	1	1	1
Start time of job 3	1	3	3	1	3	1	3
Start time of job 4	1	1	1	1	1	1	1
Start time of job 5	8	6	1	3	1	3	1
Start time of job 6	3	1	3	3	3	3	3
Start time of job 7	8	6	5	5	5	5	5
Start time of job 8	7	5	6	6	6	6	6
Start time of job 9	12	10	9	9	9	7	8
Start time of job 10	12	10	10	10	9	9	9
Start time of job 11	21	12	13	11	11	11	11
Start time of job 12	10	8	7	7	7	8	7
Start time of job 13	15	13	12	11	10	10	9
Start time of job 14	12	10	9	9	8	9	9
Start time of job 15	17	15	12	12	12	10	11
TTS:	**277**	**70**	**44**	**37**	**35**	**34**	**34**

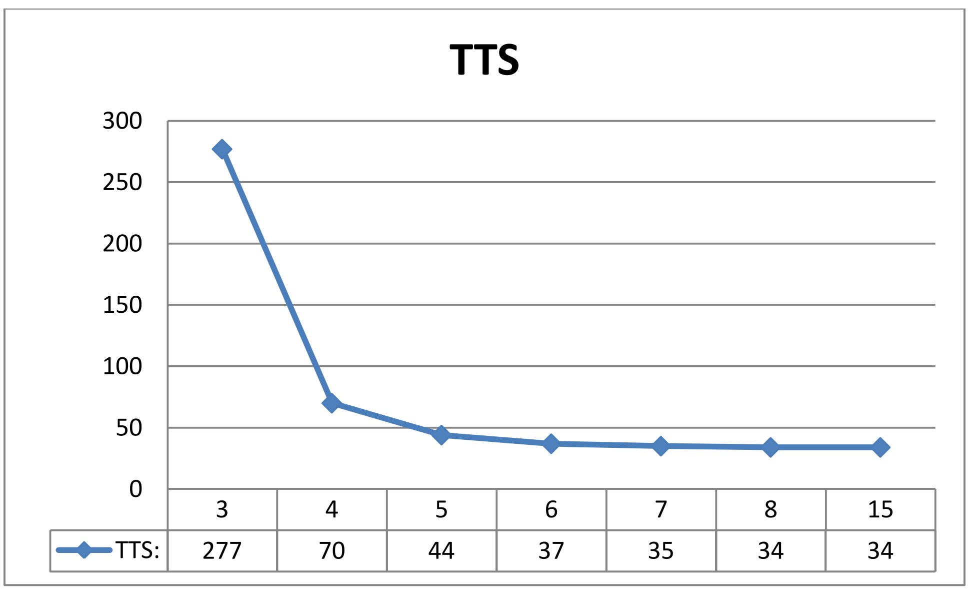

Figure 3.10 Resource limit scenario: 15-task problem

3.2.5 Extended Heuristic Algorithm (with empty capsule routing)

The mathematical Model 2 presented above has the complexity of $O(K^2T^2)$ for both the variables and constraints where K is the number of stations and T is the number of time periods. Similar to Model 1, Model 2 can only be solved to optimality for small problems using commercial solvers. As the problem size increases, it becomes impractical to obtain the optimal solution within a reasonable time. Based on the heuristic algorithm designed for Model 1, an extended heuristic that considers the issue of empty capsule routing is proposed.

This extended heuristic also has two phases. Phase I decides which transportation tasks to put in a queue by means of rolling of a timer which indicates the next available time and line fill open space. Then the tasks in the queue are compared iteratively by the net increase in Total Tardiness Square (TTS), the tasks with the least increase in TTS are put into the schedule. The heuristic not only considers the pipeline line fill capacity, but also considers the total number of capsule in the system. When a task's origin station does not have enough idling capsules, capsules in the neighboring stations need to be routed to the job's origin station. Similarly, in case that several tasks compete for the limited number of idling capsules, the net increase of TTS is used to decide the priority. In phase II, the result job schedule generated in phase I is set as the initial solution, and a pair wise interchange procedure is applied to improve the objective function value. The algorithm logic is shown below:

3.2.5.1 Phase I: the initial step

Sets:

Timer[N] (N is the pipeline's capacity): an array used to track the next available time point that is available to schedule the next job.

ASJS: already scheduled job set.

NSJS: not scheduled job set.

CJS: candidate job set containing jobs available to be scheduled

AVA: real time pipeline line-fill capacity

CapsulesST[s][t]: number of capsules in station S at time T

Step 0: Let timer[N]={0,…,0} (N is the pipeline's capacity), ASJS={Ø}, NSJS=IJS, CJS={Ø}, AVA=N

Step 1: confirm the start time for the next scheduled jobs

Let min_release_time = $min_{i \in NSJS}\{r_i\}$

If timer[0] < min_release_time, then let timer[0] = min_release_time

Step 2:

Pick up jobs from NSJS, whose release times are smaller than or equal to timer[0] and add them to CJS, $CJS = \{Job_i, i \in NSJS \; and \; r_i \leq timer[0]$

Step 3: Judge and select the BJS from CJS

If the candidate job set, CJS has m jobs ($m \leq AVA$), then go to step 3.1;

Otherwise ($m > AVA$), go to step 3.2

Step 3.1:

If all jobs in CJS have enough capsules in each of its origin station:

Declare the m jobs in CJS be the BJS and eliminate them from CJS, go to step 3.1.1.

Otherwise (one or more jobs in CJS do not have enough capsules in its origin station):

go to step 3.1.2;

Step 3.1.1: add the most appropriate jobs, BJS, to the set of jobs already scheduled ASJS

CJS={Ø}, $\mathrm{ASJS} = \mathrm{ASJS} \cup \mathrm{BSJ}$, NSJS=NSJS-BSJ;

Go to Step 4;

Step 3.1.2: Declare the jobs with enough capsules in its origins (let the number be *n*) be the BJS and eliminate them from CJS

Add the most appropriate jobs, BJS, to the set of ASJS;

Update Timer[N];

Update CapsulesInST[S][T];

The rest (m-n) jobs need to compete for (AVA-n) opening time points.

①: Let the first job out of m-n jobs be the defender job (j_α);

②: Let the second job out of m-n jobs be the challenger job (j_β);

Use the performance index (Δ) to determine the winner:

If the sequence is (j_α, j_β), j_α has the higher priority to use neighbors' free capsules:

Search j_α's origin station's closest station with free capsules at earliest time possible:

Let the start time of j_α be s_α, the empty capsule travel time be e_α, and the travel time of j_α be t_α;

Then the end time of j_α, $end_\alpha = s_\alpha + e_\alpha + t_\alpha$;

Similarly, the end time of j_β, $end_\beta = s_\beta + e_\beta + t_\beta$;

$\Delta_{\alpha\beta}$= [max $(end_\alpha - d_\alpha, 0)]^2$ + [max $(end_\beta - d_\beta, 0)]^2$

If the sequence is (j_β, j_α), j_β has the higher priority to use neighbors' free capsules:

Search j_β's origin station's closest station with free capsules at earliest time possible:

Let the start time of j_β be s_β, the empty capsule travel time be e_β, and the travel time of j_β be t_β;

Then the end time of j_β, $end_\beta = s_\beta + e_\beta + t_\beta$;

Similarly, the end time of j_α, $end_\alpha = s_\alpha + e_\alpha + t_\alpha$;

$\Delta_{\alpha\beta}$= [max $(end_\beta - d_\beta, 0)]^2 +$ [max $(end_\alpha - d_\alpha, 0)]^2$

We can say the challenger defeats the defender if $\Delta_{\beta\alpha} < \Delta_{\alpha\beta}$ and the defender job will be replaced.

Determine whether we have found out the most appropriate next scheduled jobs BJS. If there are other jobs in CJS then continue executing step ②

If there were not any other jobs in CJS then the final defender j_α will be in BJS, let AVA $=$ AVA, and AVA $=$ AVA $- 1$;

If AVA $= 0$, go to step 4;

Otherwise, go back to step ①;

Step 3.2: ($m > AVA$) Check the capsule availability of m jobs' origin, let the number of jobs that have available capsules in their origins be *n*;

If $n < AVA$, go to step 3.2.1;

If $n \geq AVA$, go to step 3.2.2;

Step 3.2.1:

Declare these n jobs be the BJS, and eliminate them from CJS;

Add the most appropriate jobs, BJS, to the set of ASJS;

Update Timer[N];

Update CapsulesInST[S][T];

The rest (m-n) jobs need to compete for (AVA-n) opening time points.

①: Let the first job out of m-n jobs be the defender job (j_α);

②: Let the second job out of m-n jobs be the challenger job (j_β);

Use the performance index (Δ) to determine the winner:

If the sequence is (j_α, j_β), j_α has the higher priority to use neighbors' free capsules:

Search j_α's origin station's closest station with free capsules at earliest time possible:

Let the start time of j_α be s_α, the empty capsule travel time be e_α, and the travel time of j_α be t_α;

Then the end time of j_α, $end_\alpha = s_\alpha + e_\alpha + t_\alpha$;

Similarly, the end time of j_β, $end_\beta = s_\beta + e_\beta + t_\beta$;

$\Delta_{\alpha\beta}$= [max $(end_\alpha - d_\alpha, 0)]^2$ + [max $(end_\beta - d_\beta, 0)]^2$

If the sequence is (j_β, j_α), j_β has the higher priority to use neighbors' free capsules:

Search j_β's origin station's closest station with free capsules at earliest time possible:

Let the start time of j_β be s_β, the empty capsule travel time be e_β, and the travel time of j_β be t_β;

Then the end time of j_β, $end_\beta = s_\beta + e_\beta + t_\beta$;

Similarly, the end time of j_α, $end_\alpha = s_\alpha + e_\alpha + t_\alpha$;

$\Delta_{\alpha\beta}$= [max $(end_\beta - d_\beta, 0)]^2$ + [max $(end_\alpha - d_\alpha, 0)]^2$

We can say the challenger defeats the defender if $\Delta_{\beta\alpha} < \Delta_{\alpha\beta}$ and the defender job will be replaced.

Determine whether we have found out the most appropriate next scheduled jobs BJS. If there are other jobs in CJS then continue executing step ②

If there were not any other jobs in CJS then the final defender j_α will be in BJS, let $\mathrm{AVA_{temp}} = \mathrm{AVA}$, and $\mathrm{AVA_{temp}} = \mathrm{AVA_{temp}} - 1$;

If $\mathrm{AVA_{temp}} = 0$, go to step 4;

Otherwise, go back to step ①;

Step 3.2.2:

①: Let the first job out of n jobs be the defender job (j_α);

②: Let the second job out of n jobs be the challenger job (j_β);

Use the performance index (Δ) to determine the winner

If the processing sequence of j_α and j_β is (j_α, j_β), the starting time of j_β is $\min\{timer[0] + p_\alpha, timer[1]\}$ (s_β), then

$\Delta_{\alpha\beta}$= [max $(timer[0] + p_\alpha - d_\alpha, 0)]^2$ + [max $(s_\beta + p_\beta - d_\beta, 0)]^2$

If the processing sequence of j_α and j_β is (j_β, j_α), the starting time of j_α is $\min\{timer[0] + p_\beta, timer[1]\}$ (s_β), then

$\Delta_{\beta\alpha}$= [max $(timer[0] + p_\beta - d_\beta, 0)]^2$ + [max $(s_\alpha + p_\alpha - d_\alpha, 0)]^2$

We can say the challenger defeats the defender if $\Delta_{\beta\alpha} < \Delta_{\alpha\beta}$ and the defender job will be replaced.

Determine whether we have found out the most appropriate next scheduled jobs BSJ. If there are other jobs in CJS then continue executing step ②.

If there were not any other jobs in CJS then the final defender j_α will be in BJS, let $\mathrm{AVA}_{\mathrm{temp}} = \mathrm{AVA}$, and $\mathrm{AVA}_{\mathrm{temp}} = \mathrm{AVA}_{\mathrm{temp}} - 1$;

If $\mathrm{AVA}_{\mathrm{temp}} = 0$, go to step 4;

Otherwise, go back to step ①;

Step 4: add the most appropriate jobs, BSJ, to the set of jobs already scheduled ASJS

CJS={Ø}, ASJS = ASJS∪BSJ, NSJS=NSJS-BSJ;

Update timer[N] (in increasing order)

Update AVA: check timer[N], see how many elements in timer[N] are equal to timer[0], say this number is b, which means b jobs can be scheduled to start at timer[0], then AVA=b;

Update CapsulesInST[s][t];

Step 5: check if the stop condition is reached

If NSJS ≠ {∅}, then go to step 1 and continue executing the algorithm;

Otherwise, the solution is obtained;

3.2.5.2 Phase II: the improvement phase (pairwise interchange)

In phase II, a pairwise interchange procedure is applied for further improvement. For N jobs, the pairwise interchange will generate N(N-1)/2 new sequence by interchanging all pairs of jobs, not just adjacent pairs. Then, the sequence with the least total tardiness square among the N(N-1)/2 new sequences is selected to be the initial sequence for the next iteration of pairwise interchange.

Let α be the improvement iteration break criteria (percentage), if the improvement from iteration m to iteration m+1 is less than or equal to α, then the best sequence from iteration m+1 is the final solution.

3.2.6 Heuristic and Optimal solution comparison

As previously stated, Model 2 can only be solved to optimality for small problems. Based on the experimental experience, within a fixed network configuration and resource limits, as the number of transportation tasks increases, the computational time increases exponentially. In this subsection, two numerical scenarios with increasing number of transportation tasks and a more comprehensive heuristic and optimal solution comparison are given. All scenarios are based on a 4-station network.

In the first scenario, the line-fill constraint is set to be 10, which denotes that at the most 10 transportation tasks are allowed being presented in the freight pipe line simultaneously. The total number of available capsules is set to be 10. The computational time versus the problem size (number of transportation tasks) are shown in Table. 3.11 and Figure 3.11. In this scenario, the problems with more than 18 transportation tasks cannot be solved within a reasonable time.

Table 3.11 Computational time versus problem size (Scenario 1)

Problem size	2	4	7	10	12	15	16	17	18
Comp. Time (seconds)	1	3	10	30	70	150	400	840	2940

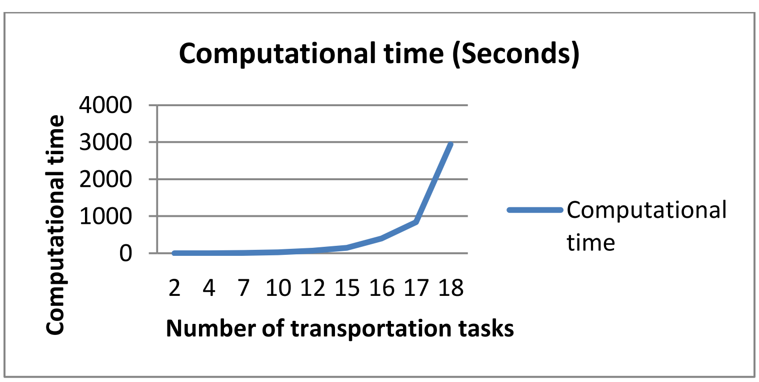

Figure 3.11 Computational time versus problem size (Scenario 1)

The second scenario is based on the same 4-station network configuration, yet the line-fill constraint is set to be 14 and total available capsules is set to be 15. As shown in Table. 3.12 and Figure 3.12, computational time increases exponentially as the number of transportation tasks increases, and problems with more than 45 transportations tasks cannot be solved within a reasonable time.

Table 3.12 Computational time versus problem size (Scenario 2)

Problem size	**2**	**5**	**8**	**15**	**20**	**25**	**30**	**35**	**40**
Comp. Time (seconds)	1	3	4	15	30	65	125	516	2635

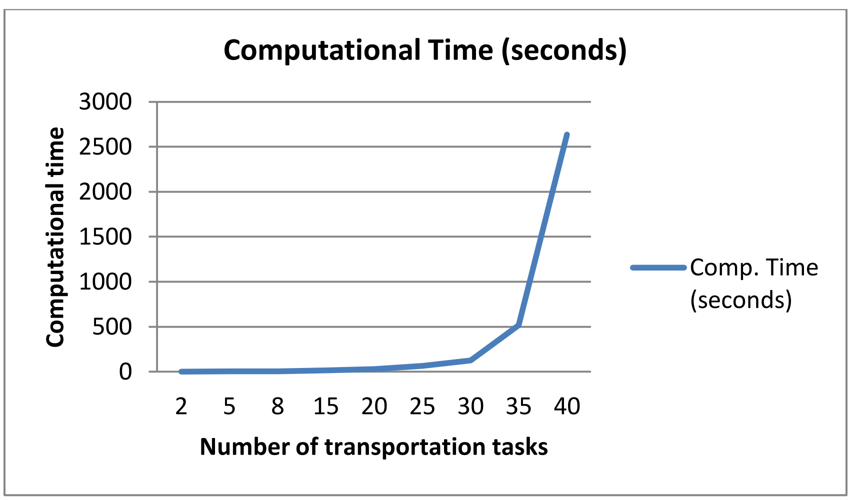

Figure 3.12 Computational time versus problem size (Scenario 2)

Even though the size of problem that can be solved with a reasonable time varies and depends on the problem parameters settings, in general, computational time increases exponentially thus we are only able to solve problems optimally with relatively small size.

Ten randomly generated data sets are solved by both commercial solver CPLEX and the heuristic algorithm and the objective function values from each approach are shown in Table. 3.13 and Figure. 3.13. We notice that the heuristic gives optimal solution for two data sets yet provides good solutions for the rest. Besides, the heuristic algorithm surpasses the commercial solver in terms of time efficiency. As shown in Table. 3.14 and Figure. 3.14, the heuristic algorithm is able to solve all the problem sets in less than 2 seconds while it takes one hour to solve certain problem with larger size, nevertheless it becomes impossible to solve the problem with more than 18 transportation tasks optimally.

Table 3.13 Optimal versus heuristic solutions

Optimal Solution	**110**	**15**	**10**	**15**	**70**	**1**	**12**	**2**	**292**	**458**
Heuristic Solution	125	15	15	23	78	6	18	2	303	476

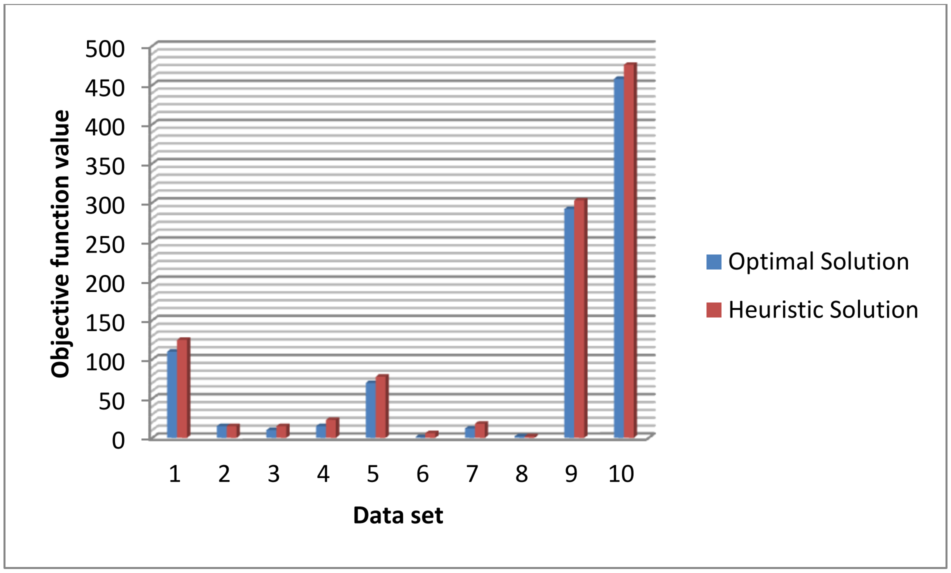

Figure 3.13 Optimal versus heuristic solutions

Table 3.14 Computational time (seconds): optimal versus heuristic

comp. time optimal	**1800**	**5**	**3**	**5**	**60**	**35**	**70**	**3**	**1900**	**3000**
comp. time heuristic	2	1	1	1	1	1	1	1	2	2

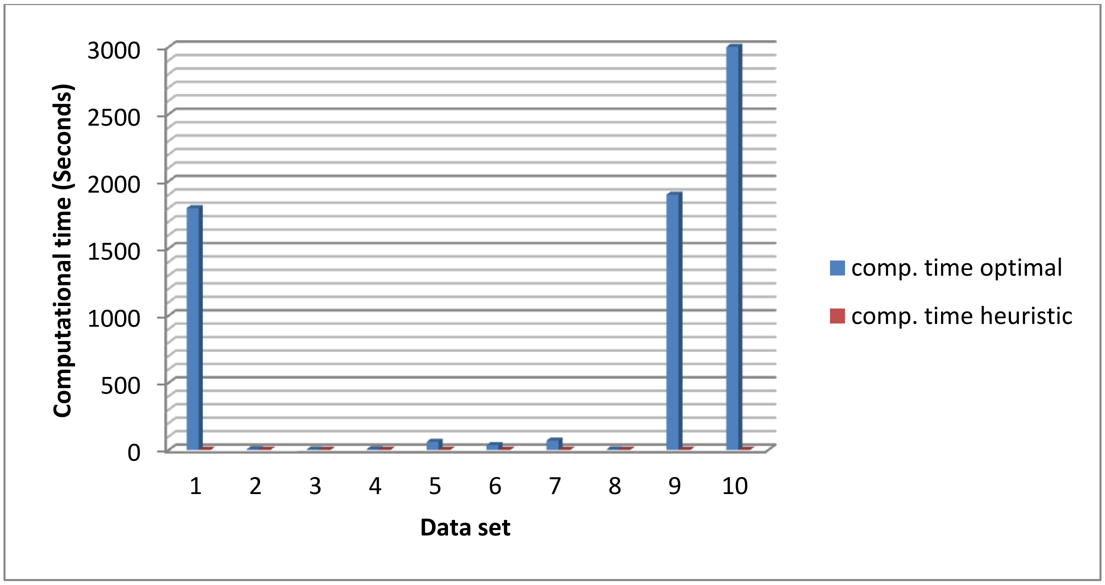

Figure 3.14 Computational time: optimal versus heuristic

In a more comprehensive comparison of heuristic and optimal solution shown below 15 data sets are randomly generated for each number of transportation tasks, 30, 35 and 40. All scenarios are based on a 4-station network shown in Figure 3.14. It is assumed that this network has the pipeline line-fill capacity of 14 and the total available number of capsules is 15.

The numerical results are given in Table 3.15, 3.16, and 3.17. The statistical result summary is shown in Table 3.18.

Table 3.15 Optimal versus Heuristic (30 transportation tasks)

	Objective function value		Difference		Computational time (second)	
Data set	Solver	Heuristic	Value	Percentage	Solver	Heuristic
1	1050	1050	0	0.00%	115	2
2	1102	1102	0	0.00%	110	1
3	750	813	63	8.40%	80	1
4	827	835	8	0.97%	95	1
5	985	985	0	0.00%	85	1
6	594	594	0	0.00%	90	1
7	670	690	20	2.99%	80	1
8	1883	1971	88	4.67%	80	1
9	911	998	87	9.55%	120	1
10	1262	1312	50	3.96%	150	1
11	1163	1163	0	0.00%	85	1
12	947	954	7	0.74%	120	1
13	808	812	4	0.50%	110	1
14	1246	1246	0	0.00%	123	1
15	1341	1341	0	0.00%	113	1
	Average:			**2.12%**	**103.73**	**1.07**

With 30 transportation tasks, the heuristic is able to obtain the optimal solution for 7 data sets out of 15, and the average difference in percentage between optimal solutions and heuristic solutions is 2.12%. For 15 data sets, the average computational time for CPLEX® is 103.73 seconds, while the heuristic's average computational time is slightly over 1 second.

Table 3.16 Optimal versus Heuristic (35 transportation tasks)

	Objective function value		Difference		Computational time (second)	
Data set	Solver	Heuristic	Value	Percentage	Solver	Heuristic
1	729	729	0	0.00%	516	1
2	871	982	111	12.74%	489	1
3	720	720	0	0.00%	495	1
4	2119	2119	0	0.00%	526	2
5	1457	1457	0	0.00%	603	1
6	1256	1354	98	7.80%	482	1
7	1427	1427	0	0.00%	1008	2
8	1023	1023	0	0.00%	690	1
9	907	907	0	0.00%	725	1
10	1155	1156	1	0.09%	256	1
11	1365	1365	0	0.00%	546	1
12	1666	1694	28	1.68%	555	2
13	2011	2037	26	1.29%	192	1
14	833	842	9	1.08%	463	1
15	985	985	0	0.00%	398	1
	Average:			**1.65%**	**529.60**	**1.20**

With 35 transportation tasks, the heuristic is able to obtain the optimal solution for 9 data sets out of 15, and the average difference in percentage between optimal solutions and heuristic solutions is 1.65%. For 15 data sets, the average computational time for CPLEX® is 529.6 seconds, while the heuristic's average computational time is 1.2 seconds.

