Integrated Optimization And Simulation Model For Resource Acquisition And Utilization

— An Application To Ocean/River Articulated Tug/Barge System

by

Ming Qi

M.A. in Marine Affairs, University of Rhode Island (1991)
M.S. in Ocean Engineering, University of Rhode Island (1990)
B.S. in Physical Oceanography, Shandong College of Oceanography (1982)

Submitted to the Department of Ocean Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Science in Ocean Systems Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 1997

AUTHOR ....
DEPARTMENT OF OCEAN ENGINEERING
DECEMBER 11, 1996

CERTIFIED BY ....
DR. ERNST G. FRANKEL
PROFESSOR OF MARINE SYSTEMS & MANAGEMENT
THESIS SUPERVISOR

ACCEPTED BY ....
DR. J. KIM VANDIVER
CHAIRMAN, DEPARTMENTAL COMMITTEE ON GRADUATE STUDIES

Integrated Optimization And Simulation Model For Resource Acquisition And Utilization

— An Application To Ocean/River Articulated Tug/Barge System

by

Ming Qi

Submitted to the Department of Ocean Engineering
on December 11, 1996, in partial fulfillment of the requirements for the degree of
Doctor of Science in Ocean Systems Management

Abstract

This thesis research develops an integrated adaptive iteration of optimization and simulation (IAIOS) approach for modeling complex systems subject to resource constraints as well as alternative acquisition and utilization. This proposed modeling approach is tested by an application to a real world ocean/river articulated tug/barge (ORATB) transport system.

The integrated modeling approach is adaptive and iterative in the sense that the optimization model first suggests the acquisition strategy, which is then tested and evaluated in the simulation model. If the performance of the acquisition strategy is inferior, constraints and cost parameters are added and/or modified in the optimization model and the procedure is iterated until no significant improvement in performance can be achieved. The distinction of this proposed IAIOS approach is that it solves resource acquisition and resource utilization simultaneously. In addition, this modeling approach provides the mechanism for decision makers to interact with the model at different levels during the whole decision making process.

The proposed IAIOS modeling approach is applied to solve the acquisition and utilization problems in designing the ORATB system for transporting iron ore and containers between a coastal port and the Yangtze River ports in China. Through this application, the IAIOS modeling approach is proved to be an effective tool for decision makers to test the performance of the ORATB operations under a wide range of anticipated conditions, and thus ensures a satisfactory deployment of the transport system.

<u>Thesis Committee members:</u>

Prof. Ernst G. Frankel, thesis supervisor	Department of Ocean Engineering, MIT
Prof. Jeremy F. Shapiro	Sloan School of Management, MIT
Prof. Gordon M. Kaufman	Sloan School of Management, MIT

Acknowledgments

First and foremost I thank my thesis advisor, Dr. Ernst G. Frankel, for his invaluable advice and guidance throughout the entire process of this study. This dissertation could not have been completed without Prof. Frankel's generous support. My sincere thanks also go to my thesis committee members, Prof. Jeremy F. Shapiro and Prof. Gordon M. Kaufman of the Sloan School of Management, for their inspiration and advice on this research. Their understanding and willingness to spend so much of their precious time on me to improve this study is highly appreciated. My deep appreciation is also extended to other faculty and stuff members in the Department of Ocean Engineering for their special concerns and encouragement during my whole study in Course 13.

I thank our administrative secretary, Ms. Sheila McNary, for her moral support and assistance at all appropriate moments. I learned many things from our daily conversation. In addition, I thank Sheila indeed for her industrious effort to proof read this manuscript.

To all my friends at MIT, especially to Mr. Richard Preston, Dr. Xiaoming Wang, Mr. Lian Shen, Mr. Yu Jin, Dr. Di Jin, Mr. Chris Hayes, Dr. Qing Zhou, Dr. Qin Chu, Mr. Matt Tedesco, Dr. Hua He, Dr. Xiaohong Yu, Dr. Ronald Chu, and Mr. Qiang Gao, thank you all for the greatly needed friendly support in many ways. You made my study at MIT a most pleasant experience in my life.

On a more personal note, I thank my wife, Yudong, from the bottom of my heart for her irreplaceable support and encouragement all my life. I dedicate this dissertation to her and our two lovely children, Bryan and Taryn.

Contents

Abstract

Acknowledgments

References

List of Tables

List of Figures

Chapter One

1. INTRODUCTION

1.1. Background

The field of transportation has been always very important from all technical, sociological and economic points of view. Though ships constitute a very old means of transportation of goods, waterborne shipping holds the first place in the world among all types of transportation modes. It is estimated that even today, waterborne shipping still accounts for about 90% of the volume of all goods transported and around 70% of their value. In fact, ships can carry more ton-miles of goods per gallon of fuel than any other mode of transportation. At the same time, historical statistics and the new trend of world trade indicate that still greater demands will be made for more and better transportation systems.

Over the last decade, the world's maritime transport systems have become more complex than ever, and have greatly increased in size. In tramp services, for example, the size of a cross ocean bulk ship has increased steadily to take advantage of the economies of scale. A typical ocean-going iron ore ship has a capacity of over 150,000 dead weight tons[1] (dwt). The problem of this increasing size is that the carrier may have a single supply

---

[1] The maximum weight of cargo and stores that a ship can carry. It is expressed in metric tons (1,000 kg) or long tons (1,016 kg). On the other hand, gross tonnage and net tonnage are defined according to formulas which take into account, among other things, the volume of the vessel's enclosed spaces (gross tonnage) and the volume of its holds (net tonnage). Gross tonnage is the basis on which manning rules and safety regulations are applied, and registration fees are reckoned. Port fees are also often reckoned on the basis of gross tonnage and net tonnage.

node at one end, and a more sporadic demand node on the other end. To overcome this problem, carriers must develop a cost effective and efficient transshipment[2] network to satisfy their customers.

Container liner carriers face similar problems. As the size of a containership has increased up to 6,000 teus[3], carriers' profits mainly depend on how many containers the ship can gather before crossing the ocean. To increase the customer base that the ship serves, carriers have only two choices: (a) make the ship call at more ports before it starts its deep-sea voyage; or (b) arrange more cargo transshipments to and from ports where the deep-sea vessel does not call. The drawback of (a) is that the more intermediate stops there are, the longer the total transit time for those containers first loaded on board will be. These multiple origin to multiple destination services are viewed as low quality service by the majority of shippers because they are concerned about the door-to-door transit time of their shipment. In addition, there are fewer ports in the world that can accommodate large container vessels mainly due to ports' physical restrictions, such as shallow water depth. The problem of (b), on the other hand, is that transshipment must be efficient both in terms of service charges and service transit time. As a result, there is again an urgent need for carriers to develop a cost effective and efficient transshipment network to satisfy their customers.

It is the objective of this study to develop a theoretical approach to design a cost effective and efficient transshipment network. To a larger extent, this study deals with resource acquisition and resource utilization problems. In other words, this thesis solves waterborne transportation resource acquisition and utilization problems through an integrated adaptive iteration of optimization and simulation (IAIOS) modeling approach. The integrated modeling approach is adaptive and iterative in the sense that the

---

[2] A shipment under one Bill of Lading, whereby waterborne transport is 'broken' into two or more parts. The port where the waterborne transport is 'broken' is the transshipment port.

[3] It is the unit of measurement equivalent to one twenty-foot container. A twenty-foot equivalent unit has an external dimension of 20' X 8' X 8'6" (length X width X height). Another popular type of container is forty-foot equivalent unit (feu), which has an external dimension of 40' X 8' X 9'6". In practice, we usually count two twenty-foot equivalent units as one forty-foot equivalent unit.

optimization model first suggests the acquisition of the ORATB system, and then the simulation model evaluates the performance of the acquisition in a simulated environment. If the detailed evaluation of the acquisition is unacceptable, constraints and cost parameters are added and/or modified in the optimization model and the procedure is repeated until no significant improvement in performance can be achieved. It is worth pointing out, however, that this integrated modeling approach can be applied to resource acquisition and utilization problems in other areas with minor modification.

1.2. Problem Statement

Although resource acquisition and utilization has been an active subject in many areas, most of the research works separate resource acquisition from resource utilization, or *vice versa*. The problem of such a separation between resource acquisition and utilization is that optimal resource acquisition solutions often find it very difficult to implement due to the changes and the uncertainties of the real world. On the other hand, resource utilization solutions only answer the question of how a fixed amount of resources are utilized, but not the question of how many resources should be acquired in the first place.

In this study, optimization of resource acquisition is integrated with the simulation of resource utilization through adaptive iterations. It is our intention to use the resource acquisition optimization model for the strategic planning, and the resource utilization simulation model for the tactical planning. The strategic planning is concerned mainly with establishing the necessary resources for a system to satisfy external requirements consistent with specific goals. Strategic decisions are extremely important because they are responsible for maintaining the competitive capabilities of the system, determining its rate of growth, and eventually defining its success or failure. One of the essential characteristics of strategic decisions is that they all have long lasting effects, thus they mandate long planning horizons in their analysis. Moreover, strategic decisions are usually resolved at fairly high managerial levels, and are affected by both internal and external information of the system. As a result, the strategic planning necessarily has a rather broad scope of information to be processed in an aggregated form, so that all the dimensions of the problem are included. Another reason for such a high level aggregation of information is not to let top level decision makers be distracted by unnecessary operational details.

Once the strategic planning on resource acquisition is accomplished, the next question is how to tactically plan on resource utilization. The purpose of the tactical planning is to ensure the implementation of the strategic planning at a more detailed operational level. In general, the tactical planning involves detailed operational information with the medium range time horizon divided into several periods. Meanwhile, the tactical planning is often performed at the middle management level. The tactical planning is important because it ensures the success of the strategic planning. Without the tactical planning, the strategic planning becomes meaningless.

It is clear that resource acquisition is quite different from resource utilization. These two types of decisions, the strategic planning and the tactical planning, differ in scope, level of management involvement, type of supporting information, and length of planning horizon. Thus, it is necessary to develop two distinct approaches to resolve resource acquisition and resource utilization decisions respectively. These two different type of decisions, however, must be integrated and interactive. The strategic decisions provide constraints for the tactical planning, and the execution of the tactical decisions determines the resource acquisition requirements to be supplied by the strategic planning.

1.3. Research Objectives And Scopes

It is the objective of this thesis to develop an IAIOS modeling approach to support both the strategic and the tactical planning. The first distinction of this IAIOS model is that it is formed by two mathematical models that interact with each other. The first mathematical model is an optimization model that deals with the long-term strategic planning associated with resource acquisition. The second one is a simulation model that deals with the short-term tactical planning associated with resource utilization. We can also say that constraints of the system are decomposed into two sets, global constraints and local constraints. Such decomposition really simplifies the model which in turn makes the decision making process much easier. More importantly, it permits decision makers to interact with the model through adaptive iterations at different time periods and horizons, at different levels of information aggregation, and at different scopes and levels of management.

The first constraint subset contains all global constraints for the optimization model. The second constraint subset constitutes all local constraints for the simulation model. The basic reason to decompose constraints into two sets is that the more constraints there are in an optimization model, the more complicated an objective function can be solved. In some cases, it may have no solution at all when there are too many constraints. Secondly, the more constraints there are under a simulation model, the less efficient and robust the model is. Needless to say, it would also take much longer to develop a simulation model when there are too many constraints. The third reason to decompose constraints into two sets is that some of the constraints are system-wide constraints, *i.e.*, global constraints, while others are only site-specific constraints, *i.e.*, local constraints. As a general guideline in decomposing in this research, global constraints are grouped together under the optimization model and local constraints are grouped together under the simulation model.

It is understood that optimal solutions may change with the composition of global constraints. Therefore, in this two-layer model, the simulation model only evaluates the performance of the optimal solutions generated by the optimization model under local constraints. Recommendations about modifying constraints, including adding and deleting them, for both the optimization and the simulation models are subsequently accomplished based on simulation evaluation process. It is very important to provide such opportunity to decision makers to interact with the system at local level. It is the task of this research to demonstrate the significance of providing different levels of interaction during the whole decision making process.

The second distinction of the proposed IAIOS modeling approach is that it deals with interdependent resource acquisition and utilization. In an interdependent resource acquisition and utilization problem, not only the level of each resource assignment but also the combined effects of the interdependencies would affect the optimal performance of the system under consideration. The performance of a tug/barge transport system, for example, is not only dependent on what and how many tugs and barges are deployed, but also on how many barges are attached to each tug. For a particular tug/barge transport system, due to the designed horsepower output of the tug, the larger the barges and the more barges attached to the tug, the slower the navigation speed of the system can be. In this sense, the proposed IAIOS model would also provide an optimal combination among all the interdependent resources.

The third distinction of this study is that the proposed IAIOS model is applied to a real world case to test its effectiveness. The case is to design an ocean/river articulated tug/barge (ORATB) system for meeting transport iron ore and container demands along the Yangtze River in central China. In the case, the river ports start from Shanghai near the coast upstream to Chongqing along the Yangtze River, while the coastal deep water port is located in Ningbo. It is assumed in this case that iron ore is imported to China from overseas and containers are exported to foreign markets. The proposed transport system is to distribute iron ore from Ningbo to various demand locations along the Yangtze River, and to collect containers from various supply locations along the Yangtze River to Ningbo. In other words, Ningbo is a transshipment center for both iron ore and containers. The transport technology employed is the ORATB system, which will be discussed in Chapter Two. The IAIOS model is applied to provide an optimal acquisition and utilization of resources. The optimal solutions include optimal number of tugs and barges in different sizes, optimal scales of port facilities, and optimal routing and scheduling of this transshipment system.

The fourth distinction of this study is that acquisition and utilization of resources are appraised through benefit and cost analyses in an economic sense. The costs are composed of all capital costs, maintenance costs, and operating costs of the infrastructure and the superstructure of the system. The benefits, however, are considered as cost savings accruing to the society through improvements of the system in effectiveness and efficiency. These cost savings, for example, may come from a reduction in congestion and turnaround time of vessel or a reduction in congestion and turnaround time of cargo.

1.4. Thesis Structure

Chapter Two reviews previous research on multiple resource acquisition and utilization, and explains the existing insufficiencies. It begins with a description of tug/barge systems with a focus on its advantages and disadvantages in water transportation. Then, it defines the proposed transshipment system, *i.e.*, the ORATB system between the Yangtze River ports and the coastal deep water port in Ningbo. The heart of this further definition is to identify the resource acquisition and resource utilization problems within the system. The detailed literature review focuses on examining both resource acquisition models and resource utilization models, and on identifying the insufficiencies and ineffectiveness of those models.

Chapter Three is fully devoted to model development. The structure of the IAIOS model is first introduced. Then, the optimization model for resource acquisition is established using mixed integer programming, including model formulation, model construction, and model execution. Next, the simulation model for resource utilization is developed, including model formulation, model construction, and model execution. Finally, the

IAIOS model is constructed and executed focusing on the sequential adaptive iterations of the optimization model and the simulation model.

Chapter Four presents computational results using the real world case of the Yangtze River ORATB system. It starts by reviewing current shipping practices, future shipping demands, and existing barriers for future development along the Yangtze River. It then introduces the proposed ORATB transport system. The proposed system is aimed at transporting iron ore upstream from the coastal deep sea port located in Ningbo to the Yangtze River ports, and transporting containers downstream from the river ports to Ningbo. The IAIOS model is applied to this case to provide numerical results on how many different sizes of tugs and barges should be deployed and what the utilization and performance levels of these resources are to satisfy the projected shipping demands of iron ore along the Yangtze River. During the IAIOS application, the nature of the adaptive iterations is illustrated through several examples. Sensitivity analysis of the IAIOS model is also performed to analyze alternative shipping investment strategies. In concluding this chapter, we present a parametric cost comparison between the current practice and the proposed ORATB system for transporting iron ore from Australia and Wuhan. The calculation shows that the savings from using the ORATB are quite significant.

Chapter Five presents major the findings and conclusions of this research. Then, it discusses the potential applications of the IAIOS model in other fields of resource acquisition and resource utilization. In addition, the proposed IAIOS modeling approach is further explored with respect to both of its advantages and drawbacks. Finally, future research is planned to improve the proposed IAIOS model.

Chapter Two

2. THE ORATB TRANSPORT SYSTEM

2.1. *Introduction To Tug/Barge System*

On the high seas and coastal waters, in every port and navigable river, tug/barge systems are found operating all the time. In fact, tug/barge systems have long performed an important, and indeed vital, role in sea transportation. In the U.S., for example, 33% of all U.S. waterborne commerce (including foreign traffic) and 66% of all U.S. domestic waterborne commerce are carried by tug/barge systems[4]. A tug/barge system makes a unique transportation team. It is not the fastest nor the most flexibly maneuvering mode of transportation we have, but for sheer efficiency in the movement of vast tonnage of freight it has no peer. At the same time, numerous studies of fuel efficiency show that shallow-draft barge transportation is the most fuel efficient mode of transportation for moving bulk raw materials. As shown in Table 1 barge transport is the least energy intensive method in freight transportation. In moving equivalent amounts of cargo, it consumes less energy than alternative modes.

A tug may pull or push (including alongside push) one barge or any multiple of barges ranging up to as many as 40 barges in push-towing operations or three to four in pull-towing operations, depending on the types of services and the characteristics of the

---

[4] Statistics of Waterborne Commerce of the United States.

Table 1 **Measures Of Freight Transportation Energy Efficiency**[5]

unit: BTUs per net ton-mile

Mode	Operating energy[6]	Line-haul energy[7]	Modal energy[8]
Rail:			
overall	660	1,130	1,720
unit coal train	370	590	890
Truck:			
intercity average	2,100	2,800	3,420
Barge:			
overall	420	540	990
upstream	580	700	1,280
downstream	220	340	620

Source: Congressional Budget Office, U.S. Congress, Energy Use in Freight Transportation, Washington, DC, February 1982.

---

[5] Net ton-miles includes weight of cargo only, excluding carrying unit(s).

[6] Propulsion energy including refinery losses.

[7] Combines operating energy with maintenance energy, vehicle manufacturing energy, and construction energy.

[8] Adjusts line-haul energy for circuitry.

waterway on which the tow is operating. In general, the type of water determines which of the two methods is used, push towing or pull towing.

On most of the inland systems where the water routes are protected by surrounding land masses and where the waters are relatively calm, push towing operations are used. For push towing , the barges are tied rigidly together by steel cables or ropes to form a single unit, and this unit is then lashed solidly against the tug's towing knees. The power unit working at the rear of the tow can handle a greater number of barges at greater speed under more absolute control than can be handled in pull towing operations. The tug with massive power in its propellers also has a set of multiple rudders which afford maximum control for forward, backing, and flanking movements such as are required to navigate the restricted channels of the rivers.

Wind, wave, and tidal actions can break up a tow of vessels lashed rigidly together as is done for push towing operations. Where these conditions exist, a pull towing method is applied in which a tug hauls barges behind on a hawser. This pull towing method naturally prevails in river delta areas and intra-coastal areas. There is a limit to the number of barges which may be pulled on a hawser, however, and it is obvious that a towing vessel can exercise little guidance control over barges being pulled except to provide propulsion power.

Navigation speed is important to tug/barge line operators. So the underwater hull shapes and operating features of barges receive the same attention from the naval architect as does the modern high powered diesel tug. Most barges are designed as single individual units, having a rake or slope on each end. For navigating singly, this form is still the most efficient. However, model testing shows that the assembly of multiple units of this form in a single tow results in greater loss of efficiency by the cumulative drag of many water breaking rakes in the middle of the tow. Some barges are then designed to be assembled into integrated tows having an underwater shape that is nearly the equivalent of a single vessel. Such an integrated assembly made up of several vessels has a lead barge with an easy rake at the bow to minimize the resistance of the water. This lead barge has a square stern for joining with the square end of another barge, thus eliminating any underwater surface break. The trailing barge in an integrated assembly has a short rake on the stern, and the bow of this barge is square. Between the lead barge and the trailing barge, double square-ended barges are inserted. The water resistance of such an integrated tow is nearly equivalent to the smooth underwater lines of a single vessel of equivalent length. A premium benefit though is the increase in capacity due to the added buoyancy of the square ends of the barges.

The fully integrated design concept, however, has the disadvantage that a single barge built for an intermediate position in such a tow, square on both ends, is extremely unwieldy to handle when separated from the other units of the tow, especially in a current. Such barges are also difficult to move around in terminal areas. The water resistance of these barges, if placed in a tow with other barges which do not have matching square ends, makes such use of them prohibitive. Fortunately, there is a useful and successful compromise with the concept of the fully integrated tow. This produces a barge with a well designed rake on one end and square on the other end. Two such barges assembled square-end to square-end provide about 8% increased capacity over two similar barges having rakes at each end. At ordinary towing speeds they have about 18% less resistance, according to the report of the American Waterways Operations, Inc. (1973). By assembling such semi-integrated barges into fleets, the combined effects of added capacity and less resistance permit a typical boat moving a typical tow of such barges to make about 25% more cargo ton-miles per hour that the same boat with the same tow of barges having a rake on each end. At the same time, by having a rake on one end these semi-integrated barges can be handled singly without difficulty.

The integrated high speed tow is generally efficient for the carriage of a large volume of a single commodity over a long distance on a continuing basis. Identical draft of all barges comprising the tow is vital to the efficiency of the operation.

Virtually any commodity can be shipped by water. The inland waterways industry has implemented this theory by developing a variety of types and sizes of barges for the efficient handling of products ranging from dry bulk in open hopper barges to liquid bulk in tank barges, and from dredged rock in dump scows to containers on deck barges. The open hopper barge is the most versatile, least costly, and most popular one in the barge family. With minor modifications it can be adapted to the transportation of literally any solid commodity in bulk or package. The hopper barge is basically a simple double-skinned, open-top box, the inner shell forming a long hopper or cargo hold. The bottom, sides, and ends of the hold are free of appendages and adapt ideally to unloading with clam-shell buckets, hooks, grabs, continuous belt buckets or pneumatic devices. They can accommodate dry bulk-loading commodities, structures and shapes, and heavy bulky vehicles with equal facility.

The open hopper barge is a multipurpose vessel in general use for transporting a wide range of commodities that need no protection from the elements. Open hoppers serve the coal industry and the steel industry by moving both raw materials and finished products. They serve the construction industry in the movement of sand, gravel, crushed rock, limestone, log, lumber and lumber products. They serve the agriculture community in the movement of fertilizer materials. Heavy equipment and machinery, oversized tanks and pressure vessels can also be transported in an open hopper. Open hoppers are generally welded plate construction, usually with double bottoms for greater safety. They are braced to resist the heaviest of external blows as well as to absorb the impact of loading and unloading buckets.

The covered dry cargo barge serves a wide variety of shippers in providing transportation for bulk-loading commodities that need protection from the elements. In general, these barges differ from the hopper barge only in that they are equipped with watertight covers over the entire cargo hold. Several types of covers have been developed. Lift covers can be adapted to any hopper barge without modification of the barge itself. Such covers are handled by shore side facilities, and when not in use can be stacked at the ends of the barge. This type of cover is ideal for barges operating in both grain and ore service, for instance, since each of the several covers can be equipped with small hinged grain hatches so that cargo can be loaded and unloaded with grain legs or pneumatic devices without removing the hold covers. Rolling covers, though more costly, are also more versatile. One telescoping type, where the covers roll for and aft on tracks installed on the barge, permits the opening of one-half of the hopper at a time. Some variations of the rolling hatch cover permit opening of the entire hopper.

Covered dry cargo barges are used for the carriage of such commodities as grain and grain products, coffee, soybeans, paper and paper products, lumber and building materials, cement, iron and steel products, dry chemicals, aluminum and aluminum products, machinery and parts, rubber and rubber products, salt, soda ash, sugar, and in some cases packaged goods. These barges, like the open hoppers, are generally of welded steel construction.

Tank barges are used for the transportation of liquid commodities. There are three basic types of tank barges, namely single skin tank barge, double skin tank barge, and cylindrical tank barge. Single skin tank barges have bow and stern compartments separated from the midship by transverse collision bulk-heads. The entire midship shell of the vessel then constitutes the cargo tanks. Hydrodynamic considerations require that this huge tank be divided by bulkheads. The hull structural framing is inside the cargo tank.

Double skin tank barges have, as the term implies, an inner and outer shell. The inner shell forms cargo tanks free of appendages and they are thus easy to clean and to line. Poisons and other hazardous liquids require the protection of the void compartments between the outer and inner shells. Moreover, the double skin limits spills in cases of accident and grounding.

Barges having independent cylindrical tanks are used to transport liquids under pressure or in cases where pressure is used to discharge the cargo. Cylindrical tank barge design is used in some cases to carry cargoes at or near atmospheric pressures because of the high efficiency of linings and/or insulation which can be incorporated. Cylindrical cargo tanks are generally mounted on the barge hopper and are thus free to expand or contract independent of the hull structure. For this reason, too, they are preferred for high temperature cargoes, *e.g.*, liquid sulfur or refrigerated cargoes such as anhydrous ammonia.

Deck barges can also serve a variety of purposes. Machinery, vehicles and heavy equipment can be moved aboard such vessels as can most any type cargo that can be tied down and which does not require protection from the elements. The deck barge is a simple box hull, generally with a heavy plated, well supported deck. The high combined center of gravity of the deck cargo and the hull can have an adverse effect on stability of this type of barge so that careful consideration must be given to hull size. More recently, deck barges have been increasingly used to carry containers stacked up to four high. For these container barges, stability is the main issue. One popular way to overcome this difficulty is to load the hull with ballast[9]. One of the advantages of container barges is that the hull can be used as covered dry cargo barges in cases where the container traffic is unbalanced or when there are not enough containers available for a full barge load.

2.2. The ORATB Transport System

One of the most dramatic developments in the barge and towing industry is the increase use of ocean-barging and its emergence as an important factor in the total transportation system. Serious research and experimentation have been going on throughout the maritime industry to develop an improved method of connecting ocean-going tugs and barges. To date there have been several different designs for the linkage of large

---

[9] Materials, usually water in tanks, solely carried to improve the trim and the stability of a vessel.

tug/barge combinations. Some of them are already in service, using carefully matched surfaces and a mechanical system of holding the separable bodies tightly together, thus replacing the traditional winches and wires which hold the tug's forebody into the barge's notched stern. Such new developments in technology and the realization by operators and shippers of its economic potential indicate even greater usage of ocean-barging in the future.

Most recently, a new type of integration between tug and barge, the articulated tug/barge system, has been introduced into the shipping industry jointly by Ocean Tug Barge Engineering of the U.S. and Marine Research Institute of the Netherlands. The articulated tug/barge unit is designed to combine the economics of tug/barge operation with the speed and weather reliability of a ship. A hallmark of this concept that makes it different from the integrated tug/barge is the fact that both the tug and the barge are truly independent vessels able to operate successfully even if not together. The tug is a full internationally classed ocean tug, capable of towing and featuring lines which, while optimized for flow when connected to the barge, are above-average in performance as a towing vessel. Articulated tug/barge systems are designed for maximum speed instead of maximum towing stability. The notch of the typical articulated tug/barge is designed such that in the absence of the "parent" connection-equipped tug, the barge can be pushed in calm waters by any tug capable of handling it with backing wires. The barge can also be towed. The tug is also capable of working with other barges, either towing, rear pushing or alongside pushing. More importantly, articulated tug/barge units can be designed around any of the existing successful connection system.

An innovative service proposed in this study is to apply the articulated tug/barge transport system between coastal ocean and inland river transportation as the technology develops. One of the major advantages of this ocean/river articulated tug/barge (ORATB) system is that it eliminates extra cargo transshipment at a river mouth port when shipments must be first received at coastal deep water port due to the shallow water depth at the river mouth port. In many instances, it is determined that this ORATB system is superior to the water/water transshipment between ocean-going vessels and river-going tug/barge systems in fulfilling transportation demand.

The first superior character of an ORATB system is its ability to separate the manned propulsion unit (tug) from cargo unit (barge) at any cargo supply/demand node. Such decoupling ability permits dropping and swapping operations whereby the manned tug does not stay idle when the cargo compartment is loaded or unloaded. Instead, the tug drops off the barge at a port to be unloaded and then proceeds independently or with an empty or a loaded barge to another port. This decouplable system offers more flexible and efficient operations, particularly when there are multiple types of commodities to be carried and partial loads to be unloaded with a minimum impact on transport efficiency. Additionally, this flexibility allows the same propulsion unit to be used in any mission area as demands and priorities dictate.

The dropping and swapping operation also increases tug/barge utilization. It allows the costly tug and its crew to be utilized more efficiently since they will be spending more time transporting cargo than awaiting cargo operations. At the same time, the barge that remains in port may be used as a waterside storage facility. That is, the barge does not have to be discharged immediately since no crew or propulsion unit is being tied up with it. Instead, the cargo can remain onboard until required so that the barge is used as a warehouse. In this manner, it is possible to save the cost of constructing shore-side storage facilities. Also, since the barge does not require rapid discharge, cargo may be handled at a slower rate using less sophisticated and less expensive port equipment.

The second superior character of an ORATB system is its shallow draft and low height clearance when compared with ships of equal cargo carrying capacity. Such superiority makes the ORATB system a unique advantage over ships in inland water transport. For example, an ORATB system is only two-thirds of the draft and one-half of the height clearance of a ship with the same deadweight carrying capacity. The low draft and height clearance would allow a significantly larger ORATB system to be served in a depth and/or height limited trade route. Table 2 presents the typical characteristics of an ORATB system.

The third superior character of an ORATB system is its low building and operating costs compared with the same sized ship. The construction cost of an ocean-going barge, for example, is usually about 35-40% that of an ocean-going ship of the same carrying capacity. The tug will usually cost about 22-28% of the cost of a ship with a similar horsepower output. As a result, the total construction cost of an ORATB system is only 57-68% of that of a similarly sized ship. In terms of operating costs, crew and supply costs are only about 50% of those of a ship of equal carrying capacity and, as a result, total daily variable costs, excluding financial, insurance, management, *etc.* costs, is normally 55-60% of that of a ship with equal carrying capacity. If we account for a 20% shorter travel distance per day, ton-mile costs of a tug/barge system, including all financial costs and insurance costs, is about 57-65% of that of a ship of equal carrying capacity. Table 3 presents a preliminary discussion on the economics of the ORATB operations.

The fourth superior character of an ORATB system is that they cannot only be built in different horsepower output and sizes, but tugs and barges can also be combined into different combinations of horsepower output and cargo carrying capacity, and thereby speed. For a particular tug, for example, its navigation speed largely depends on the size and the number of barges being attached and its designed horsepower output. The larger the size and the number of barges attached, the slower the ORATB system could navigate. Such a combination of tugs and barges would also have an impact on the frequency of meeting demand on time. On the other hand, a larger ORATB system becomes less flexible to navigate, and may further require a higher standard on berths and berthing facilities.

The fifth superior character of an ORATB system is that of its large cargo carrying capacity compared with rail and road transport. In terms of capacity, a 1,500-ton barge

Table 2 Typical Characteristics Of An ORATB System

Carrying capacity (ton)	10,000	20,000	30,000
Barge length (m)	108	142	169
Barge draft (m)	5	6	7
Barge height (m)	14	16	17
Barge width (m)	26	29	32
Tug towing capacity (kw)	4,200	6,000	7,200
Tug length (m)	18	22	23
Total integrated length (m)	120	158	186
Total integrated height above water (m)	20	20	20
Total integrated width (m)	26	29	32
Total integrated draft (m)	5	6	7
Traveling speed (knot)	10~12	10~12	10~12
Size of crew	10	10	10
Diesel consumption at full horsepower (ton/day)	15.0	21.4	25.7
Diesel consumption at traveling speed (ton/day)	10.3	16.0	18.2

Source: Personal communications with Prof. Ernst G. Frankel.

Table 3 Capital And Operating Costs Of ORATB Systems

Bulk capacity[10] (dwt)	10,000	20,000	30,000
Container capacity[11] (teu)	520	800	1,016
Loaded displacement (ton)	13,400	25,800	37,900
Barge building cost ($ million)	9.61	14.80	18.91
Tug towing capacity (kw)	4,200	6,000	7,200
Tug building cost ($ million)	4.28	5.94	6.60
Daily financial cost[12] ($)	2,610	3,900	4,790
Daily insurance cost ($)	100	140	170
Daily crew cost ($)	400	400	400
Other daily fixed cost ($)	250	250	250
Total daily fixed cost[13] ($)	*3,360*	*4,690*	*5,610*
Daily fuel cost underway[14] ($)	1,430	2,220	2,520
Daily fuel cost at port ($)	170	250	310
Other daily variable cost ($)	250	280	300
Total daily variable cost underway ($)	*1,680*	*2,500*	*2,820*
Total daily variable cost at port ($)	*420*	*530*	*610*
Total daily cost underway ($)	***5,040***	***7,190***	***8,430***
Total daily cost at port ($)	***3,780***	***5,220***	***6,220***

Source: Personal communications with Prof. Ernst G. Frankel.

---

[10] Capable to carry ballast, dry bulk, or liquid bulk under deck.

[11] Capable to carry containers of four-high on deck.

[12] Assuming the cost of financing is 10% and a 20-year amortization.

[13] Assuming there are only one tug and one barge per barge train.

[14] Assuming the average speed is 11 knots, recognizing down and up river speeds differ.

carries as much as fifteen 100-ton jumbo hopper rail cars or sixty 25-ton trailer trucks. A standard barge is 195 feet long; the fifteen rail cars would be 825 feet long; and the sixty trucks would be over a half mile long. A typical size barge tow consists of fifteen barges that have a capacity of 22,500 tons and is approximately one-quarter mile in length. The equivalent capacity of the other modes would be two hundred twenty-five rail cars measuring two and three-quarters miles long, and nine hundred 25-ton trailer trucks stretching 36 miles[15].

Some of the disadvantages of an ORATB system include lower navigation speed and less maneuvering capability. As shown in Table 2, the typical operating speed of an ORATB system is usually between 10 and 12 knots while the speed of a tanker or bulk carrier is about 15 knots. The less maneuvering capability occurs more significant when towing than pushing operations. One of the major difficulties in towing operations is that cargo on deck has less stability, especially when containers are stacked. Another major difficulty in towing operations is that it requires a more effective traffic control system.

2.3. Problem Definition

The ORATB problem described in this research involves both resource acquisition and resource utilization problems. The objective is to determine the optimal acquisition and utilization schedule of the ORATB transport system. The capacity of an ORATB system is directly related to the number and size of the allocated tugs and barges, and their service frequency. Owners and operators of the system invest in order to provide the capacity to meet demands. Determining the optimal number and size of a particular ORATB system requires a trade-off between the costs of the vehicles and the potential costs or penalties associated with not meeting some demands. Serving demands also results in the relocation of tugs and barges. The demand for movements between various locations is often unbalanced, and this implies the need for redistribution of tugs and

---

[15] Assuming 150 feet between trucks.

barges over the network from locations at which they have become idle to locations at which they can be reused. Thus, the availability of an ORATB system which is available for service at any given time also depends upon the vehicle redistribution strategy.

In this research, we consider the ORATB transport problem as two separate problems defined by different time horizons. In the long-term, we consider the strategic planning problem which is concerned with the size and number of tugs and barges, subject to the projected demand constraints. In the shorter-term, we consider the tactical planning problem which is concerned with how best to utilize the ORATB system, given their compositions under local conditions. The interaction between the strategic planning and the tactical planning, and their combined effects on the capacity and efficiency of a transportation system are therefore the focus in this research.

2.4. Literature Review On Resource Acquisition And Utilization

Since Dantzig's (1954) benchmark initiatives of programming techniques over forty years ago, mathematical programming procedures have made great progress, both in terms of an ever wider span of applications and with respect to methodological elaborations and refinements. While optimization techniques have been applied extensively to strategic and utilization problems, their applications in the shipping field have been less numerous. In this section, however, we focus on reviewing methodologies and applications of optimization techniques in the shipping industry.

Many variables affect the total cost of ocean transportation, ranging from selection of design parameters to reduce resistance to selection of power plant and combustibles to reduce operation costs. Among all the factors, however, routing and scheduling are recognized to be of the greatest importance. As a matter of fact, routing and scheduling of vehicles have been the focus of much research and application in the past. A routing problem represents the resource acquisition problem, where an optimal number of vessels with different characteristics are optimized among the selected routings. A scheduling problem, on the other hand, is under the category of resource utilization, where optimal schedules of each ship in a fleet are determined.

While routing and scheduling planning techniques have been applied successfully mainly to land and air transportation system, limited research has been done in the area of managing waterborne shipping systems. Some explanations for this phenomenon may be the following:

1. Waterborne shipping operations are complex and lightly structured. Basic modeling cannot easily represent all of the various aspects of this trade;
2. Uncertainty is an important factor in those operations. The probability of meeting a schedule is rather low mainly due to weather conditions, mechanical or labor problems and port congestion;
3. The charter markets, representing a substantial portion of the shipping markets, are very competitive, and the chartering rates fluctuate significantly. The factors greatly determined the success of operations are the selection of the type of trade and operation, and the capital investment decisions; and
4. The industry has a long tradition of being conservative. Few people are aware of sophisticated modeling techniques that can be applied to routing and scheduling decision making processes.

For a rather wide audience, including naval architects, marine engineers, meteorologists, naval officers and ship masters, the word "routing" in ship operations usually means "weather routing". The weather routing problems, however, deal with the selection of a track between two given ports on the basis of expected weather conditions to achieve a minimum transit time and avoid the negative effects of storms. In this research, however, the words "routing" and "scheduling" are taken as what they mean to mathematical programmers and operations research practitioners, *i.e.*, resource acquisition and resource utilization.

Dantzig (1954) formulated the first linear programming model in the maritime field. The objective function was to minimize the number of tankers needed to meet a required schedule, giving the loading and discharging schedules. Later on, many efforts were made to either refine such an application or expand such an application to other related areas in the ocean shipping industry. Flood (1954) treated the same problem by minimizing the total distance in ballast. Further extensions to Dantzig's model were addressed by Briskin (1965), Bellmore (1968), and McKay (1974) who considered multiple products in tanker scheduling. Bellmore, for example, considered a fixed fleet of tankers with different carrying capabilities, speeds, and operational costs. The delivery dates were allowed to be within a prescribed interval of time, and partial loadings were permitted. The objective of the problem was to determine a schedule and a routing for the fleet that had maximal utility. The author defined a utility that reflected the desirability to deliver on time and the cost associated with the empty legs (penalty). In this model, the feasible shipping schedules and tanker routes were equivalent to a set of flows in a network. This network was acyclic, *i.e.*, all arcs represented a forward progression in time. Some arcs had bundle capacities which represented the amount to be shipped. Others were restricted from use by tankers of some type, as was the case in real transportation problems.

Other linear programming models were developed by Laderman (1965) and Applegren (1969) to assign shipments to available dry bulk-cargo ships. Later, Applegren (1971) and Ronen (1979) first solved the same problem using integer programming models. Rana (1985) expanded the application of mathematical programming to containership operations using a mixed integer non-linear programming model. Rao (1968) developed the first linear programming model that minimizes the chartering plus the operating costs. Two decades later, Rana (1988) developed the first mixed integer programming model on the routing of time-chartered ships available in the market. Magnanti (1981) was the first to study the perspectives and prospects for combinatorial optimization and vehicle fleet planning. He also described several alternative models for vehicle routing problems and exhibited the relations that existed among those approaches. For a more complete summary of literature on ship routing and scheduling models, see Ronen (1982), Alexis (1982), and Rana (1985).

The review of ship routing and scheduling models presented by Ronen (1982) and Alexis (1982) are important references in this literature review. A typical example of a linear programming model is presented below, where a single commodity is shipped by k vessels ($k = 1, \ldots, K$) from origin port i ($i = 1, \ldots, I$) to destination port j ($j = 1, \ldots, J$). Other notations are the following:

A_{ij}: amount of commodity to be shipped from port i to j;

V_k: capacity of vessel k;

T_{ij}^k: total time for vessel k to load at port i, travel from i to j and unload at j;

t_{ij}^k: time for vessel k to travel from port i to j in ballast;

T_k: time available for vessel k during the planning season;

C_{ij}^k: cost of a trip from port i to j for vessel k; and

c_{ij}^k: cost of a trip from port i to j for vessel k in ballast.

The decision variables are:

X_{ij}^k: number of trips from port i to j for vessel k; and

x_{ij}^k: number of trips from port i to j for vessel k in ballast.

Possible objective functions are:

1. Minimize total costs: $\sum_k \sum_{i,j} (X_{ij}^k C_{ij}^k + x_{ij}^k c_{ij}^k)$;

2. Minimize total operating time: $\sum_k \sum_{i,j} (X_{ij}^k T_{ij}^k + x_{ij}^k t_{ij}^k)$; or
3. Any other combination.

The constraints are:

1. Ship time constraints: $\sum_{i,j} (X_{ij}^k T_{ij}^k + x_{ij}^k t_{ij}^k) \leq T_k$ for all *k*;
2. Meet demand constraints: $\sum_k (X_{ij}^k V_k) \leq A_{ij}$ for all *i* and *j*;
3. For each ship, the number of trips to a port equals to the number of trips from this port: $\sum_i X_{ij}^k = \sum_i x_{ij}^k$ for all *j* and *k*,

   $\sum_i X_{ij}^k = \sum_i x_{ij}^k$ for all *i* and *k*; and
4. Non-negativity constraints: $X_{ij}^k \geq 0$, and $x_{ij}^k \geq 0$ for all *i*, *j* and k.

Other linear constraints may be added, depending on the assumptions of the model.

In the problem where a fleet of vessels has to meet a fixed schedule, a network formulation is convenient. A delivery is defined by a loading port, a loading time and an unloading port. We can build a "space time" network $G = (N, A)$ with nodes N consisting of a source node s, a sink node t, nodes (*i*, t_i^l) for every loading port *i* and associated loading date t_i^l, and (*j*, t_j^m) for every unloading port *j* and associated loading date t_j^m. For each delivery, the departure date t_j^m from the unloading port is obtained by adding the transit time (including loading and unloading time) from port *i* to *j* to t_i^l. Concerning the arcs of the network, their associated upper capacities c, and utilities v are:

s - i, t_i^l	for every loading port i and associated date t_i^l, where $c = \infty$, $v = 0$;
i, $t_i^l - j, t_j^m$	for every pair corresponding to a delivery, where $c = c_{ij}^{ll}$, $v = v_{ij}^{ll}$;
j, t_j^m - i, t_i^l	for every pair for which it is possible to travel from node j at time t_j^m to node i at time t_i^l (corresponding to ballast legs), where $c = \infty$, $v = 0$; and
j, t_j^m-t	for every unloading port j and associated date t_j^m, where $c = \infty$, $v = 0$.

After defining for every node e, a potential $\pi(e)$ as the length of the longest path from e to t, reduced costs are then defined as:

$$a(e,f) = \pi(e) - \pi(f) - v(e,f).$$

The resulting transshipment problem is solved using an algorithm to obtain the maximum utility solution, and individual ship schedules are constructed from the optimal arc flows in G.

For liner operations[16], the operator's objective is to maximize profits per unit of time. Scheduling liner vessels is a complex task which comprises selecting cargoes, keeping track of them, taking into account port delays and cargo handling rates, as well as other uncertainties. Due to the complexities and the uncertainties involved, most models, especially those which have been implemented, consist of simulation and heuristic procedures.

A simple deterministic simulation model was first developed by Olson (1969) to provide a medium term regular schedules rather than a long range planning for a fleet of vessels involved in liner shipping. In the model, Olson described the main elements that

---

[16] Liner vessels operate on a given route under a fixed schedule.

influenced the operations as follows: customer seasonal demands, ship capacities, frequency of service, shore-side facilities, dry docking, and lay-up. Revenues and direct operation costs were displayed using tables of average values. Parameters were chosen to define the level of profit above which a ship would sail. Arbitrary, but reasonable operating constraints were selected. The model was also used to investigate the effects of various factors such as waiting in port for additional cargo or increasing competition in the trade under concern.

More elaborate models were developed using stochastic simulation processes. Kydland (1969) formulated such a model for planning purposes. Given a sailing frequency, Kydland determined the optimal number of required ships by using linear programming in a sub-problem of a stochastic simulation model. Almogy (1970) considered the problem of cargo selection for a ship which had to visit N ports in a given order. In Almogy's model, cargoes were available for shipping from a port to those the ship visited later following the schedule. The objective function was to maximize the profit per unit of time for the rest of the route under following assumptions:

- A single type of cargo was shipped from one port to another (this caused no loss of generality since a port could be split to accommodate several types of cargoes);
- Direct costs, loading, and unloading times were linear functions of cargo volume carried by the ship;
- The speed of the ship, the fixed time in ports, and the operating costs were given constants; and
- The amount of cargo available from port i to j was an independent random variable.

At each stage, the contribution to the total objective function was expressed in parametric form. The objective function was modified as a sum of separable linear functions. In fact, this representation was close to that of a linear multi-stage stochastic programming problem which emerged later.

Nemhauser (1972) developed dynamic programming for the case of bulk shipping. The objective was to maximize revenues over the planning horizon. One of its contribution was the condition for discrete time approximations to converge to an optimal continuous time solution. The model offered *n* different types of services with a discrete grid of planning horizons. For each discrete value of time, a vector of dimension *n* indicated whether a service was offered or not. The demand for service depended on its frequency and arrival time. The dynamic programming algorithm was used to recursively compute the maximum profit and thus determined the optimal frequency of the service.

Boffey (1979) provided a new description of the real world scheduling problem by developing an interactive computer program and heuristic optimization model for scheduling containerships on the North Atlantic route. In the model the scheduler chose the speed of the ship and the ports to be visited. Then, for each route, the timing, the minimum transit time, and the total slack were explored. The program indicated whether it would be possible or not to add another port call. The heuristic model was used to alter the routes by adding one port at a time which yielded the largest increase in profit. Boffey's model was not a true optimization tool, but instead a method that provided information on profitability, timing, transit times and total slack for different inputs of ship speeds and combinations of ports to be called. This interactive program, however, was rather unsophisticated, but it facilitated communications among decision makers at different divisions of the shipping line. Moreover, the model results could be easily understood and implemented.

Baker (1981) presented an interactive vessel scheduling system put into service by Exxon Oil Company. The problem was to schedule vessels for the distribution of product oils from a single source to 23 destinations in a three-month time period. The demands were deterministic and a ship could partially unload the cargo at several ports. The model consisted of a linear programming formulation for voyage selection and a network formulation which checked terminal inventories to determine optimal vessel loading and unloading operations for a given schedule. The interaction was achieved through a sequence of runs of the program and analysis of the results to generate new data.

Scott (1981) developed another interactive ship scheduling system which had been used by Bethlehem Steel Corporation. This interactive time-sharing system was built to assist decision makers in both long-term planning and day-to-day operations. It also provided information on financial opportunities. In this case, the company undertook shipping raw material for its own plants, but could also contract to transport non-company bulk commodities. Factors considered in this model include characteristics of the vessels, their abilities to carry alternative commodities, as well as the cost and size of port facilities. The main objective was to minimize the ballast legs' length but the system intended to be flexible enough to analyze the potential financial opportunities and quickly compute the effects of any change in the plan. This flexibility was achieved by means of the following four major subsystems:

1. Analyze and compare the voyage costs of different vessels on different routes;
2. Analyze the number of trips for each vessel that will maximize profit (with the use of a linear program optimization module);
3. Compute the critical dated (arrival and departure times at ports); and
4. Generate fleet management reports.

Though the evaluation of benefits from this interactive model was difficult, management was convinced that shipping resources were being assigned in a near-optimal manner and further development was expected. As Magnanti (1981) pointed out, heuristic models were more readily accepted by decision makers because they were easier to program and understand. As a matter of fact, heuristic models often can provide fairly good results for a wide range of practical transportation problems. Nevertheless, in some situations, such as the scheduling of large tankers, the additional effort of developing the exact methods is justified by the large savings through using a more exact model. Moreover, exact procedures are useful in providing sensitivity analysis results. The size of a realistic model in ship scheduling is enormous, but new advances in integer programming and evolution of optimization methods are expected to make the solution of those problems possible.

Perakis (1987) developed a rather comprehensive model in fleet deployment, and obtained appreciable improvement. The model focused on the problem of minimum cost operation of a fleet of ships that had to carry a specific amount of cargo between two ports in a given time period for a specific and fixed contract price. The operating costs were modeled as a non-linear function of the ship's speed and the non-linear constrained optimization problem was solved with a non-linear optimization algorithm and Lagrange multiplier techniques. Sensitivity analyses were performed to study the effects of small or large changes in one or more cost components on the total cost. Later on Perakis (1991) systematically presented the background, problem formulation and solution approaches on fleet deployment optimization for liner shipping. He further developed a detailed and realistic model for the estimation of the operating costs of liner ships on various routes, and presented another comprehensive linear programming formulation for liner fleet deployment problem. Independent approaches for fixing both service frequencies and vessel speeds in different routes were also presented in this paper.

Bremer (1992) and Perakis (1992) expended Perakis' original models into tanker scheduling optimization problems, focusing on short-term operational scheduling for the Chevron Shipping Company. An integer programming formulation for schedule optimization was developed and successfully implemented. In this model, optimization of a given schedule involved three stages. In the first stage, input files were prepared which contained all the necessary information concerning vessels, cargoes and ports. In the second stage, a self-coded FORTRAN program took the input files and generated a list of feasible schedules and the associated costs for each tanker, and then prepared a file suitable for input into the LINDO[17] software. In the last stage, LINDO took the input file

---

[17] Linear, Interactive, Discrete Optimizer.

and attempted to solve the integer program for the overall optimal schedule. Use of the model and the potential for cost savings were demonstrated with a realistic scheduling situation based on Chevron's practice information.

In summary, we can see that while optimization techniques have been applied extensively to strategic and utilization problems, their applications in the maritime shipping industry have been less numerous. In the shipping industry, we see that most of the research efforts have been focused on large liner services instead of regional transportation system planning. Moreover, no attempts have been made in establishing transshipment network between the coast and the inland waterway. In past decades, routing and scheduling problems have been the focus of much research, but most of the attention has been devoted to scheduling vehicles under given information on cargo and vessel availability and/or port and time constraints. Throughout virtually all these works, the focus has been on the efficient utilization of a given fixed vessel fleet. The number of available vessels of various types are assumed to be specified as known data, and the models attempt to find the most efficient routing and scheduling for those vessels. Such formulation can achieve benefits associated with reduced operating costs, but do not address the longer term decisions on investment of the shipping fleet.

2.5. Chapter Summary

In this chapter, we briefly introduced the tug/barge system with a focus on its advantages and disadvantages in water transportation. A tug/barge system makes a unique transportation team. It is not the fastest nor the most flexibly maneuvering mode of transportation we have, but it has no doubt on sheer efficiency in the movement of vast tonnage of freight. In fact, tug/barge systems are extensively used in the shipping industry all over the world. New innovative uses between coast and river transport and new technologies in forming the articulated tug/barge system surely will bring a new bright future for tug/barge systems in ocean/river transshipment service.

The proposed ORATB system combines the economics of tug/barge operation with the speed and weather reliability of a ship. It eliminates redundant transshipment nodes and therefore saves time and costs for shippers in door-to-door transportation service. Other benefits for using the ORATB system include the following:

- The ORATB system increases the flexibility and efficiency of the shipping service because it has the ability to uncouple the manned propelling unit from the cargo compartment unit;
- The ORATB system has a unique advantage over ships in river transport because it has a shallower draft and a lower height clearance than ships of equal carrying capacity;
- The ORATB system is more economic to establish and maintain because it has a low building and operating costs compared with the same sized ships;
- The ORATB system has the flexibility in meeting demand on time and at low costs because tugs and barges can be easily combined into different combinations of horsepower output and cargo carrying capacity; and
- The ORATB system is superior than rail and road transport because it has a large carrying capacity and low fuel consumption rate.

Some of the disadvantages of an ORATB system, however, include lower navigation speed and less maneuvering capability when compared with same sized ships. The less maneuvering capability occurs more significant when towing than pushing operations. One of the major difficulties in towing operations is that cargo on deck has less stability, especially when containers are stacked. Another major difficulty in towing operations is that it requires a more effective traffic control system.

Then, we presented problem definition in this chapter. The ORATB transport problem described in this research involves both resource acquisition and resource utilization. The objective is therefore to determine the optimal acquisition and utilization schedule of the

ORATB transport system. In this research, we consider the ORATB transport problem as two separate problems defined by different time horizons. In the long-term, we consider the strategic planning problem which is concerned with the number and size of tugs and barges, subject to the projected cargo requirements. In the shorter-term, we consider the tactical planning problem which is concerned with how best to utilize the ORATB system, given their compositions. The interaction between the strategic resource acquisition planning and the tactical resource utilization planning, and their combined effects on the capacity and efficiency of a transportation systems are therefore the focus in this research.

The last part of this chapter focused on reviewing previous research on resource acquisition and utilization. In the shipping industry, most of the research effort has been focused on large liner services instead of regional transportation system planning. In the past decades, routing and scheduling problems have been the focus of much research, but most of the attention has been devoted to scheduling vehicles under given information on cargo and vessel availability and/or port and time constraints. Throughout virtually all these works, the focus has been on efficient utilization of a given fixed vehicle fleet. The number of available vehicles of various types are assumed to be as known data, and the models attempt to find the most efficient routing for those vehicles. Such formulation can achieve benefits associated with reduced operating costs, but does not address the longer term decisions on investment of the shipping fleet. In addition, there are very few models that focus on both of the acquisition and utilization problems simultaneously.

Finally, it is necessary to point out that owing to the complexity and the uncertainty of the operations in the shipping industry, most models consist of simulation and heuristic procedures. Heuristic and simulation models are more likely to be accepted by decision makers because they are easier to program and understand. In addition, there are less number of simplified assumptions about the real transport problem in a heuristic or simulation model than in an optimization model. As a matter of fact, heuristic and simulation models often can provide fairly good results, *i.e.*, near-optimal solutions, for a wide range of practical transportation problems.

Chapter Three

3. MODEL DEVELOPMENT

3.1. Introduction

The ORATB problem described in this research deals with the determination of the optimal acquisition and utilization of a tug/barge transport system. The ORATB problem can also be characterized as a capacity planning problem in general, where a great deal of uncertainty exists as to the demands and the operations of the system. It is the goal of this thesis research to solve the ORATB resource acquisition and utilization problems through the proposed IAIOS modeling approach.

The IAIOS modeling approach illustrates how to complement the strengths of two important modeling techniques: mathematical programming and simulation. The adaptive and iterative nature of this IAIOS modeling approach is that the optimization model first suggests the acquisition of the ORATB system, and then the simulation model evaluates the performance of the acquisition in a simulated environment. If the detailed evaluation of the acquisition is unacceptable, constraints and cost parameters are added and/or modified in the optimization model and the procedure is repeated until no significant improvement in performance can be achieved. As a result, the IAIOS model is embellished through simple and direct modifications. Such an adaptive iteration process can also be used to generate recommendations to improve conditions that affect the performance of the ORATB system, particularly those site-specific conditions or local constraints.

The IAIOS approach provides a viable way of eliminating the weaknesses inherent in both of the optimization and simulation modeling approaches. On one hand, the optimization model cannot incorporate all the detailed local constraints, sequential relationships, as well as explicit uncertainties, without becoming so large and so complicated that it would be impossible to solve. On the other hand, the simulation model does not generate any alternative acquisition schedule, but merely evaluates those presented to it. Although it is theoretically possible to develop a single model to support these two levels of decisions, that approach seems unacceptable for the following two reasons:

- Present computer and methodological capabilities do not permit solution of such a large detailed mathematical model; and
- A single mathematical model does not provide sufficient cognizance of the distinct characteristics of time horizons, scopes, and information content of the various decisions.