Table 3.17 Optimal versus Heuristic (40 transportation tasks)

	Objective function value		Difference		Computational time (second)	
Data set	Solver	Heuristic	Value	Percentage	Solver	Heuristic
1	1187	1192	5	0.42%	2260	2
2	1405	1684	279	19.86%	2080	3
3	1292	1301	9	0.70%	1955	2
4	1292	1293	1	0.08%	2020	2
5	1576	1587	11	0.70%	2055	2
6	1909	1925	16	0.84%	2221	2
7	1308	1389	81	6.19%	2432	3
8	1396	1396	0	0.00%	2096	3
9	1744	1864	120	6.88%	2962	3
10	1629	1735	106	6.51%	2035	3
11	2212	2245	33	1.49%	2085	3
12	1066	1212	146	13.70%	2315	3
13	1374	1431	57	4.15%	2350	3
14	1015	1015	0	0.00%	2200	3
15	1246	1282	36	2.89%	2260	3
Average:				**4.29%**	**2221.73**	**2.67**

With 40 transportation tasks, the heuristic is able to obtain the optimal solution for 2 data sets out of 15, and the average difference in percentage between optimal solutions and heuristic solutions is 4.29%. For 15 data sets, the average computational time for CPLEX® is 2221.73 seconds, while the heuristic's average computational time is 2.67 seconds.

Table 3.18 Optimal versus Heuristic (result summary)

	Average computational time

Number of tasks	Average difference (%)	Solver	Heuristic
30	2.12%	103.73	1.07
35	1.65%	529.60	1.20
40	4.29%	2221.73	2.67

The result summary shown in Table 3.17 indicates that for a fixed network configuration as well as fixed resource limits, the computational time for CPLEX® solver increases quickly and for problems with more than 40 transportation tasks it becomes not practical to solve them optimally as the computational time is beyond reasonable range.

3.2.7 Heuristic solutions

Two larger size problem, respectively, with 20 and 40 transportation tasks are solved and the results are given in this subsection.

The 20-task problem is generated based on a 4-station network shown in Figure. 3.15. It is assumed that this network has the pipeline line-fill capacity of 4 which means only 4 or fewer capsules are allowed in the pipeline simultaneously. The travel time between each pair of stations is given in Table. 3.19. And there are 10 capsules in the system. The transportation tasks related data is given in Table. 3.20.

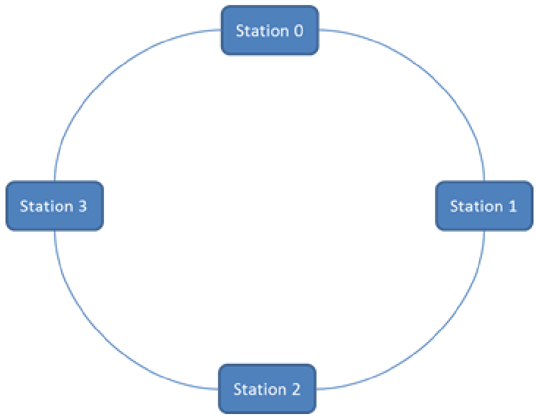

Figure 3.15 Underground pipeline network with 4 stations

Table 3.19 Travel time between stations

	Station 0	Station 1	Station 2	Station 3
Station 0	0	2	4	7
Station 1	7	0	2	5
Station 2	5	3	0	3
Station 3	2	4	6	0

Table 3.20 Data: a 20-task problem

Task	Origin	Destination	Release Time	Due Date
0	0	1	0	1
1	2	0	0	2
2	1	2	0	4
3	3	2	0	5
4	3	0	0	4
5	1	3	1	5
6	0	2	2	9
7	2	3	3	9
8	2	1	3	15
9	0	3	4	16
10	0	2	5	15
11	0	1	5	10
12	1	3	5	15
13	1	2	6	12
14	1	0	6	18
15	2	0	6	15
16	3	2	7	20
17	2	1	7	18
18	0	2	8	18
19	1	3	10	20

The developed heuristic solved the problem in less than 1 second with 3 iterations in heuristic Phase II. The solution with total tardiness square of 67 is shown in Table. 3.21 and Figure. 3.16 is a Gantt chart of the tasks schedule.

Table 3.21 Solution: a 20-task problem

Task	Start time	Duration	End time
0	0	2	2
1	0	5	5
3	0	6	6
4	0	2	2
2	2	2	4
5	2	5	7
6	4	4	8
7	5	3	8
10	6	4	10
13	7	2	9
8	8	7	15
11	8	2	10
12	9	5	14
15	10	5	15
9	10	7	17
14	14	7	21
18	15	4	19
17	15	7	22
16	17	6	23
19	19	5	24

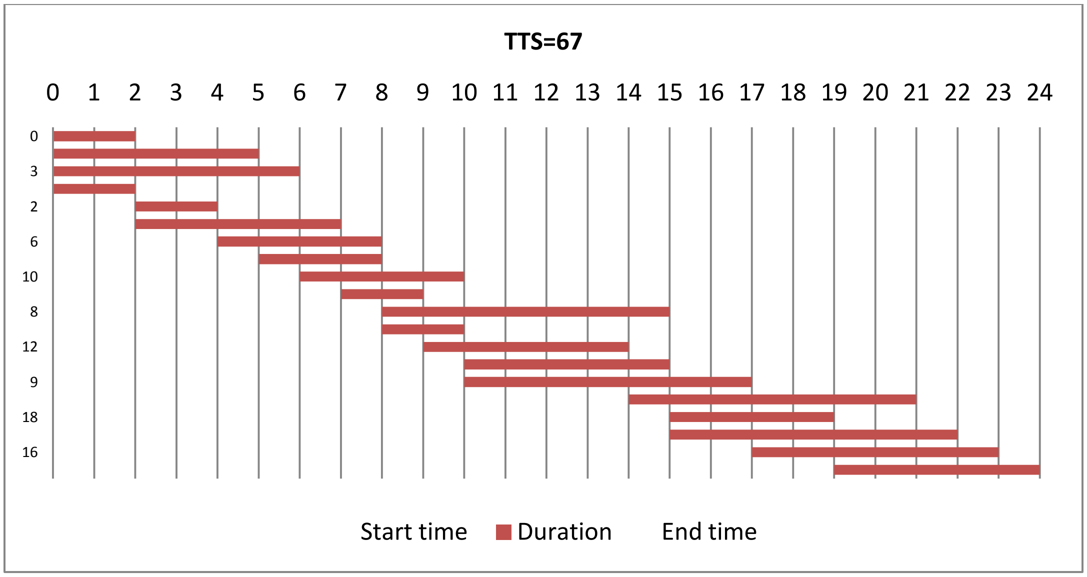

Figure 3.16 Gantt chart: a 20-task problem

The 40-task problem is generated based on an 8-station single loop network. Table 3.22 is the travel time between each pair of stations. And we assume that this specific pipeline network has the pipeline line-fill capacity of 10 and all transportation tasks share the total of 12 capsules.

The developed heuristic obtained the solution with the total tardiness square value of 15886 and the transportation tasks schedule Gantt chart is given in Figure. 3.17. It is clearly showed that at any given time point, there are at the most 10 transportation tasks are being processed, which satisfied the pipeline line-fill constraint.

Table 3.22 Travel time between stations

	Station 0	Station 1	Station 2	Station 3	Station 4	Station 5	Station 6	Station 7
Station 0	0	7	15	24	26	20	15	8
Station 1	7	0	8	17	25	27	22	15
Station 2	15	8	0	9	17	23	28	23
Station 3	24	17	9	0	8	14	19	26
Station 4	26	25	17	8	0	6	11	18
Station 5	20	27	23	14	6	0	5	12
Station 6	15	22	28	19	11	5	0	7
Station 7	8	15	23	26	18	12	7	0

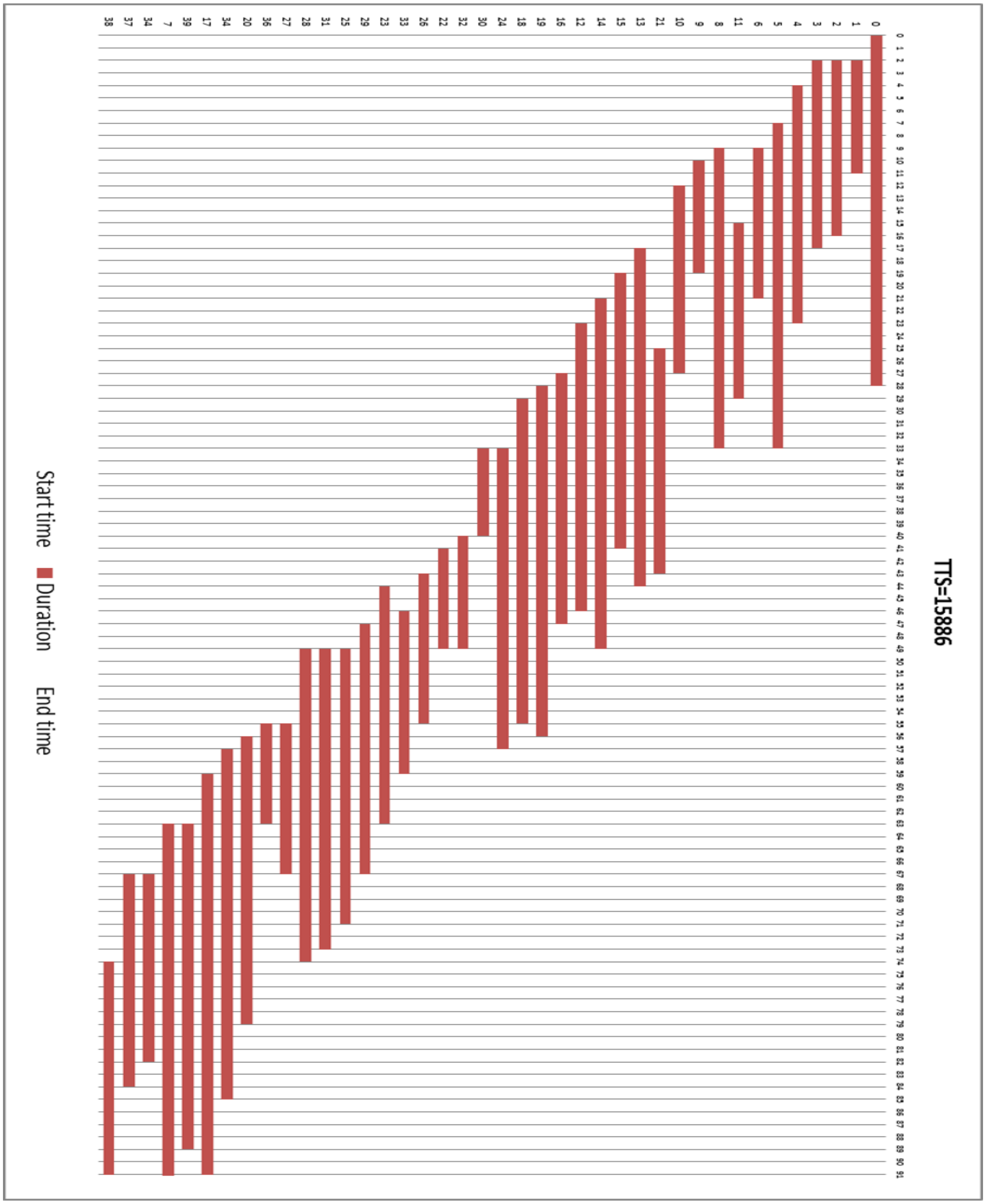

Figure 3.17 Gantt chart: a 40-task problem

3.2.8 Results comparison of the developed heuristic and other simple scheduling heuristics

The developed heuristic provides an efficient way to obtain a good solution to the proposed scheduling problem. In this subsection, the developed heuristic solutions are compared with solutions from other widely applied simple scheduling heuristics including Earliest Release Time (ERT), Earliest Due Date (EDD) and Shortest Processing Time (SPT).

10 problem data sets with a range of 30 to 73 transportation tasks are solved by the developed heuristic and other simple scheduling heuristics. The data sets are randomly generated by the following criterion:

- Consider a single loop underground pipeline system with 8 stations and 10 pipeline line-fill capacity. Travel time between each pair of stations is given in Table. 3.23
- The inter arrival time of all transportation tasks follows the exponential distribution with an average of 1 time unit.

As shown in Table 3.23 and Figure 3.18, with the systematical consideration of allocating pipeline line-fill capacity and capsules resource capacity to a set of transportation tasks, the developed heuristic generated better solutions than ERT, EDD and SPT scheduling heuristics. Note that, in this experimentation, the improvement iteration break criteria α is set to be 0.01%.

Table 3.23 Solution comparison

	set 1	set 2	set 3	set 4	set 5	set 6	set 7	set 8	set 9	set 10
Number of Tasks	30	35	40	40	38	50	60	65	70	73
Comp.Time (Minitues)	<1	<1	1	2	1	5	5	1	6	11
Number of iterations	6	5	6	18	6	41	36	26	37	51
Heuristic	3051	15300	12409	15886	7527	19055	37685	53819	38772	64750
ERT	10390	24660	23051	29082	17301	49981	81710	85275	75953	124154
EDD	5723	16126	14898	24387	9702	35169	63185	66670	57304	90340
SPT	14754	38442	49147	80259	35920	178931	211578	292513	184123	310172

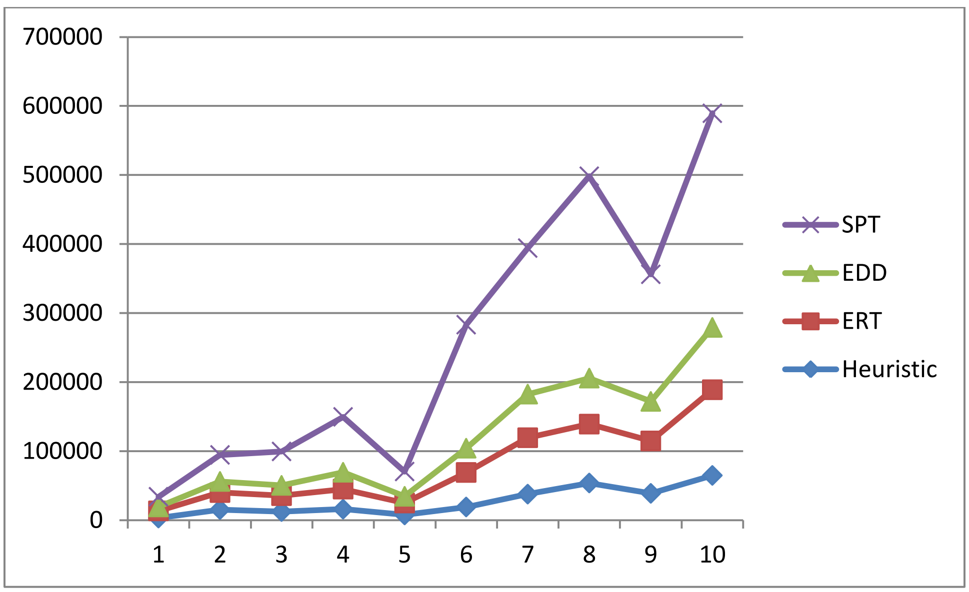

Figure 3.18 Solution comparison

Chapter 4 Underground Freight Pipeline Network Design

The scheduling problem mathematical models presented in the previous chapter focuses on the operational level of the overall underground freight pipeline system, which is based on a single loop network with loading/unloading stations that is initially considered as a portion of a more complex network configuration. Since the performance of the overall underground freight pipeline system relies not only on the optimization of the operational level, we found the necessity to extend the research to a higher level to the fundamental network configuration design. In this particular chapter, first we give a review of mathematical models in the similar rapid transit network design problem, and then we propose the underground freight pipeline network design problem and the associated mathematical model.

One issue that needs to be pointed out is that we realize that building an underground freight pipeline system in a metropolitan area is a major endeavor that requires long-term planning. These types of projects are very costly without mentioning the uncertainties, schedule and budget constraints. The decision making process always involve a number of stakeholders such as politicians, urban planners, engineers, management consultants and citizen groups. These players often have different or even conflicting objectives and constraints, which make it hard to solve the problem by a direct application of mathematical models and optimization algorithms. However, given the multi-player and multi-objective nature of the problem, we still believe and see a clear potential for the use of operations research methods and systems optimization techniques to construct and assess potential solutions to be later submitted to the decision makers.

4.1 Mathematical models of rapid transit network design problem

The rapid transit network design problem is to construct a set of interconnected transit lines within an undirected network $\mathrm{G} = (\mathrm{N}, \mathrm{E})$, where $\mathrm{N} = \{1, \dots, \mathrm{n}\}$ is the node set and $\mathrm{E} = \{(\mathrm{i}, \mathrm{j}): \mathrm{i}, \mathrm{j} \in \mathrm{N}, \mathrm{i} < j\}$ is the edge set. The nodes represent the population center in a metro area, while the edges represent the potential connections to be selected and built between node pairs. Each node is associated with a population number p_i and station construction cost c_i, and each edge is associated with a construction cost c_{ij}.

There are three main criteria used in mathematical models for rapid transit network design: (1) the total construction cost; (2) the total population covered by the network; and (3) the total traffic captured by the network. The problem can be modeled as a multi-objective optimization problem. Alternatively, it can be modeled with the objective of minimizing cost subjected to a population or traffic coverage constraint or with the objective of maximizing coverage subjected to a budget constraint.

The network location models with the objective of minimizing cost or maximizing coverage belong to the class of Steiner tree problem with profits (STPP) (Costa et al. 2006). Below are some classical examples:

4.1.1 Prize-collecting STPP (Costa et al. 2006)

Let x_{ij} be a set of binary variable equal to 1 if and only if edge (i, j) belongs to the network, let y_i be a binary variable equal to 1 if and only if vertex i belong s to the network, and let α be a user-controlled positive parameter. Let T ⊆ N be a set of nodes that must belong to the network. The mathematical model is:

Minimize $\sum_{(i,j)\in E} c_{ij}x_{ij} + \sum_{i\in N} c_i y_i - \alpha \sum_{i\in N} p_i y_i$ (1)

$\sum_{(i,j)\in E} x_{ij} = \sum_{i\in N} y_i - 1$ (2)

$\sum_{i,j\in S} x_{ij} \leq y_i \; k \in S \subseteq N, |S| \geq 2$ (3)

$y_i = 1 \; i \in T$ (4)

$x_{ij} = 0 \; or \; 1 \;\; (i,j) \in E$ (5)

$y_i = 0 \; or \; 1 \; i \in N$ (6)

The objective function of Prize-collecting STPP is to minimize the network construction cost while maximize the population covered by the network. Constraints (2) and (3) force the network to be a tree.

4.1.2 Quota STPP (Costa et al. 2006)

The objective of Quota STPP is to minimize construction cost while maintaining minimal population coverage β:

Minimize $\sum_{(i,j)\in E} c_{ij}x_{ij} + \sum_{i\in N} c_i y_i$

Quota STPP shares all the constraints of Prize-collecting STPP but with an additional constraint of: $\sum_{i \in N} p_i y_i \geq \beta$.

4.1.3 Fractional STPP (Klau et al. 2003)

Fractional STPP shares all the constraints of Prize-collecting STPP with the objective of maximizing the population-to-cost-ratio:

Maximize $\alpha \sum_{i \in N} p_i y_i / (\sum_{(i,j) \in E} c_{ij} x_{ij} + \sum_{i \in N} c_i y_i)$

4.1.4 Line location models

The previously mentioned STPP models are able to locate a network but cannot decompose it into distinct transit lines. Laporte et al. (2011) introduce a new model to locate a set L of lines covering a part of G. The objective of the model is to minimize a linear combination of the construction cost and the population covered by the transit network.

The following binary variables are used in the model:

$x_{ij}^l = 1$ if and only if edge (i, j) belongs to line l ∈ L

$x_{ij} = 1$ if and only if edge $(i, j) \in E$ belongs to a line

$y_i^l = 1$ if and only if a station is built at node i on line l

Let z_k^l be the number of edges incident to a node k belonging to line l excluding the edges of line l. c_{ij} is the edge construction cost and c_i^l is the station construction cost at node i on line l. Then the prize-collecting line location model is:

Minimize $\sum_{(i,j)\in E} c_{ij}x_{ij} + \sum_{l\in L}\sum_{i\in N} c_i^l y_i^l - \alpha \sum_{i\in N} p_i y_i$ (1)

Subject to

$x_{ij}^l \leq y_i^l, \forall(i,j) \in E, l \in L$ (2)

$x_{ij}^l \leq y_j^l, \forall(i,j) \in E, l \in L$ (3)

$x_{ij}^l \leq x_{ij}, \forall(i,j) \in E, l \in L$ (4)

$\sum_{i<k} x_{ik}^l + \sum_{j>k} x_{kj}^l \leq 2, k \in N, l \in L$ (5)

$\sum_{i,j\in S} x_{ij}^l \leq |S| - 1, \text{s} \subset \text{N}, |S| \geq 2, l \in L$ (6)

$\sum_{i,j\in N\ i<j} x_{ij}^l \geq \sum_{i\in N} y_i^l - 1, \text{l} \in \text{L}$ (7)

$z_k^l \geq \sum_{h\in L/\{l\}}(\sum_{i<k} x_{ik}^l + \sum_{j>k} x_{kj}^l) - M(1 - y_i^l), k \in N, l \in L$ (8)

$z_k^l \leq \sum_{h\in L/\{l\}}\sum_{i<k} x_{ik}^l + \sum_{j>k} x_{kj}^l) + M(1 - y_i^l), k \in N, l \in L$ (9)

$z_k^l \leq M y_k^l, k \in N, l \in L$ (10)

$\sum_{k\in N} z_k^l \geq 1$ (11)

$x_{ij}^l, x_{ij}, y_i^l = 0\ or\ 1, (i,j) \in E, i \in N, l \in L$ (12)

$z_k^l \geq 0\ k \in N, l \in L$ (13)

Constraints (2) and (3) force that edge (i, j) cannot be part of the network if no station is built at i and j. constraint (4) states that line l can use edge (i, j) only if $x_{ij} = 1$. Constraints (5) – (7) ensure that each line is made up of a single path. Constraints (8) – (11) prevent the formation of lines disconnected from the others.

The class of Steiner Tree Problem with Profits provides a basic approach to address the rapid transit network design problem. STPP answers the question of where to locate stations and how to connect them to form a network, yet it is not able to decompose the network into distinct transit lines which are critical to the complex networks due to the fact that it is easier to control and schedule since each line/alignment is relatively independent and also such a network can easily adapt a hub system to eliminate large number of point to point LTC transportation and consolidate cargo in the hub to cut cost.

Laporte et al. (2011)'s model took a step further to include the alignment/line allocation with the same objective function as the STPP to minimize the overall construction cost and maximize the population coverage. Its objective function is well suited to the rapid transit network design problem. However, if we change the scenario to cargo transportation instead of passenger movement, the objective function needs to be modified and additional issues need to be considered.

In the setting of designing a network configuration for an underground freight pipeline system, the overall construction cost remains a critical issue, yet the population coverage is no longer in consideration.

Besides the initial construction cost another concern of the users and operator of such a system is that they need the cargo to be transported in an efficient way so as to save time and money. One way to address this issue is that once a network configuration is decided, engineers need to design such a transportation routing plan that makes sure cargo is transported through the shortest possible path between their origin and destination. Even though this way of design is easy to execute, it does not consider the fact that a good network design should not only minimize the initial construction cost but also need to accommodate the major transportation corridor to enhance the transportation efficiency of the potential major cargo flow. Thus an alternative and better way to address this issue is not to think of these two design criteria separately but to integrate them together.

Above all, the new underground freight pipeline network design model's objective function not only includes the overall construction cost but also the operational cost. More specifically, the overall construction costs consist of both the station construction cost and the freight pipeline tunnel construction cost.

With the objective of minimizing the overall system cost, three major issues are addressed in the proposed network design model. Firstly, the network design model selects a set of network edges from the pool of all possible edges connecting the predefined station nodes. The selected network edges connect the predefined station nodes to form a connected network and incur the freight pipeline tunnel construction cost. Secondly, the network design model allocates the selected network edges into several transit lines. Each line has to be interconnected with at least another line so that the whole network is connected. Note that the intersect station of two or more lines is considered as a transfer station which has a higher construction cost than the regular stations to respect the fact that a transfer station has a more complex structure and is usually larger than a regular station. Thirdly, the operational cost depends not only on the fundamental network structure, i.e. the directness of the freight pipeline tunnel and the number of transfer stations between cargo's origin and destination stations, but also on the actual routing plan cargo follows. Thus, the network design model makes sure that cargo from each origin and destination station pair travels the shortest possible route.

Overall, these three design and modeling issues are not independent but interact with each other. Freight pipeline tunnel selection and line allocation together define the network structure thus incur the overall construction cost including the freight pipeline tunnel, station, and transfer station construction cost. However, if we consider the fact that in reality cargo flows are usually not evenly distributed, it is possible that certain major business/industrial districts have higher volume of cargo flow. Then, the network designer needs to consider such fact and generate the network that not only minimize the overall construction cost but also provide more freight pipeline tunnel directness and less transfer operations for high volume cargo flow.