By integrating these two approaches through adaptive iterations, it is possible to generate acquisition schedules and evaluate the performance of each acquisition against a set of detailed and site-specific conditions sequentially.

The adaptive iteration approach also facilitates decision makers to interact with the model at different levels. The IAIOS approach allows for comprehensive testing of a wide variety of options that can be used to improve the robustness and efficiency of the system. Therefore, we solve the ORATB transport problem by means of the proposed IAIOS modeling approach.

3.2. Structure Of The IAIOS Model

The structure of the IAIOS model is presented in Figure 1. The adaptive and iterative nature of the modeling process is indicated by the feedback branches in the figure. There are two distinct levels of decisions in the design of an ORATB transport system. At the first level, the optimization model solves the tug/barge acquisition problem subject to the projected demands. At the second level, the simulation model utilizes the tug/barge acquisition information derived from the optimization model as inputs to evaluate the utilization of the acquired resources and the performance of each activity in the system.

The time horizon of the optimization model is at least as long as the deployment period of the ORATB system, typically one year, whereas the simulation model usually addresses decisions on a weekly or daily basis. The two models are coupled and highly interactive through adaptive iterations. The optimization model is oblivious to daily or weekly changes in the demand patterns, and does not consider bottlenecks or queue formation in front of berthing facilities. It does, however, bound the daily or weekly operation of the facilities. On the other hand, in the scheduling of the ORATB system and assigning of berth facilities, the simulation model determines the utilization of the facilities (undertime or overtime) and recognizes how demand uncertainties affect the measures of the ORATB system. This information can alter the acquisition of the ORATB system determined by the optimization model. It is proposed that the two models be solved sequentially, the optimization model first, with adaptive iterations of the two models as necessary to address the interaction.

3.3. The Optimization Model

The primary objective of the optimization model is to provide a preliminary acquisition of tugs and barges in different numbers and sizes. This acquisition is made by attempting

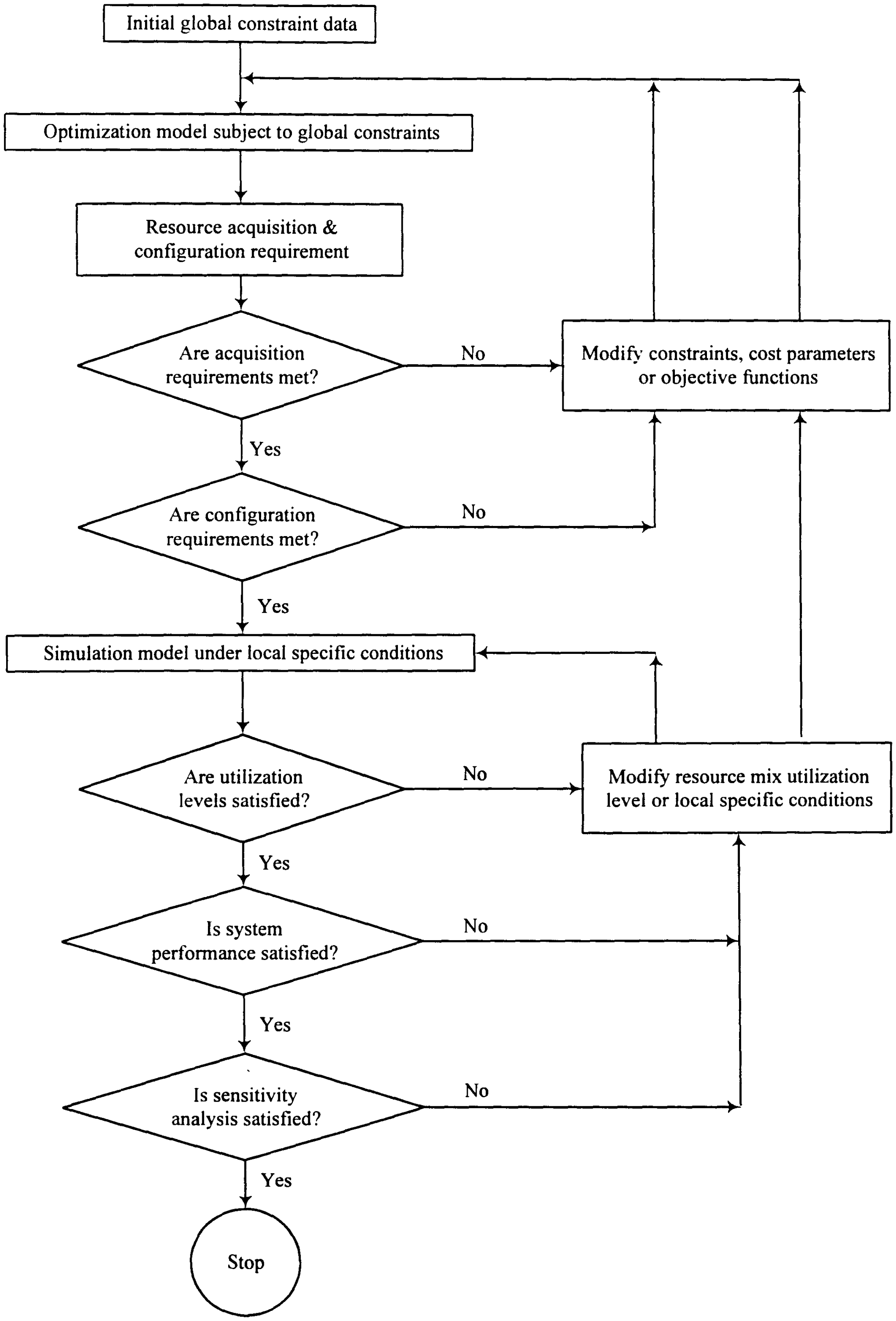


Figure 1 The Structure Of The Proposed IAIOS Model

to minimize the relevant costs associated with the recommended ORATB system, while observing several aggregate constraints on transport demands, tug/barge interdependent relations, and draft limitations, *etc.*

Several assumptions have been made to simplify the model structure, while maintaining an acceptable degree of realism in the problem representation. First, demand requirements are assumed to be known and deterministic. This assumption, however, is relaxed in the simulation model where the impact of uncertainties in transport demands are evaluated.

Second, the size and composition of the ORATB system are assumed to be fixed throughout the planning horizon once they are determined by the optimization model. Hiring and firing options are precluded in this study. Moreover, we have not allowed for overtime to be used as a method for absorbing demand fluctuations in the optimization model. Rather, overtime is reserved as an operational device to deal with operational uncertainties in the simulation model.

Finally, under performance of the system due to operator errors or severe weather conditions is not explicitly considered. Rather, the productivity figures that are used include allocations for a normal amount of such incidents. Similarly, no provision is made for mechanical breakdowns or preventive maintenance.

A linear mixed integer programming model is developed to obtain the optimum composition and mission of the ORATB system. Some basic characteristics of the model are as follows.

1. The mission is satisfied by allowing the ORATBs to make the required number of voyages per year, *i.e.*, the planning period in this model is one year;
2. These voyages may include one load-carrying leg and a return leg in ballast or two load-carrying legs plus one or two in ballast;

3. The number of tugs and the number of barges required depend on the number of voyages and their duration;
4. For each possible voyage, various types of tugs and barges may be deployed. The role of the linear mixed integer program is to select the combination of voyages and tugs/barges that executes the assigned mission at minimum total annualized average cost (*TAAC*);
5. For each leg of a voyage, a calculation is needed to determine the maximum amount of cargo that the candidate barge may carry. This is a function of the characteristics of the tug and the barge, including tug's towing restriction, barge carrying capacity, tug/barge draft restrictions, and the characteristics the cargo;
6. For each voyage, a calculation is required to ascertain the time taken by any eligible ORATB system. The time at sea depends on the system's speed and voyage length. The time in port depends on the quantity of cargo and the rates of loading and discharging, when a dropping and swapping is not performed. The rates are considered to depend primarily on the port and the type of cargo in question, and differential loading or discharging rates between different barges are not considered; and
7. As implied in the foregoing, the *TAAC* of the candidate ORATB system to fulfill the mission is required. The *TAAC* consists primarily of capital costs, the annual fixed costs, and operating costs of the ORATB system. In the linear mixed integer programming model, it assumes constant costs per tug and per barge, independent of the number of tugs and barges.

It is important to point out that most of these characteristics of the ORATB system are altered and tested in the simulation model. These alteration and testing results are then feedbacked to the optimization model iteratively so that the IAIOS model is refined.

3.3.1. Optimization Procedures

The optimization technique is backed by many mathematical theories and algorithms. The application of the optimization model presents a scientific approach to decision making of how best to design and operate a system with limited resources. When the optimization model is used to solve a system's problem. the following seven-step procedure, as shown in Figure 2, should be followed.

Step 1 Formulate the Problem

The problem of the system must be first defined. Defining the problem includes specifying the objectives and the parts of the system that must be studied before the problem can be solved;

Step 2 Observe the System

Next, we collect data to estimate the values of parameters that affect the system. These estimations are used to develop (in Step 3) and evaluate (in Step 4) the optimization model of the problem;

Step 3 Formulate the Problem

In this step, we develop an optimization model of the problem. When no analytic model exists due to the complexity of the problem, we must often develop a simulation model which enables a computer to approximate the behavior of the system;

Step 4 Verify the Model and Use the Model for Prediction

We now try to determine if the optimization model developed in Step 3 is an accurate representation of reality. To determine how well the model fits reality, we first determine the validity of the model for the current situation. If the model's predictions are not close to the actual values, a new model is surely needed. In such case, we return to either Step 2 or 3 and develop a new model that better describes the actual situation. If the

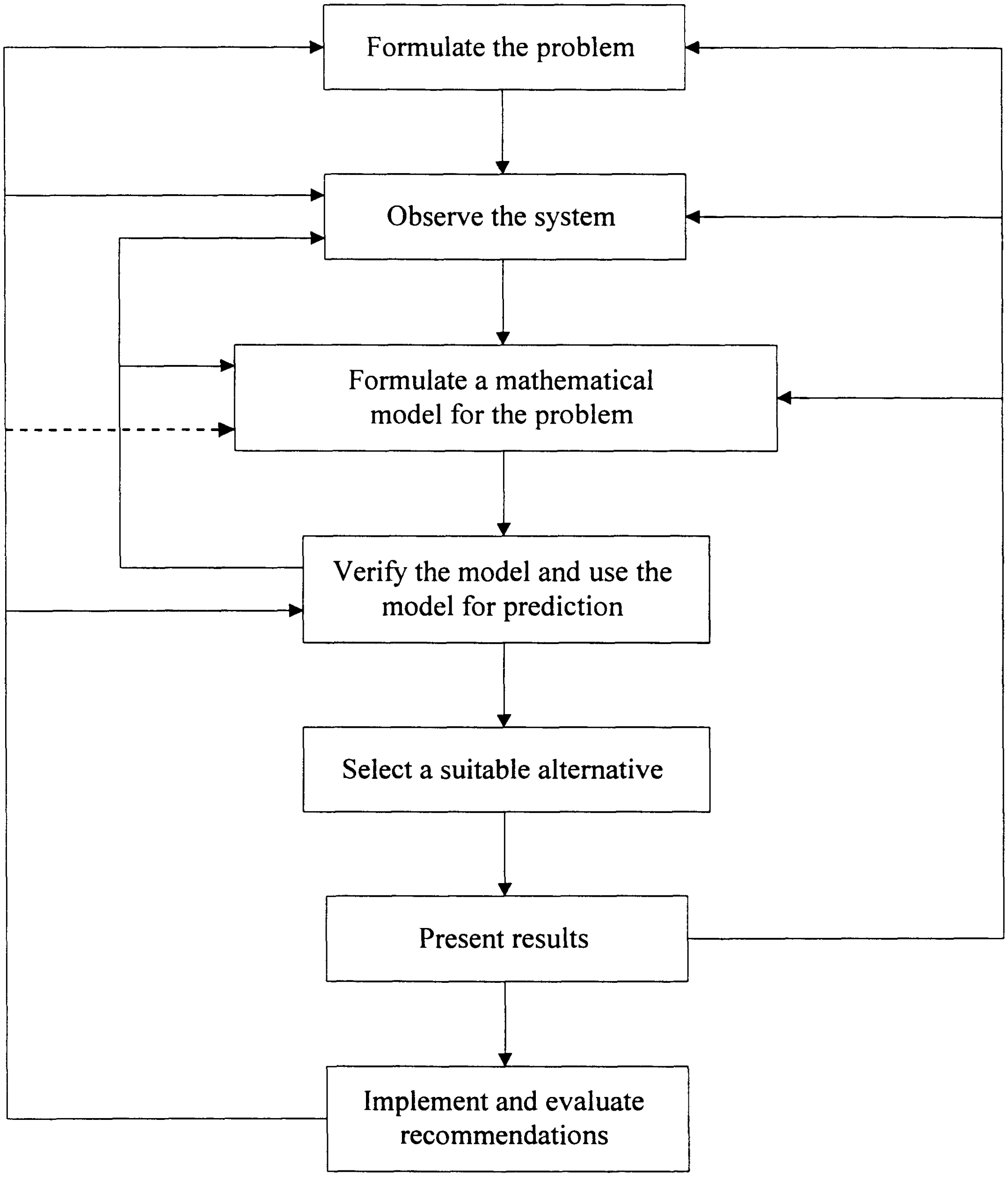


Figure 2 **General Optimization Procedures**

Source: Winston (1990).

first model fits the current situation, we then check it under different situations before applying it;

Step 5 Select a Suitable Alternative

Given a model and a set of alternatives, we now choose the alternative that best meets the objectives. Sometimes the set of available alternatives is subject to certain external restrictions or constraints. And in some other cases, the best alternative may be impossible or too costly to determine based on current development of the optimization algorithms;

Step 6 Present the Results and Conclusions of the Study

In this step, we present the model results and recommendations from Step 5 to decision makers. After presenting the results, we may find that the decision makers are not satisfied with the results or do not approve of the recommendations. These may result from incorrect definition of the problem or from failure to involve the decision makers from the beginning of the project. If this is the case, we should return to Step 1, 2 or 3; and

Step 7 Implement and Evaluate Recommendations

We must aid to implement the recommendations once they are accepted. The system should be constantly monitored and updated dynamically as the environment changes to ensure that the recommendations are enabling the system to meet its objectives. If the objectives are not met after the implementation, we should again return to Step 1, 2 or 3 and re-examine the model.

The optimization modeling enables the user to optimize the objective function under the restriction that all constraints are applied. This method features the advantage that it quickly resolves the problem without having to simulate all possible combinations between the decision variables. This does not, however, mean that simulation becomes an unnecessary game; because there are many cases where optimization modeling cannot deliver satisfied results.

The input format of an optimization model first requires that each equation, *i.e.*, the objective function and the constraints, is expressed in a common dimension, such that each item which conditions the optimization process is formulated in a consistent way. These dimensions, however, may vary across the equations. Thus the objective function may take a monetary dimension (costs, revenues, profits or opportunity costs) whereas the constraints express operating frequencies, demand units of a variety of operating conditions.

Secondly, all variables introduced in the constraints must feature in the objective function. Otherwise, the optimization could not be reached in a comprehensive way, *i.e.*, taking into account the potential activation of constraints.

Thirdly, constraints are either:

- Inequalities, which express the necessity that a number of quantified items should not exceed a given ceiling or violate a minimum value. For each inequality constraint, the program itself calculates the extent to which the inequality applies through the introduction of a so-called slack variable. The program thus calculates those slacks for each inequality, of which the economic meaning denotes idle capacity; or
- Equalities, which state that a number of items have to equal a given amount. Generally, those equalities follow a concept, similar to the inequalities, except for the slacks which are replaced by the explicit introduction of a penalty.

Lastly, the objective function and the set of constraints must be a linear combination of the introduced variables for linear programming. Otherwise, a different approach, non-linear programming, is required. It should nevertheless be mentioned that a substantial number of so-called non-linearities can be linearized through either adjustment procedures (stepped functions) or the organization of the linear programming format itself.

3.3.2. Formulation Of The Optimization Model

In the proposed linear integer programming model, the objective is to minimize the *TAAC* of the ORATB system. The following decisions are to be made in the model simultaneously.

1. Upstream and downstream routing;
2. Number and type of tugs to be deployed in each route;
3. Number and type of barges to be deployed in each route; and
4. Service frequency in each route.

Then, the corresponding decision variables are:

X^t_{ijk}	A 0-1 variable of downstream routing indicating whether that cargoes of port *i* are transshipped through *j* to *k* for period *t*, $X_{ijk} \in [0, 1]$. If $X^t_{ijk} = 1$, the transport scheme is accepted, and if $X^t_{ijk} = 0$, it is rejected;
Y^t_{ijk}	A 0-1 variable of upstream routing indicating whether that cargoes of port *i* are transshipped through *j* to *k* for period *t*, $Y^t_{ijk} \in [0, 1]$. If $Y^t_{ijk} = 1$, the transport scheme is accepted, and if $Y^t_{ijk} = 0$, it is rejected;
M^t_{ijks}	A 0-1 variable of tug type, $M^t_{ijks} \in [0, 1]$. If $M^t_{ijks} = 1$, the *s*-type tug is to be deployed in route *i-j-k* for period *t*, and if $M^t_{ijks} = 0$, it is rejected;
N^t_{ijkl}	A 0-1 variable of barge type, $N^t_{ijkl} \in [0, 1]$. If $N^t_{ijkl} = 1$, the *l*-type barge is to be deployed in route *i-j-k* for period *t*, and if $N^t_{ijkl} = 0$, it is rejected;
MM^t_{ijks}	An integer variable indicating number of *s*-type tug to be deployed in route *i-j-k* for period *t*, $MM^t_{ijks} \in [0, S]$; and

NN^{t}_{ijkl}	An integer variable indicating number l-type of barge to be deployed in route i-j-k for period t. $NN^{t}_{ijkl} \in [0, L]$.

where,

i	origination port; $i \in [0, I]$, a discrete combination;
j	transshipment port; $j \in [0, J]$, a discrete combination;
k	destination port; $k \in [0, K]$, a discrete combination;
s	type of tug, $s \in [0, S]$, a discrete combination;
l	type of barge, $l \in [0, L]$, a discrete combination; and
t	time period, $t \in [0, T]$, a discrete combination.

On the other hand, cost coefficients are:

CT_i	annualized average fixed and variable costs of port side container infrastructures at origination port i, including berthing and loading and unloading facilities, *i.e.*, $CT_i = CTBH_i + CTBE_i$, where $CTBH_i = CTBHF_i + CTBHV_i$, and $CTBE_i = CTBEF_i + CTBEV_i$;
CT_j	annualized average fixed and variable costs of port side container infrastructures at transshipment port j, including berthing and loading and unloading facilities, *i.e.*, $CT_j = CTBH_j + CTBE_j$, where $CTBH_j = CTBHF_j + CTBHV_j$, and $CTBE_j = CTBEF_j + CTBEV_j$;
CT_k	annualized average fixed and variable costs of port side container infrastructures at destination port k, including berthing and loading and unloading facilities, *i.e.*, $CT_k = CTBH_k + CTBE_k$, where $CTBH_k = CTBHF_k + CTBHV_k$, and $CTBE_k = CTBEF_k + CTBEV_k$;

BK_i	annualized average fixed costs of port side bulk infrastructures at origination port i, including berthing and loading and unloading facilities, *i.e.*, $BK_i = BKBH_i + BKBE_i$, where $BKBH_i = BKBHF_i + BKBHV_i$ and $BKBE_i = BKBEF_i + BKBEV_i$;
BK_j	annualized average fixed and variable costs of port side bulk infrastructures at transshipment port j, including berthing and loading and unloading facilities, *i.e.*, $BK_j = BKBH_j + BKBE_j$, where $BKBH_j = BKBHF_j + BKBHV_j$ and $BKBE_j = BKBEF_j + BKBEV_j$;
BK_k	annualized average fixed and variable costs of port side bulk infrastructures at destination port k, including berthing and loading and unloading facilities, *i.e.*, $BK_k = BKBH_k + BKBE_k$, where $BKBH_k = BKBHF_k + BKBHV_k$ and $BKBE_k = BKBEF_k + BKBEV_k$:
TG_s	s-type tug related annualized average costs, including fixed and variable costs of the tug, *i.e.*, $TG_s = TGF_s + TGV_s$; and
BG_l	l-type barge related annualized average costs, including fixed and variable costs of the barge, *i.e.*, $BG_l = BGF_l + BGV_l$.

Finally, the given transport demands are:

$CTDM_{ik}^{t}$	container flow from port i to k for period t; and
$BKDM_{ik}^{t}$	bulk flow from port i to k for period t.

Then, the objective function is:

MIN (TAAC) *Equation 1*

where,

$$TAAC = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\{[(CT_i + CT_j + CT_k)(\sum_{t=1}^{T} X_{ijk}^{t}) + (BK_i + BK_j + BK_k)(\sum_{t=1}^{T} Y_{ijk}^{t})] + \sum_{s=1}^{S}[TG_s(\sum_{t=1}^{T} M_{ijks}^{t})(\sum_{t=1}^{T} MM_{ijks}^{t})] + \sum_{l=1}^{L}[BG_l(\sum_{t=1}^{T} N_{ijkl}^{t})(\sum_{t=1}^{T} NN_{ijkl}^{t})]\}$$

Equation 2

subject to:

$$CTDM_{ik}^{t} = \sum_{j=1}^{J}\{X_{ijk}^{t}[\sum_{l=1}^{L}(N_{ijkl}^{t} NN_{ijkl}^{t})]\} \quad \forall\, i, k, t$$

Equation 3

$$BKDM_{ik}^{t} = \sum_{j=1}^{J}\{Y_{ijk}^{t}[\sum_{l=1}^{L}(N_{ijkl}^{t} NN_{ijkl}^{t})]\} \quad \forall\, i, k, t$$

Equation 4

$$\sum_{s=1}^{S} M_{ijks}^{t} = 1 \quad \forall\, i, j, k, t$$

Equation 5

$$\sum_{l=1}^{L} N_{ijkl}^{t} = 1 \quad \forall\, i, j, k, t$$

Equation 6

$X_{ijk}^{t} \in [0, 1]$ and integers $\forall\, i, j, k, t$ *Equation 7*

$Y_{ijk}^{t} \in [0, 1]$ and integers $\forall\, i, j, k, t$ *Equation 8*

$M_{ijks}^{t} \in [0, 1]$ and integers $\forall\, i, j, k, s, t$ *Equation 9*

$N_{ijkl}^{t} \in [0, 1]$ and integers $\forall\, i, j, k, l, t$ *Equation 10*

$MM_{ijks}^{t} \geq 0$ and integers $\forall\, i, j, k, s, t$ *Equation 11*

$NN^t_{ijkl} \geq 0$ and integers $\forall\, i, j, k, l, t$ *Equation 12*

$M^t_{ijks} \leq X^t_{ijk} \quad \forall\, i, j, k, s, t$ *Equation 13*

$M^t_{ijks} \leq Y^t_{ijk} \quad \forall\, i, j, k, s, t$ *Equation 14*

$N^t_{ijkl} \leq X^t_{ijk} \quad \forall\, i, j, k, l, t$ *Equation 15*

$N^t_{ijkl} \leq Y^t_{ijk} \quad \forall\, i, j, k, l, t$ *Equation 16*

$MM^t_{ijks} = \gamma\, NN^t_{ijkl} \quad \forall\, i, j, k, s, l, t$ *Equation 17*

where,

$\gamma \in [1, \Gamma]$ and integers *Equation 18*

$MM^t_{ijks} - 100 M^t_{ijks} \leq 0 \quad \forall\, i, j, k, s, t$ *Equation 19*

$NN^t_{ijkl} - 100 N^t_{ijkl} \leq 0 \quad \forall\, i, j, k, l, t$ *Equation 20*

Figure 3 represents the distribution and collection network for this optimization model based on the analysis presented in Section 4.1.1.

3.3.3. General Algebraic Modeling System

The General Algebraic Modeling System (GAMS) was first developed by the World Bank in 1988. GAMS is designed to make the construction and solution of large and complex mathematical programming models more straightforward for programmers and more comprehensive to users of models from other disciplines, *e.g.*, managers. Because it can make concise algebraic statements of models in a language that is easily read by both modelers and computers, GAMS can substantially improve the productivity of

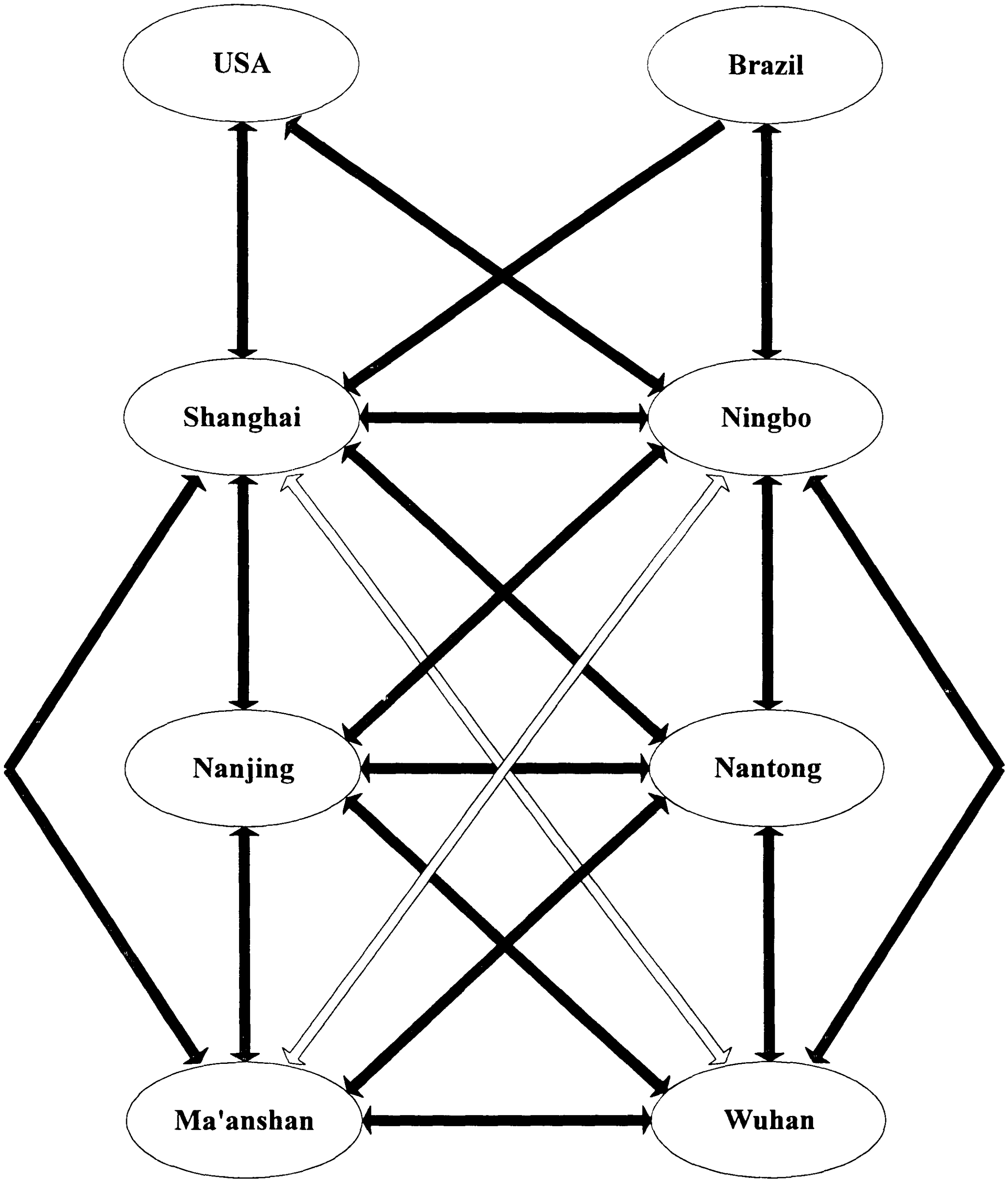


Figure 3 Distribution And Collection Network In The Yangtze Valley

modelers and greatly expand the extent and usefulness of mathematical programming applications in decision making analysis.

Substantial progress has made in the past decades with the development of algorithms and computer codes to solve large mathematical programming problems. The number of applications of these tools is less than expected because the solution procedures form only a small part of the overall modeling effort. A large part of the time required to develop a model involves data preparation and transformation and report preparation. Almost every model requires many hours of analysis and programming time to organize the data and write the programs that transform the data into the form required by the mathematical programming optimizers. Furthermore, it is difficult to detect and eliminate errors because the programs that performed the data operations are only accessible to the specialists who write them and not to the analysts who are in charge of the project.

GAMS is developed to improve on this situation by

- Providing a high level language for the compact representation of large and complex models;
- Allowing changes to be made in model specifications simply and safely;
- Allowing unambiguous statements of algebraic relationships; and
- Permitting model descriptions that are independent of solution algorithms.

The design of GAMS has incorporated ideas drawn from relational database theory and mathematical programming and has attempted to merge these ideas to suit the needs of strategic modelers. A relational database provides a structured framework for developing general data organization and transformation capacities. On the other hand, mathematical programming provides a way of describing a problem and a variety of methods for solving it. The following principles are used in designing the system:

- All existing algorithmic methods should be available without changing the user's model representation. Introduction of new methods or of new implementations of existing methods should be possible without requiring changes in existing models. Linear, non-linear, and mixed integer optimizations can currently be accommodated, as well as the special cases of simultaneous linear or non-linear systems; extensions are planned to linear and non-linear complementarity problems;
- The optimization problem should be expressible independently of the data it uses. This separation of logic and data allows problems to be increased in size without causing an increase in the complexity of the representation; and
- The use of the relational data model requires that the allocation of computer resources be automated. This means that large and complex models can be constructed without the user having to worry about details such as array sizes and scratch storage.

The GAMS model representation is in a form that can be easily read by people and by computers. This means that the GAMS program itself is the documentation of the model, and that the separation description required in the past is no longer needed. Moreover, the design of GAMS incorporates the following features that specifically address the user's documentation needs:

- A GAMS model representation is concise, and makes full use of the elegance of the mathematical representation;
- All data transformations are specified concisely and algebraically. This means that all data can be entered in their most elemental form and that all transformations made in constructing the model and in reporting are available for inspection; and
- Explanatory text can be made part of the definition of all symbols and is reproduced whenever associated values are displayed.

Of course some discipline is needed to take full advantage of these design features, but the aim is to make models more accessible, more understandable, more verifiable, and hence more credible.

The GAMS system is also designed in a way that models can be solved on different types of computers with no change. A model developed on a small personal computer can later be solved on a large mainframe. One person can develop a model that is later used by others who may be physically distant from the original developer. In contrast to other approaches, only one document needs to be moved -- the GAMS statement of the model, which rarely exceeds a few hundred lines of information and always fits on a diskette. It contains all the data and logical specifications needed to solve the model.

Portability concerns also have implications for user interface. The basic GAMS system is file-oriented and no special editor or graphical input and output routines exist. Rather than burden the user to learn yet another set of editing commands, GAMS offers an open architecture in which each user can use his word processor or editor of choice. This basic user interface facilitates the integration of GAMS with a variety of existing user environments.

3.3.4. The ORATB GAMS Model Construction

In this section, we discuss the basic components of a GAMS model before the construction of a GAMS model for the ORATB problem. The basic components are

Inputs

- SETS: declaration and assignment of members;
- Data (PARAMETERS, TABLES, SCALARS): declaration and assignment of values;

- VARIABLES: declaration and assignment of type;
- EQUATIONS: declaration and definition; and
- MODEL and SOLVE statements.

Outputs

- Echo Print;
- Reference Maps;
- Equation Listings;
- Status Reports; and
- Results.

There are optional input components, such as checks for bad data and requests for customized results. Other optional advanced features include saving and restoring old models, and creating multiple models in a single run. A general description on the structure of a GAMS model is listed below.

1. A GAMS model is a collection of statements in the GAMS language. The only rule governing the ordering of statements is that an entity of the model cannot be referenced before it is declared to exist;
2. GAMS statements may be laid out typographically in almost any style that is appealing to the user. Multiple lines per statement, embedded blank lines, and multiple statements per line are allowed;
3. GAMS statements are all terminated with a semicolon;
4. The GAMS compiler does not distinguish between upper and lower case letters. The style adopted and recommended is to always use upper case for any word or symbol that is part of the GAMS language or that is an entity declared to exist in a particular model, and reserve lower case only for words that appear in the GAMS input documentation;
5. There are at least two ways to insert documentation into a GAMS model. First, any line that starts with an asterisk in column one is disregarded as a comment

line by the GAMS compiler. Second, documentary text can be inserted within specific GAMS statements. All the lower case words are recommended to be reserved for documentation;

6. As we can see from the list of input components above, the creation of GAMS entities involves two steps: a declaration and an assignment or definition. "Declaration" means declaring the existence of something and giving it a name. "Assignment" or "definition" means giving something a specific value or form. In the case of equations, we must make the declaration and definition in separate GAMS statements. For all other GAMS entities, however, we have the option of making declarations and assignments in the same statement or separately; and
7. The names given to the entities of the model must start with a letter and can be followed by up to nine more letters or digits.

A complete text of the GAMS model developed for the proposed ORATB transport system is presented in Table 4.

3.3.5. GAMS Model Execution

Linear and mixed integer models created with GAMS are solved with a specially modified version of an optimizer which is called GAMS/OSL[18]. It is intended for medium-sized problems with no special structure and up to about 200 zero/one variables. GAMS/OSL handles integer variables by expressing them as combinations of 0-1 variables.

The mathematical form of a general linear mixed integer programming problem can be expressed as the following.

---

[18] OSL stands for optimization subroutine library.

Table 4 The ORATB Optimization Model Input Statements

```
*************************************************************************
SETS
   I  origination nodes       /O1/
   K  destination nodes       /D1,D2,D3,D4,D5/
   T  time periods            /T1,T2,T3,T4,T5,T6,T7,T8,T9,T10,T11,T12/
   L  barge sizes (kdwt)      /BG1,BG2,BG3/
   S  tug types (kkw)         /TG1,TG2,TG3/;

TABLE
   BKDM(I,K,T)     iron ore demand at k from i for period t (kton)
                   T1  T2  T3  T4  T5  T6  T7  T8  T9  T10  T11  T12
      O1.D1        14  14  15  16  15  16  15  14  13  14   15   15
      O1.D2        4   5   5   6   6   5   4   3   4   6    6    5
      O1.D3        5   5   4   4   3   6   5   4   4   5    6    6
      O1.D4        4   4   5   4   6   5   5   5   4   5    6    5
      O1.D5        9   9   10  11  10  10  9   9   9   10   11   11;

TABLE
   GAMA(S,L)       match-up relations between tug and barge
                   BG1   BG2   BG3
            TG1    1     0     0
            TG2    2     1     0
            TG3    4     2     1;

PARAMETERS
   BKBHO(I)        AAC of berthing facilities at origination port i ($m)
   /O1             30/

   BKBHD(K)        AAC of berthing facilities at destination port k ($m)
   /D1             20
    D2             5
    D3             5
```

Table 4 (continued)

```
 D4             5
 D5             10/

BKBEO(I)        AAC of berth equipment at origination port i ($m)
/O1             8/

BKBED(K)        AAC of berth equipment at destination port k ($m)
/D1             6
 D2             4
 D3             4
 D4             4
 D5             5/

TGF(S)          AAC fixed of s-type tug ($m)
/TG1            4
 TG2            6
 TG3            8/

TGV(S)          AAC variable of s-type tug ($m)
/TG1            1.0
 TG2            1.1
 TG3            1.2/

BGF(L)          AAC fixed of l-size barge ($m)
/BG1            1.0
 BG2            1.5
 BG3            2.0/

BGV(L)          AAC variable of l-size barge ($m)
/BG1            0.1
 BG2            0.2
 BG3            0.3/
```

Table 4 (continued)

```
   SZ(L)          capacity of I-size barge (kton)
   /BG1           5
    BG2           10
    BG3           15/;

VARIABLES
   Y(I,K,T)       0-1 decision variable to select routing for period t
   M(I,K,S,T)     0-1 decision variable to select tug type for period t
   N(I,K,L,T)     0-1 decision variable to select barge size for period
                  t
   MM(I,K,S,T)    integer decision variable to determine numbers of tugs
                  for period t
   NN(I,K,L,T)    integer decision variable to determine numbers of
                  barges for period t
   TAAC           total annual average costs
   TAAC11         AAC of berthing facilities at origination port i
   TAAC12         AAC of berthing facilities at destination port k
   TAAC21         AAC of berth equipment at origination port i
   TAAC22         AAC of berth equipment at destination port k
   TAAC31         AAC of tug related
   TAAC41         AAC of barge related;

INTEGER VARIABLES
   MM(I,K,S,T)    integer decision variable to determine numbers of tugs
                  for period t
   NN(I,K,L,T)    integer decision variable to determine numbers of
                  barges for period t;

BINARY VARIABLES
   Y(I,K,T)       0-1 decision variable to select routing for period t
   M(I,K,S,T)     0-1 decision variable to select tug type for period t
   N(I,K,L,T)     0-1 decision variable to select barge size for period
                  t;
```

Table 4 (continued)

```
EQUATIONS
   TAACDEF                    as equation 2
   TAAC11DEF                  as equation 2
   TAAC12DEF                  as equation 2
   TAAC21DEF                  as equation 2
   TAAC22DEF                  as equation 2
   TAAC31DEF                  as equation 2
   TAAC41DEF                  as equation 2
   BKDEMAND(I,K,T)            as equation 4
   BALANCE1(I,K,T)            as equation 5
   BALANCE2(I,K,T)            as equation 6
   BALANCE3(I,K,S,T)          as equation 14
   BALANCE4(I,K,L,T)          as equation 16
   BALANCE5(I,K,S,L,T)        as equation 17
   BALANCE6(I,K,S,T)          as equation 19
   BALANCE7(I,K,L,T)          as equation 20;

   TAACDEF..      TAAC =E= TAAC11+TAAC12+TAAC21+TAAC22+TAAC31+TAAC41;
   TAAC11DEF..    TAAC11 =E= SUM((I,K),BKBHO(I)*SUM(T,Y(I,K,T)));
   TAAC12DEF..    TAAC12 =E= SUM((I,K),BKBHD(K)*SUM(T,Y(I,K,T)));
   TAAC21DEF..    TAAC21 =E= SUM((I,K),BKBEO(I)*SUM(T,Y(I,K,T)));
   TAAC22DEF..    TAAC22 =E= SUM((I,K),BKBED(K)*SUM(T,Y(I,K,T)));
   TAAC31DEF..    TAAC31 =E=
                  SUM((I,K,S),(TGF(S)+TGV(S))*SUM(T,M(I,K,S,T)));
   TAAC41DEF..    TAAC41 =E=
                  SUM((I,K,L),(BGF(L)+BGV(L))*SUM(T,N(I,K,L,T)));
   BKDEMAND(I,K,T)..          SUM(L,N(I,K,L,T)*SZ(L)) =E= BKDM(I,K,T);
   BALANCE1(I,K,T)..          SUM(S,M(I,K,S,T)) =E= 1;
   BALANCE2(I,K,T)..          SUM(L,N(I,K,L,T)) =E= 1;
   BALANCE3(I,K,S,T)..        M(I,K,S,T) =L= Y(I,K,T);
   BALANCE4(I,K,L,T)..        N(I,K,L,T) =L= Y(I,K,T);
   BALANCE5(I,K,S,L,T)..      MM(I,K,S,T) =E= GAMA(S,L)*NN(I,K,L,T);
   BALANCE6(I,K,S,T)..        MM(I,K,S,T)-100*M(I,K,S,T) =L= 0;
```

Table 4 (continued)

```
   BALANCE7(I,K,L,T)..         NN(I,K,L,T)-100*N(I,K,L,T) =L= 0;

OPTION ITERLIM = 5000

OPTION OPTCR = 0.005

OPTION RESLIM = 10000

MODEL ORATB /ALL/;

SOLVE ORATB USING MIP MINIMIZING TAAC;

DISPLAY
   Y.L,M.L,N.L,MM.L,NN.L;
***********************************************************************
```

$$\begin{array}{ll} \text{Maximize} & c_1x_1 + c_2x_2 + \cdots + c_Nx_N \\ \text{Subject to} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,N}x_N \le b_1 \\ & \quad\vdots \qquad \vdots \qquad\quad \vdots \qquad \vdots \\ & a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,N}x_N \le b_M \\ & L_j \le x_j \le U_j \quad \text{for } j = 1, \ldots, N \\ & L_b = 0,\ U_b = 1, \text{ and } x_b = 0 \text{ or } x_b = 1 \text{ for the set of binary variables.} \end{array}$$

GAMS automatically converts integer variables into sums of 0-1 variables. For example, if x_j can take on values 0, 1, , U_j, we define *K* as a new 0-1 variables, where *K* is chosen so that:

$$2^{K-1} \le U_j < 2^K .$$

Then, the integer variable x_j is represented as:

$$x_j = y_{j,1} + 2y_{j,2} + 4y_{j,3} + 8y_{j,4} + \cdots + 2^{K-1}y_{j,K},$$

where $y_{j,k}$ is a 0-1 variable.

As shown in the above, the data for a given problem consists of the $a_{i,j}, b_j, c_j, L_j$, and U_j. These will all have been specified as parts of the GAMS model. GAMS/OSL begins by solving the problem as a linear program, and uses the Pivot and Complement (P&C) heuristic to find an initial integer feasible solution. It then uses a Branch-and-Bound (B&B) search to find improved solutions and to verify optimality. It also has several other features that are not found in other programs. These are: fixed order branching, multilevel expansion, the resource space tour, and "cheating." The whole optimization procedure is explained in detail below.

Initial Linear Program

The first step in solving a 0-1 mixed integer program is to solve the linear programming relaxation. This is obtained by relaxing the $x_j = 0$ or 1 conditions to $0 \leq x_j \leq 1$. GAMS/OSL solves this LP relaxation by calling XMP. If the LP relaxation is infeasible, so is the original mixed integer program, and it is finished. If the LP relaxation has a natural integer optimal solution, then this solution is also optimal for the original mixed integer program, and it is finished. Notice that this is the case that if all of the 0-1 variables are non-basic or basic and degenerate in the LP solution.

If neither of the above events occurs, we still get a bound on the optimal value of the mixed integer program. Let v_{LP} denote the optimal value of the LP relaxation, and let v_{MIP} denote the optimal value of the mixed integer program. Then we have $v_{LP} \geq v_{MIP}$, assuming maximization.

Pivot and Complement Heuristic

In general terms, the P&C heuristic method searches in the vicinity of the optimal solution of the LP relaxation for an integer feasible solution. It conducts this search by trying to force any basic integer variable, which is necessarily fractional unless it is degenerated, out of the basis. This is done by performing pivots. Non-basic integer variables can also be complemented, *i.e.*, flipped from zero to one or one to zero. Both one-at-a-time and two-at-a-time complements are attempted. If a feasible solution is found, then additional complements are performed in an attempt to improve it.

This P&C heuristic has worked very well in our experience to date, and it is quite fast (the pairwise complements are not done on more than 500 zero/one variables). Good solutions can be found for many large mixed integer problems by doing just the initial linear program and the P&C heuristic. On the other hand, doing branch-and-bound generally takes far longer than the initial LP + P&C heuristic combination.

If the P&C heuristic finds a feasible integer solution with value *INCUMBENT*, and if the relative gap between *INCUMBENT* and v_{LP} is less than or equal to a user specified *TOLERANCE*, *i.e.*,

$$(v_{LP} - INCUMBENT) \div (1 + |v_{LP}|) \le TOLERANCE,$$

then it is finished. *TOLERANCE* corresponds to the GAMS *OPTCR* and its default value is 0.1. GAMS/OSL also uses an absolute tolerance to independently control termination.

Branch-and Bound

If the LP relaxation is feasible but does not have a natural integer solution, and if the heuristic does not find an integer feasible solution that satisfies the *TOLERANCE*, then we have to resort to B&B.

The B&B can be described in general terms in the following way. Assume that we are maximizing, and that *INCUMBENT* is the objective function value of the best integer feasible solution that has been found so far. We begin by creating a search tree consisting of a single node that represents the original MIP problem. This node is the permanent root of the tree and is temporarily a leaf as well. The value of the LP relaxation, v_{LP}, is attached to this node as a label. One "iteration" procedure of the B&B method is as follows:

BB1 Select some leaf node in the current search tree. This node represents a problem which we denote CP for "candidate problem." Initially, CP = MIP, and if the search tree is empty, it is done;

BB2 Choose a 0-1 variable, say x_k, and split CP into two new problems NP_1 and NP_2 by appending the mutually exclusive constraint: $x_k = 0$ and $x_k = 1$. Add two new leaf nodes to the tree for NP_1 and NP_2 as shown in Figure 4. Now, the node for CP is no longer a leaf; and

BB3 Solve the LP relaxation LP_1 of NP_1. Eliminate the node for NP_1 if LP_1 is infeasible or if $v_{LP_1} \le INCUMBENT$, or if LP_1 has a natural integer solution.

In the latter case, update the value of the *INCUMBENT*. If the NP_1 node survives, then attach v_{LP_1} to it as a label. Do the same for NP_2. If NP_1 and NP_2 are both eliminated, then CP may be eliminated as well.

This cycle, select, split, and solve, as shown in Figure 4, is then repeated to complete the B&B procedure. The B&B search is finished when the search tree is empty or the *INCUMBENT* value satisfies the *TOLERANCE*, whichever comes first. The largest LP-value label over all leaves in the current tree is the best available upper bound on v_{MIP}, and we call this bound *GLOBAL*.

$$INCUMBENT \leq v_{LP} \leq GLOBAL.$$

Initially, $GLOBAL = v_{LP}$. The search is halted if either of the termination criteria is satisfied.

If the search tree is empty, then the *INCUMBENT* value is guaranteed to be optimal, assuming we really have a solution with value *INCUMBENT* and are not bluffing. Thus B&B is a race between creating nodes at step BB_2 and eliminating them at step BB_3.

Fixed Order Branching

In the general B&B procedure described above, we can use any 0-1 variable to split node CP into two new nodes. More precisely, we split any 0-1 variable that has not already been fixed on the path from the root of the tree to the node for CP. In GAMS/OSL, there is a restriction of this freedom of choice. Any node in the search tree has a "level," which is the number of arcs on the path from that node up to the root of the tree or the number of 0-1 variables already fixed in the problem associated with that node. So we designate a specific 0-1 variable for each level in the tree. For level *d*, for example, we denote the index of this variable as $SORT_d$. Then, whenever any node at level *d* is split into two new nodes, this will be done with the constraints:

$$x_{SORT_{(d+1)}} = 0 \text{ and } x_{SORT_{(d+1)}} = 1.$$

In GAMS, we also use "depth" to mean "level."

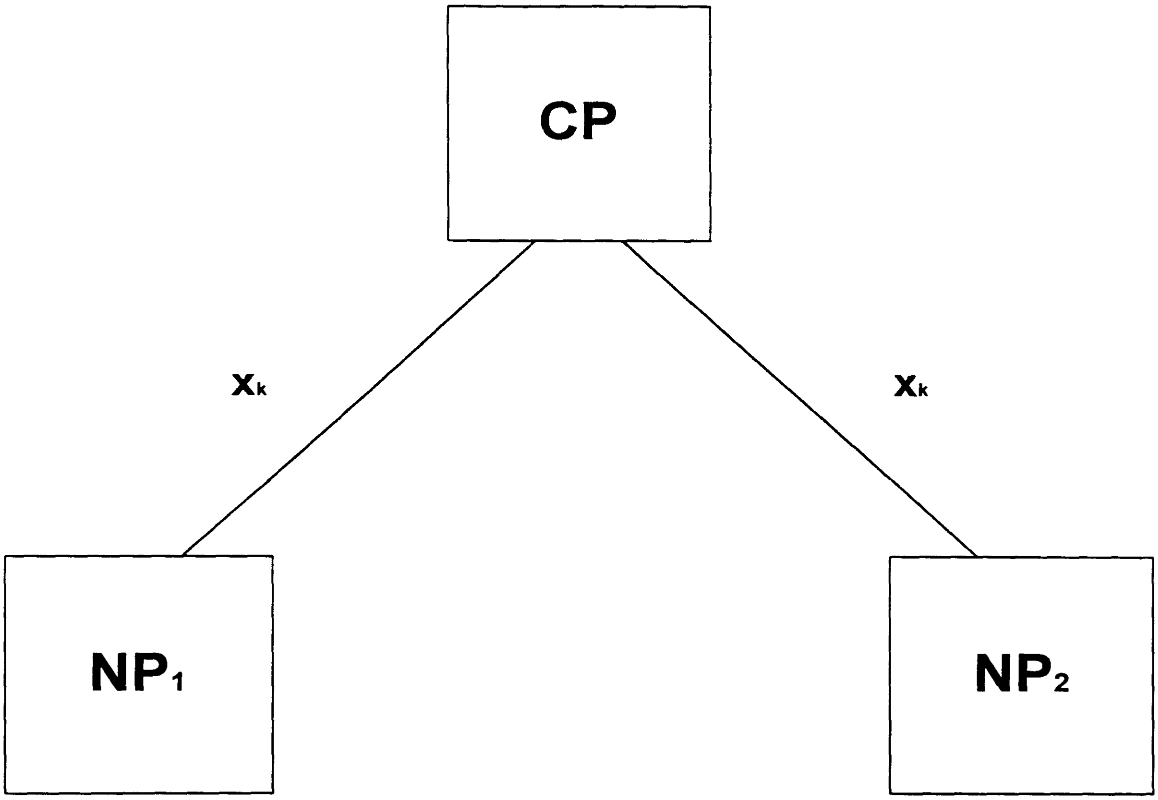


Figure 4 GAMS Schematic Of Branch-And-Bound Tree

Source: Brooke (1992).

The reason for imposing a fixed branching order will be explained below in the section on the resource space tour. The fixed order is determined in the following way. Consider the LP relaxation of the original MIP problem. The optimal LP solution partitions the 0-1 variables into three groups: basic (BS), non-basic at upper bound (UB), and non-basic at lower bound (LB) variables. The BS variables are sorted according to their values, the UB are sorted according to their relative profits, and the LB are sorted according to the negatives of their relative profits. The relative profit or reduced cost, for variable x_j is defined as:

$$d_j = c_j - u_1 a_{1,j} - \cdots - u_M a_{M,j},$$

where the $u_1 \ldots, u_M$ are the shadow prices. Then, if *ORDER* = 1 in the options file, the branching order is taken as first the BS variables as sorted, then the UB as sorted, and finally the LB as sorted. If *ORDER* = 2, then the UB variables are placed before the BS variables.

Multi-level Expansion

In the B&B procedure described above, we always select one node and split it into two nodes at the next level. A more general strategy is employed in GAMS/OSL. We may select several nodes which are at the same level in the current tree and split them several times to create new nodes which are several levels below the selected nodes. In such a case, we may specify two parameters, *SELECT* and *EXPAND* in the options file. GAMS/OSL will then take $SELECT{*}2^{EXPAND}$ nodes at level (d + *EXPAND*). For example, the default values are *SELECT* = 2 and *EXPAND* = 3. This causes two nodes at level d to be expanded into 16 nodes at level (d + 3). The standard strategy is obtained by setting *SELECT* = 1 and *EXPAND* = 1.

It is empirically advantageous to use values greater than one for *SELECT* and *EXPAND* in an algorithm based on fixed order branching and resource space tours. The question of what level the *SELECT* nodes should be taken from is addressed in the next section. As to which nodes are actually selected, we always take the ones with the *SELECT* best LP-value labels. If there are fewer *SELECT* nodes, we then take all of them.

Diving *vs.* Best Bound

The first step in each B&B iteration is deciding on the level in the tree from which the nodes to be expanded are to be selected. GAMS/OSL permits two alternatives strategies: diving and best bound. In the diving strategy, it chooses the deepest level in the current tree. This tends to produce natural integer solutions as soon as possible. This is particularly useful if the *INCUMBENT* value is far from the optimum.

In the best bound strategy, it chooses the level that has the leaf node with the largest LP-value label in the current tree. That is, the leaf node with LP-value = *GLOBAL*. This tends to reduce *GLOBAL* as fast as possible. This is particularly useful when we have an optimal or near-optimal solution and are trying to satisfy the tolerance. The choice between diving and best bound is made in the options file.

The Resource Space Tour

Now, we explain the motivation for the fixed order branching. Suppose *SELECT* = 2 and *EXPAND* = 3, we select the two best nodes from some level d and expand them into 16 nodes at level (d + 3). These 16 nodes have 16 associated linear programs: LP_1, …, LP_{16}. We apply bounding tests to these 16 nodes and eliminate as many as we can. The bounding tests are based on LP generated information, but we are not solving the 16 LPs completely separate from one another. We are making a "resource space tour" described below.

Because of the fixed order branching, the 16 linear programs are identical except for their right-hand-sides. They all have the same set of fixed 0-1 variables: $SORT_1,\ldots,SORT_{(d+2)}$. The fixed variables are moved from the left-hand-side of the constraints to the right-hand-side, giving 16 different reduced right-hand-side vectors. We imagine the 16 nodes as represented by 16 points in the M-dimensional space of right-hand-side vectors. Starting at the point for LP_1, we make a tour that visits some or all of these 16 points. We begin by solving LP_1. Then we use parametric programming to obtain the solution of LP_2. On the other hand, the 16 linear programs have the same dual feasible region, hence any dual feasible solution can be used to compute an upper bound v_{LP_q} for every q, and hence to perform a bounding test on all 16 nodes. Since we are doing parametric programming on the right-hand-side, every pivot produces a new dual feasible solution that can be used to perform a bounding test on all of the still surviving nodes.

If we are traveling by parametric programming from the point for LP_q to the point for LP_r and we succeed in eliminating some other points, say LP_S $(q < r < s)$, we call it an "indirect hit." If we eliminate the current destination, LP_r, we call it a "direct hit." If LP_r is eliminated, we immediately change direction and head toward the next still surviving point. On the other hand, if LP_r survives, we then check to see if its LP solution is naturally integer and strike off for the next still surviving node, which will be $(LP_r + 1)$ if it still exists. At the end of the tour we have eliminated some of the nodes or points and obtained the LP solutions of the others, and some of which may be naturally integer and hence new incumbents.

It has been discovered empirically that with fixed order branching and the resource space tour, the standard *SELECT* = 1, *EXPAND* = 1 strategy gives poor results. It is much more effective to spread the LP computational overhead among more nodes and take advantage of indirect hits. In a series of experiments, the best strategy is *SELECT* = 5 and *EXPAND* = 4, giving 80 nodes for each tour.

If *EXPAND* = 3, then we only perform tours at levels 3, 6, 9, ..., in the search tree. GAMS/OSL saves one "hot" LP basis for each of these levels. A tour at level 9 will start with the hot basis for level 9 if one is available, otherwise with the hot basis for level 6, or level 3, or level 0 (root) in that order of performance. The *"MAXSAVE"* option in the options file allows us to decide how many hot bases should be saved. If *"MAXSAVE 5"* is specified, then the root basis and four other hot bases are held at any one time. Saving more hot bases should speed up the tours, but each one saved takes up (COLUMNS/2 + ROWS) of work space.

Locking 0-1 Variables

GAMS/OSL locks 0-1 variables automatically based on the solution to the initial LP relaxation. After solving the initial LP we compute the relative profits d_j for each variable x_j, j in binary. Assuming a maximization problem, if x_j is at its UB and

$$d_j \geq v_{LP} - INCUMBENT,$$

then x_j can be locked permanently at one. On the other hand, if x_j is at its LB and

$$-d_j \geq v_{LP} - INCUMBENT,$$

then x_j can be locked permanently at zero. The d_j and v_{LP} values are saved and whenever a new *INCUMBENT* value is found during the heuristic or the B&B search we attempt to lock additional variables.