4.2 Underground Freight Pipeline Network Design Problem

As presented in the previous chapters, the underground freight pipeline network considered so far consists of only a single loop with loading/unloading stations. Similar to an urban rapid transit system, a single loop or a single line is a fully functional component of a more complex network configuration which usually has several lines (loop lines and/or single lines). In each line, there are a number of loading/unloading stations and cargo is able to move from one station to the other. Lines are interconnected with each other. The intersection of two or more lines is considered as a transfer station which not only has the basic loading/unloading function of a regular station but also the ability of transferring cargo from one line to the other, i.e. as shown in Figure 4.1, if the cargo from station A needs to be transported to station B which is located in a different line, it needs to be transported to transfer station S first and then transported to station B.

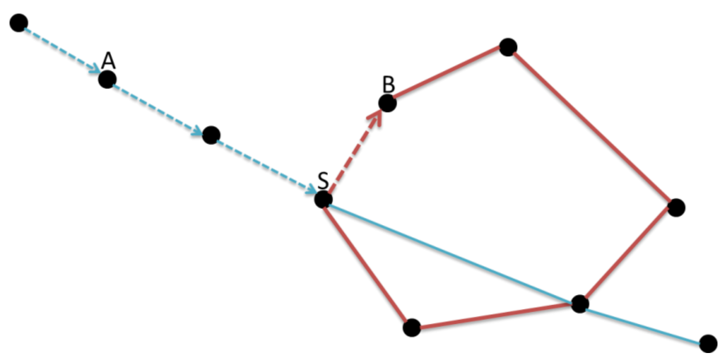

Figure 4.1 Transfer cargo between transit lines

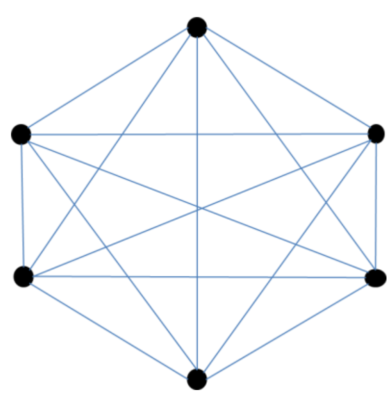

Figure 4.2 Station nodes and all possible connections

Consider the potential network shown in Figure 4.2, which includes a set of predefined nodes and all possible edges connecting each pair of nodes. The nodes in the figure represent the stations that have cargo flow in and out, and each station node is associated with a station construction cost and a unit cargo transfer cost in case that the node is determined to be a transfer station. Each edge is associated with a tunnel construction cost and a unit capsule transportation cost. We also assume that the cargo flow information between each pair of stations is given. Then, the objective is to find a network configuration with a number of lines (loop or single line) that connects all predefined station nodes to minimize the total construction cost as well as the operational cost. The issue that how the operational cost affects the overall network design is shown as follows: let A, B, C, D and E be the predefined station nodes as shown in Figure 4.3 and the cargo flow information is given in Table 4.1. We notice that the flow between each of station A, B, and E is dominant. Intuitively, one would like to locate a line following the route from station A, B to E and leave the edge BC and BD as branch lines. In such a way, no transfer operations are needed for dominant cargo flow.

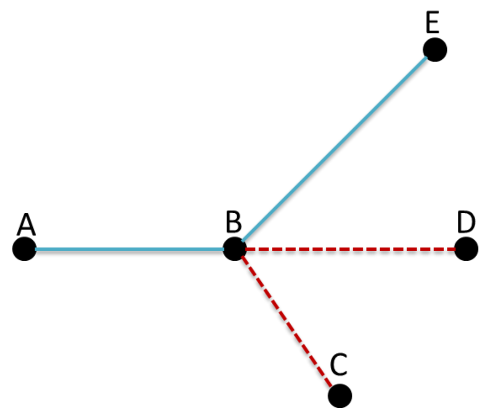

Figure 4.3 Operational cost affects the overall network design

Table 4.1 Cargo flow sample data

	A	B	C	D	E
A	0	50	10	10	100
B	50	0	10	10	80
C	10	10	0	10	20
D	10	10	10	0	20
E	100	80	20	20	0

The proposed underground freight pipeline network design model is able to answer the following questions:

- Which edges are included in the network?
- Which line does each edge belong to?
- Which edges and lines does each cargo use (Shortest path)?

Section 4.2.1 shows a comprehensive network design mathematical model, and presents the findings from some initial testing results. Due to the exponential increase of computational time of solving this model, it is not practical to use this model for larger sized network design problem, therefore, in Section 4.2.2, based on the comprehensive model, we propose *the UFP Network Design Two Steps Model* which can greatly reduce the computational effort while still are able to generate optimal or close to optimal solutions

4.2.1 Comprehensive UFP Network Design Model

Sets:

N: Station nodes n

L: Line l

E: Edges e

Parameters:

c_{ij}^{c}: Construction cost of edge (i, j)

c_{i}^{l}: Construction cost of station i on line l

c_{ij}^{T}: Unit transportation cost on edge (i, j)

c^{o}: Unit operational cost in transfer station

n_{uv}: Number of cargo from station u to station v

Decision variables:

$$x_{ij}^{l} = \begin{cases} 1, \text{if edge (i, j) belongs to line l} \\ 0, otherwise \end{cases}$$

$$x_{ij} = \begin{cases} 1 \text{ if edge (i, j) belongs to a line} \\ 0, otherwise \end{cases}$$

$$y_{i}^{l} = \begin{cases} 1, \text{if a station of line l is built at node i} \\ 0, otherwise \end{cases}$$

$$z_{uvij}^{l} = \begin{cases} 1, \text{if cargo from station u to v use edge (i, j) on line l} \\ 0, otherwise \end{cases}$$

$$b_{uvi}^{l} = \begin{cases} 1, \text{cargo from station u to v use line l either entering} \\ \quad\quad \text{node i or leaving node i, but not both} \\ 0, otherwise \end{cases}$$

$$x_{uvi}^{T} = \begin{cases} 1, \text{if cargo from station u to v is transfered in node i} \\ 0, otherwise \end{cases}$$

Objective function:

Minimize:

$$\sum_{(i,j)\in E} c_{ij}^{C} x_{ij} + \sum_{l\in L} \sum_{i\in N} c_i^l y_i^l + \sum_l \sum_u \sum_v \sum_{(i,j)} c_{ij}^{T} n_{uv} z_{uvij}^{l} + c^o \sum_u \sum_v \sum_i n_{uv} x_{uvi}^{T} + \sum_u \sum_v \sum_i \sum_l b_{uvi}^{l}$$

(1)

Subject to:

$$x_{ij}^{l} \le y_i^l, \forall (i,j) \in E, l \in L \quad (2)$$

$$x_{ij}^{l} \le y_j^l, \forall (i,j) \in E, l \in L \quad (3)$$

$$x_{ij}^{l} = x_{ji}^{l} \quad (4)$$

$$\sum_{j\in N, j\neq i} x_{ij}^{l} \le 2 \; \forall i \in N, \forall l \in L \quad (5)$$

$$\frac{1}{2} \sum_{\{i,j\}\in E} x_{ij}^{l} \ge \sum_{i\in N} y_i^l - 1, \forall l \in L \quad (6)$$

$$b_{uvi}^{l} \ge \sum_j z_{uvji}^{l} - \sum_j z_{uvij}^{l}, \forall u, v \in N, u \neq v, \forall i \in N, i \neq u, v, \forall l \in L \quad (7)$$

$$b_{uvi}^{l} \geq \sum_j z_{uvij}^{l} - \sum_j z_{uvji}^{l}, \forall u,v \in N, u \neq v, \forall i \in N, i \neq u,v, \forall l \in L \quad (8)$$

$$x_{uvi}^{T} = \frac{1}{2}\sum_l b_{uvi}^{l}, \forall u,v \in N, u \neq v, \forall i \in N, i \neq u,v \quad (9)$$

$$x_{ij}^{l} \geq z_{uvij}^{l} \quad (10)$$

$$\sum_l \sum_j z_{uvij}^{l} - \sum_l \sum_j z_{uvji}^{l} = \begin{cases} 1 \ if \ i = u \\ -1 \ if \ i = v \\ 0 \ otherwise \end{cases} \quad (11)$$

The objective function (1) has two major components: construction cost and operational cost.

$\sum_{(i,j)\in E} c_{ij}^{C} x_{ij}$ denotes the total tunnel construction cost for all edges that are included in the final network configuration.

$\sum_{l\in L} \sum_{i\in N} c_i^l y_i^l$ is the station construction cost. Note that the cost of constructing a station at node i is counted for each line to which the node v belongs, which corresponds to the fact that the construction cost of a transfer station is higher than a regular station.

$\sum_l \sum_u \sum_v \sum_{(i,j)} c_{ij}^{T} n_{uv} z_{uvij}^{l}$ is the transportation cost occurred as cargo moves through the selected tunnels (edges).

$c^o \sum_u \sum_v \sum_i n_{uv} x_{uvi}^{T}$ is the operational cost in transfer stations.

Constraint (2) and (3) ensures that edge (i, j) cannot be part of the network if no station is built at node i and node j. Constraint (4) ensures that the constructed tunnel in edge (i, j) is bidirectional. Constraints (5) forces that the degree of each node is 0, 1 or 2.

Constraints (5) and (6) together imply that each line is a collection of continuous edges which is either a single line or a loop line.

Since constraint (6) defines the relationship between the number of stations and links of each line, it is worth noticing that this model allows non-connected lines consisting of one non-circular sub-line and various circular sub-lines. Figure 4.4 shown below gives an example that non-connected lines are formed. This would increase the number of lines for the network, but it would not affect the construction cost and constraints (7), (8) and (9) make sure the transfer operations are correctly calculated even in case that non-connected lines are in place.

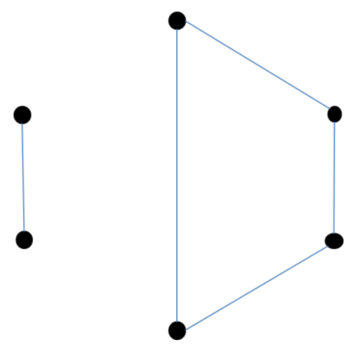

Figure 4.4 Non-connected lines are allowed

Constraints (7) and (8) ensure the correct value of b_{uvi}^{l}. Constraint (9) calculates the number of transfer operations. Below is an example that shows how these constraints work:

There are two lines shown in the Figure 4.5, dashed line denotes line 1 and solid line represents line 2, note that line 2 is a non-connected line with one non-circular sub-line and one circular sub-line.

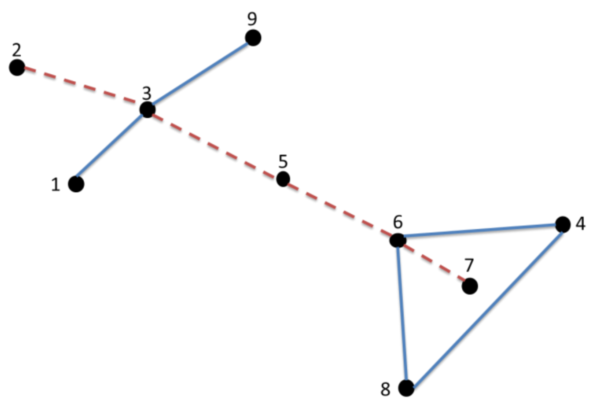

Figure 4.5 Non-connected lines

Assume that cargo from station 1 to station 8 follows the path 1-3-5-6-8, then cargo needs to be transferred in station 3 and 6 which is captured in constraint (7) and (8):

For O/D pair station 1 and 8, at station 3 in line 2 (solid line), constraint (7) and (8) are:

$$b_{183}^{2} \geq \sum_{j} z_{18j3}^{2} - \sum_{j} z_{183j}^{2} = 1 - 0 = 1$$

$$b_{183}^{2} \geq \sum_{j} z_{183j}^{2} - \sum_{j} z_{18j3}^{2} = 0 - 1 = -1$$

Then: $b_{183}^{2} \geq 1$, since the objective function has an additional term to minimize the total sum of b_{uvi}^{l}, we have $b_{183}^{2} = 1$ which denotes that cargo from station 1 moving to station 8 use line 2 either entering or leaving station 3 ("entering" in this example).

Similarly, for O/D pair station 1 and 8, for station 3 and line 1 (dashed line), constraint (7) and (8) are:

$$b_{183}^{1} \geq \sum_{j} z_{18j3}^{1} - \sum_{j} z_{183j}^{1} = 0 - 1 = -1$$

$b_{183}^{1} \geq \sum_j z_{183j}^{1} - \sum_j z_{18j3}^{1} = 1 - 0 = 1$

Then: $b_{183}^{1} \geq 1$, since the objective function has an additional term to minimize the total sum of b_{uvi}^{l}, we have $b_{183}^{1} = 1$ which denotes that cargo from station 1 moving to station 8 use line 1 either entering or leaving station 3 (leaving in this example).

Then for O/D pair station 1 and 8, for station 3, constraint (9) is:

$x_{183}^{T} = \frac{1}{2}(b_{183}^{1} + b_{183}^{2}) = \frac{1}{2} \times 2 = 1$ This indicates that cargo from station 1 to station 8 is transferred in station 3.

Constraint (10) ensures that cargo be transported through edge (i, j) of line l only if edge (i, j) is constructed and is assigned to line l.

Constraints (11) is inherited from the classic Shortest Path Problem to ensure that all cargo follow the shortest path from its origin station node to its destination station node. Three parts as illustrated below:

- For the origin node u, the difference in the number of edges leaving u and entering into u is 1
- For the destination node v, the difference in the number of edges entering into v and leaving v is 1
- For all other nodes, the number of edges of outgoing and incoming are equal

4.2.1.1 Initial test result

Figure 4.6 shows a network with 5 station nodes as well as all possible edges that connect any pair of station nodes. We assume that in this test problem, the single station construction cost is $1,000,000 and the edge construction cost and transportation cost are proportional to the length of the edge.

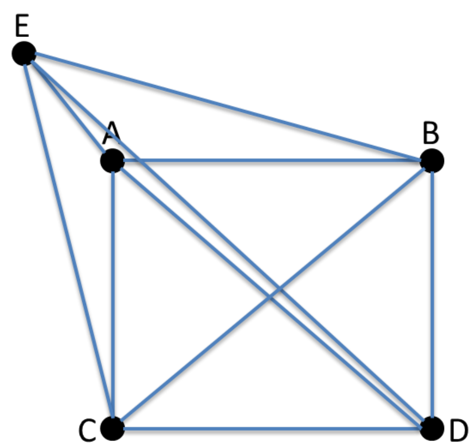

Figure 4.6 5-station network

Table 4.2 Initial test data

		Scenario 1	Scenario 2	Scenario 3
Origin	Destination	cargo Quantity	cargo Quantity	cargo Quantity
A	B	200000	200000	2500000
A	C	200000	200000	2500000
A	D	2500000	2500000	200000
B	A	200000	200000	2500000
B	C	200000	200000	200000
B	D	5000	5000	2500000
C	A	200000	200000	2500000
C	B	200000	200000	200000
C	D	210000	2500000	2500000
D	A	2500000	2500000	200000
D	B	5000	5000	2500000
D	C	210000	2500000	2500000
E	A	30000	30000	30000
E	B	20000	20000	20000
E	C	20000	20000	20000
E	D	200000	200000	200000
A	E	30000	30000	30000
B	E	20000	20000	20000
C	E	20000	20000	20000
D	E	200000	200000	200000

With all the cost components fixed (station construction cost, tunnel edge unit length construction and transportation cost, transfer cost in transfer station), then the cargo flow profile plays a big role in shaping the structure of the UFP network. Table 4.2 shows 3 cargo flow profile scenarios for the network in Figure 4.6. Assumed that we need to build two lines for this 5-station network, in scenario 1, the cargo flow between station node A and D are dominant, the network design solution is presented in Figure 4.7, we notice that node A and node D are directly connected to provide the shortest possible route for the dominant cargo flow. Also, as the cargo flow between node B and node D is relatively small, the connection between B and D is indirect and two nodes are assigned to different lines.

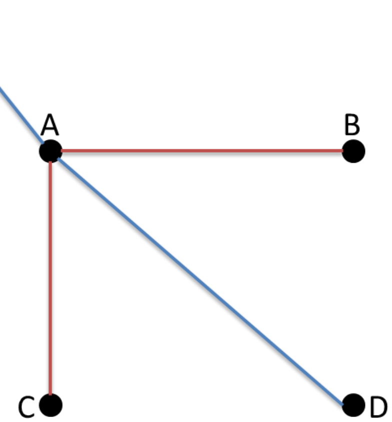

Figure 4.7 Initial result - Scenario 1

In scenario 2, since cargo flow between A and D, C and D are dominant, as the network design solution shown in Figure 4.8, node A and D, node C and D are directly connected and are assigned to the same line, in such a way, cargo flow between them can shipped directly without going through any transfer stations.

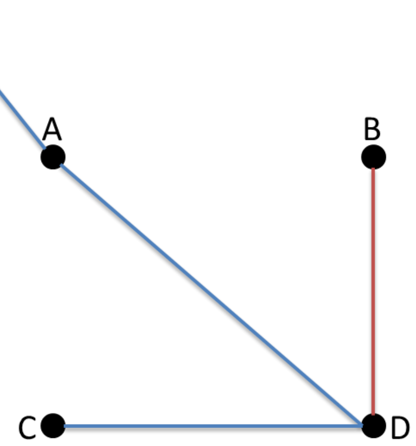

Figure 4.8 Initial result - Scenario 2

In scenario 3, as shown in Table 4.2, the cargo flow between O/D pairs of AB, AC, BD and CD are dominant. The network design model gives us the solution, as shown in Figure 4.9, that OD pair of AB, AC, BD and CD are directly connected and form a loop line.

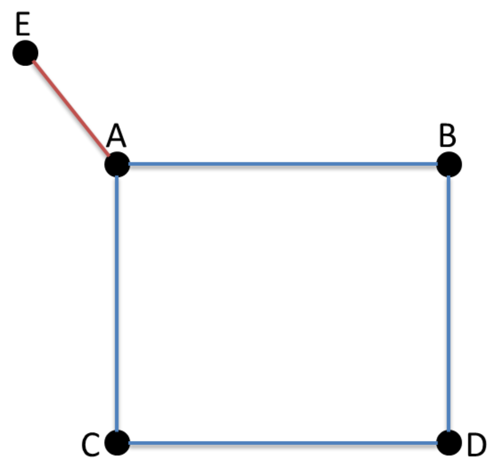

Figure 4.9 Initial result - Scenario 3

To summarize, the initial analysis of a 5-station test problem show us that the UFP network work design model is able to generate optimal network design solution with the respect to different cargo flow profile to provide shipping directness for dominant cargo flow.

Furthermore, several larger sized test problems are tested to evaluate the computational performance of the UPF network design model. Figure 4.10 shows a 10-station network with edges assigned to 4 lines generated from UFP network design model. Figure 4.11

shows a 12-station network with 3 lines. And in Figure 4.12 is a 15-station network with 3 lines. The computational time is summarized in Table 4.3 and Figure 4.13.

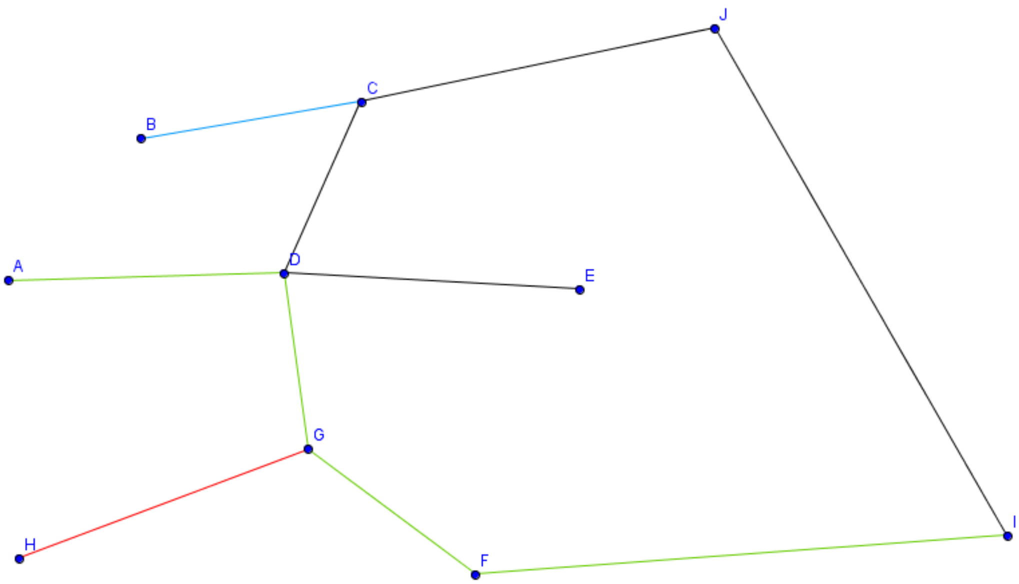

Figure 4.10 Test result: 10 station network

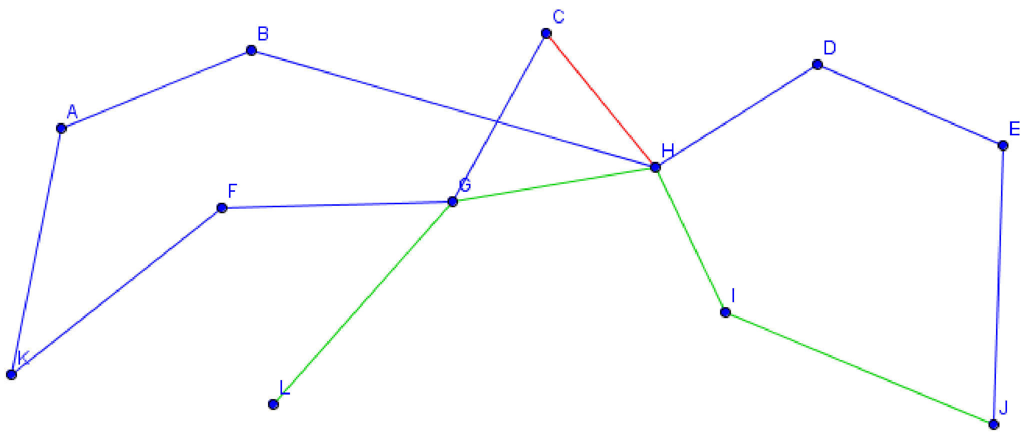

Figure 4.11 Test result: 12 station network

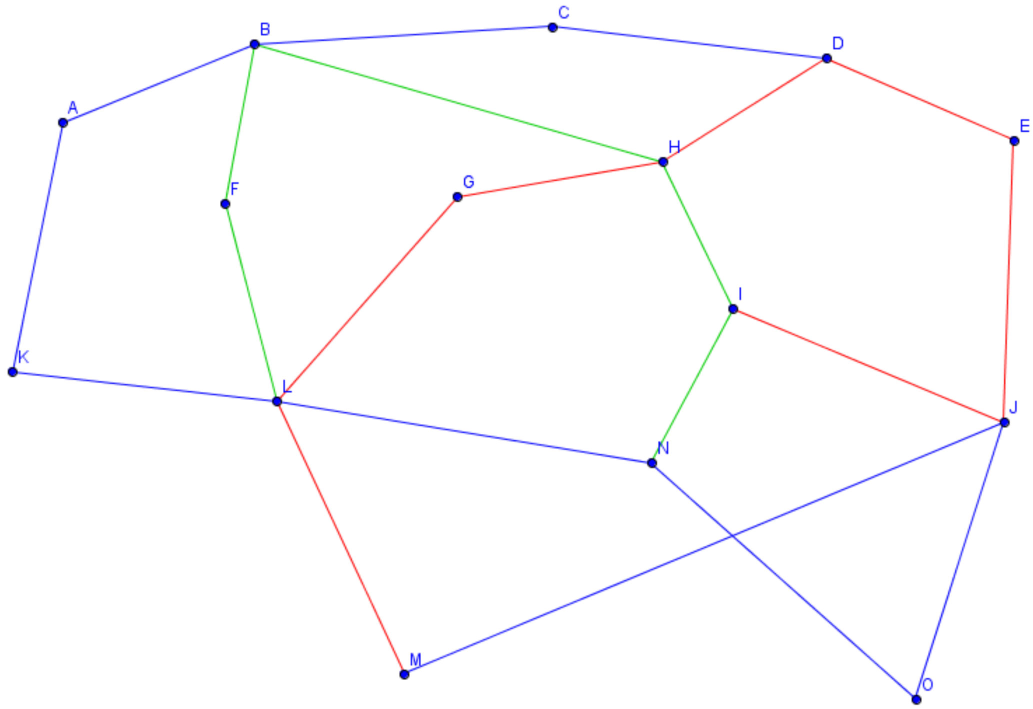

Figure 4.12 Test result: 15 station network

Table 4.3 Computational time

	Computational time (seconds)
5-station problem	3
10-station problem	821.2
12-station problem	5634
15-station problem	18000 (Preset Time Limit)

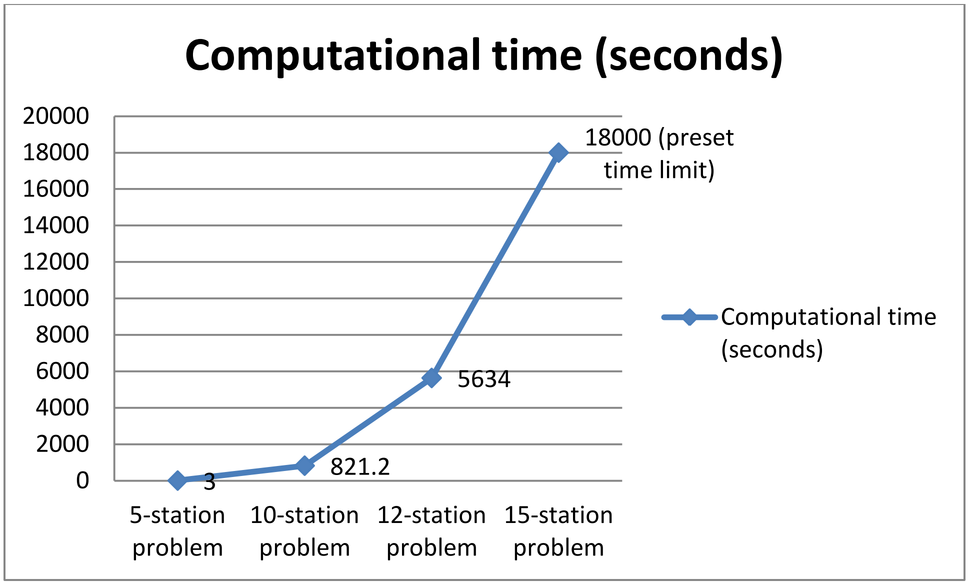

Figure 4.13 Computational time

We notice that the computational time of solving the comprehensive UFP network design model increases in an exponential fashion as the number of stations increases. For a 15-station problem, even though the commercial solver GUROBI® was able to generate a feasible solution after running for 5 hours, the optimality gap remains 5.29%, comparing with the default optimality gap value of 0.01%. There is still plenty of room to reach the optimal solution.