The Algorithm

The following is a summary of the above discussion in a concise statement of the main steps in the algorithm. Capitalized names refer to options and parameters from the options file or to program variables. It is assumed maximization, and an *INCUMBENT* value.

Step 1 Solve the initial linear program. If it is infeasible, stop. If it has a natural integer solution, stop. Otherwise initialize the search tree by creating a node with LP-value label = v_{LP} (If the problem is a pure LP, stop);

Step 2 Compute the relative profits, d_j, of all the 0-1 variables and lock as many of them as possible using v_{LP} and *INCUMBENT*;

Step 3 If *HEURISTIC* = 'YES', then use the P&C heuristic to find an integer feasible solution. If P&C finds an improved *INCUMBENT*, try to lock additional variables;

Step 4 If *INCUMBENT* satisfies either of the termination tolerance (*ORTCA* or *ORTCR*), stop;

Step 5 If *BRANCH* = 'NO', stop. Otherwise use *ORDER* to determine the branching order. Set *INPROV* = .FALSE.;

Step 6 If the search tree is empty, stop. The *INCUMBENT* value is optimal unless it is a bluff;

Step 7 Set *GLOBAL* = the largest LP-value label over all leaves in the current tree. If (*GLOBAL* - *INCUMBENT*) $\div$ (1 + |*GLOBAL*|) $\leq$ *TOLERANCE*, stop;

Step 8 Stop, if *INPROV* = .TRUE. and *QUIT* = 'YES';

Step 9 Stop, if the *LIMIT* on LP iterations has been exceeded;

Step 10 If *DIVE* = 'YES', choose *d* = the deepest level in the current tree. Otherwise, choose *d* = the level that has the leaf with LP-value label = *GLOBAL*;

Step 11 Take the *SELECT* nodes at level *d* that have the best LP-value labels, and expand them into $SELECT{*}2^{EXPAND}$ nodes at level (*d* + *EXPAND*). If there are fewer than *SELECT* nodes at level *d*, then just take all of them.

Step 12 Do a resource space tour at level (*d* + *EXPAND*). If a natural integer solution is found that gives an improved *INCUMBENT* value, then set *INPROV* = .TRUE. Keep track of all LP iterations for comparison with *LIMIT*. Label each surviving node with its LP-value;

Step 13 If *INPROV* = .TRUE., purge the tree. That is, eliminate any leaf that has LP-value label $\leq$ *INCUMBENT*;

Step 14 If *INPROV* = .TRUE., use the saved d_j and v_{LP} values to lock additional variables; and

Step 15 Go to Step 6.

Cheating

In this section we examine the bounding test that is used in the B&B procedure. Consider any node in the search tree. Suppose it represents sub-problem CP_q with LP relaxation LP_q. We may discard this node if

$$UB_q \leq INCUMBENT,$$

where UB_q is an upper bound on v_{CP_q}, the optimal value of CP_q. UB_q may be v_{LP_q}, or it may be an upper bound on v_{LP_q} computed using a dual feasible solution of LP_q during a resource space tour. It may happen that

$$v_{CP_q} \leq INCUMBENT \leq \mathrm{UB}_q,$$

i.e., CP_q does not contain any solution better than the current incumbent, but we cannot discard it because UB_q is too loose an estimate of v_{CP_q} which is unknown. It may also happen that

$$INCUMBENT \leq \mathrm{UB}_q \leq v_{MIP},$$

i.e., CP_q does not contain an optimal solution of MIP, but we cannot discard it since it may contain a solution that is better than the current poor incumbent.

Because of the error involved in making the LP relaxation, UB_q is too big. And *INCUMBENT*, because it is not in general optimal, is too small. We would like to decrease UB_q and increase *INCUMBENT* to correct for these errors. Of course the amount of correction needed is unknown, but the form of the correction would be:

$$\mathrm{UB}_q \bullet (1 - A) \leq INCUMBENT \bullet (1 + B),$$

where $0 \leq A < 1$ and $B > 0$. This is equivalent to:

$$\mathrm{UB}_q \leq INCUMBENT \bullet (1 + \varepsilon).$$

If $\varepsilon > 0.0$, then we may discard nodes that would not normally ($\varepsilon = 0.0$) be discarded. Clearly if ε is large, we will eliminate too many nodes and almost surely miss the optimal solution. We can, however, determine the maximum possible error in our final answer.

During the B&B search we simply keep track of the largest UB_q value over all discarded nodes. That is, the best bound that gets thrown away. We call this value *DISCARD*. During the search, we modify the definition of *GLOBAL* to be either *DISCARD* or the largest LP-value label over all the leaves in the tree, whichever is larger, *i.e.*,

$$GLOBAL = \mathrm{MAX}(DISCARD, INCUMBENT).$$

Thus the incumbent is optimal if *DISCARD* ≤ *INCUMBENT*, and otherwise the relative error is no greater than

$$(GLOBAL - INCUMBENT) \div GLOBAL.$$

So we can determine the consequences of cheating, *i.e.*, using $\varepsilon > 0.0$, simply by keeping track of the single number *DISCARD*.

In deciding on the numerical value to use for ε, our first observation is that ε should be smaller when we are deep in the tree than when we are near the top of the tree. We would also like ε to depend on the apparent relative gap between the MIP and LP optimal values for the problem at hand. GAMS/OSL uses the following method for computing ε values when *CHEAT* = 'YES' is specified in the options file.

Let *NBR* be the number of 0-1 variables that have to be branched on, *i.e.*, the original number minus any that are locked. Let *SHRINK* and *EXPAND* be as given in the options file, or by default. Suppose we are performing a resource space tour at level (*d* + *EXPAND*). Then we use the bounding test:

$UB_q \leq INCUMBENT \bullet (1 + \varepsilon)$, where

$\varepsilon = GAP_d \bullet [1 - (SHRINK + EXPAND - 1) \div (NBR - d)$

if $(SHRINK + EXPAND - 1) < (NBR - d)$; or

$\varepsilon = 0.0$ otherwise; and

$GAP_d = (BESTUB_d - INCUMBENT) \div INCUMBENT$; and

$BESTUB_d =$ the largest LP-value label obtained so far at depth d.

Thus we cheat more if the estimate GAP_d of the relaxation error at depth d is large. The *SHRINK* parameter is required to be at least 1.0, and the larger it is, the less we are cheating.

GAMS/OSL Options Specified In The Program

The most important of the options used to control the behavior of GAMS/OSL must be included in GAMS program as *OPTION* statement, rather than in the options file. These are listed below in the sequence *option*, *value*, and *default*.

OPTCR	A real number in the range 0 to 1	Default = 0.1

Controls the termination of GAMS/OSL. If, assuming maximization, an integer feasible solution is found in which

$$v_{MIP} \geq BESTUB(1 - OPTCR),$$

then GAMS/OSL is stopped and this solution reported. *BESTUB* is the best objective value possible. It decreases during B&B. Notice that setting $OPTCR = 0.0$ can result in extremely long-run times for even small problems that have a great many optimal and near optimal solutions, unless *OPTCR* has been set different from zero. It is much safer to use a very small value like 0.0001 than an exact 0.0.

OPTCA A positive real number Default = 0.0

Also controls the termination of GAMS/OSL. If, assuming maximization, an integer feasible solution is found in which

$$v_{MIP} \geq BESTUB(1 - OPTCA),$$

then GAMS/OSL is stopped and this solution reported. This control operates independently of that provided by *OPTCR*.

ITERLIM A positive integer Default = 1,000

Limits the number of iterations that will be performed by GAMS/OSL. If the problem is an MIP this limit will be approximate and not exact.

RESLIM A positive real number Default = varies, often 1,000

Controls the amount of computer time used by GAMS/OSL. The units are machine specific.

SYSOUT ON or OFF Default = OFF

If set to ON, we will receive a listing of all GAMS/OSL output on the output file. This information is essential for investigating or changing the behavior of GAMS/OSL. Use the options file *PRINT* keyword to control the amount and type of output.

WORK A positive integer Default = estimated by GAMS/OSL

Controls the size of the size of the table of waiting B&B nodes. If we have trouble with the node table size, messages will tell us how big it was.

Format Of GAMS/OSL Options File

The options file contains general information about the problem to be solved, and the user's choices for the many algorithmic parameters and options. The options file is a file with a fixed name, and it must be in the current directory if you have a directory-based file system. The name is constructed from two parts. The first is most likely OSL: the name we would use in our program to specify that GAMS/OSL be used, *e.g.*, *OPTION* MIP = OSL. The second is a machine-specific suffix or type that would apply to all solvers on a particular machine: on personal computers "OPT," giving a complete filename "OSL.OPT."

The first line of the options file must contain the keyword *'BEGIN'*, and the last line must contain the keyword *'END'*. Every line between the *'BEGIN'* and *'END'* must either be a comment line (first non-blank character '*' or '!') or contain one of the keywords listed in the following section as its first non-blank characters.

GAMS/OSL Options

The following is the list in sequence of valid keywords with their associated modifiers, values, and default values.

1. The GAMS options *RESLIM* and *ITERLIM* may be used to control resource usage, *OPTCA* and *OPTCR* to control termination. The node table size can be set with the GAMS *WORK* option. The OSL options *TOLERANCE* and *MAX NODES* are not recognized by GAMS/OSL;
2. GAMS/OSL is not able to use basis or tree information from a previous solve to provide an advanced start to the problem under consideration;

3. GAMS/OSL accepts only a subset of keywords available to programmers using OSL. None of the problem size parameters are applicable, and several of the methods and settings are not available. They are:
    a) The dual method;
    b) The OSL *SAVE/RESTORE* capability; and
    c) The ability to provide a user-specified branching order; and
4. GAMS/OSL provides an additional pre-specified branching order (*ORDER* = 3) that has been found to work well with general integer variables. The meaning of (*ORDER* = 3) is thus different from the meaning in the FORTRAN program OSL.

GAMS/OSL Option List

BRANCH 'YES', 'NO' Default = 'YES'

Specifies whether or not a B&B search is to be performed.

CHEAT 'YES', 'NO' Default = 'NO'

Specifies whether or not the cheating strategy is to be used during the B&B search.

DIVE 'YES', 'NO' Default = 'YES'

Specifies whether or not to use diving strategy in the B&B search. Diving is recommended as a first attack on the problem, especially if the heuristic fails or finds a relatively poor solution. 'NO' means use the best-bound strategy. In the diving strategy, we always select nodes from the deepest level in the current search tree. In the best-bound strategy, we choose the level in the tree that has the best associated LP bound. If GAMS/OSL is diving and it finds an improved integer feasible solution, *i.e.*, new incumbent, it will automatically switch over to the best-bound strategy.

EXPAND 1, ..., 6 Default = 3

During the B&B search, each node that is selected is expanded *EXPAND* levels down in the search tree, resulting in 2^{EXPAND} new nodes.

FACTOR 1, ..., 100 Default = 50

The number of iterations to be allowed between refactorizations of the basis matrix. A value of 50 is suitable for most problems. It should be reduced in the presence of serious numerical instability. The accuracy of the current solution is checked every *FACTOR*/2 iterations and an early refactorization done if necessary.

GAP Non-negative real Default = +∞

An estimate of the relative gap between the value of the integer program, v_{MIP}, and the value of the linear program, v_{LP}. In other words, *GAP* is an *a priori* estimate of $(v_{LP} - v_{MIP}) / (1+|v_{LP}|)$. After computing v_{LP}, the code computes an estimate of v_{MIP} using *GAP*. It uses this as the incumbent value at the beginning of the B&B search, unless a better incumbent value has been found by the heuristic. If the true gap is larger than *GAP*, and the corresponding estimate of v_{MIP} is not improved by the heuristic, then the B&B search finds no solutions. Guessing a *GAP* can save a lot of work in the early part of the search, until a true incumbent solution is found. It is a bluff, however, and fails if it is too optimistic. A value specified with the *INCUMBENT* keyword always overrides the value of *GAP*. The default of +∞ has the effect of not using the *INCUMBENT* feature at all. Suggested value is 0.25.

HEURISTIC ‘YES’, ‘NO’ Default = ‘YES’

Specifies whether or not the search for an integer feasible solution starts with the P&C heuristic.

INCUMBENT Real Default = ∓∞ for maximization and minimization

The objective function value of the best integer feasible solution that is known so far. This is used to eliminate nodes in the search tree whose LP values are not better than *INCUMBENT*. We can also bluff and hope that the search turns up a solution with a better value than *INCUMBENT*. If we bluff, however, and the value we give for *INCUMBENT* is better than the true optimal value, then the B&B search finds no solutions. As stated before, a value specified with the *INCUMBENT* keyword always overrides the *GAP* value.

MAX Positive integer Default = 5

An upper limit on the number of hot LP bases saved during the B&B search. Saving more bases speeds up the search, but each one saved takes up (COLUMNS/2 + ROWS) words of work space.

MULTIPLE Positive integer Default = 5

Multiple pricing parameter for the simplex method. This is the number of attractive non-basic variables saved during a major iteration for basis entry during the subsequent series of minor iterations.

ORDER 1, 2, 3 Default = 1 or 3 if integer variables

Specifies the branching order for B&B. If BS, UB, and LB denote the sets of 0-1 variables that are basic, at upper bound, and at lower bound respectively in the optimal solution of the initial linear program, and if the variables in these sets are ordered, then: '1' means BS, UB, LB; '2' means UB, BS, LB; and '3' means an order that may provide improved performance with integer variables.

PARTIAL Positive integer Default = 500

Partial pricing parameter for the simplex method. The number of columns priced out during a major iteration. A major iteration consists of pricing out *PARTIAL* columns and saving the *MULTIPLE* best ones.

QUIT 'YES', 'NO' Default = 'NO'

Specifies whether or not the B&B search should be halted when an improved incumbent solution is found.

SELECT Positive integer Default = 2

The number of nodes to be selected for each expansion during the B&B search. One expression consists of choosing a level in the search tree, selecting the *SELECT* best nodes at that level, and expanding them done *EXPAND* levels, resulting in $SELECT*2^{EXPAND}$ nodes.

SHRINK Positive integer ≥ 1.0 Default = 5

The control parameter for the cheating strategy. The smaller the value of *SHRINK*, the more we are cheating. The more we cheat, the faster the B&B search goes, but the less likely we are to find the optimal solution.

Controlling B&B

The time and computational effort required to solve a mixed integer program by B&B is notoriously unpredictable. GAMS/OSL has many options and parameters that we can set in our search for an optimal, or near-optimal, integer solution. Here, in the currently fashionable form of *if-then* rules, are some suggestions about using these parameters.

1. If the model we want to solve is new, unfamiliar, or large, then plan on making more than one run;
2. If we are going to make more than one run and this is our first run, then use *HEURISTIC* = 'YES' and *BRANCH* = 'NO';
3. If the heuristic has found a feasible integer solution or an earlier B&B search has found a feasible integer solution or we know a feasible integer solution, then set *INCUMBENT* to the value of that solution and compute:
   CRITERION = (v_{LP} - INCUMBENT) ÷ (1 + | v_{LP} |);
4. If we have not found any feasible integer solution, then set *CRITERION* = 100;
5. If *CRITERION* < 0.10, then make a B&B search with *BRANCH* = 'YES', *DIVE* = 'NO' and *QUIT* = 'NO';
6. If 0.10 < *CRITERION* < 0.25, then make a B&B run with *BRANCH* = 'YES', *DIVE* = 'NO' and *QUIT* = 'YES';

7. If *CRITERION* > 0.25 and we have not tried B&B yet, then make a B&B run with *BRANCH* = 'YES', *DIVE* = 'YES', *QUIT* = 'YES', *SELECT* = 1, *EXPAND* = 4;
8. If *CRITERION* > 0.25 and we have already tried B&B, then bluff, *i.e.*, use the *INCUMBENT* or *GAP* parameter to pretend that we have a feasible integer solution that is within 25% of v_{LP};
9. If *CRITERION* > 0.25 and we have already tried bluffing, then cheat, *i.e.*, use the *CHEAT* and *SHRINK* parameters to find a better feasible integer solution; and
10. If we found an improved feasible integer solution during a B&B run with cheating, then make another B&B run with cheating, but cheat less, *i.e.*, increase *SHRINK*.

If we have made several B&B runs and we have a feasible integer solution with *CRITERION* > 0.50 and the current upper bound on v_{MIP} is almost the same as v_{LP}, then we should give up on trying to satisfy our stopping *TOLERANCE*. In this event, B&B is just not going to bring the upper bound down in any reasonable amount time. Any further B&B runs should be pure diving raids *i.e.*, *DIVE* = 'YES' and *QUIT* = 'YES', perhaps with bluffing or cheating, looking for improved integer solutions, *i.e.*, increase *INCUMBENT*.

The output results and sensitivity analysis of the ORATB optimization model are discussed in Section 4.4.1.

3.4. Simulation Model

The objective of this simulation model is to test the preliminary recommendations obtained from the optimization model with regard to tug/barge acquisition against a more realistic environment, which include the uncertainties present in the daily operation of the tug/barge system, the sequential relationships exist in scheduling, and the congestion generated by executing the ORATB transport system.

Although there are many mathematical programming models which can provide optimal solutions, because of complexity, uncertainties, stochastic relations, and so on, few real world problems can be fully represented and solved by optimization models. In other words, optimization models usually require so many simplified assumptions that the solutions are likely to be inferior or inadequate for implementation. Often, in such instances, the only alternative form of modeling and analysis available to decision makers is simulation model. For instance, in modeling a queuing system, we have to assume that random variables, such as interarrival times and service times, have specific probability distributions. In many situations, though, these assumptions are not appropriate, *i.e.*, the distributions of these random variables cannot be specified precisely. In these cases, empirical distributions of interarrival times and service times must be used, which imply that analytical models from queuing theory are no longer applicable. In simulation models, however, any distribution of interarrival time and service time may be used, thereby giving much more flexibility to the decision making process.

In the transportation field, many problems can be solved by network programming, but the mixture of integer and non-integer variables and the mixture of continuous and discrete variables prohibit an effective use of network programming in a large-scale transportation planning study. At the same time, the dynamic characteristics in transport demand also hamper the effective use of network programming. Such dynamics include both stochastic and conditional changes in transport demand. The stochastic character is mainly associated with uncertainties in operations and related economic development. The conditionality behavior, on the other hand, is preliminary due to the nature that transport demand is highly affected by its supplies. A better service with respect to transit times and transportation costs often attracts demand from other modes of transportation. To my knowledge, it is very difficult to incorporate these dynamic features into ordinary network programming.

A simulation model is an ideal tool for network system design. It is flexible and easy to modify, which progresses in the level of detail model modeled, from preliminary design to detailed design. Simulation is particular useful for making comparisons, robustness studies, and the like. Trade-off analysis between various scenarios and their implications can be easily studied through simulation models. The essence of simulation is to provide a realistic and detailed representation of the problem under study. It allows decision makers to test various alternatives that they might want to consider. On the other hand, simulation evaluates the impact of the decision makers' planning strategies.

A simulation model can be defined as a technique that imitates the operations of a real world system as it evolves over a period of time. There are basically two types of simulation models: a static model and a dynamic model. A static model represents a system at a particular point in time, while a dynamic model represents a system as it evolves over time. At the same time, simulation models can be deterministic or stochastic. A deterministic simulation contains no random variables, whereas a stochastic model includes one or more random variables. Finally, simulations may be represented by either discrete or continuous models. A discrete simulation is one in which the state variables, which describe the status of the simulated system, change only at discrete points of time.

As with most other modeling techniques, simulation has its advantages and disadvantages. One of the major advantages is that simulation model is rather straightforward. Generally speaking, simulation models are relatively easier than optimization models to apply to the real world problems. Whereas optimization models often require many simplified assumptions, simulations have few such restrictions, thereby permitting much greater flexibility in representing the real system. Moreover, once a simulation model is built, it can be used repeatedly to analyze different parameters, designs, and managerial policies.

On the disadvantage side, it must be emphasized that simulation is not an optimization technique. It does not generate an optimum solution, but simply permits the evaluation of alternative solutions supplied externally by decision makers. In other words, simulation models are often used to analyze "what if" types of questions by calculating the corresponding measures of performance. Simulation can also be costly in terms of both model construction and model execution. However, with the development of special purpose simulation languages, advances in simulation methodology, and decreasing computational costs, the problem of cost is becoming less important. Finally, in contrast to exact mathematical solutions available in optimization models, simulation only generates samples of the measures of performance.

3.4.1. Simulation Process

The successful development of a simulation model consists of beginning with a simple model which is embellished in an evolutionary fashion to meet a problem solving requirement. Within this process, the following stages of development can be identified as described by Pritsker (1986).

1. Problem formulation: the definition of the problem to be studied including a statement of the problem solving objective. The construction of a clear definition of the problem and an explicit statement of the objectives of the analysis are crucial. In addition, because of the evolutionary nature of simulation, problem definition is a continuing process which typically occurs throughout the duration of the study. As additional insights into the problem are gained and additional questions become of interest, the problem definition is revised accordingly;
2. Model building: the abstraction of the system into mathematical logical relationships in accordance with the problem formulation. The model of a system may consist of both static and dynamic description. The static description defines the elements of the system and the characteristics of the elements. The dynamic

description defines the way in which the elements of the system interact to cause changes to the state of the system over time;

3. Data acquisition: the identification, specification, and allocation of data. Typically, input data are initially hypothesized or based on a preliminary analysis. In some cases, the exact values for one or more of the input parameters may have little effect on the simulation results. The sensitivity of the simulation results to changes in the input data to the model can be evaluated by making a series of simulation runs while varying the input parameter values. In this way, the simulation model can be used to determine how best to allocate money and time in refining the input data to the model;
4. Model translation: the preparation of the model for computer processing. Although a simulation model can be programmed using a general purpose language, there are distinct advantages to using a simulation language. In addition to the savings in programming time, a simulation language also assists in model formulation by providing a set of concepts for articulating the system description;
5. Verification: the process of establishing that the computer program executes as intended. The verification task consists of determining that the translated model executes on the computer as the modeler intended. This is typically done by manual checking of calculations;
6. Validation: the process of establishing that a desired accuracy or correspondence exists between the simulation model and the real system. The validation task consists of determining that the simulation model is a reasonable representation of the system. Validation is normally performed in levels on data inputs, model elements, subsystems, and interface points. Validation of simulation models, although difficult, is a significantly easier task than validating other types of models, for example, validating a linear programming formulation;
7. Strategic and tactical designing: the process of establishing the experimental conditions for using the model. The strategic designing task consists of developing an efficient experimental design either to explain the relationship

between the simulation response and the controllable variables, or to find the combination of values for the controllable variables which either minimize or maximize the simulation response. In contrast, the tactical designing is concerned with how each simulation within the experimental design is to be made to glean the most information from the data. Two particularly important issues in tactical designing are the starting conditions for simulation runs and methods for reducing the variance of the mean response;

8. Experimentation: the execution of the simulation model to obtain output values. The simulation experimentation involves the exercising of the model and the interpretation of the outputs;
9. Analysis of results: the process of analyzing the simulation outputs to draw inferences and make recommendations for problem solution. When simulation results are used to draw inferences or to test hypotheses, statistical methods should be employed; and
10. Implementation and documentation: the process of implementing decisions resulting from the simulation and documenting the model and its use. No simulation project should be considered complete until its results are used in the decision making process. The implementation of recommendations to improve system performance is an integral part of the simulation process. At the same time, documentation is required to ensure that the results can be understood, replicated, criticized, and extended by others.

Figure 5 presents the proposed simulation model for the ORATB transport system based on the procedures listed above. The iterative nature of the process is indicated by the feedback branches in the figure.

3.4.2. Simulation Modeling Perspectives

In developing a simulation model, we need to select a conceptual framework for describing the system to be modeled. The framework or perspective contains a world

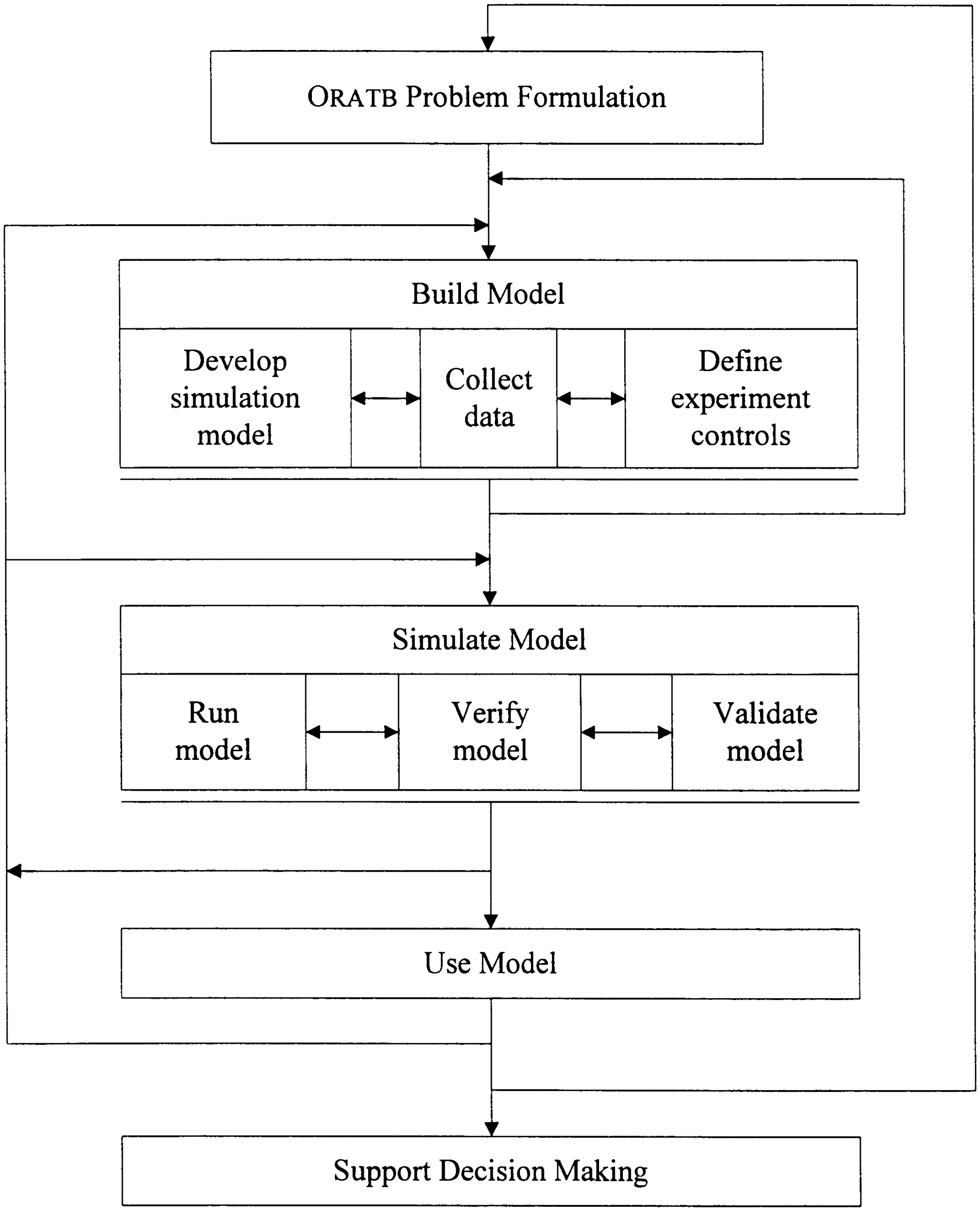


Figure 5 **The Structure Of The ORATB Simulation Model**

view within which the functional relationships of the system are perceived and described. If we employ a simulation language , then the world view will be implicit within the language. However, if we select a general purpose language, then we are responsible for the perspective of organizing the system description. In either case, the world view employed provides a conceptual mechanism for articulating the system description.

Modeling World Views

Simulation models are classified as either discrete change or continuous change. In most simulations, time is the major independent variable, while other variables are functions of time and are the dependent variables. The words 'discrete' and 'continuous' refer to the behavior of the dependent variable rather than the real system. In fact, a real system can be modeled with either a discrete or a continuous model.

Discrete simulation occurs when the dependent variables change discretely at specified points in simulated time. On the other hand, in continuous simulation the dependent variables of the model change continuously. In combined simulation the dependent variables of a model may change discretely, continuously, or continuously with discrete jumps superimposed. In any case, however, the time variable may be either continuous or discrete in the model, depending on whether the discrete changes in the dependent variable can occur at any point in time or only at specified points.

Discrete Simulation Modeling

The objects in a discrete system, such as people, equipment, and raw materials, are called entities. There are many types of entities and each has various attributes. The aim of a discrete simulation model is to reproduce the activities that the entities engage in and thereby discover the behavior and performance potential of the system. This is done by defining the states of the system and constructing activities that move it from state to state. The state of a system is defined in terms of the numerical values assigned to the attributes of the entities. A system is said to be in a particular state when all of its entities are in states consistent with the range of attribute values that define that state. Thus, simulation is the dynamic portrayal of the states of a system over time.

In discrete simulation, the state of the system can change only at event times. Since the state of the system remains constant between event times, a complete dynamic portrayal of the state of the system can be obtained by advancing simulated time from one event to the next. A discrete simulation model can be formulated by:

- Defining the changes in state that occur at each event time, *i.e.*, event oriented;
- Describing the activities in which the entities in the system engage, *i.e.*, activity scanning oriented; or
- Describing the process through which the entities in the system flow, *i.e.*, process oriented;

where

- An event takes place at a point in time at which activities start or end; and
- A process is a time-ordered sequence of events that encompass several activities.

These concepts are the three alternative world views for discrete simulation modeling, and are commonly referred to as the event, activity scanning, and process orientations.

In the event orientation, a system is modeled by defining the changes that occur at event times. The task of the modeler is to determine the events that change the system's state and then to develop the logic associated with each event. If we employ a general purpose language to code a discrete event model, then a considerable amount of programming effort is directed at developing the event calendar and a timing mechanism for processing the events in their proper chronological order. Since this function is common to all discrete event models, a number of simulation languages have been developed which provide special features for event scheduling, as well as other functions which are commonly encountered in discrete event models.

In the activity scanning orientation, we describe the activities in which the entities in the system engage and prescribe the conditions which cause an activity to start or end. The events which start or end the activity are not scheduled by the modeler, but are initiated by the conditions specified for the activity. As simulated time is advanced, the conditions for either starting or ending an activity are scanned. If the prescribed conditions are satisfied, then the appropriate action for the activity is taken. To ensure that each activity is accounted for, we must scan the entire set of activities at each time. This approach is well suited for situations where an activity duration is indefinite and is determined by the state of the system satisfying a prescribed condition. However, because of the need to scan each activity at each time advance, this approach is relatively inefficient when compared to the discrete event orientation. As a result, the activity scanning orientation has not been widely adopted as a modeling framework for discrete simulation.

Many simulation models include sequences of elements which occur in defined patterns. The logic associated with such a sequence of events can be generalized and defined by a single statement. A simulation language can then translate such statements into the appropriate sequence of events. A process oriented language employs such statements to model the flow of entities through a system. Theses statements define a sequence of events which are automatically executed by the simulation language as the entities move through the process. The simplicity of the process orientation is derived from the fact that event logic associated with the statements is contained within the symbols of the simulation language. However, since we are normally restricted to a set of standard symbols provided by the simulation language, our modeling flexibility is not as great as with the event orientation.

Continuous Simulation Modeling

In a continuous simulation model, the state of the system is represented by dependent variables which change continuously over time. To distinguish continuous change variables from discrete change variables, the former are referred to as state variables. A continuous simulation model is thus constructed by defining equations for a set of state variables whose dynamic behavior simulates the real system. Continuous models are often written in terms of the derivatives of the state variables. The reason for this is that it is often easier to construct a relationship for the rate of change of the state variable than to devise a relationship for the state variable directly.

During the 1950's and 1960', analog computers were the primary means for performing continuous simulations. An analog computer represents the state variables in the model by electrical charges. The dynamic structure of the system is modeled using circuit components such as resistors, capacitors, and amplifiers. The principal shortcoming of an analog computer is that the quality of these components limits the accuracy of the results. In addition, the analog computer lacks the logical control functions and data storage capacity of the digital computer.

A number of continuous simulation languages have been developed for use on digital computers. It is necessary to recognize that a digital computer is technically discrete in its operation. As a practical matter, however, any variable whose possible values are limited only by the word size of the computer is considered continuous. A digital computer performs the common mathematical operations with great speed and accuracy. By using these operations, a digital computer can perform the numerical integration required in continuous simulation.

Continuous simulation languages for digital computers normally employ either a block or statement orientation. The block oriented languages employ a set of blocks which functionally emulate the circuit components of an analog computer. Most of the recently developed continuous simulation languages employ an equation orientation. In these languages, the differential or difference equations are explicitly coded in equation form. An advantage of the equation orientation is the increased flexibility afforded by the algebraic and logical features of these languages.

Combined Discrete-Continuous Models

In combined discrete-continuous models, the dependent variables may change both discretely and continuously. The world view of a combined model specifies that the system can be described in terms of entities, their associated attributes, and state variables. The behavior of the system model is simulated by computing the values of the state variables at small time steps and by computing the values of attributes of entities at event times.

There are three fundamental interactions which can occur between discretely and continuously changing variables, as listed below.

- A discrete change in value may be made to a continuous variable;
- An event involving a continuous state variable achieving a threshold value may cause an event to occur or to be scheduled; and
- The functional description of continuous variables may be changed at discrete time instants.

There are also two types of events that can occur in combined simulations. Time-events are those events which are scheduled to occur at specified points in time. They are commonly thought of in terms of discrete simulation models. In contrast, state-events are not scheduled, but occur when the system reaches a particular state. The idea of a state-event is similar to the concept of activity scanning in that the event is not scheduled but is initiated by the state of the system.

3.4.3. Simulation Language On Alternative Modeling

One of the most important aspects of a simulation study is computer programming. Writing computer code for a complex system used to be a difficult and arduous task. Because of this, several special purpose computer simulation languages have been developed to lighten the burden in computer programming. One of the best known and the most readily available simulation languages is the Simulation Language On Alternative Modeling (SLAM), which was developed by the Pritsker & Associates, Inc.

In SLAM, the alternate modeling world views are combined to provide a unified modeling framework. A discrete change system can be modeled within an event orientation, process orientation, or both. Continuous change systems can be modeled using either differential or difference equations. Combined discrete-continuous change systems can be modeled by combining the event and/or process orientation with the continuous orientation. In addition, SLAM incorporates a number of features which correspond to the activity scanning orientation.

The process orientation of SLAM employs a network structure which consists of specialized symbols called nodes and branches. These symbols model elements in a process such as queues, servers, and decision points. The modeling task consists of combining these symbols into a network model which pictorially represents the system. In short, a network is a pictorial representation of a process. The entities in the system flow through the network model.

In the event orientation of SLAM, we define the events and the potential changes to the system when an event occurs. The mathematical-logical relationships prescribing the changes associated with each event type are coded as FORTRAN subroutines. A set of standard subprograms is provided by SLAM for use in performing common discrete event functions, such as event scheduling, file manipulations, statistics collection, and random sample generation. The executive control program of SLAM controls the simulation by advancing time and initiating calls to the appropriate event subroutines at the proper points in simulated time. Hence, the modeler is completely relieved of the task of sequencing events to occur chronologically.

A continuous model is coded in SLAM by specifying the differential or difference equations which describe the dynamic behavior of the state variables. These equations are coded by the modeler in FORTRAN by employing a set of special SLAM defined storage arrays. When differential equations are included in the continuous model, they are automatically integrated by SLAM to calculate the values of the state variables within an accuracy prescribed by the modeler.

An important aspect of SLAM is that alternate world views can be combined within the same simulation model. There are six specific interactions which can take place between the network, discrete event, and continuous world views of SLAM:

1. Entities in the network model can initiate the occurrence of discrete events;
2. Events can alter the flow of entities in the network model;
3. Entities in the network model can cause instantaneous changes to values of the state variables;
4. State variables reaching prescribed threshold values can initiate entities in the network model;
5. Events can cause instantaneous changes to the values of state variables; and
6. State variables reaching prescribed threshold values can initiate events.

In summary, SLAM is an advanced FORTRAN-based simulation language that allows models to be built based on three different world views. It provides network symbols for building graphical models that are easily translated into input statements for direct computer processing. It contains subprograms that support both discrete event and continuous model developments. By combining network, discrete event, and continuous modeling capabilities, SLAM truly represents a simulation language for alternative modeling. The interfaces between the alternative modeling approaches are explicitly defined to allow new conceptual views of systems to be explored.

3.4.4. The ORATB SLAM Model Construction

A simulation model usually takes a set of assumptions about the operation of the system which are expressed as mathematical or logic relations among the interested objectives of the system. The first step in the simulation process is to formulate the problem by understanding the ORATB transport system context, identifying goals, specifying system performance measures, setting specific modeling objectives, and defining the system to be modeled. These functions serve to guide and bind the whole simulation process.

The goals of performing this simulation model are to evaluate the detailed characteristics of the ORATB system once its acquisition is generated by the optimization model. At the same time, simulation results are examined to re-evaluate the performance of the optimization model. The simulation model allows us to incorporate a number of characteristics of the ORATB system performance that were not taken into account in the optimization model. The most important of these characteristics are uncertainties in completion time, priority rules associated with cargo handling, sequential relationships associated with the various activities or tasks that are part of an individual job, alternative ways of executing these activities, and so on.

In the simulation model, it identifies each node that is part of the ORATB system and each job that has to be processed in the ORATB operations. The model also identifies dispatching rules that govern the order in which cargoes are handled. These rules include the inter-arrival times of the ORATB system, and service times at each node when cargoes are handled. Then, the simulation model specifies performance measures for the ORATB system. Common measures of performance include percentage of cargoes to be handled on time; total tardiness in port operation; utilization levels of berths and berth facilities, and so forth. Some specific results expected in the simulation model include the following:

- Occupancy and utilization levels of the tugs in the ORATB system;
- Occupancy and utilization levels of the barges in the ORATB system;
- Average waiting times for the ORATB system at each port;
- Occupancy and utilization levels of the berths at each port;
- Utilization levels of the loading and unloading facilities at each port;
- Assessment of likely down-times of the ORATB system; and
- Assessment of likely down-time of all related port facilities.

These performance measures are then analyzed to support decision making.

Decisions can be categorized into three levels: strategic planning, management control, and operational control. The strategic planning is the process of deciding on the objectives of an organization, on changes in these objectives, on the resources needed to achieve these objectives, and on the policies that are to govern the acquisition, use, and disposition of resources. The management control is the process by which managers assure that the required resources are obtained and used effectively and efficiently in the accomplishment of the organization's objectives. The operational control is the process of assuring that specific tasks are carried out effectively and efficiently. A list of problem situations categorized by these three levels is presented in Table 5.

3.4.5. SLAM Model Execution

Once the optimization model suggests an ORATB fleet to meet the demand, we evaluate the fleet's performance through the simulation model. Suppose there is a fleet of 15 ORTAB systems carrying iron ore from Ningbo to Wuhan. It is assumed that all ORATBs can be loaded simultaneously in Ningbo, if necessary[19]. At Wuhan, there is

---

[19] This is mainly because there are other ORATBs that are also loaded in Ningbo but unloaded in ports other than Wuhan.

Table 5 Areas Of Decision Making For Procedural Systems

Strategic planning

1. Design of new process
2. Design of new policies
3. Determination of effect of different priorities
4. Design of new systems
5. Forecast of production levels
6. Determination of required resources
7. Estimation of cost of alternatives

Management control

8. Determination of how to improve throughput
9. Determination of effect of changes in resource capacities
10. Determination of effect of delays in raw materials
11. Determination of how to relieve bottlenecks
12. Determination of effect of change in demand
13. Determination of effect of equipment failures
14. Determination of system efficiency

Operational control

15. Determination of capacity
16. Determination of bottlenecks
17. Determination of operational requirements
18. Assessment of in-process inventories
19. Determination of utilization
20. Determination of critical operation rates
21. Determination of best staffing configurations
22. Scheduling jobs
23. Scheduling resources

Source: Pritsker (1989).

only one iron ore unloading dock which supplies a storage yard that feeds a steel mill nearby through a conveyor system. The yard receives iron ore from a barge at the dock at a constant rate of 18,000 tons/day. At the same time, the yard supplies iron ore to the steel mill continuously at a constant rate of 9,000 tons/day. The unloading dock is open between 6 a.m. and 12 p.m. The completion of unloading occurs when the amount of iron ore remaining in the barge is less than 450 tons.

It is assumed that the yard has a capacity of 120,000 tons. When it is full, unloading is halted until the amount of iron ore in the yard decreases to 80% of capacity. On the other hand, when the yard is nearly empty, say less than 300 tons, supply to the mill is halted until 3,000 tons is reached to avoid frequent start-ups and shut-downs of the steel mill. The characteristics associated with the ORATB are listed in detail as the following.

1. Normal carrying capacity of the barge is 9,000 tons;
2. Travel time loaded is normally distributed with a mean of five days and a standard deviation of one and a half days, according to the tug selected;
3. Travel time unloaded is normally distributed with a mean of four days and a standard deviation of one day, assuming the same tug is deployed; and
4. Time to load is uniformly distributed in the interval of 2.9 and 3.1 days.

The initial conditions for the simulation are that the storage yard is half empty, *i.e.*, 60,000 tons, and the ORATBs are to arrive at their loading points at 0.5-day intervals, starting with the first at time 0.

The objective of this model is to simulate the above system for 365 days to obtain estimates of the following quantities:

1. Unloading dock utilization at Wuhan;
2. Loading dock utilization at Ningbo;
3. The time that there is iron ore input available to the steel mill;

4. The amount of iron ore in the storage yard;
5. The ORATB round trip traveling time;
6. The ORATB waiting time for unloading; and
7. The number of ORATBs waiting for unloading.

The ORATB system is simulated using a combined network-continuous model. The continuous variables are used to represent the level of ore in the barge being unloaded and in the storage yard. The network is used to model the movement of the ORATBs through the system and the interactions between the continuous and discrete elements of the system.

Two state variables are used in this simulation: *SS(1)*, the amount of iron ore in the barge available for unloading; and *SS(2)*, the amount of iron ore in the storage yard. *SS(1)* is zero when no barge is in the unloading dock; otherwise it is equal to the amount of iron ore in a barge that is in the unloading dock. When a barge leaves the unloading dock, *SS(1)* either becomes zero or is equal to the amount of iron ore in the next waiting barge to be unloaded. By defining *SS(1)* in this manner, a separate state variable for the amount of iron ore in each barge needs not to be defined.

There are three *XX* variables which are used in the simulation model to control the flow of iron ore between the unloading barge and the steel mill. Each of these variables represents a switch which is open when it is equal to one and closed when equal to zero. *XX(1)* represents the dock input switch and is open between the dock opening hours of 6 a.m. to 12 p.m., and is closed otherwise. *XX(2)* models the storage yard input switch and is closed whenever the yard iron ore level, *SS(2)*, reaches its capacity of 120,000 tons. It is re-opened when the level of iron ore decreases to 96,000 tons. *XX(3)* is the storage yard output switch and is closed whenever the yard ore level has decreased to less than 300 tons, thereby halting the flow to the steel mill. *XX(3)* is re-set to open when the iron ore level in the yard has increased to 3,000 tons, thereby restoring the flow of iron ore to the steel mill. A schematic diagram depicting the arrangement of these three switches is provided in Figure 6.

The equations describing the state variables *SS(1)* and *SS(2)* are coded in subroutine *STATE* shown in Table 6. The variable *RATIN* represents the ore flow rate into the yard. It is set to zero if *XX(1)*, *XX(2)*, or *SS(1)* is zero, and is set equal to 18,000 otherwise. The variable *RATOUT*, representing the ore flow rate from the yard to the mill, equals 9,000 if *XX(3)* = 1, and equals 0 if *XX(3)* = 0. Equations for state variables *SS(1)* and *SS(2)* are written as difference equations in terms of *RATIN* and *RATOUT*. In this study, we are integrating the state equations explicitly in subroutine *STATE*, as shown in Table 6.

The network model is structured with three sub-processes consisting of the barge flow through the system, the start-up/shut-down of dock operations, and the state events. The network for the barge flow sub-process is depicted in Figure 7. The initial 15 ORATBs are created by the *CREATE* node at 0.50-day intervals, beginning with the first at time 0. The ORATB proceeds to the *ASSIGN* node labeled *NB* where their arrival time to Ningbo is marked as *ATRIB(1)*. The ORATBs then undertake the loading activity which is represented by *ACTIVITY 1*. The trip from Ningbo to Wuhan is modeled by *ACTIVITY 2*, the ORATBs then wait in file 1 at the *AWAIT* node for the resource *DOCK*. A single unit of resource *DOCK* is available as specified by the resource block. When the *DOCK* becomes available, the state variable *SS(1)* is set to 9,000 at the *ASSIGN* node indicating that there is 9,000 tons of ore available for unloading. The barge then undergoes *ACTIVITY 3* which represents the unloading activity. This *ACTIVITY* is completed at the next release of the node labeled *ENDU*. The node labeled *ENDU* is a *DETECT* node which is released when *SS(1)* crosses, in the negative direction, the threshold value of 450 which indicates that the state-event "end-of-unloading' has occurred.

At the completion of unloading, the barge entity is routed to the *ASSIGN* node where *SS(1)* = 0, and then releases the *DOCK* at the *FREE* node. The return trip to Ningbo is modeled by *ACTIVITY 4*. At the *COLCT* node, statistics are collected on the round trip

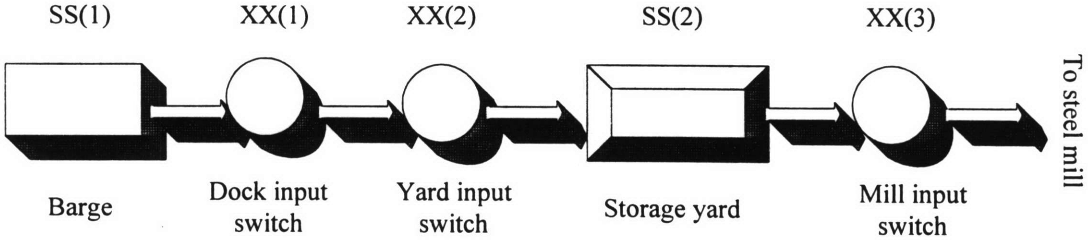


Figure 6 Iron Ore Flow From Barge To Mill In The ORATB Simulation Model

---

Table 6 The ORATB Simulation Model *STATE* Subroutine

```
***********************************************************************
      SUBROUTINE STATE
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,
     1NCLNR,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100), SSL(100),TNEXT,
     2TNOW,XX(100)
C***RATIN=0 IF DOCK OR YARD INPUT CLOSED OR NO WAITING BARGE, ELSE 18000
      RATIN=18000.
      IF(XX(1)*XX(2)*SS(1).EQ.0.0) RATIN=0.
C***RATOUT=0 IF MILL INPUT OFF, ELSE 9000
      RATOUT=9000.*XX(3)
      SS(1)=SSL(1)-DTNOW*RATIN
      SS(2)=SSL(2)+DTNOW*(RATIN-RATOUT)
      RETURN
      END
***********************************************************************
```

time for the ORATB which is then routed to the *ASSIGN* node labeled *NB* to repeat the cycle through the network.

The second network is for the startup/shut-down sub-process as depicted in Figure 8. The *CREATE* node inserts an entity into the network beginning at time 0.25 days, *i.e.*, 6 a.m., and then daily thereafter. At the *ASSIGN* node, the dock input switch is opened by setting *XX(1)* = 1. The dock remains open during the 0.75 days required for the entity to transverse the *ACTIVITY* before being closed at the *ASSIGN* node where *XX(1)* is reset to 0. Then, the dock remains closed until the next entity is inserted into the network at 6 a.m. the next day.

There are five possible conditions that could result in a state-event during the simulation process, as listed below.

Condition	State-event
The ore level in the unloading barge, *SS(1)*, decreases to 450	Barge unloading is completed
The ore level in the yard, SS(2), decreases to 300	Stop supply to the mill by setting *XX(3)* = 0
The ore level in the yard increases to 3,000	Start supply to the mill by setting *XX(3)* = 1
The ore level in the yard reaches its capacity of 120,000	Close input to the yard by setting *XX(2)* = 0
The ore level in the yard decreases to 96,000	Open input to the yard by setting *XX(2)* = 1

These five state-events are modeled by the five sub-networks depicted in Figure 9. The first sub-network is used to detect the end of unloading state-event and causes the completion of *ACTIVITY 3* whose duration is keyed to the release of the node labeled *ENDU*. The other four sub-networks are used to detect and process state-events which cause the opening and closing of the yard and the mill input valves. The tolerance of each state-event is set at 300. The value prescribed for a tolerance is set according to the

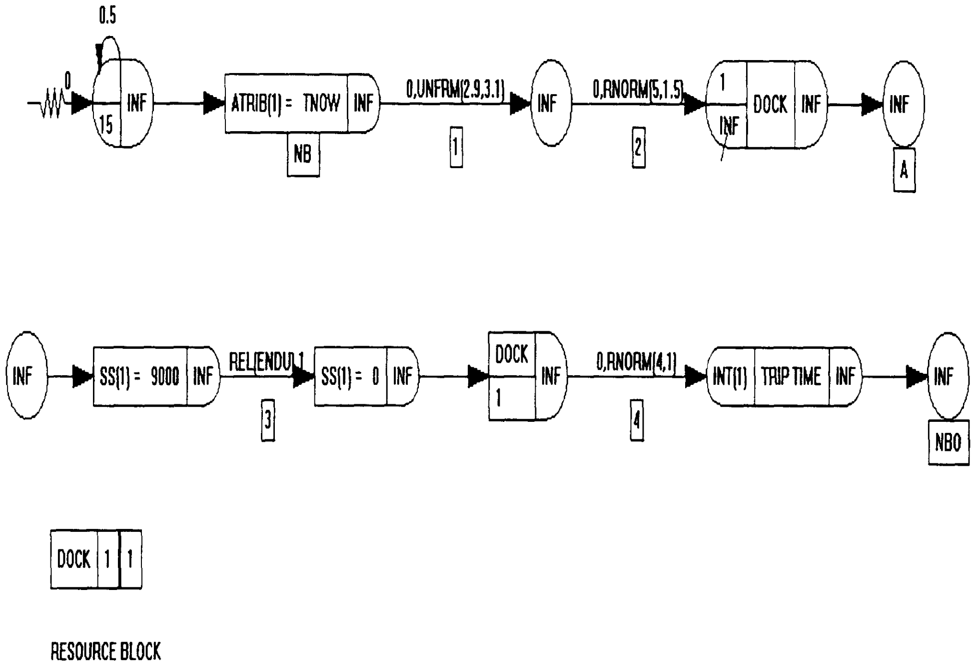


Figure 7 Iron Ore Flow Sub-Process In The ORATB Simulation Model

---

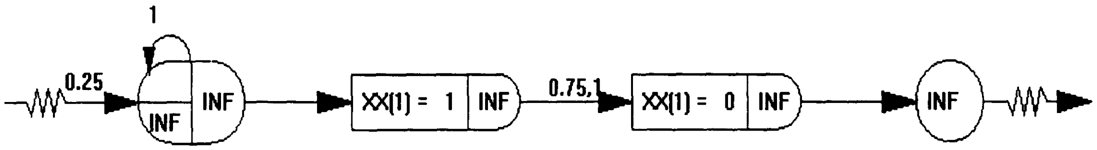
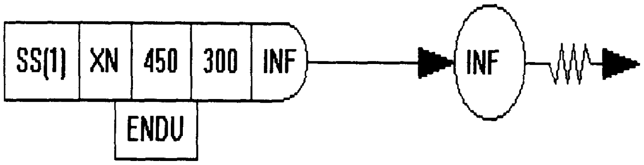
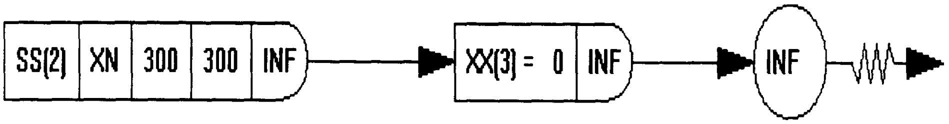


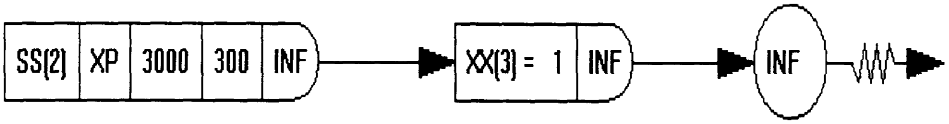
Figure 8 Start-Up/Shut-Down Sub-Process In The ORATB Simulation Model



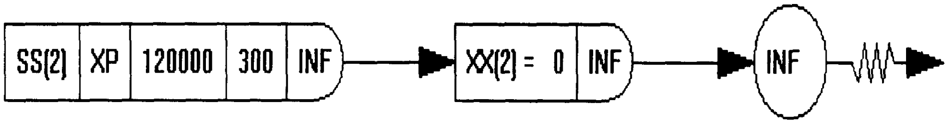
End of unloading activity



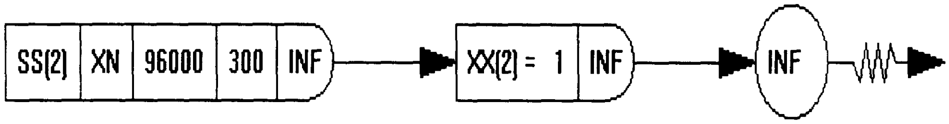
Yard is empty, close mill input valve



Yard byackup to 3000, open mill input valve



Yard capacity reached, close yard input valve



Yard level dropped below 80%, open yard input valve

Figure 9 State-Event Sub-Processes In The ORATB Simulation Model accuracy with which a state-event should be detected. The value of the tolerance should also consider the value given to *DTMIN* and the maximum rate of change of the state variable. In this case, *DTMIN* = .0025 days and the maximum rate is 18,000 tons/day, hence tolerances of 45 tons or greater should enable detection of state-events within tolerance.

The input statements for this case are listed in Table 7. In addition to the network statements, the necessary control statements are included to obtain: a plot of the ore level on an unloading barge and the ore level in the yard; and time-persistent statistics on the mill input availability and the average ore level in the yard. The *INITIALIZE* statement specifies that the model is to be simulated for 365 days. *MONTR* statements with the *CLEAR* option are used to clear statistics at time 65 and 165.

The output results and sensitivity analysis of the ORATB simulation model will be presented in Section 4.4.2.

3.5. Adaptive Iteration Mechanism Of The IAIOS Model

In the previous two sections, we presented that the resource acquisition and utilization decisions associated with the ORATB transport problem have been partitioned into two manageable models. The ORATB optimization model deals with the long-term strategic planning associated with resource acquisition, while the simulation model deals with the short-term tactical planning associated with resource utilization. In this section, we analyze the adaptive iteration mechanism of the IAIOS model.