The fact that the comprehensive UFP network design model is hard to solve for large size real-world problem has motivated us to find a better way of modeling in order to greatly reduce the computational effort but meanwhile remain the ability of generating the optimal or near-optimal network design solution.

4.2.2 UFP Network Design Two-Step Model

As previously stated and also shown in Figure 4.14, the comprehensive UFP network design model has three decisions to make: edge selection, cargo route selection and line assignment. All these decisions are based on the balancing and minimizing the total cost which includes edge construction cost, transportation cost, station construction cost and transfer cost in transfer stations.

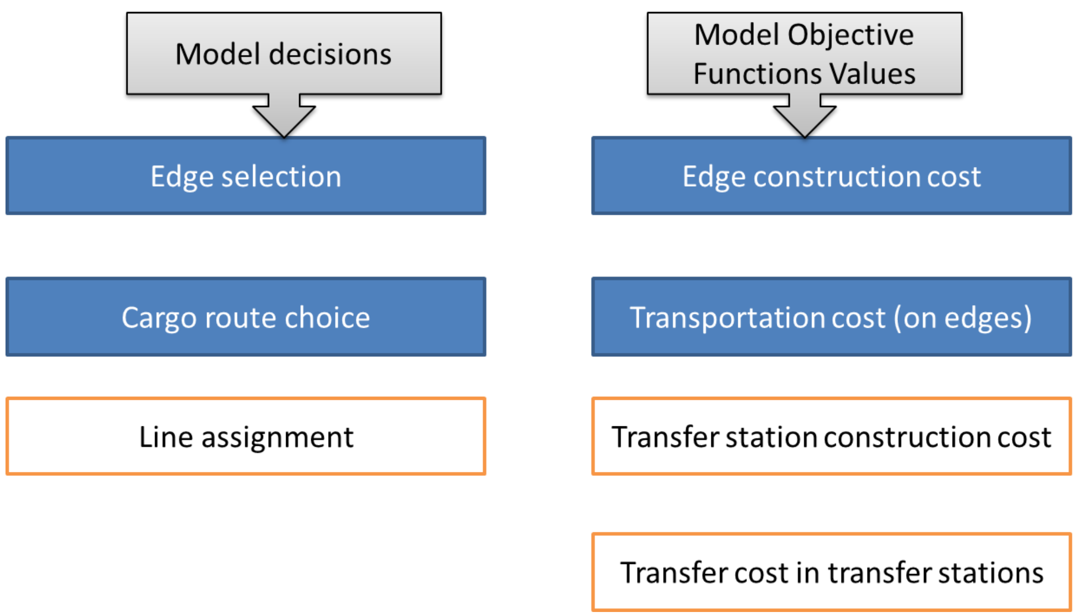

Figure 4.14 Model decisions VS. Objective function values

Intuitively, instead of considering all decisions and costs components together, one can solve the problem in two consecutive steps. As illustrated in Figure 4.15, in Step one, we only consider the decisions of edge selection and cargo route choice, and the optimal solution is achieved by minimizing the edge construction cost and transportation cost on tunnel edges. After Step one, we obtain a set of tunnel edges out of the pool of all possible edges. Then in Step two, taking the output of the step one as the input, we then consider assigning tunnel edges into different transit lines with respect to the minimization of the transfer station construction cost as well as the cargo transfer cost in transfer stations. For the ease of illustration, we call this way of modeling as *Intuitive Two Step Model*.

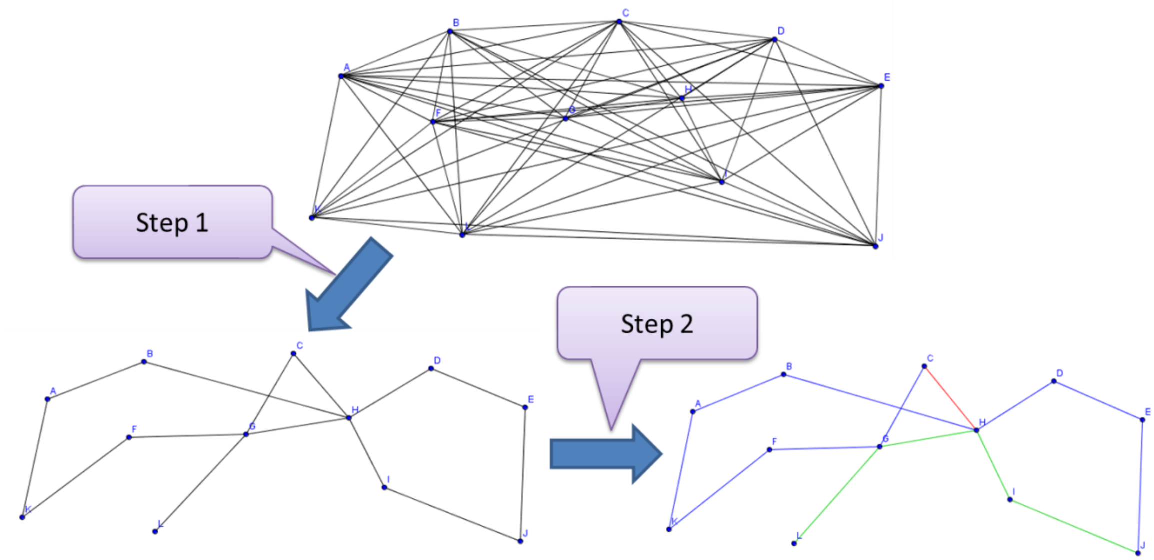

Figure 4.15 Intuitive two step modeling

However, even though *Intuitive Two Step Model* can greatly reduce the computational effort, it is lack of the ability to optimize the network comprehensively. There is a missing link between these two steps: the transfer station construction cost. The transfer station construction cost does affect the decision of selecting which tunnel edges to build. Here is an example: shown in Figure 4.16 and 4.17 is a 13-station node UFP network in New York area connecting major sea ports with nearby industrial districts. We consider two scenarios for this network. We assumed that all the parameters are the same for these two scenarios except the transfer station construction cost. As given in Table 4.4, in scenario 1, the unit station construction cost per transit line is $10 million. In scenario 2, this number is $13 million. The network configurations generated from the comprehensive UFP network design model based on these two scenarios are slightly different around the station node H. As shown in Figure 4.16, in scenario 1, there are three lines intersect in station node H. Yet, in scenario 2, as shown in Figure 4.17, two lines intersect in station node H. Note that it is the assumption in this research that more transit lines intersect at one station node results in a more complex station structure, which is translated into higher construction cost. The difference between these two scenarios is explained that the UFP network design model is trying to find the best trade-off between the increases of all other cost components and the decrease of transfer station construction cost. In this example, as the unit station construction cost increases from $10 million to $13 million, the number of intersected transit lines in station node H decreased from 3 to 2, even though this results in the increase of both edge construction cost and the transportation cost. In Table 4.5 is the summary of all cost components.

The example presented above shows that the station construction cost does affect the overall network configuration. Therefore, it is necessary to add certain enhancement to the *Intuitive Two Step Model* to make sure that the contribution of station construction cost to the overall network design is considered.

Table 4.4 Different unit station construction cost

	Scenario 1	Scenario 2
Unit station construction cost	$10 million	$13 million

Table 4.5 Analysis of station construction cost: Cost components comparison

	Scenario 1	Scenario 2
Edge construction cost	868749515	874133792
Transportation cost	1051133306	1051854479
Cargo transfer cost	55346500	52205500
Station construction cost	180000000	221000000

Figure 4.16 Scenario 1: Unit station construction cost equals $10 million

Figure 4.17 Scenario 2: Unit station construction cost equals $13 million

4.2.2.1 Enhanced UFP Network Design Two Step Models

The Enhanced *UFP Network Design Two Step Model* presented in this sub-section is based on the *Intuitive Two Step Model* illustrated in the previous section but with an important enhancement which estimates the approximate number of transit lines intersected at the same station node based on the degrees of each station node. The two consecutive models, named *Step One Model* and *Step Two Model*, are illustrated as the following. The details of how the enhancement works are illustrated in the *Step One Model*, specifically, constraints (7), (8) and (9).

Among two consecutive models, *Step One model* decides which tunnel edges are constructed and makes sure that cargo always travels the shortest path from its origin station to its destination station. It minimizes the total tunnel construction cost, total cargo transportation cost and the approximated total station construction cost. *Step Two model* takes the output of *Step One model* as the input and assign each tunnel edges into transit lines, and decides how many transfer stations are to be constructed with the objective function of minimizing the total transfer cost in transfer stations.

Step One Model:

<u>Sets:</u>

N: Station nodes n

E: Edges e connecting station node i and j

Parameters:

c_{ij}^{c}: Construction cost of edge (i, j)

c_i: Construction cost of station i

c_{ij}^{T}: Unit transportation cost on edge (i, j)

n_{uv}: Number of cargo from station u to station v

Decision variables:

$$x_{ij} = \begin{cases} 1 \text{ if edge (i, j) belongs to a line} \\ 0, otherwise \end{cases}$$

$$y_i = \begin{cases} 1, \text{if a station of is built at node i} \\ 0, otherwise \end{cases}$$

$$z_{uvij} = \begin{cases} 1, \text{if cargo from station u to v use edge (i, j)} \\ 0, otherwise \end{cases}$$

a_i: Approximated number of additional stations (integer) needs to be built at station node i.

For instance, if there is only one transit line passing through node i, then $a_i = 0$, which indicates that node i is just a regular station. If there are two transit lines intersect at node i, then $a_i = 1$ which denotes one additional station needs to be built at node i.

Objective function:

Minimize:

$$\sum_{(i,j)\in E} c_{ij}^{C} x_{ij} + \sum_{l\in L} \sum_{i\in N} c_i y_i + \sum_l \sum_u \sum_v \sum_{(i,j)} c_{ij}^{T} n_{uv} z_{uvij} + \sum_i c_i a_i \quad (1)$$

Subject to:

$$x_{ij} \leq y_i, \forall (i,j) \in E \quad (2)$$

$$x_{ij} \leq y_j, \forall (i,j) \in E \quad (3)$$

$$x_{ij} = x_{ji} \quad (4)$$

$$x_{ij} \geq z_{uvij}, \forall (i,j) \in E, u \in N, v \in N, u \neq v \quad (5)$$

$$\sum_j z_{uvij} - \sum_j z_{uvji} = \begin{cases} 1 \; if \; i = u \\ -1 \; if \; i = v \\ 0 \; otherwise \end{cases} \quad (6)$$

$$d_i = \sum_j x_{ij}, \forall i \in N \quad (7)$$

$$a_i \geq \frac{1}{2}(d_i - 2), \; \forall i \in N \quad (8)$$

$$a_i \leq \frac{1}{2}(d_i - 1), \; \forall i \in N \quad (9)$$

The objective function (1) has two major components: construction cost and operational cost.

$\sum_{(i,j)\in E} c_{ij}^{C} x_{ij}$ denotes the total tunnel construction cost for all edges that are included in the final network configuration.

$\sum_{l\in L}\sum_{i\in N} c_i y_i$ is the station construction cost. Note that the cost of constructing a station at node i is counted for each line to which the node v belongs, which corresponds to the fact that the construction cost of a transfer station is higher than a regular station.

$\sum_l \sum_u \sum_v \sum_{(i,j)} c_{ij}^{T} n_{uv} z_{uvij}$ is the transportation cost occurred as cargo moves through the selected tunnels (edges).

$\sum_i c_i a_i$ is the approximated transfer station construction cost.

Constraint (2) and (3) ensures that edge (i, j) cannot be part of the network if no station is built at node i and node j. Constraint (4) ensures that the constructed tunnel in edge (i, j) is bidirectional.

Constraint (5) ensures that cargo be transported through edge (i, j) only if edge (i, j) is constructed.

Constraints (6) is inherited from the classic Shortest Path Problem to ensure that all cargo follow the shortest path from its origin station node to its destination station node. Three parts as illustrated below:

- For the origin node u, the difference in the number of edges leaving u and entering into u is 1
- For the destination node v, the difference in the number of edges entering into v and leaving v is 1
- For all other nodes, the number of edges of outgoing and incoming are equal

Constraint (7) calculates the degree of each station node.

Constraint (8) and (9) calculate the approximated number of transfer stations of each node based on the number of degree of each station node, which works as the following:

Figure 4.18 Node degree equals 2

As shown in Figure 4.18, the degree of the node is 2 ($d_i = 2$), then constraint (8) and (9) are:

$a_i \geq 0$

$a_i \leq \frac{1}{2}$

Then: $a_i = 0$, which denotes that this node is not a transfer station but a regular station.

Figure 4.19 Node degree equals 3

As shown in Figure 4.19, the degree of the node is 2 ($d_i = 3$), then constraint (8) and (9) are:

$a_i \geq \frac{1}{2}$

$a_i \leq 1$

Then: $a_i = 1$, which denotes that this node is a transfer station. Thus, one additional station needs to be built at node i.

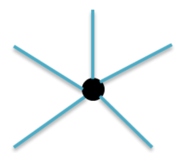

Figure 4.20 Node degree equals 5

As shown in Figure 4.20, the degree of the node is 5 ($d_i = 5$), then constraint (8) and (9) are:

$a_i \geq \frac{3}{2}$

$a_i \leq 2$

Then: $a_i = 2$, which denotes that this node is a transfer station and two additional stations needs to be built at node i.

To summarize, a_i is calculated as the floor of half the number of degrees of node i ($\left\lfloor \frac{d_i}{2} \right\rfloor$) and is used to estimate the number of transfer stations needs to be built at node i.

Step Two Model:

Sets:

N: Station nodes n

L: Line l

E: Edge e, a set of tunnel edges selected by Step One Model

Parameters:

c^o: Unit operational cost in transfer station

n_{uv}: Number of cargo from station u to station v

z_{uvij}: = 1 if cargo from station u to station v uses edge (i, j), the value of this parameter is decided by the output from Step One Model

Decision variables:

$$x_{ij}^{l} = \begin{cases} 1, \text{if edge (i, j) belongs to line l} \\ 0, otherwise \end{cases}$$

$$y_{i}^{l} = \begin{cases} 1, \text{if a station of line l is built at node i} \\ 0, otherwise \end{cases}$$

$$b_{uvi}^{l} = \begin{cases} 1, \text{cargo from station u to v use line l either entering} \\ \quad \text{node i or leaving node i, but not both} \\ 0, otherwise \end{cases}$$

$$x_{uvi}^{T} = \begin{cases} 1, \text{if cargo from station u to v is transfered in node i} \\ 0, otherwise \end{cases}$$

Objective function:

Minimize:

$$\sum_{l \in L} \sum_{i \in N} c_i^l y_i^l + c^o \sum_u \sum_v \sum_i n_{uv} x_{uvi}^T + \sum_u \sum_v \sum_i \sum_l b_{uvi}^l \tag{1}$$

<u>Subject to:</u>

$$x_{ij}^l \leq y_i^l, \forall (i,j) \in E, l \in L \tag{2}$$

$$x_{ij}^l \leq y_j^l, \forall (i,j) \in E, l \in L \tag{3}$$

$$x_{ij}^l = x_{ji}^l \tag{4}$$

$$\sum_{j \in N, j \neq i} x_{ij}^l \leq 2 \ \forall i \in N, \forall l \in L \tag{5}$$

$$\frac{1}{2} \sum_{\{i,j\} \in E} x_{ij}^l \geq \sum_{i \in N} y_i^l - 1, \forall l \in L \tag{6}$$

$$b_{uvi}^l \geq \sum_j z_{uvji}^l - \sum_j z_{uvij}^l, \forall u, v \in N, u \neq v, \forall i \in N, i \neq u, v, \forall l \in L \tag{7}$$

$$b_{uvi}^l \geq \sum_j z_{uvij}^l - \sum_j z_{uvji}^l, \forall u, v \in N, u \neq v, \forall i \in N, i \neq u, v, \forall l \in L \tag{8}$$

$$x_{uvi}^T = \frac{1}{2} \sum_l b_{uvi}^l, \forall u, v \in N, u \neq v, \forall i \in N, i \neq u, v \tag{9}$$

The objective function (1) minimizes the total transfer cost in transfer stations.

$\sum_l \sum_u \sum_v \sum_{(i,j)} c_{ij}^T n_{uv} z_{uvij}^l$ is the transportation cost occurred as cargo moves through the selected tunnels (edges).

$c^o \sum_u \sum_v \sum_i n_{uv} x_{uvi}^T$ is the operational cost in transfer stations.

Constraint (2) and (3) ensures that edge (i, j) cannot be part of the network if no station is built at node i and node j. Constraint (4) ensures that the constructed tunnel in edge (i, j) is bidirectional. Constraints (5) forces that the degree of each node is 0, 1 or 2. Constraints (5) and (6) together imply that each line is a collection of continuous edges which is either a single line or a loop line.

Since constraint (6) defines the relation between the number of stations and links of each line. Constraints (7), (8) and (9) make sure the transfer operations are correctly calculated even in case that non-connected lines are in place.

Table 4.6 Improved computational performance

	Computational time (seconds)	
	Comprehensive model	Two steps model
5-station problem	3	0.21
10-station problem	821.2	0.34
12-station problem	5634	0.51
15-station problem	18000	2.8

Recall the discussion in section 4.2.1.1 regarding to the computational time of the UFP *comprehensive network design model*, same testing problems are used to test the computational performance of the *UFP network design Two Step model*. The computational time comparison between these two modeling approaches is listed in Table 4.6. It is very clear that the UFP *Two Steps Model* outperforms the UFP *Comprehensive Model* tremendously in terms of computational time, which makes the *Two Steps Model* a better choice for larger sized UFP network design problem.

Table 4.7 Result comparison: Comprehensive model vs. Two-step model

	Total Cost	
	Comprehensive model	Two step model
5-station problem	235894226	235894226
10-station problem	392995732	392995732
12-station problem	360362624	360362624
15-station problem	618797336	615835610

Yet, the more important issue is whether the *Two Step Model* is able to generate a good UFP network design. In Table 4.7 summarizes the total cost values of 4 test problems from both the *Comprehensive Model* and the *Two Steps Model*. For the test problems with 5, 10 and 12 station nodes, both *Comprehensive Model* and *Two Steps Model* generate exact the same network configuration as shown in Figure 4.9, 4.10 and 4.11. For the test problem with 15 station nodes, *Two Steps Model* generated a better network design with the objective function value of 615835610 (Total Cost). The *Comprehensive Model* could only generate a feasible solution with the objective function value of 618797336 after 18000 seconds of computational time in GUROBI®.

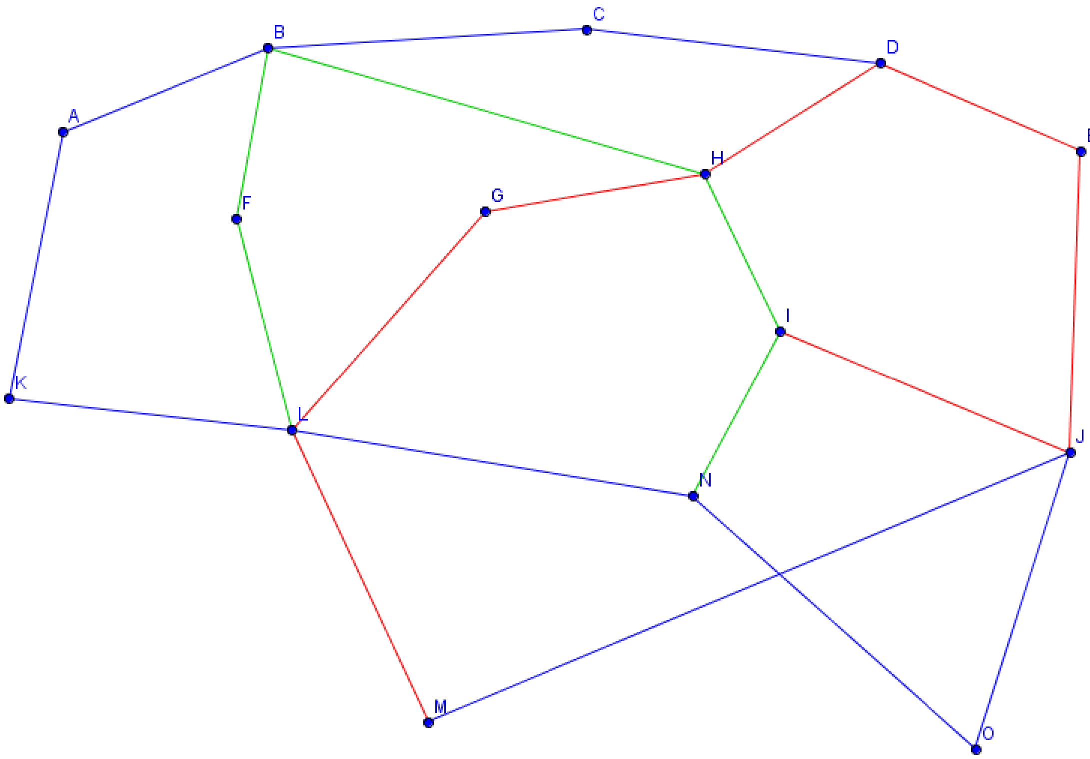

Figure 4.21 15-station network design result from Comprehensive Model

Overall, the experiment shows that the UFP *Two Steps Model* is not only able to solve the UFP network design problem with much less computational time but also maintain the ability of generating decent network design solutions.