The adaptive iterations occur sequentially between the optimization model and the simulation model. As shown in Figure 10, the optimization model is first solved, obtaining an initial recommendation for tug/barge acquisition. Then, these acquisition requirements are examined in the simulation model to check their consistency with

Table 7 The ORATB Simulation Model Input Statements

```
************************************************************************
      GEN,MINGQI,ORATB FLEET,11/18/96,1;
      LIMITS,1,2,100;
      TIMST,XX(3),STML INPUT AVAIL;
      CONT,0,2,.0025,.25,.25;
      RECORD,TNOW,DAYS,O,P,.25;
      VAR,SS(1),T,BARGE LEVEL,0,9000;
      VAR,SS(2),S,YARD LEVEL,0,120000;
      TIMST,SS(2),YARD LEVEL;
      INTLC,SS(2)=60000,XX(2)=1,XX(3)=1,XX(1)=0;
      NETWORK;
      ;
      ;BARGE FLOW SUBPROCESS
      ;---------------------
           RESOURCE/DOCK(1),1;
           CREATE,.5,0,,15;               CREATE INITIAL ARRIVALS
      NB   ASSIGN,ATRIB(1)=TNOW;          MARK ARRIVAL TIME TO NINGBO
           ACT/1,UNFRM(2.9,3.1);          LOADING
           GOON;                          END OF LOADING
           ACT/2,RNORM(5.,1.5);           TO WUHAN
           AWAIT(1),DOCK/1;               AWAIT THE DOCK
           ASSIGN,SS(1)=9000;             RESET BARGE IRON ORE LEVEL
           ACT/3,REL(ENDU);               UNLOADING
           ASSIGN,SS(1)=0;                SET BARGE IRON ORE LEVEL TO 0
           FREE,DOCK;                     FREE THE DOCK
           ACT/4,RNORM(4,1);              RETURN TRIP TO NINGBO
           COLCT,INT(1),TRIP TIME;        COLLECT STATISTICS
           ACT,,,NB;                      BRANCH TO NINGBO
      ;
      ;SHIFT START UP/SHUT DOWN SUBPROCESS
      ;-----------------------------------
           CREATE,1,.25;
```

Table 7 (continued)

```
     ASSIGN,XX(1)=1;               BEGIN SHIFT AT 6 A.M.
     ACT,.75;                      CONTINUE FOR ¾ DAYS
     ASSIGN,XX(1)=0;               CLOSE SHIFT AT 12 P.M.
     TERM;
;
;STATE EVENT SUBPROCESSES
;------------------------
ENDU DETECT,SS(1),XN,450,300;      END OF UNLOADING ACTIVITY
     TERM;
     DETECT,SS(2),XN,300,300;      YARD IS EMPTY
     ASSIGN,XX(3)=0;               CLOSE MILL INPUT SWITCH
     TERM;
     DETECT,SS(2),XP,3000,300;     YARD BACKUP TO 50
     ASSIGN,XX(3)=1;               OPEN MILL INPUT SWITCH
     TERM;
     DETECT,SS(2),XP,120000,300;   YARD CAPACITY REACHED
     ASSIGN,XX(2)=0;               CLOSE YARD INPUT SWITCH
     TERM;
     DETECT,SS(2),XN,96000,300;    YARD DROPPED BELOW 80
     ASSIGN,XX(2)=1;               OPEN YARD INPUT SWITCH
     TERM;
     ENDNETWORK;
;
INITIALIZE,0,365;
MONTR,CLEAR,65;
MONTR,SUMRY,165;
MONTR,CLEAR,165;
FIN;
************************************************************************
```

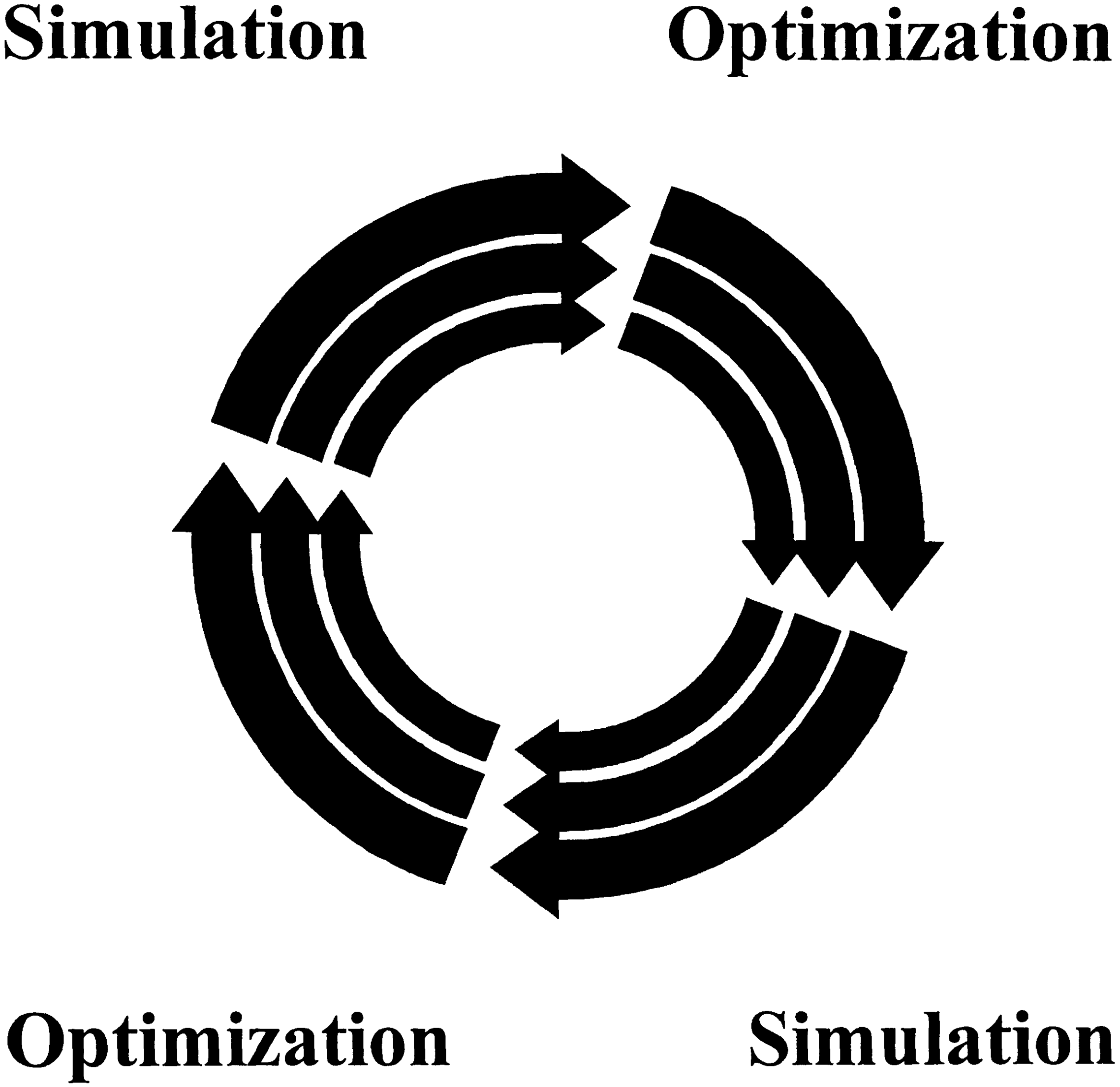


Figure 10 Sequential Adaptive Iterations Of The IAIOS Model

existing managerial policies that have not been included explicitly in the initial optimization model formulation. Constraints may be changed and/or added to the optimization model to eliminate the found inconsistencies. For example, in order to prevent excessive undertime for all the tugs and barges, the following constraints might be added to the optimization model:

$$\sum_{t=1}^{T} M_{ijks}^{t} - T(0.75)\, M_{ijks} \geq 0; \text{ and}$$

$$\sum_{t=1}^{T} N_{ijkl}^{t} - T(0.75)\, N_{ijkl} \geq 0.$$

These constraints would then require the average utilization of all tugs of class s and all barges of class l to be at least 75% over the whole T time period.

Likewise, if the utilization levels of certain size tugs and barges seemed to be excessive, leaving little or no room for absorbing demand uncertainties, the following constraints could be added to the optimization model:

$$\sum_{t=1}^{T} M_{ijks}^{t} - T(0.90)\, M_{ijks} \leq 0; \text{ and}$$

$$\sum_{t=1}^{T} N_{ijkl}^{t} - T(0.90)\, N_{ijkl} \leq 0.$$

These two constraints would then force the average utilization of all tugs of class s and barges of class l to be less than 90% over the whole T time period.

Then, the optimization model is run again with these newly adapted constraints to generate a new acquisition strategy. Again, the new acquisition strategy is evaluated in the simulation model to check inconsistencies in utilization level. This adaptive iteration continues until no inconsistencies can be found or no improvement can be achieved for a new acquisition strategy.

Adding new constraints to the problem allows decision makers to explore the cost sensitivity to the proposed changes. In a linear programming model, most of this information is provided directly by the shadow prices associated with the original model constraints. Our optimization model, however, is of a linear mixed integer programming type, which does not generate similar shadow price information. On the other hand, although there are no integer variables involved, we cannot use the shadow price for the sensitivity analysis for the IAIOS model. This is because once constraints are changed and/or added, the optimization model becomes a different one from the original model. Therefore, we conduct the sensitivity analysis for the IAIOS model in the simulation model.

Once a satisfactory tug/barge acquisition has been obtained, a simulation is conducted with these data as input parameters to the simulation model. The tug/barge utilization levels obtained from the simulation model then are examined. If these levels are not considered acceptable, new changes in the tug/barge composition and/or cost structure may be changed. These changes modify the optimization model formulation, and a new iteration is taken place. After the utilization levels are satisfied, the performance in terms of delivery time of the ORATB system is evaluated. For example, we may have early deliveries which result in longer period of storage time at the yard. In that case, we can insert yard capacity constraints into the optimization model. and have a new iteration.

Once both acceptable utilization and performance levels are obtained, extensive sensitivity analysis can be performed in the simulation model to test how robust the recommended tug/barge acquisition is to the changes in the problem parameters, such as demands, and loading and unloading rates. These sensitivity analysis results may indicate that some of the parameters, constraints or demand characteristics need to be modified, and the problem is run again starting with the optimization model.

The proposed IAIOS modeling approach provides the decision makers with an effective tool to test the performance of the ORATB operations under a wide range of anticipated conditions, and thus permits a satisfactory acquisition of the ORATB transport system.

3.6. Chapter Summary

In this chapter, we began with the introduction of the integrated modeling approach to the ORATB transport problem. The proposed iterative model consists of an optimization model for tug/barge acquisition and a simulation model for tug/barge utilization. The proposed iterative approach eliminates the weaknesses inherent in both of optimization and simulation approaches. It also facilitates the interaction of the decision makers with the model at different levels, and allows comprehensive testing of a wide variety of options.

The structure of the IAIOS model is as shown in Figures 1 and 10, and the sequential adaptive iteration processes are indicated by the feedback branches in the figures. The two sub-models are conducted at distinct levels sequentially. At the first level, the optimization model solves the broad acquisition of resources. At the second level, the simulation model deals with the utilization of the acquired resources and the detailed performance of activities.

We then introduced the optimization model and the simulation model in Sections 3.3 and 3.4, respectively. In the introduction of the optimization model, we presented the optimization procedures, formulation of the model, GAMS, as well as the ORATB GAMS model construction and its execution. For the simulation model, we discussed simulation process, simulation modeling perspectives, SLAM, and the ORATB SLAM model construction and its execution. Both of these two introductions are in great detail.

In Section 3.5, we focused on the sequential adaptive iteration of the IAIOS model. The adaptive iterations occur between the optimization model and the simulation model sequentially. The optimization model is first solved to obtain an initial recommendation for tug/barge acquisition. Then, these acquisition requirements are examined in the simulation model to check their consistency with existing managerial policies that have not been explicitly included in the initial optimization model formulation. If there are inconsistencies, constraints and/or cost structures are changed and/or added to the optimization model. Then, the optimization model is run again with these newly adapted constraints and cost structures to generate a new acquisition strategy. Again, the new acquisition strategy is evaluated in the simulation model to check inconsistencies in utilization and performance level. This adaptive iteration continues until no inconsistencies can be found or no improvement can be achieved for a new acquisition strategy.

Chapter Four

4. CASE DESIGN AND NUMERICAL RESULTS

4.1. The Case: The Yangtze Valley ORATB Transport System

The Yangtze River, the longest river in China, is about 6,300 km (3,915 miles) long and flows through the middle of China from the western to the eastern border. It is also one of the longest inland waterways in the world, with a total navigable distance of 2,660 km (1,653 miles). Being the principal economic hinterland of China, as shown in Figure 11, the Yangtze Valley plays a very important part in Chinese economy. More than one-half of China's industrial and agricultural production is from the Yangtze Valley according to the study of the World Bank (1992). At the same time, the Yangtze River is the main transportation artery in China. In 1995, the total cargo throughput along all the major Yangtze ports reached 323.4 million tons[20].

The T-shape distributed geographical locations of coastal and river ports, as shown in Figure 12, makes transshipment between coastal and river ports so vital for the economic development of the Yangtze Valley. On the other hand, the long distances between major coastal ports and major Yangtze ports, as shown in Tables 8 and 9, make such transshipment necessary for bulk cargoes going into the Yangtze Valley and container cargoes out of the region. Therefore, it is the objective of this case study to establish an

---

[20] Including all the throughput statistics of the Port of Shanghai.

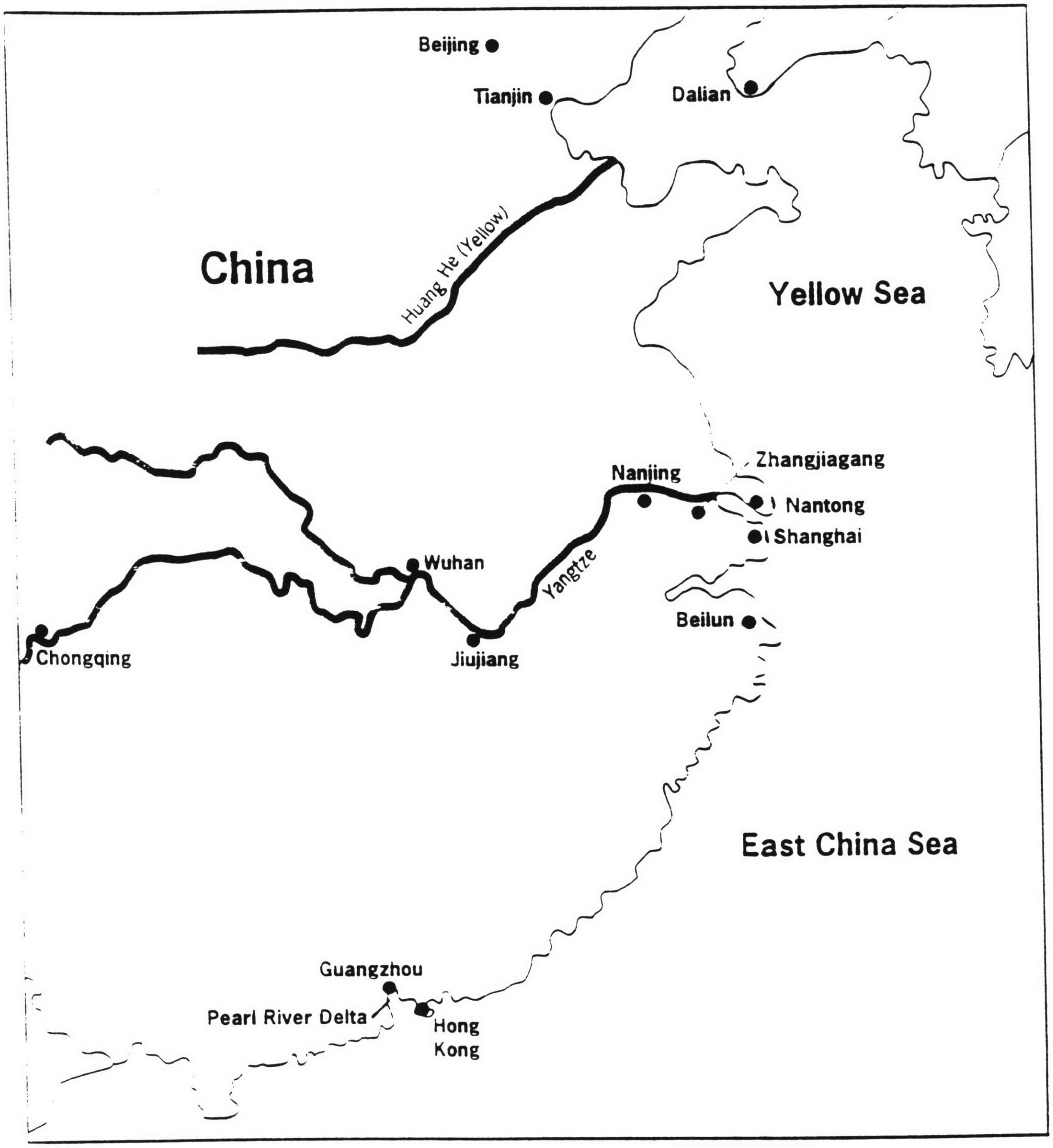


Figure 11 The Geographic Location Of The Yangtze Valley In China

Source: Atlas of China. 1995.

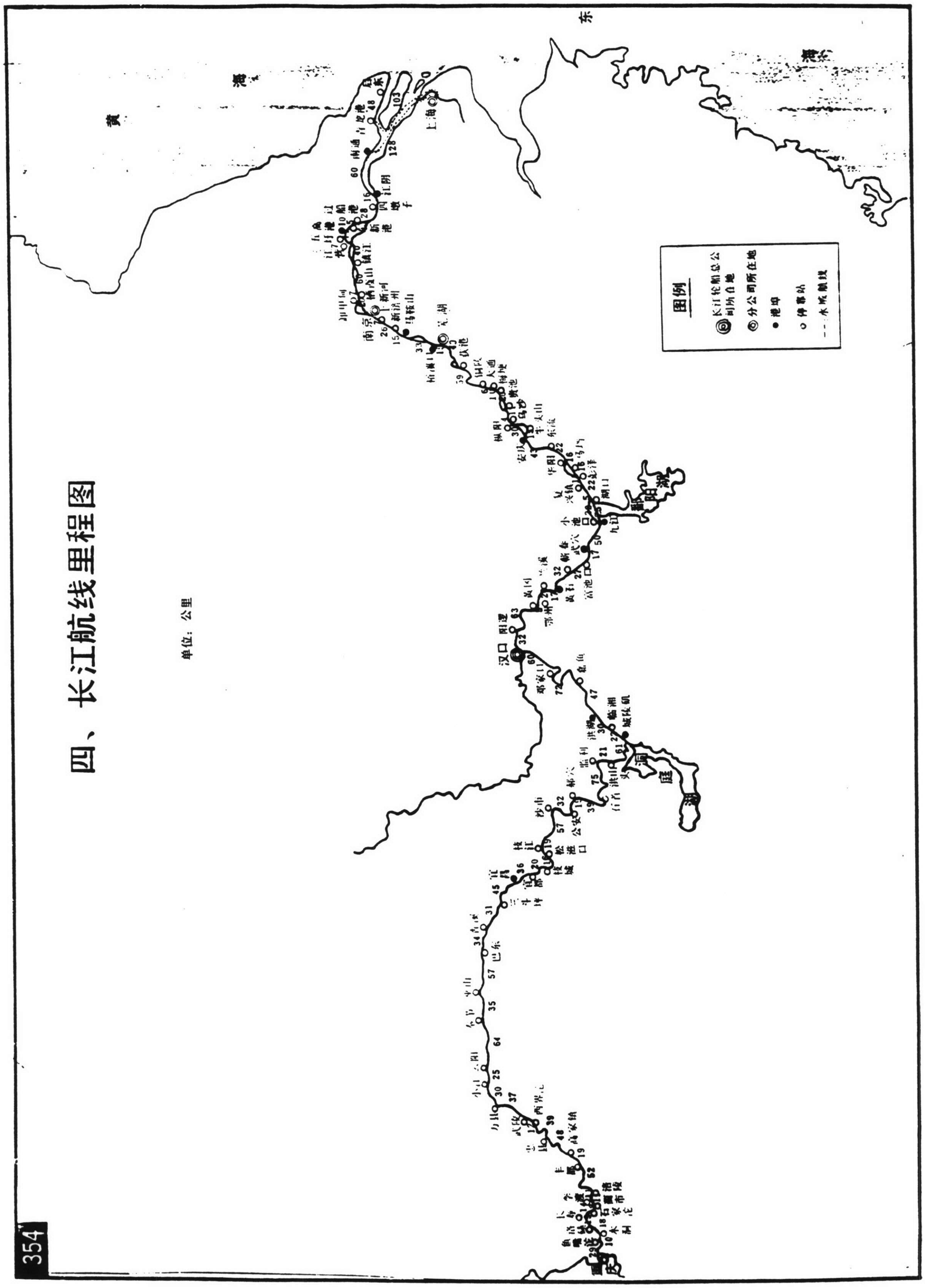


Figure 12 Port Distribution Along The Yangtze River

Source: China Transportation Atlas, 1995.

Table 8 **Distance Table Between Ningbo And Major Yangtze Ports**

unit: nautical miles

Origin	Destination	Distance
Ningbo, Zhejiang	Shanghai	140
	Nantong, Jiangsu	209
	Zhenjiang, Jiangsu	305
	Nanjing, Jiangsu	352
	Ma'anshan, Anhui	378
	Wuhu, Anhui	404
	Tongling, Anhui	462
	Chizhou, Anhui	481
	Anqing, Anhui	514
	Huayang, Anhui	549
	Jiujiang, Jiangxi	602
	Wuxue, Hubei	629
	Huangshi, Hubei	670
	Wuhan, Hubei	747
	Chenglingji, Hubei	872
	Shashi, Hubei	1,006
	Zhicheng, Hubei	1,055
	Yichang, Hubei	1,085
	Badong, Hubei	1,145
	Wushan, Sichuan	1,176
	Fengjie, Sichuan	1,195
	Yunyang, Sichuan	1,229
	Wanxian, Sichuan	1,269
	Zhongxian, Sichuan	1,306
	Wanjiazhen, Sichuan	1,332
	Fengdu, Sichuan	1,342
	Fuling, Sichuan	1,371
	Chongqing, Sichuan	1,435

Source: China Transportation Atlas, 1995.

Table 9 Distance Table Between Ningbo And Major Sea Ports

unit: nautical miles

Origin	Destination	Distance	Notes
Ningbo, Zhejiang	Dalian, Liaoning	670	North of Ningbo
	Yingkou, Liaoning	862	North of Ningbo
	Qinhuangdao, Hebei	800	North of Ningbo
	Tanggu, Tianjin	833	North of Ningbo
	Tianjin	826	North of Ningbo
	Yantai, Shandong	636	North of Ningbo
	Qingdao, Shandong	516	North of Ningbo
	Lianyungang, Jiangsu	495	North of Ningbo
	Shanghai	140	North of Ningbo
	Zhapu, Zhejiang	70	North of Ningbo
	Dinghai, Zhejiang	34	North of Ningbo
	Zhenhai, Zhejiang	11	North of Ningbo
	Shipu, Zhejiang	87	South of Ningbo
	Haimen, Zhejiang	142	South of Ningbo
	Wenzhou, Zhejiang	219	South of Ningbo
	Fuzhou, Fujian	372	South of Ningbo
	Shantou, Guangdong	586	South of Ningbo
	Huangpu, Guangdong	807	South of Ningbo
	Guangzhou, Guangdong	824	South of Ningbo
	Zhanjiang, Guangdong	967	South of Ningbo
	Haikou, Hainan	995	South of Ningbo
	Basuo, Hainan	1,138	South of Ningbo
	Sanya, Hainan	1,157	South of Ningbo
	Beihai, Guangxi	1,112	South of Ningbo
	Fangcheng, Guangxi	1,174	South of Ningbo

Source: China Transportation Atlas, 1995.

effective and efficient transshipment system between one of the major coastal ports and all major Yangtze ports.

In this case study, the Port of Ningbo is selected as the transshipment center between ocean shipping and inland waterway transport. The main reason is that the Port of Ningbo, located in Beilun, has an open access to ocean-going vessels through a naturally deep navigation channel (-22 m). Although the Port of Shanghai is geographically closer to the Yangtze ports than Ningbo, its shallow water depth (-9.5 m maximum) does not permit large ocean-going vessels to access directly. Other reasons for us not to choose the Port of Shanghai are the traffic congestion in the port and rapid urban development around the port area.

In this case study, we also propose to establish inland waterway distribution hubs, such as in Chongqing, Wuhan and Nanjing, along the Yangtze River. As shown in Table 10, the fact of the short inter-port distances between these large industrial cities and their surrounding ports and the long inter-port distances between these large industrial city ports and the Port of Ningbo makes it natural to have a transport network with several intermediate distribution hubs.

To achieve the above objectives, we propose to use the ORATB transport system to fulfill the transshipment demand between the coastal port in Beilun and the Yangtze River ports. In this chapter, we apply the proposed IAIOS modeling approach to obtain the optimal resource acquisition and utilization strategy for the development of the proposed ORATB transport system.

4.1.1. Shipping Demand In The Yangtze Valley

Overseas trade in the Yangtze Valley developed rapidly during 1990 and 1995, as shown in Table 11, and transportation demands are continuously increasing. Figure 13 shows

Table 10 Distance Table Of Major Ports Along The Yangtze River

(downstream from the west to the east)

unit: nautical miles

Chongqing ~65~ Fuling ~28~ Fengdu ~10~ Wanjiazhen ~26~

Zhongxian ~48~ Wanxian ~30~ Yunyang ~35~ Fengjie ~19~

Wushan ~31~ Badong ~59~ Yichang ~30~ Zhicheng ~50~

Shashi ~133~ Chenglingji ~125~ Wuhan ~77~ Huangshi ~41~

Wuxue ~27~ Jiujiang ~54~ Huayang ~35~ Anqing ~32~

Chizhou ~20~ Tongling ~58~ Wuhu ~26~ Ma'anshan ~26~

Nanjing ~47~ Zhenjiang ~96~ Nantong ~69~ Shanghai

Note: Major industrial cities are in bold italic.

Source: China Transportation Atlas, 1995.

Table 11 1990-95 Throughput Statistics At Major Yangtze Ports

unit: 1,000 tons

	1990	1991	1992	1993	1994	1995
Ningbo	25,535	33,899	43,669	53,214	58,498	68,528
Shanghai	139,590	146,788	162,968	175,956	165,809	165,672
Nantong	15,727	15,420	17,636	18,069	19,869	20,890
Zhangjiagang	3,601	4,569	5,323	7,413	6,878	8,784
Zhenjiang	13,468	13,467	15,614	18,322	17,431	18,461
Nanjing	42,123	44,003	46,716	47,149	45,602	50,404
Ma'anshan	2,713	2,929	3,531	3,884	4,563	5,348
Wuhu	3,408	3,154	3,353	3,910	4,006	3,395
Tongling	1,293	3,188	1,742	1,960	1,935	2,116
Chizhou	979	1,200	1,374	1,475	1,126	1,123
Anqing	6,208	6,804	6,849	6,707	5,537	6,254
Jiujiang	4,877	6,375	6,486	6,696	5,975	6,608
Huangshi	1,044	1,168	1,280	1,298	842	1,168
Wuhan	15,504	15,545	16,721	17,505	16,092	16,280
Chenglingji	4,043	4,610	5,610	5,870	5,398	5,988
Shashi	1,084	1,272	1,390	1,540	1,325	1,422
Zhicheng	1,170	1,402	1,988	2,247	2,075	1,946
Yichang	1,785	2,088	2,390	1,878	2,769	3,108
Chongqing	2,446	3,526	4,046	4,254	3,874	4,439
Total above	**286,598**	**311,407**	**348,686**	**379,347**	**369,604**	**391,934**

Source: Statistical Yearbooks of the Ministry of Communications, 1991-1996.