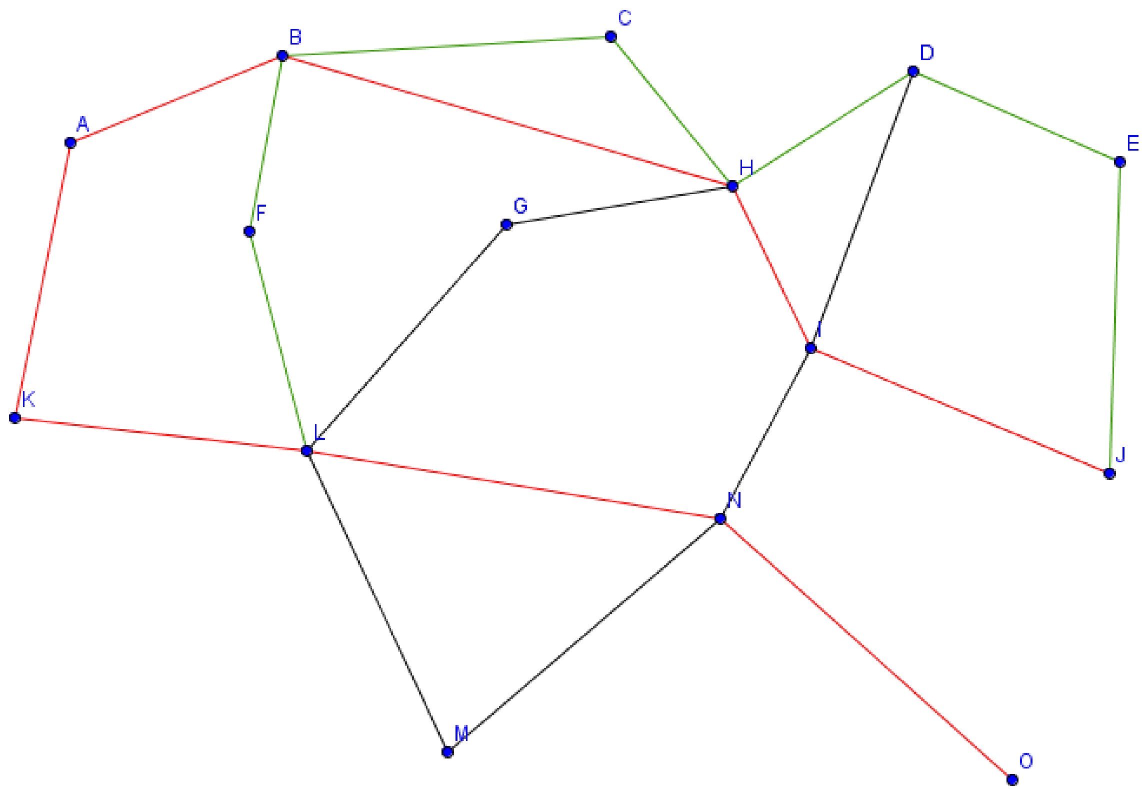

Figure 4.22 15-station network design result from Two-step Model

4.3 A case study: New York UFP network:

Among the potential applications of UFP system proposed by Liu (2004), which include pallet-tube system for transporting pallets and other goods, dispatching containers, truck-ferry system, solid waste transportation, and mail and parcel transportation, dispatching containers attracts the most attentions. Such a system was not only proposed for New York City but similar systems are also proposed to Shanghai, China and Sydney (Sydney Freight Circle), Australia. In this research, we extend Liu's proposal of dispatching containers (2004) to a more complex UFP system that connects the major ocean ports of

New York City with the regions where major industrial districts and distribution centers in the Greater New York Region are located.

New York City has some of the nation's busiest ports, with thousands of containers lying on the waterfront waiting to be shipped either to inland places by trucks and trains, or to be loaded on outbound ships. The presence of such large numbers of idled containers at any harbor wastes the precious space at the busy harbor. And there is the ever increasing environmental damage and pollution due to diesel truck/train surface container movement from and to the ports which also are making the metro area more and more congested. The dilemma can be solved by constructing a UFP system to dispatch the incoming containers to their individual destinations within New York City metro areas and vice versa. The UFP system will have immense value to New York City. It helps to reduce the number of trucks that clog the city's streets and reduce air pollution, noise and accidents generated by trucks. It provides economic development and creates jobs. Such a system is able to transport containers much faster and more reliable than trucks can as it is unaffected by inclement weather, traffic jam etc. and it also increases safety and security.

In the remaining of this chapter, we consider a network in the Greater New York region and the following section illustrates the critical parameters in this network and how they are calculated.

4.3.1 New York City UFP network parameters

As shown in Figure 4.23, there are in total 24 station nodes which are classified into 2 categories. 4 station nodes colored in red (node A, B, C and D) are the locations where the major ocean ports are located. Other nodes colored in blue are the locations where the major industrial districts and distribution centers are located. For the ease of illustration, we named the red station nodes as ocean port stations and the blue station nodes as customer station nodes.

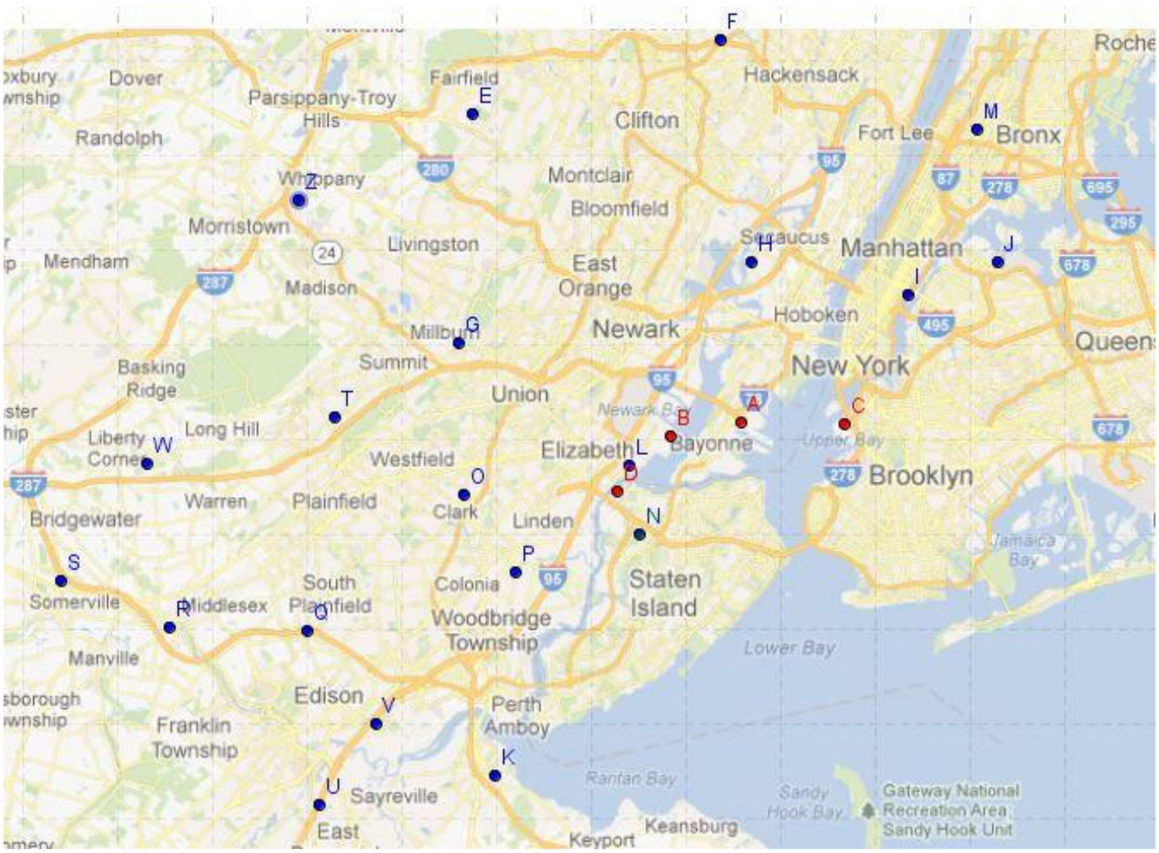

Figure 4.23 Twenty four station nodes in Greater New York

It is reported that approximately 3,200,000 TEU, a shipping measurement that stands for Twenty-foot Equivalent Units, are handled in 2011 in the port of New York and New Jersey. During the minimum life span of the UFP system of 30 years, with a conservative 2% increase of TEU quantity increase per year, we project that approximately 130,000,000 TEUs will be handled by this UFP system.

Liu (2004) proposed that the UFP system for container dispatching will require the use of large underground tunnels or conduits. For the region near the port and in urban areas, a round tunnel bored in hard bedrock 100 to 150 feet below the water level is required. As soon as the tunnel reaches rural areas, it should rise to only 5 feet below ground level and then change to a rectangular cross section which can be more easily and economically constructed. Liu (2004) also presented the way to calculate and estimate the per-mile construction cost of both the underground tunnel and conduit.

For the tunnel near the port and in urban areas, the construction cost is based on the following formula:

$$C = KD^N \tag{1}$$

In equation (1), the quantity C is the construction cost in dollars per linear foot ($/ft) of the underground tunnel. K is a constant determined from existing cost data. D is tunnel diameter in feet. And N is a constant greater than 1 but less than 2. All these parameters are provided by Liu (2004) and the equation is:

$$C = 78D^{1.7} \tag{2}$$

Using equation (2), the cost of constructing a tunnel of 15-ft diameter for dispatching containers is \$7790/ft or \$41million/mile, approximately.

For the tunnel in the rural area, the large rectangular reinforced concrete conduits are used. And the cost of unit length of such prefabricated conduit is calculated based on the following formula:

$$C = KB^N \tag{3}$$

In equation (3), C is the unit cost in dollars per linear foot (\$/ft), K and N are different constants, and B is the size of the rectangular conduit which is defined as the arithmetic mean of the width, w and the height, h, of the rectangle, namely,

$$B = \frac{w+h}{2} \tag{4}$$

For the underground conduits for dispatching containers, with the parameters provided by Liu (2004), the equation (3) is:

$$C = 12.7 * 10^{1.7} \tag{5}$$

Using equation (5), the cost of laying the prefabricated conduit is \$3.36million/mile.

4.3.2 New York UFP network analysis:

In this section, a series of experiments and analysis are conducted to provide more insights of how different parameters and variables affect the strategic planning of the UFP system for dispatching ocean containers in New York area, which includes analysis of container flow pattern, analysis of varied station construction cost and analysis of central inspection stations.

4.3.2.1 Analysis of container flow pattern

As stated previously, this UFP system is designed for dispatching containers in and out of the major ocean ports in New York and New Jersey. A good network configuration of such a system must provide a fast, convenient, and cost effective way of moving cargo containers for the customers. In reality, it is common that some customers have higher volume of container flow due to larger business size etc. Thus, the question is that how the network configuration accommodates the cargo flow pattern and provides transportation convenience to customer with more container flow accordingly. In the following are the experiments of how container flows affect the network configuration.

Figure 4.24 shows the result of the base scenario, in which we assume that container from and to the four ocean-port stations A, B, C and D are evenly distributed to all customer stations. The result network configuration of the base scenario is consists of 7 transit lines as shown in Figure 4.24 in different colors. For instance, the red line starts from station node C and ends at station node U.

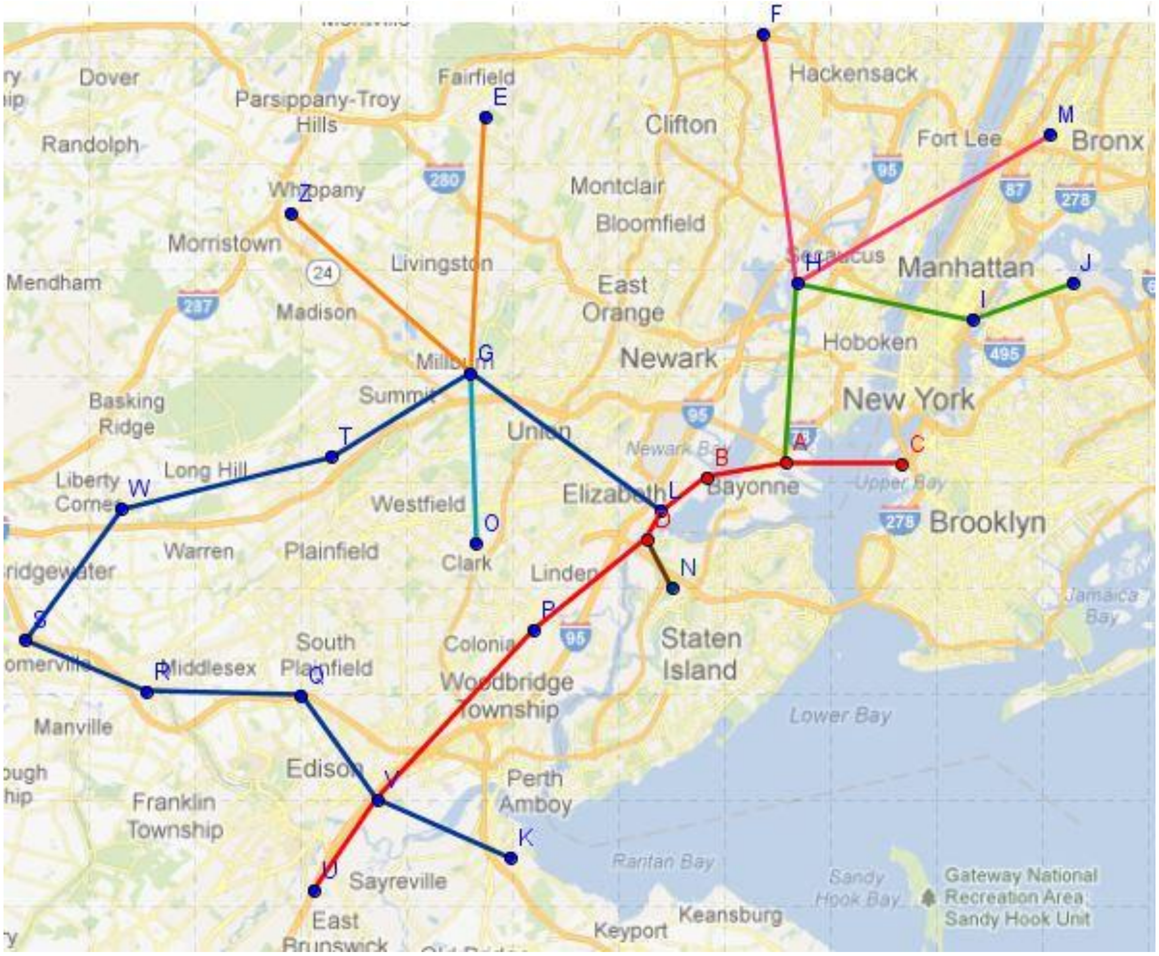

Figure 4.24 Analysis of container flow pattern: Base scenario

To compare with the base scenario, more experiments are conducted by varying the cargo flow. In each of the following scenarios, we assume that 30% of the cargo flow from/to the ocean port nodes is sent to/from two or three geometrically adjacent stations. All scenarios are summarized in Table 4.8.

Table 4.8 Analysis of container flow pattern: Scenarios

	30% container flow from ocean port stations to Station
Scenarios 1	F, H
Scenarios 2	I, J, M
Scenarios 3	E, Z
Scenarios 4	G, T
Scenarios 5	Q, V
Scenarios 6	U, K
Scenarios 7	O, P
Scenarios 8	W, S, R

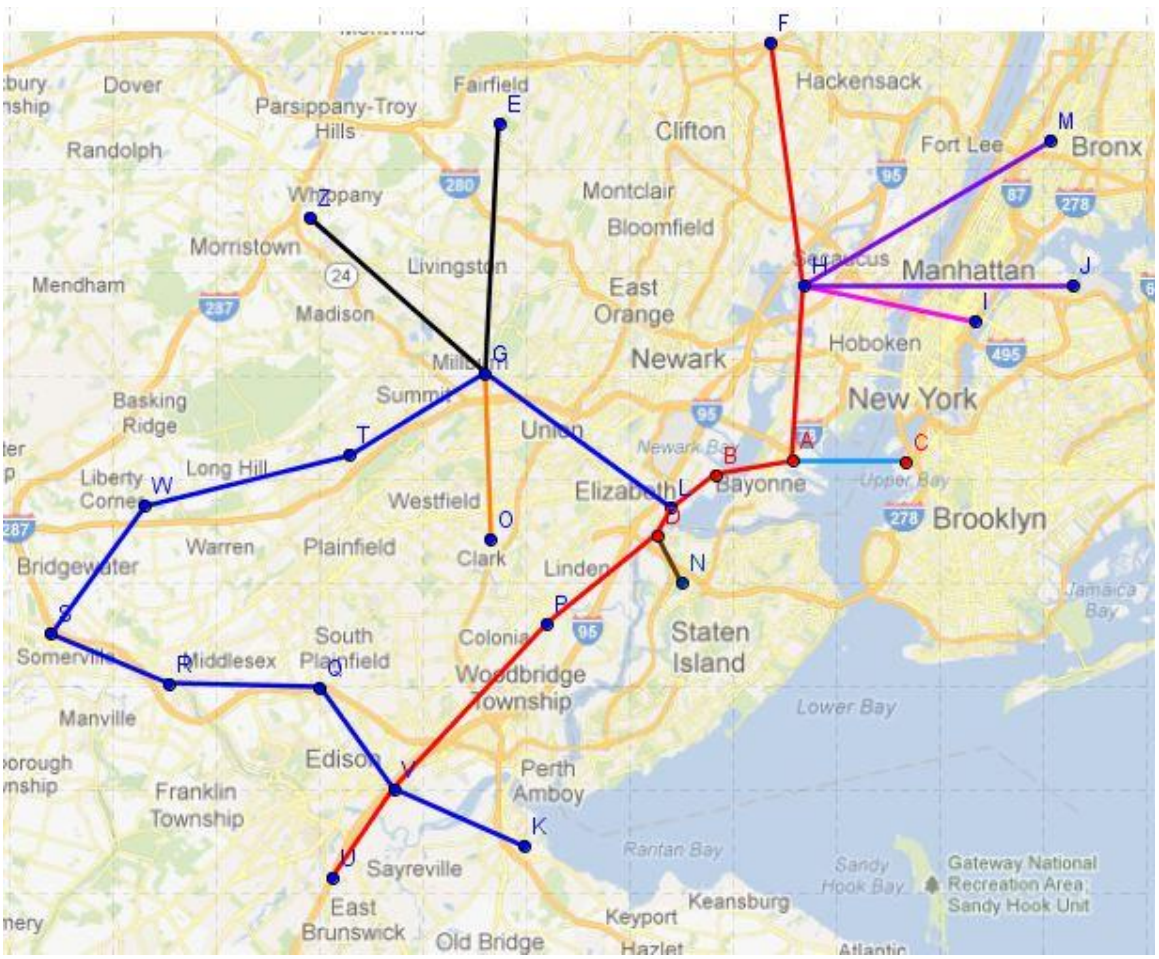

Figure 4.25 Analysis of container flow pattern: Scenario 1

In Figure 4.25 is the result network configuration of the scenario 1, in which 30% of container flow from/to the ocean port stations are sent to/from the customer station F and H. Notice that tunnel edge FH is assigned to the same transit line as the ocean port stations of A, B and D. In this way, customers in station F and H are directly connected to ocean port stations and no transfer operations are needed when container travels from ocean port stations to/from customer stations F and H, which is translated into more convenient and cost effective transportation service.

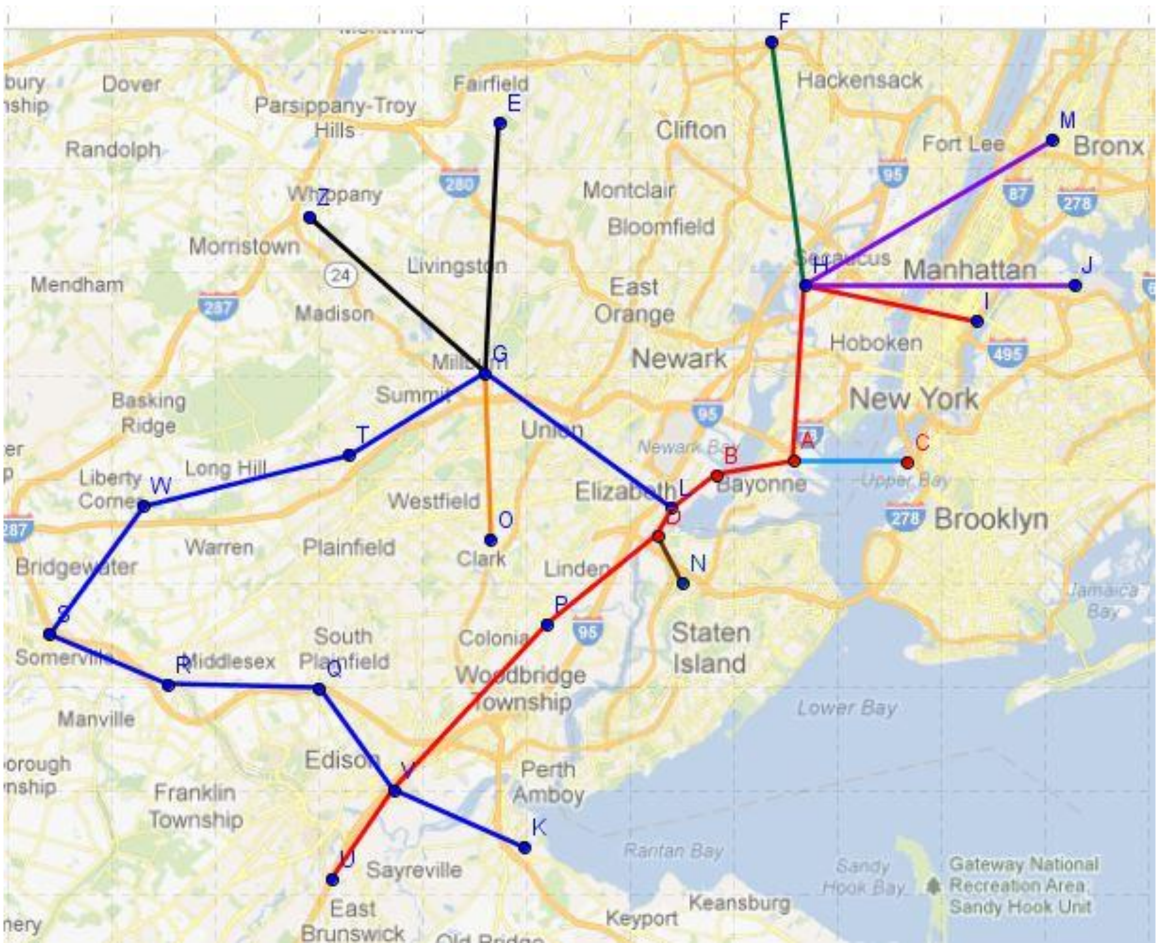

Figure 4.26 Analysis of container flow pattern: Scenario 2

The result network configuration of the scenario 2 is shown in Figure 4.26, in which station I, J and M account for 30% of the container flow between ocean port stations and the customer stations. Comparing with the base scenario in which station I, J, and M are all assigned to branch lines, in this scenario we notice that station I is assigned to the same transit line as the ocean port stations. And station J and M are assigned to a branch line that is directly connected to red line which also provides relatively more direct access to the ocean port stations.

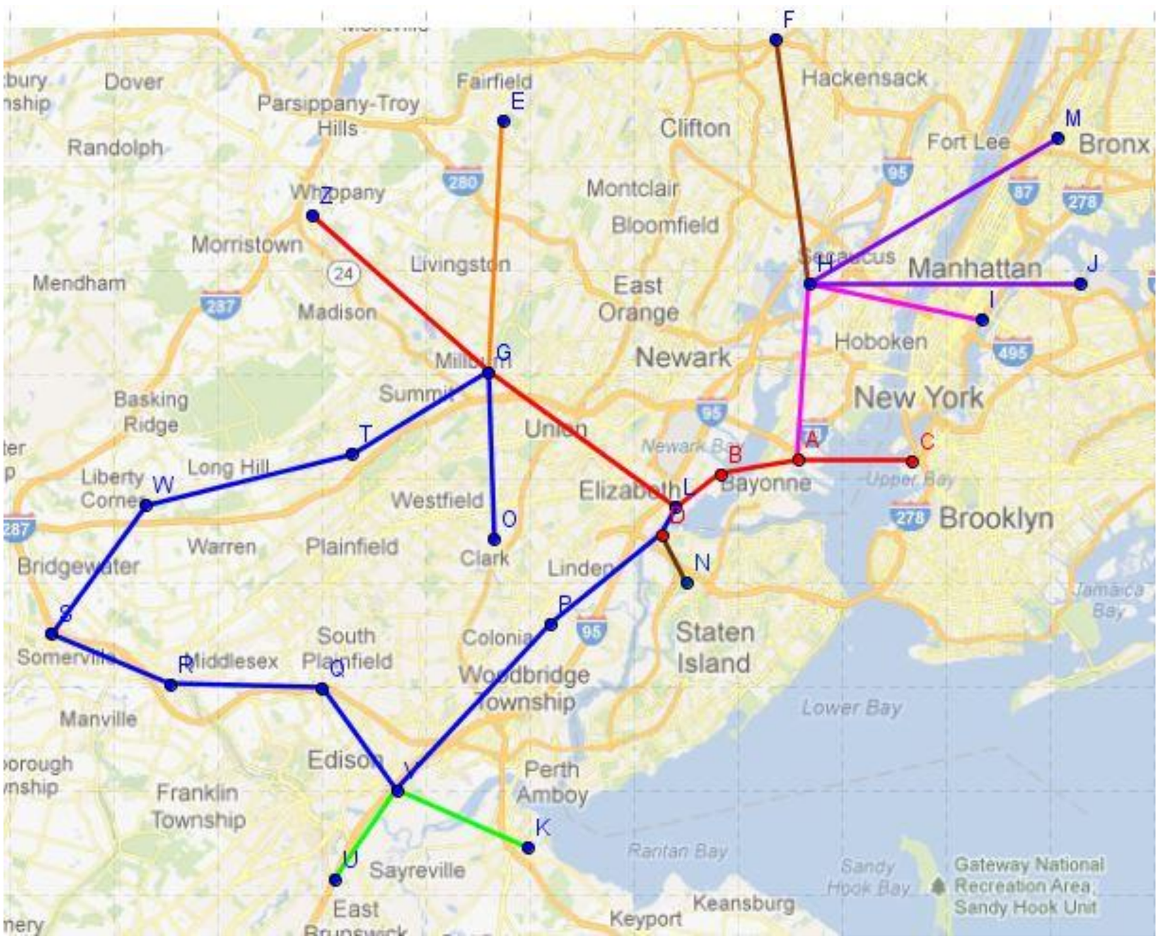

Figure 4.27 Analysis of container flow pattern: Scenario 3

In scenario 3, recall that station Z and E are the customer stations with major container flow. As shown in Figure 4.27, we notice that station Z is assigned to the same transit line as the ocean port stations and station E is in a branch line that is directly connected to the red line.

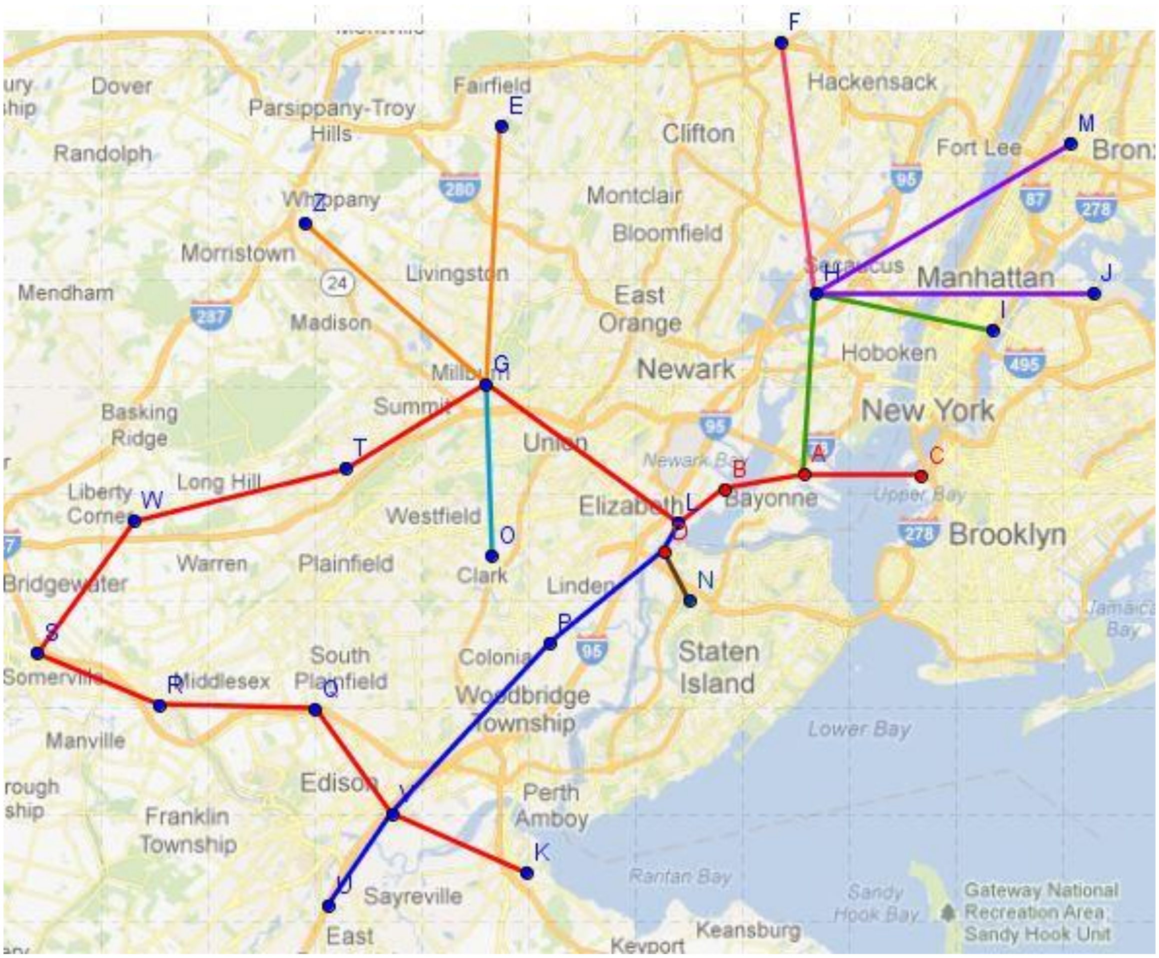

Figure 4.28 Analysis of container flow pattern: Scenario 4

In scenario 4, as station G and T has the major container flow, as shown in Figure 4.28, a long transit line is formed starting from ocean port station C, A and B all the way to station K, in which station G and T are also included and cargo from/to station G and T can reach the ocean port stations directly.

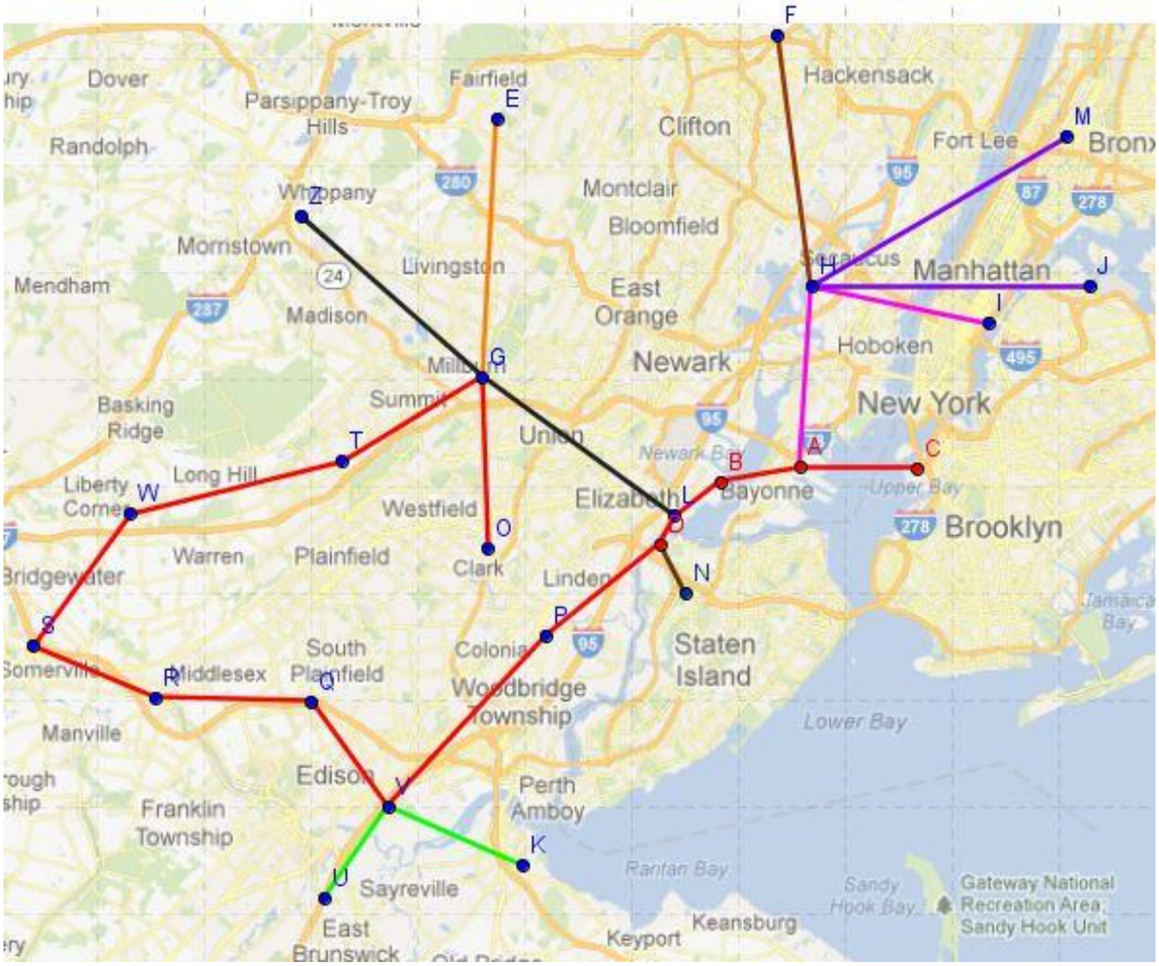

Figure 4.29 Analysis of container flow pattern: Scenario 5

Similarly, in scenario 5, since 30% of the cargo flow from/to the ocean port is shipped to/from station Q and V, as shown in Figure 4.29, these two stations are assigned to the same transit line as the ocean port stations.

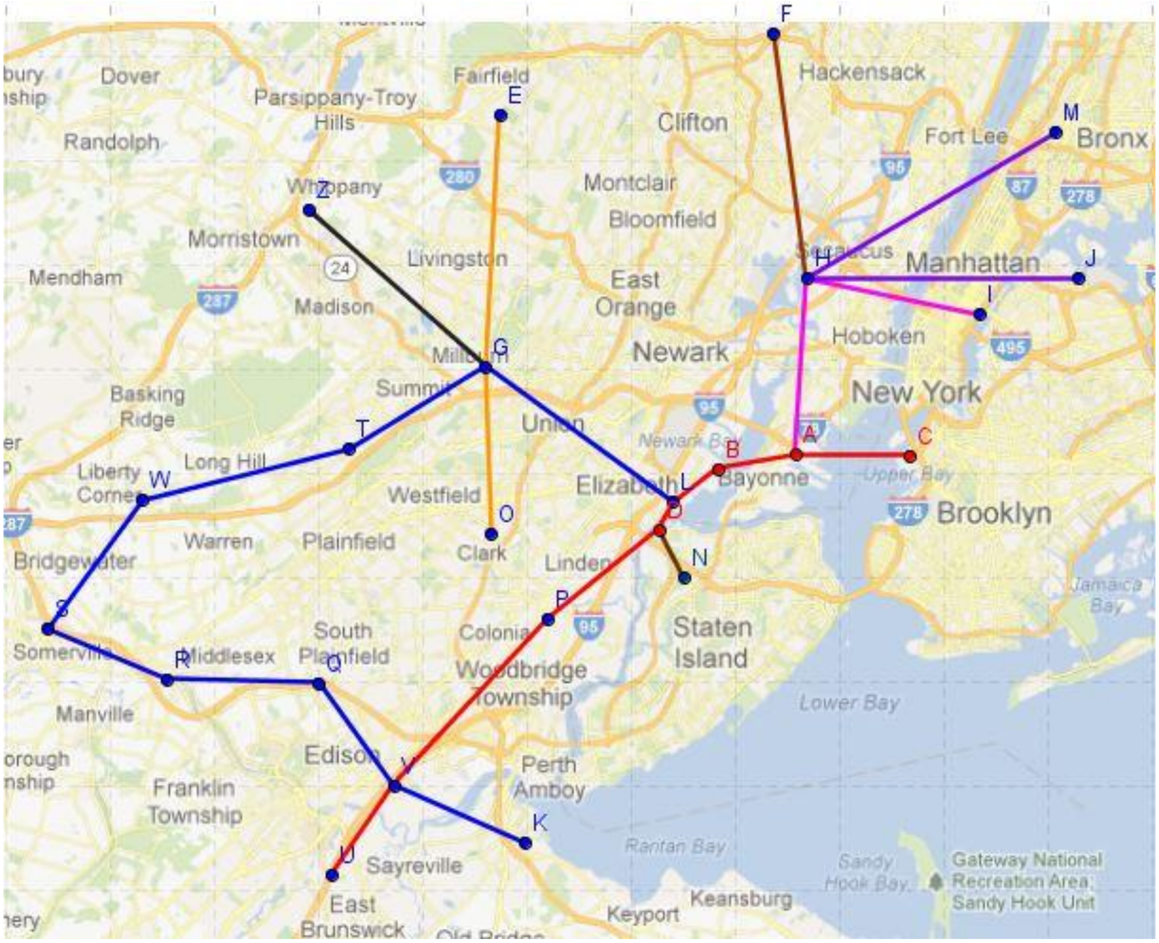

Figure 4.30 Analysis of container flow pattern: Scenario 6

In scenario 6, shown in Figure 4.30, station U is assigned to the same transit line as the ocean port stations, and station K is assigned to the blue transit line which is directly connected to the red transit line.

It is worth noticing that, among scenario 1 to scenario 6, the network structure does not change, in other words, all tunnel edges remain the same. What is changing is the transit line assignment. The UFP network design model assigns the tunnel edges to different lines to accommodate the changing cargo flow pattern so as to provide more direct and cost effective transportation service to the customers with larger container flow volume.

However, that is not always the case, in scenario 7 and 8, as shown in Figure 4.31 and 4.32, the changes of cargo flow pattern do affect the network structure.

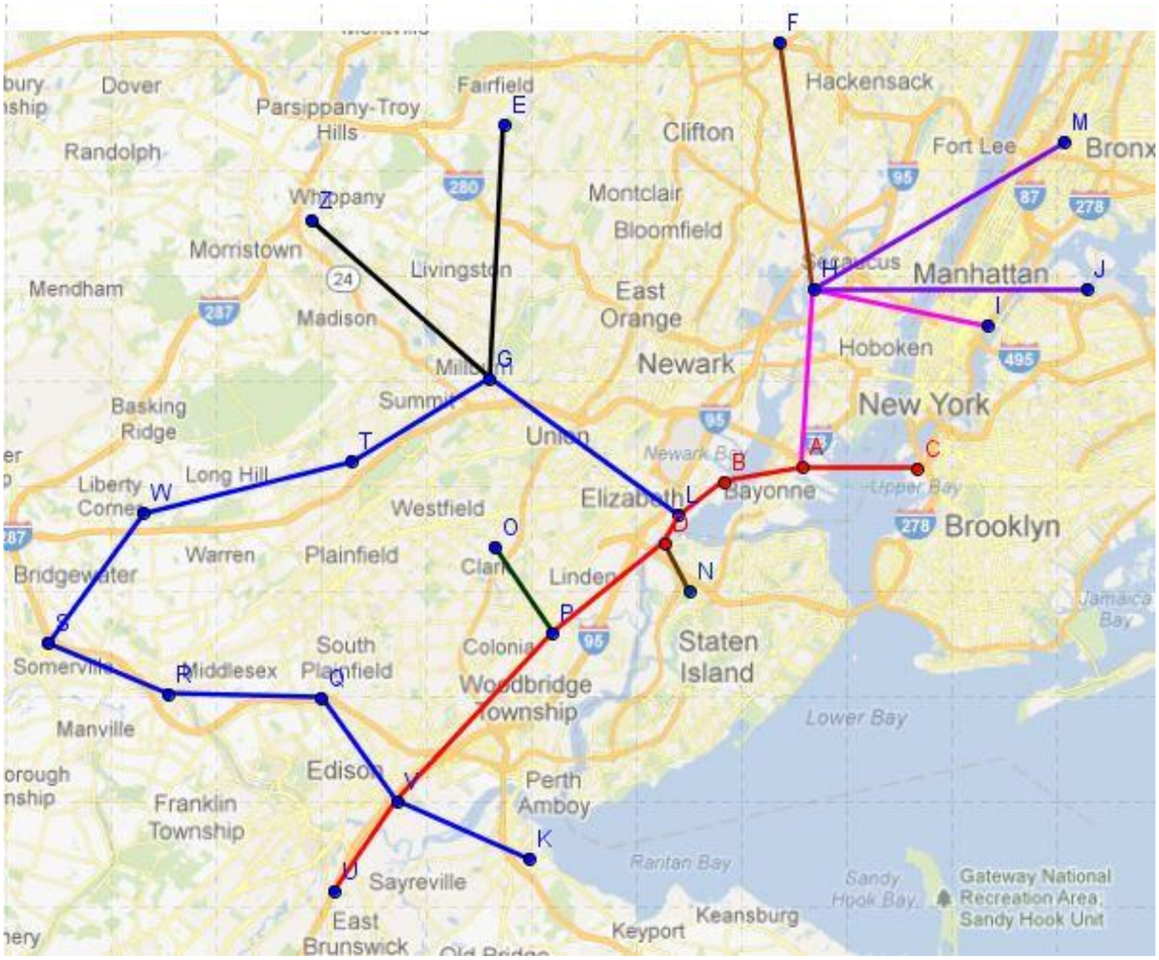

Figure 4.31 Analysis of container flow pattern: Scenario 7

In scenario 7, as 30% of the cargo flow from/to the ocean port is shipped to/from station O and P, comparing with previous 6 scenarios and the base scenario, station G and O is no longer directly connected. Instead, station O is connected to station P. In this way, the travel distances in and out of station O and P to ocean port stations are much shorter, which helps to bring down the transportation cost for the major container flow from station O and P to the ocean port stations.

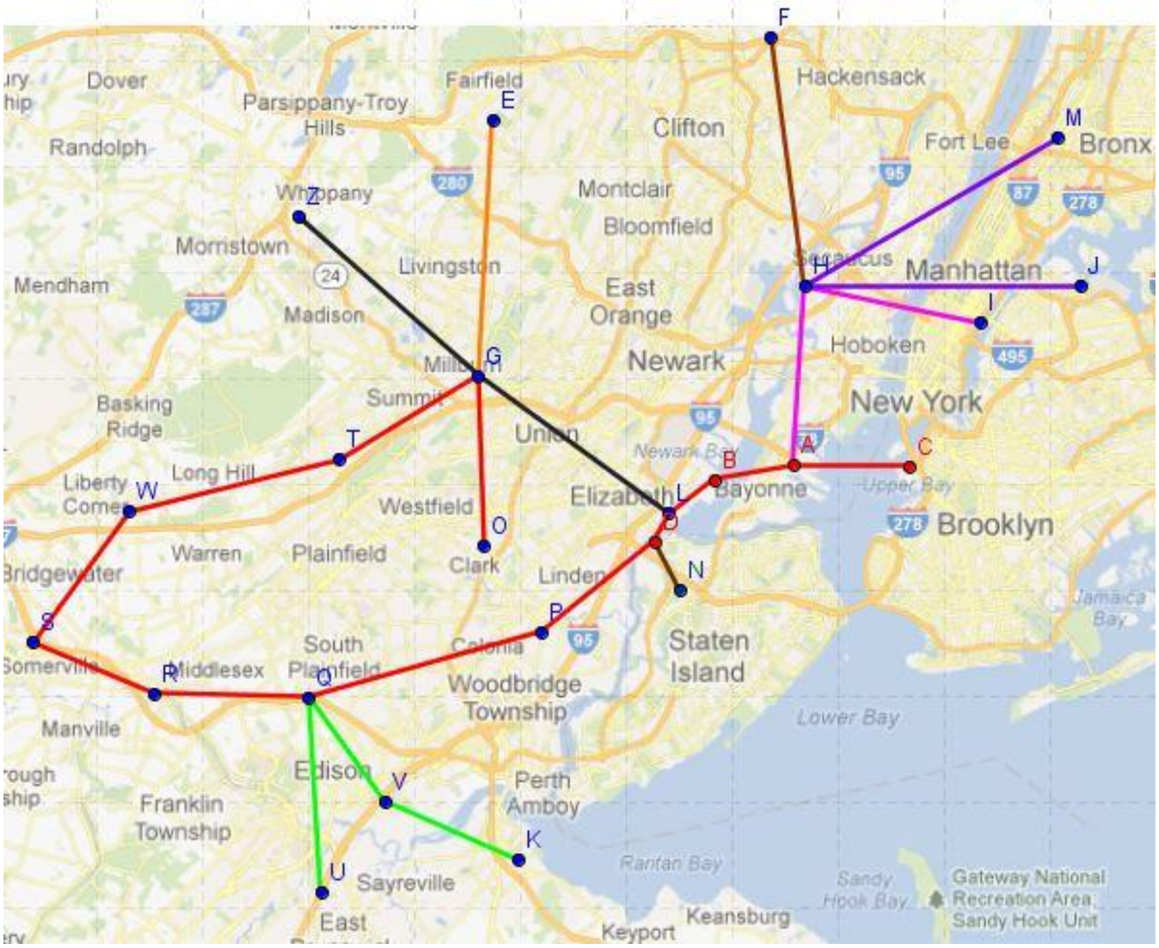

Figure 4.32 Analysis of container flow pattern: Scenario 8

In scenario 8, since station W, S and R account for 30% of the total cargo flow in and out of ocean port stations, these three stations are assigned to the same transit line as the ocean port stations. Additionally, in this scenario, the edge PV is no longer connected. Instead, station P and Q is connected. In such a way, the travel distance for cargo in and out of station W, S, R to the ocean port stations is reduced.

Overall, we analyzed a series of scenarios, each with a different container flow pattern. The container flow patterns are varied from evenly distributed from ocean port stations to customer stations, to the pattern that a specific region accounts for 30% of container flow to/from the ocean port stations. We see that the result network design, either the transit line assignment or the network configuration, changes accordingly to provide the customers the most direct access to the ocean port stations.

4.3.2.2 Analysis of per station construction cost

As discussed briefly in Section 4.2.2, the station construction cost does affect the overall network configuration. As the per station construction cost increases, the optimal network design needs to be adjusted to balance off the increasing station construction cost, possibly with fewer transfer stations, even if this may lead to the increase of other cost components. In this subsection, we present the scenario analysis of how the varied station construction cost affects the optimal network design. Note that we assume that in the case two or more transit lines intersects at the same station, this specific station is a transfer station which has a higher construction cost than a regular station.

In the following, four scenarios are presented, each with a different per station construction cost while other parameters remain the same. The varied unit station construction costs are summarized in Table 4.9.

Table 4.9 Scenarios of analysis of per station construction cost

	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 4**
Transfer station construction cost	$10 million	$13 million	$ 30 million	$ 35 million

Shown in Figure 4.33 is the result network design of scenario 1. Since 3 transit lines intersect at station G, we say that the number of intersected transit lines in station G is 3.

Then, total number of intersected transit lines in scenario is 14. The result network configuration of Scenario 2, 3 and 4 are shown in Figure 4.34, 4.35, and 4.36. It is noticed that as the per station construction cost increases, the number of intersected transit lines decreases, which is translated into fewer transfer stations and less total station construction cost. For all four scenarios, the numbers of intersected transit lines are summarized in Table 4.10.

Table 4.10 Number of intersected transit lines

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Number of intersected transit lines	14	13	11	10

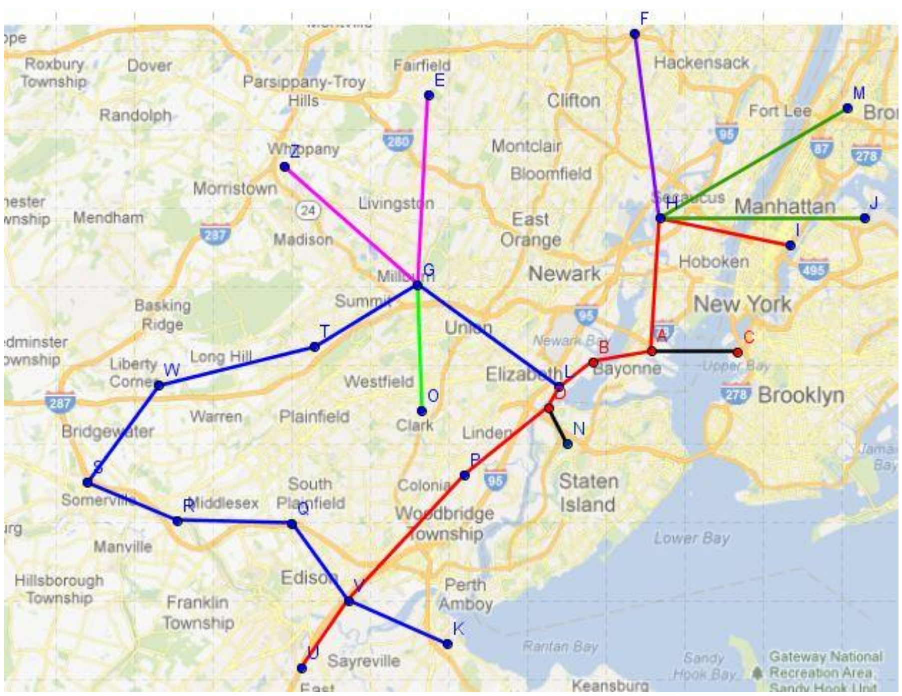

Figure 4.33 Analysis of varied station construction cost: Scenario 1

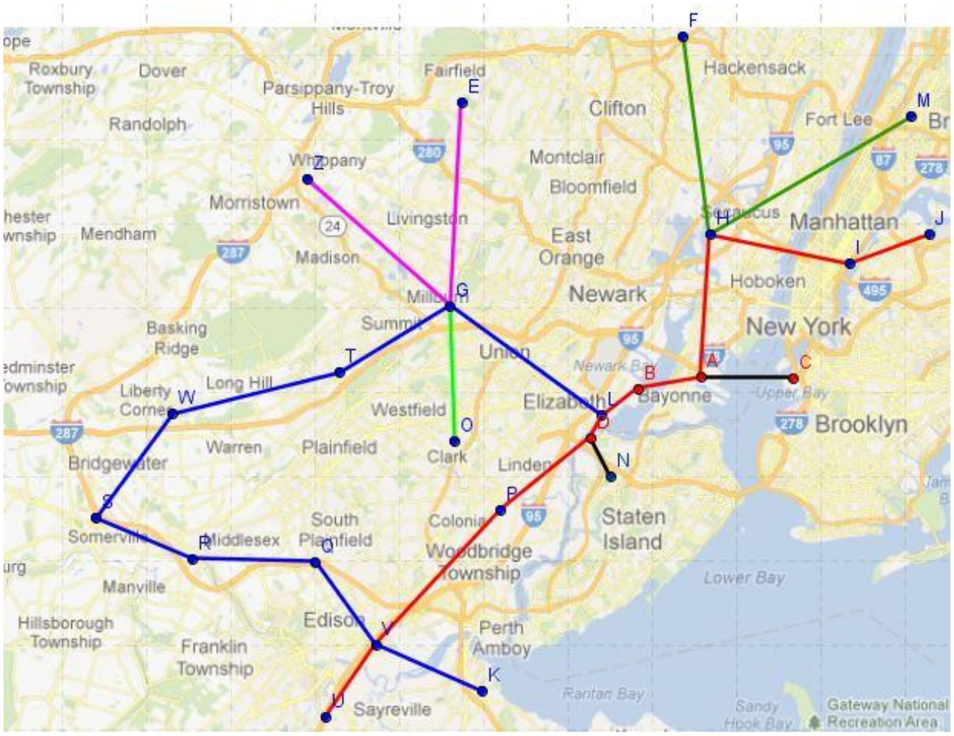

Figure 4.34 Analysis of varied station construction cost: Scenario 2

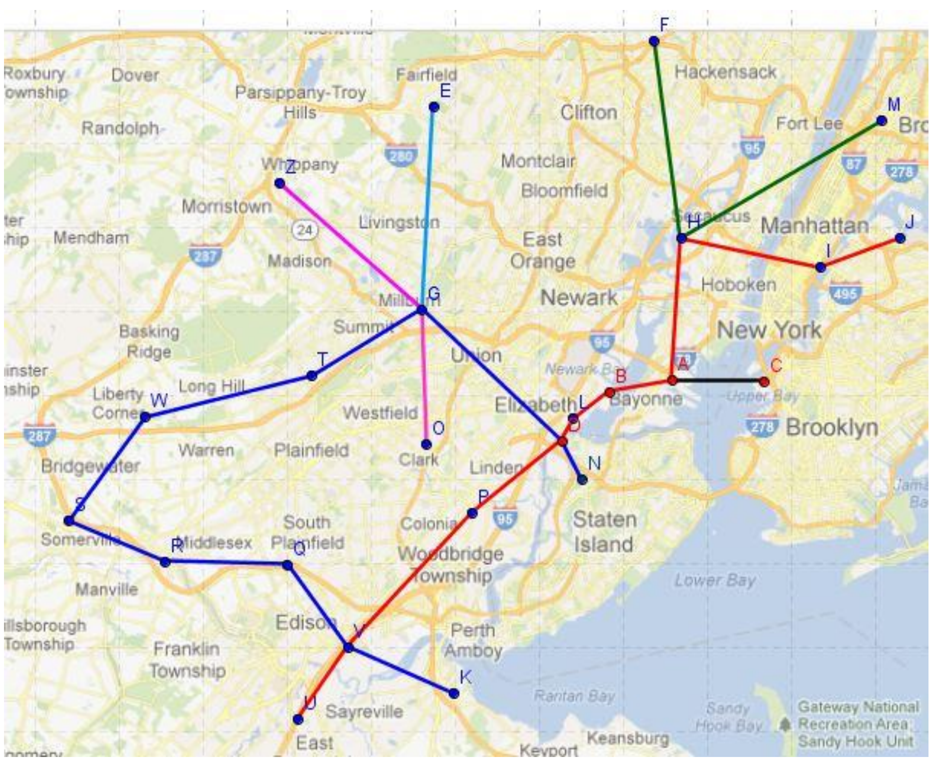

Figure 4.35 Analysis of varied station construction cost: Scenario 3

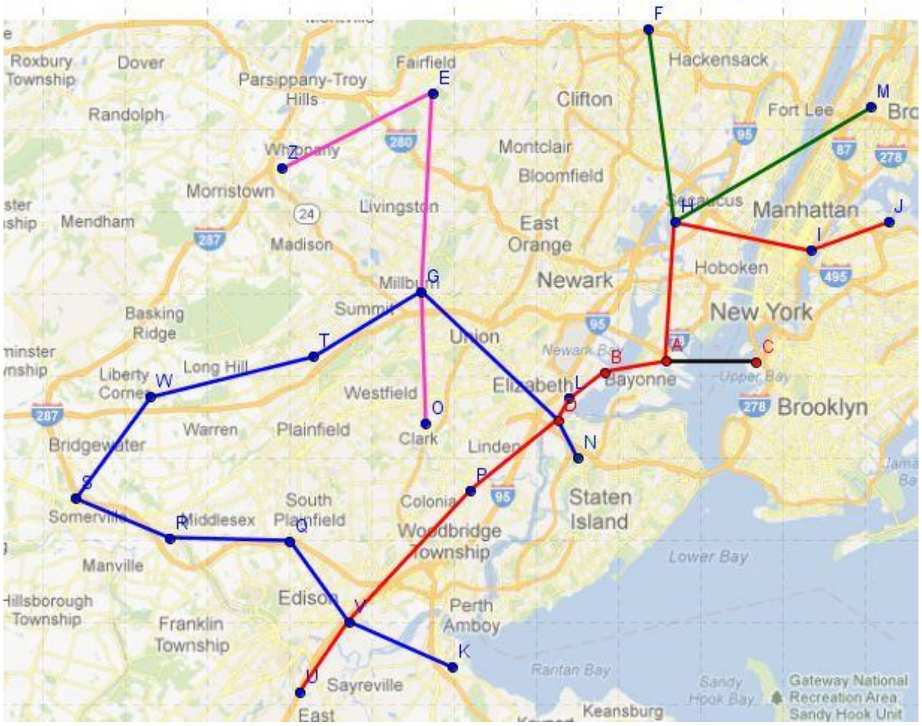

Figure 4.36 Analysis of varied station construction cost: Scenario 4

The rationale behind these changes is that the UFP network design model tries to adjust the network configuration the find the best tradeoff between all cost components.

Table 4.11 summarized all costs components of all four scenarios. As the per station construction cost increases, the optimized network configuration changes to reduce the total number of intersected lines, in such a way that the number of transfer stations is reduced to compensate the increase of unit station construction cost.

Another finding from Table 4.11 is that the sum of the edge construction and the operational cost is increasing, which indicates that the directness of the network is decreasing and containers need to travel longer distance from its origin to destination. However, the loss of network directness can help to reduce the number of transfer stations which can greatly reduce the overall cost of the whole system.

Table 4.11 Analysis of container flow pattern: Cost comparison

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
station construction cost	320000000	403000000	900000000	1015000000
edge construction cost	1737499030	1748267584	1751416812	1748784650
operational cost on edges	2102266612	2103708958	2124812829	2160519809
transfer cost	110693000	104411000	95520500	95336000
Sum of edge construction and operational cost	3839765642	3851976542	3876229641	3909304459
Total cost	4270458642	4359387542	4871750141	5019640459

To summarize, as the unit station construction cost increases, the optimal network configuration evolves to reduce the number of transfer stations, even if these changes lead to the increase of other cost components such as the total operational cost. Overall, the optimal UFP network is always the one with the best tradeoff between all cost components.

4.3.2.3 Analysis of central inspection stations

Liu (2004) mentioned possible homeland security issues in New York ports. New York City has some of the nation's busiest ports, with thousands of containers lying on the waterfront waiting to be shipped either to inland by trucks and trains, or to be loaded on outbound ships. The presence of such large number of idled containers at any harbor not only wastes the precious space at the busy harbor but also causes security concerns. The nation's port authorities have been criticized for not inspecting every container shipped into the nation. However, to inspect every container would cause even more delays and greater number of containers waiting at each container port to be inspected would exasperate the security problem.

The UFP system provides a perfect solution for this dilemma: One or several central inspection stations could be setup in less crowded inland safe locations so that containers going through the UFP system are inspected and processed by the US Customs and then transshipped to their individual destination stations.

For all the analysis that has been presented so far, we assumed that all container shipments are direct, i.e. from their origin station directly to destination station. In this subsection, we introduce a new scenario that one or more stations shown in Figure 4.23 are chosen to be the central inspection stations. We assume that all container flow between ocean ports stations and customer stations are trans-loaded in the central inspection stations. The central inspection stations are responsible to process and inspect the containers. Once inspected, containers resume their routes to their final individual destination stations.

Based on the geographical information, 4 stations are selected as candidate central inspection station locations and 6 scenarios are analyzed, which are summarized in Table 4.12. An additional scenario that has no central inspection stations is also included as the base scenario.

Table 4.12 Analysis of central inspection stations: Scenarios

	Central Inspection Stations

Base Scenario	None
Scenario 1	Station P
Scenario 2	Station O
Scenario 3	Station G
Scenario 4	Station H
Scenario 5	Station G & H
Scenario 6	Station P & H

In Figure 4.37 is the result network configuration from the base scenario where there are no central inspection stations and all containers are shipped directly from their origin stations to destination stations.

In scenario 1, Station P is selected as the central inspection station. All container flow between ocean port stations A, B, C, D and the customer stations are all intermediately sent to station P, after inspected and processed in station P by U.S. custom, all containers resume their routes and are sent to their destination stations. The result network configuration of scenario 1 is presented in Figure 4.38. And we notice that station P becomes a hub station that the customer stations located in the west and south regions are connected to ocean port stations through station P.

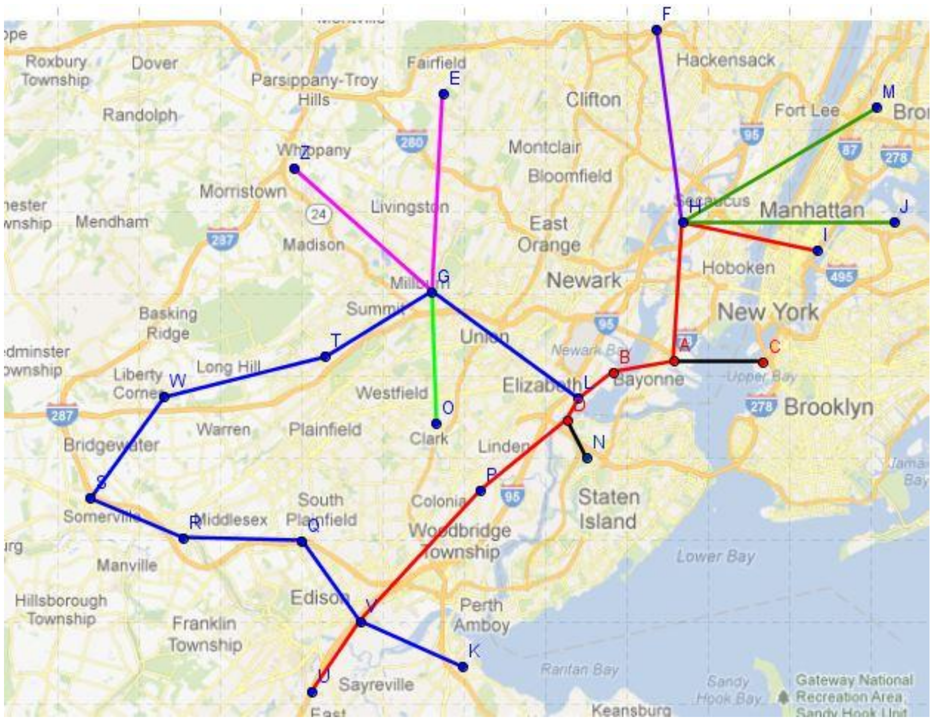

Figure 4.37 Analysis of central inspection stations: Base scenario

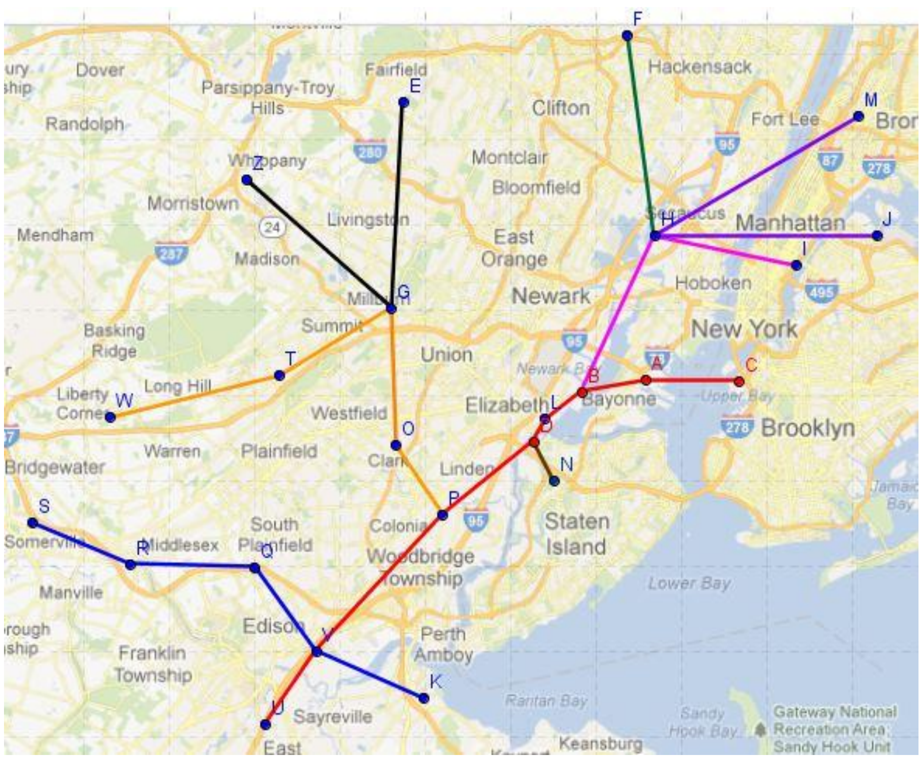

Figure 4.38 Analysis of central inspection stations: Scenario 1

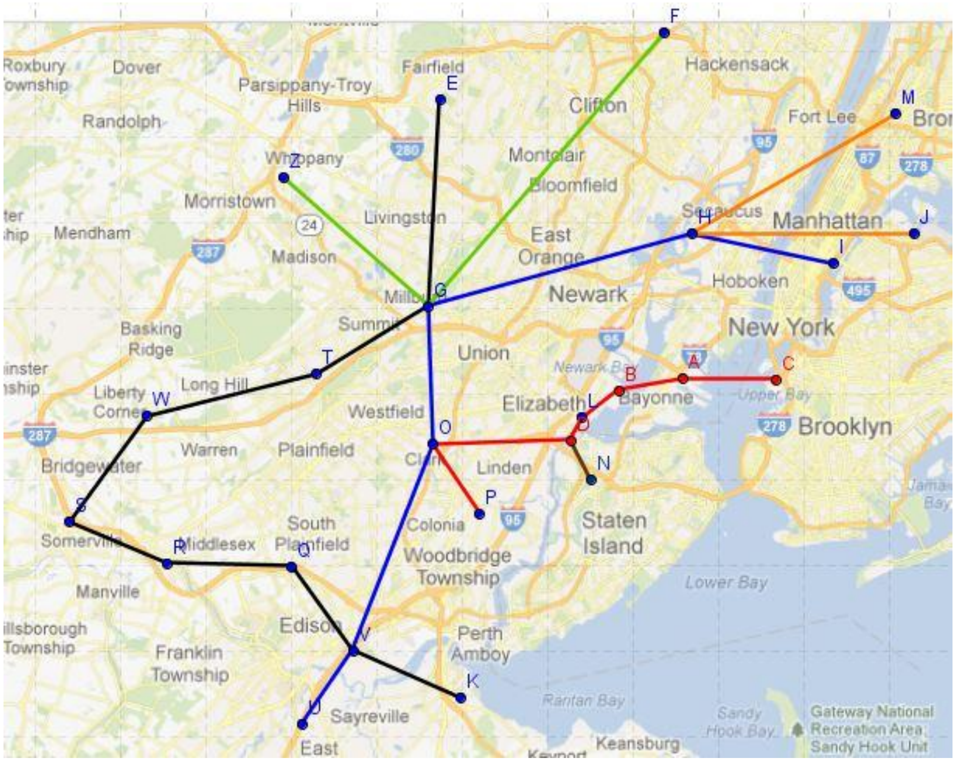

Figure 4.39 Analysis of central inspection stations: Scenario 2

In scenario 2, station O is selected as the central inspection station. It is noticed that, in the result optimal network configuration as shown in Figure 4.39, a tunnel is constructed between station O and ocean port station D, which provides a direct access from station O to ocean port stations. And almost all customer stations (except station L and N) are connected to ocean port stations through central inspection station O.

In scenario 3, recall that station G is selected as the central inspection station, the UFP network design model generates the optimal network configuration as shown in Figure 4.40, in which Station G becomes a major hub that connects most customer stations to ocean port stations. Similarly, in the result network configuration of scenario 4, as shown in Figure 4.41, the selected central inspection station H becomes a connecting station between the customer stations and the ocean port stations.

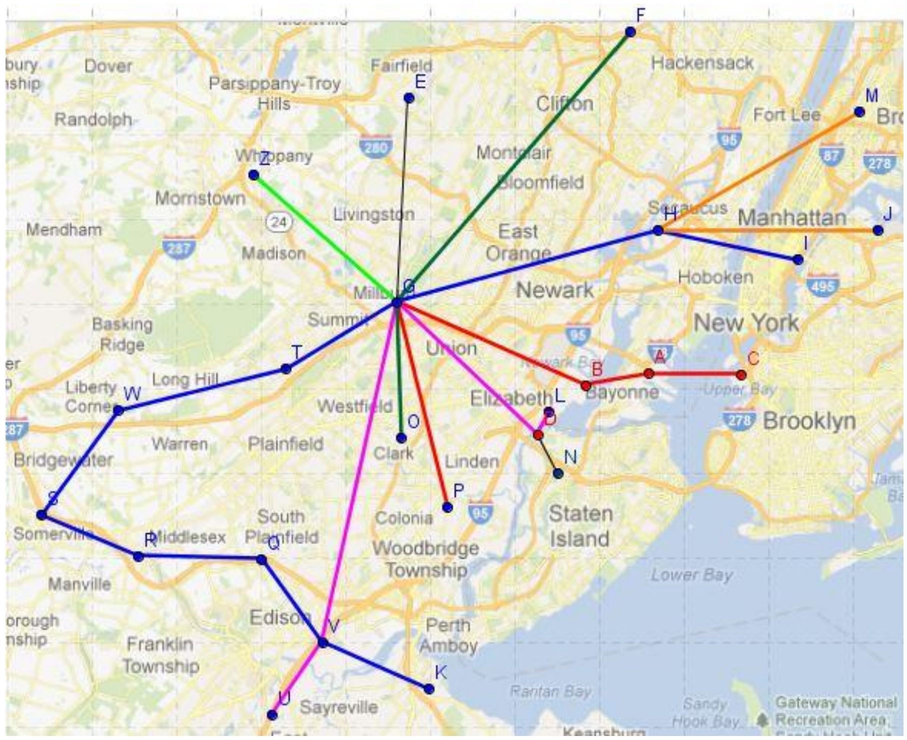

Figure 4.40 Analysis of central inspection stations: Scenario 3

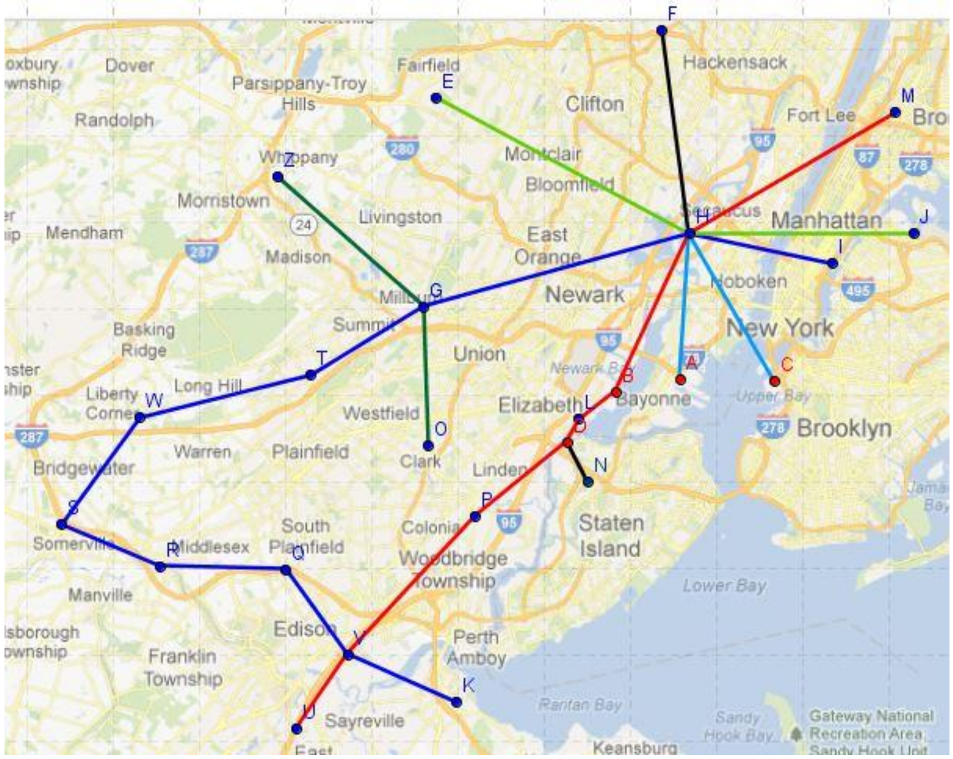

Figure 4.41 Analysis of central inspection stations: Scenario 4

In each of 4 scenarios presented so far, only one station is appointed as the central inspection station which is responsible to inspect and process all the containers to and from the ocean port stations. However, it is worth noticing that all the stations are quite spread out within the region. The scenarios that all containers are inspected and trans-loaded in only one central inspection station makes container flows between certain origin and destination pairs traveling unreasonable excessive distance. For instance, in scenario 2 shown in Figure 4.39, containers from ocean port station A to customer station H need to first travel west to station O and travel north to station G and then travel west bound to station H. Comparing with the base scenario shown in Figure 4.37, these containers are traveling directly from station A to station H through tunnel edge AH.

To address this issue, two more scenarios, scenario 5 and 6 are analyzed. In these scenarios, two stations instead of one are selected as central inspection stations so that container flows traveling to different directions are taken care of by the proper central inspection stations.

In scenario 5, the result network as shown in Figure 4.42, station G and H are appointed as the central inspection stations. Containers from customer stations in the west region are inspected and trans-loaded in station G. Containers from customer stations in the east region are trans-loaded in station H.

In scenario 6, the result network as shown in Figure 4.43, station P and H are selected as the central inspection stations. Containers from customer stations in the north region are inspected in station H, while containers from customer stations in the south region are trans-loaded in station P.

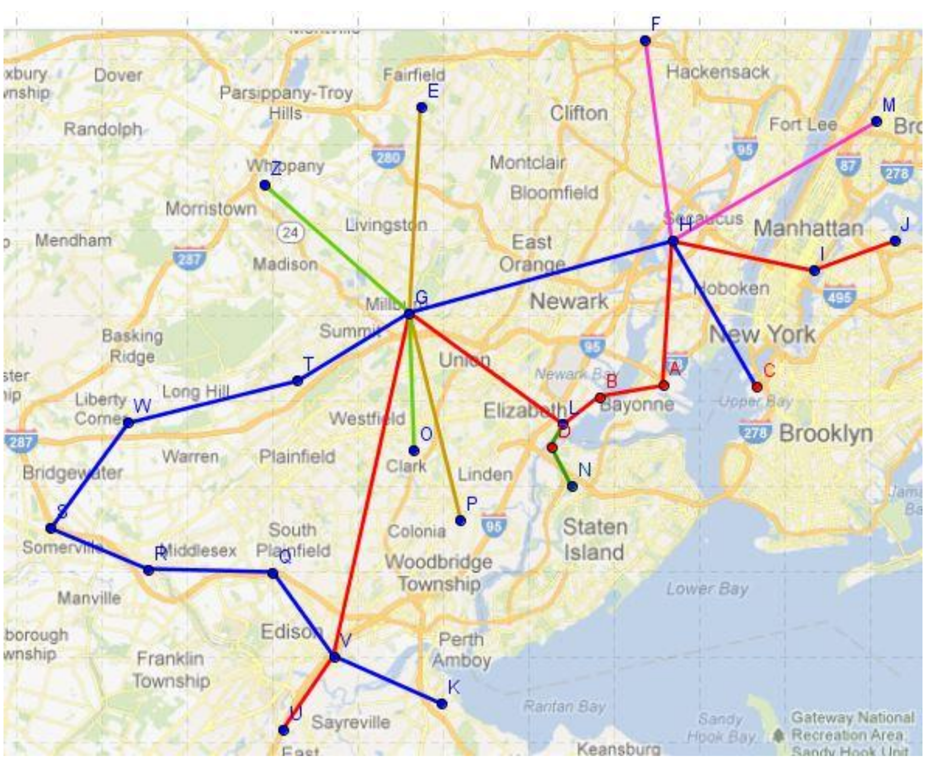

Figure 4.42 Analysis of central inspection stations: Scenario 5

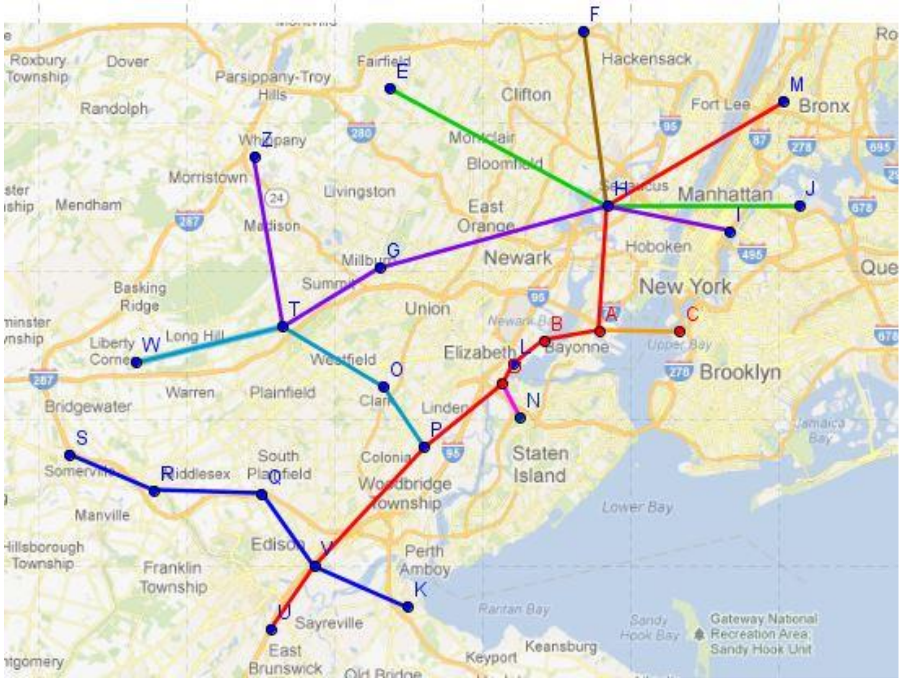

Figure 4.43 Analysis of central inspection stations: Scenario 6

Overall, as the locations of the central inspection stations changes, so as the container flow pattern. Thus, the UFP network design model adjusts the network configuration to accommodate the changing container flow pattern. The question is that which scenario is the better choice in terms of cost as well as other considerations.

Table 4.13 Analysis of central inspection stations: Cost comparison

Scenario	Base	1	2	3	4	5	6
Central inspection stations	None	P	O	G	H	G & H	P & H
Edge constr. cost	1.74×10^{9}	1.77×10^{9}	1.80×10^{9}	1.62×10^{9}	1.60×10^{9}	1.59×10^{9}	1.80×10^{9}
Station constr. cost	3.20×10^{8}	3.10×10^{8}	3.00×10^{8}	3.10×10^{8}	3.10×10^{8}	4.03×10^{8}	3.20×10^{8}
Oper. cost on edges	2.10×10^{9}	2.70×10^{9}	2.79×10^{9}	2.82×10^{9}	2.76×10^{9}	2.55×10^{9}	2.23×10^{9}
Transfer cost	1.11×10^{8}	8.81×10^{7}	5.93×10^{7}	3.58×10^{7}	3.17×10^{7}	5.85×10^{7}	6.01×10^{7}
Total cost	4.27×10^{9}	4.87×10^{9}	4.94×10^{9}	4.79×10^{9}	4.70×10^{9}	4.60×10^{9}	4.41×10^{9}
Cost changes (%)		13.96%	15.78%	12.08%	10.07%	7.73%	3.30%

All the cost components of the optimal network configurations of all 7 scenarios are presented in Table 4.13. These cost components include Tunnel edge construction cost, Station construction cost, Operational cost on edges, and Transfer cost in transfer stations.

By examining each individual cost component, there are some interesting findings. Scenarios 1, 2, and 6 have larger edge construction cost comparing with the base scenario. And this cost is smaller in scenario 3, 4, and 5 comparing with the base scenario. Note that the decrease of tunnel edge construction cost does not always mean the decrease of the total length of the constructed underground tunnel. It is actually the operational cost on edges that indicates the total length of the constructed. For all six scenarios presented, the operational cost on edges increases comparing with the base scenario which indicates that the total length of constructed tunnel is indeed longer than the counterpart of the base scenario. Thus, the decrease of the edge construction cost in scenario 3, 4, and 5 is due to the fact that more tunnels are constructed within the rural area and are much cheaper to build. Also, since all containers are mandatorily sent to central inspection stations and then transferred to their final destination, the associated inspection and transfer cost occurred in the central inspection stations are not included in the total transfer cost.

Among all six scenarios, no matter setting up one or two central inspection stations, it is shown in Table 4.13 that the total costs of the result UFP network design increase comparing with the base scenario. More specifically, for the first four scenarios with one station selected as central inspection station, scenario 4 is the most cost effective which incurs 10.07% increase of the total cost. For scenario 5 and 6 in which two stations are selected as central inspection stations, scenario 6 with station P and H as central inspection stations is a better choice as this scenario incurs only 3.3% increase of the total cost.

One must also keep in mind that whether one or two central inspection stations need to be setup also depends on other factors. One of them is the cost of setting up the custom facilities which is beyond the scope of this research. More importantly, what the decision makers also need to consider is the huge social benefits gain with just a relatively small increase of economic cost. The UFP system with central inspection stations will not only greatly improve port security, but also eliminate the need for trucks to enter ports, therefore transforming the waterfront from a container storage yard and truck depot to a quiet and nice place with shops and restaurants for the enjoyment of the local residents and tourists. Such a system will have immense value to a coast city like New York City.

4.4 Summary

In this chapter, we presented the Underground Freight Pipeline Network Design model. The UFP network design model takes into account the minimization of four cost components: tunnel construction cost, station construction cost, transportation cost and the operational cost in transfer stations.

More specifically, three versions of the UFP network design model are discussed. First, the UFP network design *comprehensive model* is introduced. This model is only able to handle small size problem due to its complexity. Second, the idea of the *Intuitive two-step model* is discussed. This way of modeling requires far less computational effort. However, it is lack of the ability to optimize the network comprehensively. There is a missing link between the two modeling steps: the transfer station construction cost. An example was given to show that the transfer station construction cost does affect overall network configuration. Then, based on the *Intuitive two-step model*, with added enhancement, we presented the *Enhanced Two Step Model.* The added enhancement estimates the approximate number of lines intersected at the same station node based on the degrees of each station node. The *Enhanced Two Step Model* is not only able to solve the UFP network design problem with much less computational time but also maintain the ability of generating decent network design solutions.

Then, we presented a case study: the New York UFP network. This UFP network is designated to transport the ocean containers between the customer stations within the Great New York region and the major ocean ports in New York and New Jersey. In this case study, a series of experiments and analysis are conducted to provide more insights of how different parameters and variables affect the strategic planning of the UFP system for dispatching ocean containers in New York area. It includes analysis of container flow pattern, analysis of varied station construction cost and analysis of central inspection stations.

Chapter 5 Summary, Contribution, Other applications and Future extension

5.1 Summary and Contributions

The effort made to the development of underground freight pipeline system since 1960s has been focused mainly on the technology side.

As the air blowing mechanism was used in the early freight pipeline system, the use of linear induction motor in the underground freight pipeline system proposed by William Vandersteel in U.S. Patent No. 4,458,602 was a major technical breakthrough, which avoids the system the restriction imposed by the airlocks and valves and allows the system to operate continuously without interruptions or distance limitation.

Liu (2009) proposed the preliminary design of several major components of LIM-PCP based underground freight transportation including capsule and guide rails, station layout, tunnel structure, and basic operations.

However, besides all the efforts put into the technological development throughout the years, little work had been done for the design of larger and more complex freight pipeline system and the logistic related issues. Because most demonstration or pre-commercialization freight pipeline systems are based on simple network configurations with a limited number of capsules as well as loading/unloading stations.

As researchers and engineers all around the world devote their intelligence and effort into the development of freight pipeline system, as well as the financial and environmental motivations, we believe that in the near future we will reach the stage of building more complex systems and making the systems work efficiently and effectively.

Among a number of potential applications of freight pipeline system such as pallet-tube system for transporting pallets and other goods, dispatching containers, truck-ferry system, solid waste transportation, and mail and parcel transportation, in this research we consider building a UFP system that connects the major ocean ports of a coastal city with the nearby regions where major industrial districts and distribution centers are located.

To satisfy the ocean container transportation needs among the ocean ports and customer stations, an underground freight pipeline system needs to be built. Then, a question was raised: how to design such a network?

In this research, we present the network design problem and the associated mathematical model for the underground freight pipeline system. The mathematical model considers a network with a set of predefined station nodes and all possible edges connecting each pair of nodes. With the knowledge of freight volume, the objective function of the model is to minimize the construction cost and the operational cost. Overall the model is able to answer the following questions:

- Which edges/tunnels needs to be constructed?
- Which line does each edge/tunnel belong to?
- Which edges and lines does each cargo use that forms a shortest path from the origin station to the destination station?

The aim of this research is to offer an overall approach to optimize the logistics performance of the underground freight pipeline system. To evaluate the performance of such a system, one needs to look at the design issues from the structure level as well as the operational level. In this research proposal, the design level of UFP system is the fundamental network structure, and the operational level of design is the capsule scheduling within one line of freight pipeline system.

The network design problem does integrate the system's operational performance issues into the strategic structure planning. It tends to find a balanced network structure which not only results the minimized construction cost but also accommodates the cargo flow during a certain period. It also makes sure that the network provides as much directness and efficiency as possible to minimize the overall operational cost. However, the operational performance in the network design model is considered from the overall strategic planning level. If we dive into the real-time operation of a freight pipeline system, more work needs to be done.

One line/alignment, as one fully functional component of the overall network, its efficiency relies heavily on the cargo transportation schedule. A good scheduling plan is able to increase the utilization of the system and reduce the tardiness of transportation tasks thus increase the revenue and customer satisfaction. The capsule control problem presented in this proposal considered a freight pipeline with a single circle along which there are loading/unloading stations. The associated mathematical model minimizes the total tardiness squared which not only minimizes the total tardiness of all transportation tasks but also distribute the possible tardiness evenly to improve the overall customer satisfaction.

Above all, both the network design model and the capsule scheduling model together give a systematic solution to the design of the underground freight pipeline system. Either of the designing models can only give a partially optimized solution, i.e. if a system has a sound network structure but the capsules routing schedule is poorly planned, the overall system performance can still be poor with low customer satisfaction. If a system's capsules and cargo flow is nicely planned but its fundamental network structure is not satisfactory, all the hard work trying to get a good schedule plan are just based on a poor foundation and the overall system performance can still be unsatisfactory.

5.2 Other applications

The UFP transportation tasks scheduling problem belongs to a more general scheduling problem class with resource limit, within which each individual candidate task has its own release time, processing time and due date. Thus, we can apply the mathematical model as well as the heuristic algorithm to any scheduling problem with the characteristics mentioned above with little or no changes. For instance, the scheduling problem in a single machine with multiple processors is well suited to our UFP transportation tasks scheduling problem if we consider the processor limit on the machine as the line fill capacity of the freight pipeline and think of the machining jobs with their own release time, processing time and due date as the transportation tasks in the freight pipeline system.

Even though the network design model in this research is designated specifically to the Freight Pipeline System, it does have the potential to be applied to other areas that requires the network design. One of the potential areas is the public transportation system. For instance, consider a problem that we need to design several bus routes. In this problem, we have the knowledge of passenger flows between stations, of which the goal is to design the bus routes that provide convenience to all passengers so that they can reach their destinations in the shortest time and with as few transfers between routes as possible. Note that since in this problem roads already exist, the application of the original FPS model needs to be adjusted to focus on the operational cost/time.

5.3 Future research extension

For the next 5 years, we envision the following as the extension to the current research:

5.3.1 Pallet tube system design

Among all the potential applications of the UFP technology, this research focuses on a UFP system that connects the major ocean ports of a coastal city with the nearby regions where major industrial districts and distribution centers are located. There is another interesting and even more complex application that we would like to research in the future – Pallet Tube System in urban area. Liu (2004) proposed such an application in New York City. The proposed pallet tube system is expected to provide transportation service for goods in pallets, boxes, crates or bags. And it will help removing a large amount of delivery trucks within New York City, which will result in much less pollution and traffic congestion. For such a system to work for a mega city such as New York, it must have an extensive network of underground tunnels or conduits with numerous inlet/outlet stations. Such a system should also have several freight transfer stations around the outer perimeter of the network. At each freight transfer station, freight carried by trucks will be unloaded onto capsules and dispatched through the underground pallet tube network to various stations in NYC.

In general, the Pallet Tube System and the UFP system for dispatching containers are similar. Both are consisted of two types of stations:

- A few number of large freight transfer stations, such as port stations in the container dispatching UFP system and freight transfer stations in Pallet Tube System.
- Numerous smaller service stations, such as customer stations in the container dispatching UFP system and the inlet/outlet stations in Pallet Tube System.

Therefore, same or similar optimization models and methodology as presented in this research are suitable to the Pallet Tube System design. This future research will also include two design components: the strategic network design and the operational capsules control design.

However, one major difference we should keep in mind is that the problem size of these two UFP applications differs greatly, i.e. the UFP system for container dispatching discussed in this research considers a network with 24 station nodes. Yet, as proposed by

Liu (2004), a Pallet Tube System is a dense underground network similar to NYC Subway system. As of 2013, the NYC Subway system has 468 stations. Thus, to cover whole New York City, the future Pallet Tube System will be the similar size and is consists of hundreds of stations. The drastic increase in problem size of the Pallet Tube System network will be a challenge to the network design model presented in this research. And a new solution methodology/strategy would be needed.

5.3.2 Preliminary study of urban freight transportation

In addition, a preliminary study of the freight transportation demand is necessary for the Pallet Tube System design. This study will include but not limited to freight demand of urban business, potential locations of both inlet/outlet stations and freight transfer station locations, and the construction cost of underground tunnel suitable to pallet transportation. Such a study will give us a solid foundation for further logistics system design by providing the critical parameter values to optimization models.

5.3.3 System design integration

In this research, we presented two stages of design: Strategic network design and the Operational capsule controlling design. It offers a systematic approach of designing the UFP system from the logistic perspective. However, how to integrate both the strategic and the operational design remains an interesting future research topic.

For instance, in the capsule control model in Chapter 3, we presume that the parameters of line-fill capacity and the number of capsules are known. And these parameters are critical to customer satisfaction level. In general, the more resource available the less delivery tardiness will be, which is translated into higher customer satisfaction level. However, at current stage of this UFP research, the appropriate resource level to support the customer satisfaction level goal remains unknown, and the future integrated system design will help to address this issue.

By integrating the strategic network configuration with the operational design of capsule control, we are not only able to find the optimal system design based on the aggregate cargo flow but also the design that supports daily operations. Therefore, it completes the system design by defining the required resource level to achieve the desired customer satisfaction level.

Reference

Ahuja, R.K., T.L. Magnanti, J.B. Orlin and M.R. Reddy (1995) Applications of network optimization, Ball, M.O., Magnanti, T.L., Monma, C.L., Nemhauser, G.L. (Eds.), Network Models, Handbooks in Operations Research and Management Science, 1995, Vol. 7, pp. 1-83.

ASCE (1998), Freight pipelines: Current status and anticipated future use, report of the Task Committee on Freight Pipelines, *ASCE Journal of Transportation Engineering*, 124(4), pp. 300-310.

Bagga P.C. and K.R. Kalra (1980) A node elimination procedure for Townsend's algorithm for solving the single machine quadratic penalty function scheduling problem, Management Science, Vol. 26, No. 6, 1980, pp. 633–636.

Beckmann, H. (2007). CargoCap: a new way to transport freight, Conference Proceedings of the Schiller Institute Conference, Kendrich, Germany. Sep. 15–16, 2007.

Blackledge D.A. and E.M.H. Humphreys (1984) The West Midland rapid transit study, Proceedings of the Planning and Transport Research and Computation Ltd., Sussex, 1984, pp. 71-84.

Boerkamps J. H. K. and A. J. van Binsbergen (1999) GoodTrip: A New Approach for Modeling and Evaluation of Urban Goods Distribution, Urban Transport Systems Conference, Lund, Sweden, June 7 and 8, 1999

Bruno G., G. Ghiani and G. Improta (1998) A multi-modal approach to the location of a rapid transit line, European Journal of Operational Research, 1998, Vol. 104, pp. 321-332.

Bruno G, M. Gendreau and G. Laporte (2002) A heuristic for the location of a rapid transit line. Computers & Operations Research, 2002, Vol. 29, pp. 1-12

Chou, F.D., T.Y. Chang and C.E. Lee (2005) A heuristic algorithm to minimize total weighted tardiness on a single machine with release times, *International Transactions in Operations Research*, 12 (2005) pp. 215–233.

Costa A.M., J.F. Cordeau, G. Laporte (2006) Steiner tree problems with profits, INFOR, 2006, No. 44, pp. 99-115

Cotana, F., Rossi. F., Marri, A., Amantini, M. (2008). Pipe§net: innovation in transport through high rate small volume payloads, Proceedings from the 5th International Symposium on Underground Freight Transportation by Capsule Pipelines and Other Tube/Tunnel Systems, 2008. The University of Texas, Arlington, Mar. 20–22.

Dufourd H., M. Gendreau and G. Laporte (1996) Locating a transit line using tabu search, Location Science Vol. 4, pp. 1-19

Emmons H. (1969) One-machine scheduling to minimize certain functions of job tardiness, Operations Research, 1969, Vol. 17, pp. 701–15.

Eusebio R.E.A., L. Soldo, R. Funiciello, and L. Solimene (2007) Geological and geotechnical conditions in underground structure design of Rome subway: the B-Line extension, Proceedings of the World Tunnel Congress 2007 and 33rd ITA/AITES Annual General Assembly, Prague, May 2007, pp. 59-64.

Fukuyama M. (1981) Optimal station location for a two hierarchy transit system, In Proceedings of the Eight International symposium on transport and traffic, 1981, pp. 264-291.

Gelders L. P.R. and Kleindorfer (1974) Coordinating aggregate and detailed scheduling in the one-machine job shop I: theory, Operations Research, 1974, Vol. 22, No. 1, pp. 46–60.

Gelders L, P.R. and Kleindorfer (1975) Coordinating aggregate and detailed scheduling in the one-machine job shop II: computation and structure, Operations Research, 1975, Vol. 23, No. 2, pp. 312–324.

Gendreau M, G. Laport, J.A. Mesa. (1995) Locating rapid transit lines. Journal of Advanced Transportation 1995,Vol. 29, No. 2, pp. 145-162

Gupta S.K. and T. Sen (1984) On the single machine scheduling problem with quadratic penalty function of completion times: an improved branching procedure, Management Science, Vol. 30, No. 5, 1984, pp. 644–647.

Hoitomt D.J., P.B. Luh, E. Max and K.R. Pattipati (1990) Scheduling jobs with simple precedence constraints on parallel machines, IEEE Control Systems Magazine, 1990, pp. 34–40.

Fang J., J. Lawson and D.B. Montgomery (2011) A new pipeline system transporting coal ores, The 21st International Conference on Magnetically Levitated Systems and Linear Drives, October 10-13, 2011, Daejeon, Korea

Kanet J.J. (2007) New precedence theorems for one-machine weighted tardiness. Mathematics of Operations Research, 2007, August, pp. 579–88.

Klau G.W., I. Ljubic, P. Mutzel, U. Pferschy, R. Weiskircher (2003) The fractional prize-collecting Steiner tree problem on trees, Lecture notes in Computer Science, 2003, vol. 2832, pp. 691-702

Laporte G, J.A. Mesa and F.A. Ortega (1994) Assessing the efficiency of rapid transit networks, Studies in Locational Analysis, 1994, No. 7, pp. 105-121

Laporte G, J.A. Mesa, and F. Perea (2010) A game theoretic framework for the robust railway transit network design problem, Transportation Research Part B, 2010, pp. 447-459.

Lawler E.L. (1979) Efficient implementation of dynamic programming algorithms for sequencing problems, Report BW 106, Mathematisch Centrum, Amsterdam, The Netherlands, 1979.

Liu, H. (2003), Pipeline Engineering (*Chapter 7 Capsule Pipelines*), CRC Press (Lewis Publishers), Boca Raton, Florida, pp.175-204.

Liu, H. and C. W. Lenau, (2009), A LIM-PCP System for Freight Transportation in Underground Cities, *Proceedings of the 12th Intl. Conference of the Associated Research Centers for Urban Underground Space*, Shenzhen, China, Nov.18-20, 2009, pp. 2*03- 210.*

Liu, H. (2004), Feasibility of Underground Pneumatic Freight Transport in New York City, Final Report Prepared for The New York State Energy Research and Development Authority, Aug. 1 2004.

Luh P.B. and D.J. Hoitomt (1993) Scheduling of manufacturing systems using the lagrangian relaxation technique, IEEE Transactions on Automatic Control, 1993, Vol. 38, No. 7, pp. 1066–1079.

Musso A. and V.R. Vuchic (1988) Characteristics of metro networks and methodology for their evaluation, Transportation Research Record, 1988, No. 1162, pp. 22-33.

Pneumatic pipeline transport system, U.S. Patent Number 4,458,602, July 10, 1984

Potts C.N. and L.N. van Wassenhove (1985) A branch and bound algorithm for the total weighted tardiness problem, Operations Research, 1985, Vol. 33, No. 2, pp. 363–377.

Rachamadugu R.M.V. (1987) A note on the weighted tardiness problem, Operations Research, 1987, Vol. 35, pp. 450–451.

Rinnooy Kan A.H.G., Lageweg JK, and Lenstra (1975) Minimizing total costs in one-machine scheduling. Operations Research, 1975, Vol. 23, No. 5, 908–927.

Schaller J. (2002) Minimizing the sum of squares lateness on a single machine, European Journal of Operational Research, 2002, 143, pp. 64–79.

Schild A. and I.J. Fredman (1962) Scheduling tasks with deadlines and nonlinear loss functions, Management Science, Vol. 9, No. 1, 1962, pp. 73–81

Schrage L.E. and K.R. Baker (1978) Dynamic programming solution of sequencing problems with precedence constraints, Operations Research, 1978, Vol. 26, No. 3, pp. 444–449.

Su L.H. and P.C. Chang (1998) A heuristic to minimize a quadratic function of job lateness on a single machine, International Journal of Production Economics, 1998, 55, pp. 169–175.

Sun X., J.S. Noble and C.M. Klein (1999) Single machine scheduling with sequence dependent setup to minimize total weighted squared tardiness, IIE Transactions, 1999, 31, pp. 113–124.

Szwarc W., M.E. Posner and J.J. Liu (1988) The single machine problem with quadratic cost function of completion times, Management Science, Vol. 34, No.12, 1988, pp. 1480–1488.

Townsend W. (1978) The single machine problem with quadratic penalty function of completion times, a branch and bound solution, Management Science, Vol. 24, No. 5, 1978, pp.530–534.

Tube Express, Ampower Corporation, 15. December. 2011, <www.tubepress.com>

Vuchic VR. Urban transit operations, planning and economics. Hoboken: Wiley; 2005.

van der Heijden, M.C., van Harten, A., Ebben, M.J.R., Saanen, Y.A., Valentin, E.C., Verbraeck, A. (2002). Using simulation to design an automated underground system for transporting freight around Schiphol Airport. Interfaces 32, 1–19.

Weber, K. M., van Zuylen, H., 2000. European Policy for Technology Innovation in Transport: Finding the Right Role and the Right Options. The Institute for Prospective Technological Studies Report, Issue 48.

Zandi, I. (1976), *Transport of Solid Commodities via Freight Pipeline, Vol.2, Freight Pipeline Technology*, U.S. Department of Transportation, Report No. DOT-TST-76T-36, Washington, D.C.

Vita

Gaohao Luo received a B.S. degree in Electrical Engineering in 2005 from Beijing Jiaotong University in Beijing, China. He joined the Industrial & Manufacturing Systems Engineering department of University of Missouri Columbia in August 2006 for graduate study and received his M.S. degree in 2008. In 2009, he started pursuing the degree of Philosophy in Industrial Engineering.