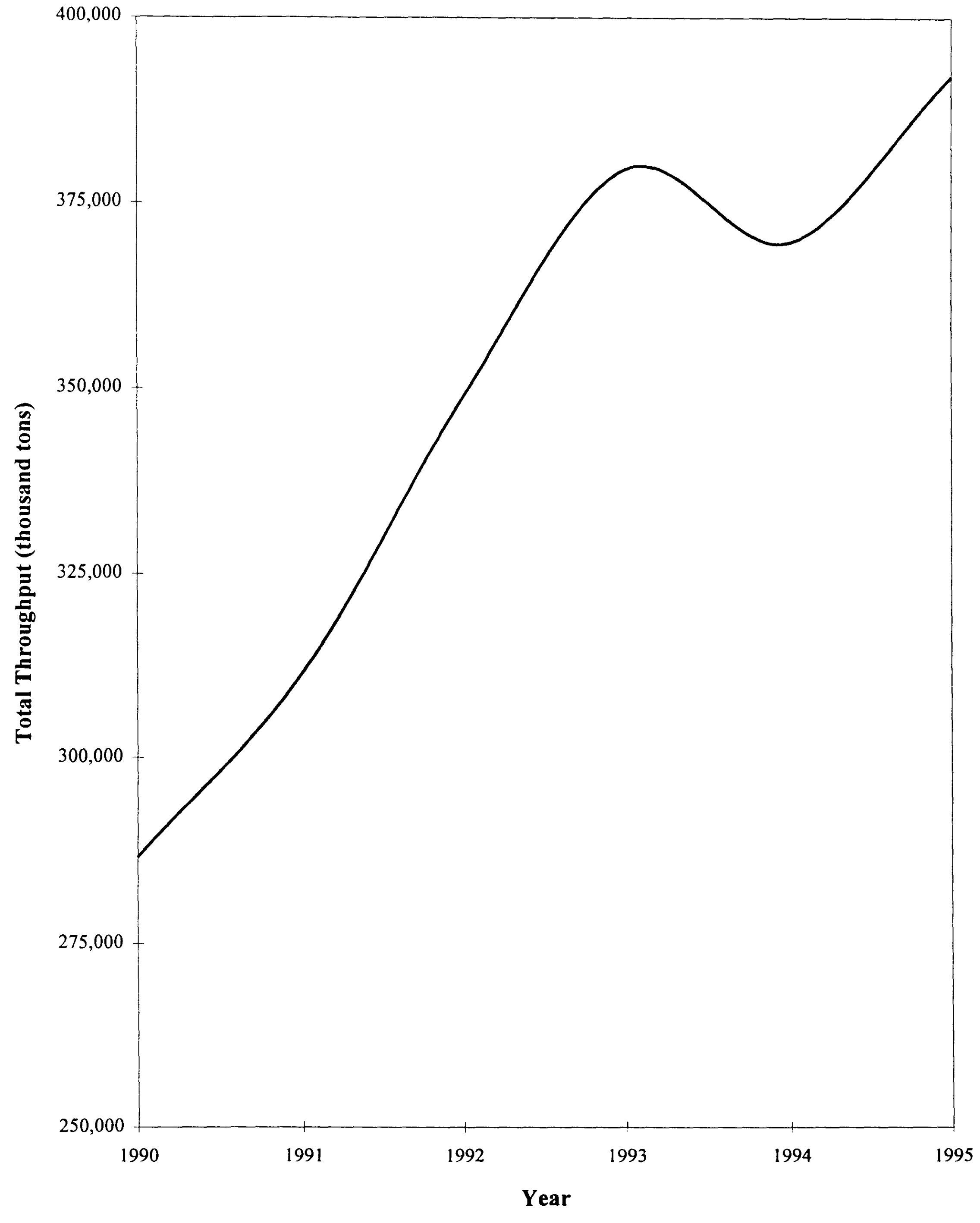


Figure 13 Iron Ore Throughput Changes Between 1990 And 1995 At All Major Yangtze Ports

Source: Statistical Yearbooks of the Ministry of Communications, 1991-1996.

that the overall throughput of major Yangtze ports increased 37% from the 1990's 286.6 million tons to 1995's 391.9 million tons. However, the growth rates are not evenly distributed. The Port of Ningbo had the fastest growth rate, 268% in five years, mainly because of its enhancing role in transshipping iron ore. In general, downstream ports had faster growth rates than upstream ports. The obvious reason for the slow growth at upstream ports is that the economy in coastal areas grew more rapidly than the inland areas. The slow growth at upstream ports is also attributed to the ineffective and inefficient waterway transport system. We believe it is the shallow water depth and low bridge clearance along the Yangtze River that puts the future growth at inland ports in danger. A new transport system must be introduced to improve transport efficiency along the Yangtze River. We believe that the proposed ORATB system can overcome the limitations in water depth and height clearance, and present an effective and efficient transport system in the Yangtze River shipping.

As a starting point, we would like to introduce the ORATB system first to iron ore and container transport along the Yangtze River. The iron ore increased dramatically in recent years mainly due to the large expansions of the steel mills along the River. As shown in Tables 12-16, the overall throughput of iron ore increased 107.1% from 38.6 million tons in 1990 to 80.0 million tons in 1994, which represented more than 20% of the total throughput for the same region for the same year. During the same period of time, total overseas imports increased 85.9% from 9.9 million tons to 18.4 million tons.

The origin and destination (O-D) patterns and their changes can also be identified from Tables 12-16. In 1990, among the 9.9 million tons of iron ore imported abroad, Shanghai had the share of 56.6% and Ningbo 43.4%. In 1994, though, Ningbo took the leading share of 82.9% of the 18.4 million tons imported, while Shanghai only 17.1%. This pattern change was also reflected by the facts that throughput increased 107.1% when overseas imports increased 85.9%. The truth was that iron ore imported to Ningbo were all transshipped to other places.

Table 12 1990 Iron Ore Traffic At Major Yangtze Ports

unit: tons

Port name	Throughput	Inbound	Outbound	Local use
Ningbo	8,572,264	4,286,132	4,286,132	0
Shanghai	13,691,348	9,873,951	3,817,397	6,056,554
Nantong	3,851,620	1,925,810	1,925,810	0
Zhangjiagang	165,560	82,780	82,780	0
Zhenjiang	2,303,946	1,151,973	1,151,973	0
Nanjing	3,142,677	2,821,875	320,802	2,501,073
Ma'anshan	839,377	839,377	0	839,377
Wuhu	234,154	173,887	60,267	113,620
Tongling	220,617	122,522	98,095	24,427
Chizhou	39,091	0	39,091	n/a
Anqing	49,767	0	49,767	n/a
Jiujiang	177,448	177,448	0	177,448
Huangshi	156,925	0	156,925	n/a
Wuhan	4,742,608	4,742,608	0	4,742,608
Chenglingji	140,979	120,096	20,883	99,213
Shashi	1,296	0	1,296	0
Zhicheng	0	0	0	0
Yichang	1,312	0	1,312	0
Chongqing	276,288	192,517	83,771	108,746
Total	**38,607,277**	**26,510,976**	**12,096,301**	**14,663,066**

Source: 1991 Statistical Yearbook of the Ministry of Communications.

Table 13 1991 Iron Ore Traffic At Major Yangtze Ports

unit: tons

Port name	Throughput	Inbound	Outbound	Local use
Ningbo	11,658,000	5,829,000	5,829,000	0
Shanghai	17,885,000	13,193,000	4,692,000	8,501,000
Nantong	3,194,000	1,597,000	1,597,000	0
Zhangjiagang	78,000	39,000	39,000	0
Zhenjiang	2,230,000	1,115,000	1,115,000	0
Nanjing	3,176,000	2,669,000	507,000	2,162,000
Ma'anshan	1,186,000	1,186,000	0	1,186,000
Wuhu	153,000	115,000	38,000	77,000
Tongling	28,000	0	28,000	n/a
Chizhou	56,000	0	56,000	n/a
Anqing	67,000	0	67,000	n/a
Jiujiang	173,000	173,000	0	173,000
Huangshi	217,000	0	217,000	n/a
Wuhan	5,198,000	5,198,000	0	5,198,000
Chenglingji	146,000	146,000	0	146,000
Shashi	0	0	0	0
Zhicheng	0	0	0	0
Yichang	0	0	0	0
Chongqing	123,000	44,000	79,000	n/a
Total	**45,568,000**	**31,304,000**	**14,264,000**	**17,443,000**

Source: 1992 Statistical Yearbook of the Ministry of Communications.

Table 14 1992 Iron Ore Traffic At Major Yangtze Ports

unit: tons

Port name	Throughput	Inbound	Outbound	Local use
Ningbo	15,116,000	7,558,000	7,558,000	0
Shanghai	19,538,000	15,434,000	4,104,000	11,330,000
Nantong	3,342,000	1,671,000	1,671,000	0
Zhangjiagang	80,000	40,000	40,000	0
Zhenjiang	2,918,000	1,459,000	1,459,000	0
Nanjing	3,879,000	3,142,000	737,000	2,405,000
Ma'anshan	1,041,000	1,041,000	0	1,041,000
Wuhu	161,000	97,000	64,000	33,000
Tongling	277,000	150,000	127,000	23,000
Chizhou	39,000	0	39,000	n/a
Anqing	136,000	0	136,000	n/a
Jiujiang	152,000	152,000	0	152,000
Huangshi	363,000	0	363,000	n/a
Wuhan	4,984,000	4,984,000	0	4,984,000
Chenglingji	221,000	221,000	0	221,000
Shashi	0	0	0	0
Zhicheng	0	0	0	0
Yichang	0	0	0	0
Chongqing	558,000	493,000	65,000	428,000
Total	**52,805,000**	**36,442,000**	**16,363,000**	**20,617,000**

Source: 1993 Statistical Yearbook of the Ministry of Communications.

Table 15 1993 Iron Ore Traffic At Major Yangtze Ports

unit: tons

Port name	Throughput	Inbound	Outbound	Local use
Ningbo	21,740,000	10,870,000	10,870,000	0
Shanghai	21,658,000	16,898,000	4,760,000	12,138,000
Nantong	4,104,000	2,052,000	2,052,000	0
Zhangjiagang	68,000	46,000	22,000	24,000
Zhenjiang	4,644,000	2,322,000	2,322,000	0
Nanjing	4,600,000	3,704,000	896,000	2,808,000
Ma'anshan	1,184,000	1,184,000	0	1,184,000
Wuhu	128,000	103,000	25,000	78,000
Tongling	243,000	155,000	88,000	67,000
Chizhou	35,000	0	35,000	n/a
Anqing	261,000	0	261,000	n/a
Jiujiang	359,000	359,000	0	359,000
Huangshi	369,000	0	369,000	n/a
Wuhan	6,100,000	6,100,000	0	6,100,000
Chenglingji	175,000	175,000	0	175,000
Shashi	0	0	0	0
Zhicheng	0	0	0	0
Yichang	0	0	0	0
Chongqing	753,000	686,000	67,000	619,000
Total	**66,421,000**	**44,654,000**	**21,767,000**	**23,552,000**

Source: 1994 Statistical Yearbook of the Ministry of Communications.

Table 16 1994 Iron Ore Traffic At Major Yangtze Ports

unit: tons

Port name	**Throughput**	**Inbound**	**Outbound**	**Local use**
Ningbo	30,432,000	15,216,000	15,216,000	0
Shanghai	23,307,000	18,365,000	4,942,000	13,423,000
Nantong	4,866,000	2,433,000	2,433,000	0
Zhangjiagang	35,000	0	35,000	n/a
Zhenjiang	5,244,000	2,622,000	2,622,000	0
Nanjing	5,419,000	4,054,000	1,365,000	2,689,000
Ma'anshan	2,040,000	2,040,000	0	2,040,000
Wuhu	143,000	124,000	19,000	105,000
Tongling	142,000	0	142,000	n/a
Chizhou	41,000	0	41,000	n/a
Anqing	254,000	0	254,000	n/a
Jiujiang	364,000	364,000	0	364,000
Huangshi	210,000	15,000	195,000	n/a
Wuhan	6,635,000	6,635,000	0	6,635,000
Chenglingji	0	0	0	0
Shashi	0	0	0	0
Zhicheng	0	0	0	0
Yichang	0	0	0	0
Chongqing	822,000	772,000	50,000	722,000
Total	**79,954,000**	**52,640,000**	**27,314,000**	**25,978,000**

Source: 1995 Statistical Yearbook of the Ministry of Communications.

The total consumption of iron ore increased 77.2% between 1990 and 1994. Major increases were in Shanghai (121.6%), Ma'anshan (143.1%), and Wuhan (39.9%). The 77.2% consumption increases closely reflect the fact that steel mill productions along the Yangtze River highly depended on the foreign iron ore imports. In fact, there were little downstream iron ore transport along the Yangtze River as a result of such heavy independence of foreign supply. In terms of local production of iron ore, as shown in Tables 12-16, domestic iron ore transport along the Yangtze were at minimum levels, only around 5% of the foreign amount, during these five years. In 1990, for example, there were only 248.4 thousand tons of iron ore generated locally, and in 1994, 665.0 thousand tons.

The iron ore transshipment patterns were also changed between 1990 and 1994, as shown in these tables. Transshipment locations did not change much between 1990 and 1994. In 1994, major transshipment ports were Ningbo, Shanghai, Nantong, Zhenjiang, and Nanjing. In 1990, the largest percentage of transshipment occurred in Shanghai, but in 1994, it was in Ningbo. As presented before, this pattern change increased total transshipment amount by about 30% due to the increase of double and triple transshipment. The increase in transshipment volume increases transit time and cost for transporting iron ore. This is exactly what the advantages are of using the proposed ORATB system. By using the ORATB system, iron ore will be transshipped only once between the coastal arrival port at Ningbo and consumption places in Shanghai, Nanjing, Ma'anshan, and Wuhan. There will be no transshipment necessary in Shanghai, Nantong, Zhenjiang, and Nanjing. As a result, these direct transshipment systems will decrease both the transit time and cost dramatically.

The analysis of container traffic shows many similarities to the conclusions found in iron ore traffic analysis. As shown in Tables 17-21 and Figure 14, container throughput increased dramatically from 584,630 teus in 1990 to 1,637,216 teus in 1994. More importantly, almost all the ports in the region experienced a sharp increase in container transport. Shanghai had been leading the trade by representing 75% of the total traffic.

Table 17 1990 Container Transport At Major Yangtze Ports

unit: teus

Port name	Throughput	Inbound	Outbound
Ningbo	22,100	10,993	11,107
Shanghai	456,129	224,298	231,831
Nantong	10,915	5,051	5,864
Zhangjiagang	48,102	23,609	24,493
Zhenjiang	537	354	183
Nanjing	42,256	20,741	21,515
Ma'anshan	0	0	0
Wuhu	1,394	613	781
Tongling	0	0	0
Chizhou	0	0	0
Anqing	50	33	17
Jiujiang	103	0	103
Huangshi	0	0	0
Wuhan	3,020	1,414	1,606
Chenglingji	0	0	0
Shashi	24	0	24
Zhicheng	0	0	0
Yichang	0	0	0
Chongqing	0	0	0
Total	**584,630**	**287,106**	**297,524**

Source: 1991 Statistical Yearbook of the Ministry of Communications.

Table 18 1991 Container Transport At Major Yangtze Ports

unit: teus

Port name	Throughput	Inbound	Outbound
Ningbo	35,487	18,079	17,408
Shanghai	575,642	280,740	294,902
Nantong	20,009	8,791	11,218
Zhangjiagang	60,375	29,880	30,495
Zhenjiang	1,801	874	927
Nanjing	50,649	25,864	24,785
Ma'anshan	0	0	0
Wuhu	1,185	563	622
Tongling	0	0	0
Chizhou	0	0	0
Anqing	25	13	12
Jiujiang	284	145	139
Huangshi	0	0	0
Wuhan	4,986	2,198	2,788
Chenglingji	0	0	0
Shashi	202	163	39
Zhicheng	0	0	0
Yichang	0	0	0
Chongqing	0	0	0
Total	**750,645**	**367,310**	**383,335**

Source: 1992 Statistical Yearbook of the Ministry of Communications.

Table 19 1992 Container Transport At Major Yangtze Ports

unit: teus

Port name	Throughput	Inbound	Outbound
Ningbo	53,250	26,967	26,283
Shanghai	729,126	338,728	390,398
Nantong	30,046	11,671	18,375
Zhangjiagang	67,017	32,244	34,773
Zhenjiang	2,692	1,055	1,637
Nanjing	72,146	36,288	35,858
Ma'anshan	0	0	0
Wuhu	1,662	808	854
Tongling	0	0	0
Chizhou	0	0	0
Anqing	44	21	23
Jiujiang	602	318	284
Huangshi	345	173	172
Wuhan	6,205	3,184	3,021
Chenglingji	0	0	0
Shashi	228	93	135
Zhicheng	0	0	0
Yichang	260	151	109
Chongqing	0	0	0
Total	**936,623**	**451,701**	**511,922**

Source: 1993 Statistical Yearbook of the Ministry of Communications.

Table 20 1993 Container Transport At Major Yangtze Ports

unit: teus

Port name	Throughput	Inbound	Outbound
Ningbo	78,847	39,485	39,362
Shanghai	932,808	447,309	485,499
Nantong	45,534	18,936	25,598
Zhangjiagang	80,344	38,060	42,284
Zhenjiang	10,588	4,900	5,688
Nanjing	104,927	51,535	53,392
Ma'anshan	0	0	0
Wuhu	1,477	629	848
Tongling	0	0	0
Chizhou	0	0	0
Anqing	186	66	120
Jiujiang	2,198	1,180	1,018
Huangshi	538	270	268
Wuhan	14,120	7,715	6,405
Chenglingji	0	0	0
Shashi	179	96	83
Zhicheng	0	0	0
Yichang	58	22	36
Chongqing	0	0	0
Total	**1,271,804**	**610,203**	**660,601**

Source: 1994 Statistical Yearbook of the Ministry of Communications.

Table 21 1994 Container Transport At Major Yangtze Ports

unit: teus

Port name	Throughput	Inbound	Outbound
Ningbo	125,135	60,174	64,961
Shanghai	1,193,112	552,488	640,624
Nantong	65,899	23,378	42,521
Zhangjiagang	94,621	44,518	50,103
Zhenjiang	12,458	5,027	7,431
Nanjing	123,712	58,790	64,922
Ma'anshan	0	0	0
Wuhu	1,335	738	597
Tongling	0	0	0
Chizhou	0	0	0
Anqing	0	0	0
Jiujiang	1,714	813	901
Huangshi	279	131	148
Wuhan	16,343	8,286	8,057
Chenglingji	0	0	0
Shashi	208	208	0
Zhicheng	169	0	169
Yichang	0	0	0
Chongqing	2,231	1,255	976
Total	**1,637,216**	**755,806**	**881,410**

Source: 1995 Statistical Yearbook of the Ministry of Communications.

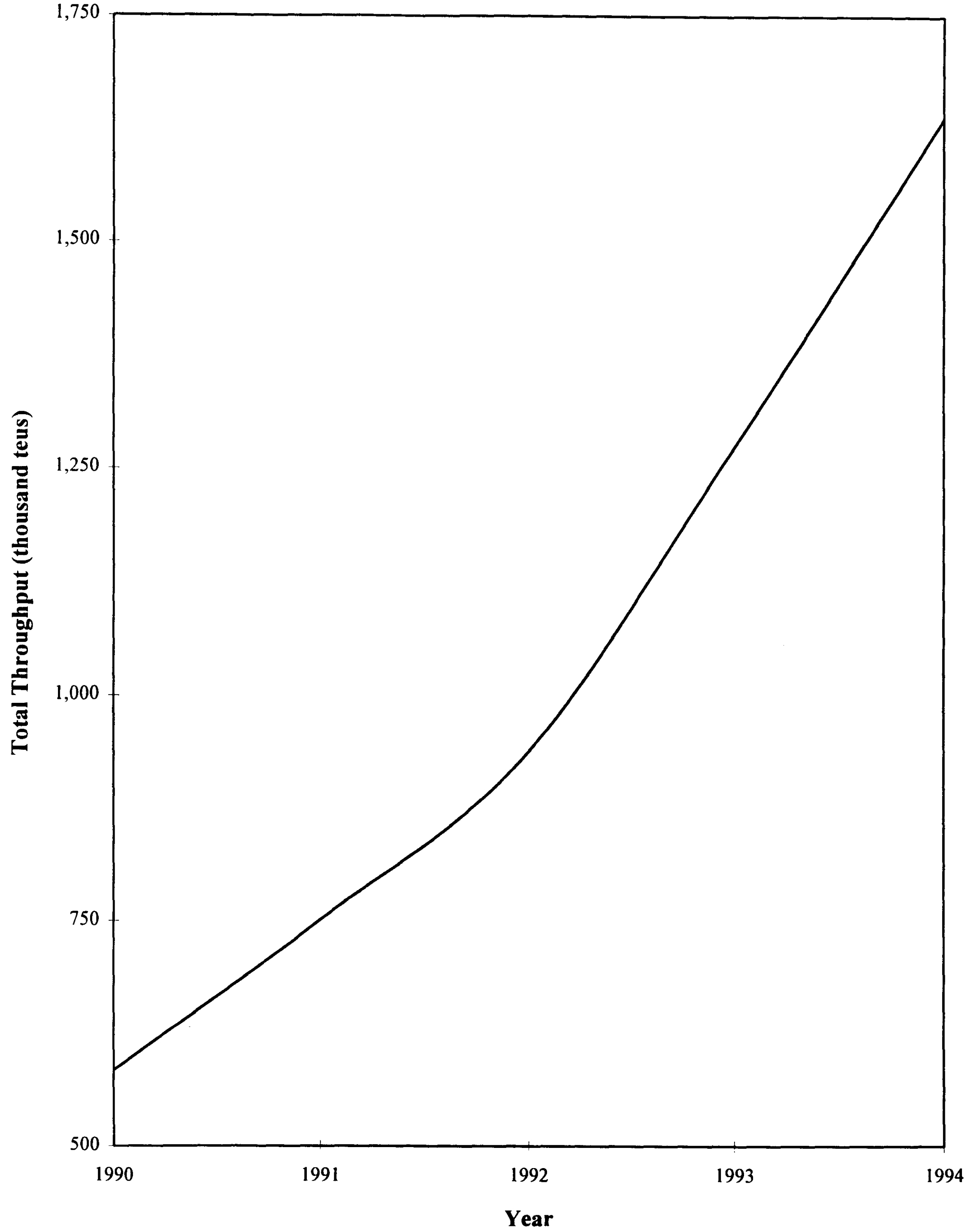


Figure 14 Container Throughput Changes Between 1990 And 1994 At All Major Yangtze Ports

Source: Statistical Yearbooks of the Ministry of Communications, 1991-1995.

Other major container ports were Ningbo, Nantong, Zhangjiagang, Nanjing, Wuhu, and Wuhan. Also, more containers went upstream in 1994 than in 1990. In 1990, for example, the most upstream traffic was only up to Wuhan and amounted only 3,020 teus. In 1994, container traffic went all the way up to Chongqing with an amount of 2,231 teus. For the same year, Wuhan handled 16,343 teus, which is more than five times the throughput in 1990.

As presented in Tables 17-21 above, the throughputs at most ports were evenly distributed between inbound and outbound traffic. This reflects the nature of container transport in that the number of containers going out of a port must be fed somehow by an equal number of containers going into the port. The relatively balanced traffic between inbound and outbound movement implies that container traffic along the Yangtze River was rather self-integrated. Waterway container transport was not making contributions to railway and/or roadway transport nor depending on them with respect to empty container relocation.

To identify some more meaningful O-D patterns about the traffic, we need to look into the detail of the statistics. Tables 22-26 present container inbound traffic at major Yangtze ports during 1990 and 1994, and Tables 27-31 present the outbound traffic. Based on these tables, we calculate the empty/heavy ratios[21] for both inbound and outbound traffic, as listed in Tables 32 and 33. The container transshipment patterns and locations were not changed between 1990 and 1994. Most of the empty containers were transshipped through Shanghai first, and then through Nantong, Nanjing, and Wuhan for the second of time. In terms of heavy containers, most of the transshipment was undertaken by Shanghai. Ningbo basically acted alone for both of its inbound and outbound container transport. As we pointed before, such a transshipment pattern increases transit time as well as cost for transporting containers. This is exactly what the advantages of using the proposed ORATB system. By using the ORATB system,

---

[21] Number of empty teus divided by number of heavy teus.

Table 22 1990 Container Inbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	10,993	1,943	4,173	865	1,204	28,880
Shanghai	224,298	16,468	48,136	37,806	67,614	1,608,981
Nantong	5,051	734	2,524	314	431	9,541
Zhangjiagang	23,609	2,701	10,194	1,543	4,927	131,969
Zhenjiang	354	31	214	0	78	1,540
Nanjing	20,741	3,140	6,802	946	5,204	113,213
Ma'anshan	0	0	0	0	0	0
Wuhu	613	48	517	0	0	0
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	33	0	33	0	0	0
Jiujiang	0	0	0	0	0	0
Huangshi	0	0	0	0	0	0
Wuhan	1,414	298	550	70	128	2,072
Chenglingji	0	0	0	0	0	0
Shashi	0	0	0	0	0	0
Zhicheng	0	0	0	0	0	0
Yichang	0	0	0	0	0	0
Chongqing	0	0	0	0	0	0
Total	**287,106**	**25,363**	**73,143**	**41,544**	**79,586**	**1,896,196**

Source: 1991 Statistical Yearbook of the Ministry of Communications.

Table 23 1991 Container Inbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	18,079	3,037	6,649	1,662	2,032	56,000
Shanghai	280,740	19,773	41,929	44,293	110,679	2,418,000
Nantong	8,791	1,404	3,644	867	605	22,000
Zhangjiagang	29,880	4,718	12,300	1,371	5,402	114,000
Zhenjiang	874	176	484	7	24	n/a
Nanjing	25,864	4,180	7,242	2,263	5,736	135,000
Ma'anshan	0	0	0	0	0	0
Wuhu	563	47	469	0	0	0
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	13	0	3	5	0	n/a
Jiujiang	145	3	135	0	4	n/a
Huangshi	0	0	0	0	0	0
Wuhan	2,198	365	1,083	66	253	4,000
Chenglingji	0	0	0	0	0	0
Shashi	163	0	156	0	7	n/a
Zhicheng	0	0	0	0	0	0
Yichang	0	0	0	0	0	0
Chongqing	0	0	0	0	0	0
Total	**367,310**	**33,703**	**74,094**	**50,534**	**124,742**	**2,749,000**

Source: 1992 Statistical Yearbook of the Ministry of Communications.

Table 24 1992 Container Inbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40’ box	20’ box	40’ box	20’ box	
Ningbo	26,967	4,559	9,194	2,778	3,099	87,081
Shanghai	338,728	31,282	48,836	52,918	121,492	2,633,262
Nantong	11,671	1,404	4,020	1,751	1,341	41,203
Zhangjiagang	32,244	5,855	11,362	2,006	5,160	101,704
Zhenjiang	1,055	123	582	50	127	1,614
Nanjing	36,288	6,675	8,796	3,900	6,342	158,193
Ma’anshan	0	0	0	0	0	0
Wuhu	808	121	558	3	2	125
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	21	1	15	1	2	18
Jiujiang	318	12	203	45	1	308
Huangshi	173	0	46	53	21	844
Wuhan	3,184	527	954	531	114	7,400
Chenglingji	0	0	0	0	0	0
Shashi	93	7	66	6	1	50
Zhicheng	0	0	0	0	0	0
Yichang	151	0	0	69	13	673
Chongqing	0	0	0	0	0	0
Total	**451,701**	**50,566**	**84,632**	**64,111**	**137,715**	**3,032,475**

Source: 1993 Statistical Yearbook of the Ministry of Communications.

Table 25 1993 Container Inbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	39,485	6,657	10,846	4,502	6,321	141,589
Shanghai	447,309	49,597	81,280	70,547	125,741	2,762,047
Nantong	18,936	2,171	6,661	2,756	2,421	69,076
Zhangjiagang	38,060	6,092	12,602	3,559	6,156	125,547
Zhenjiang	4,900	977	1,988	288	382	7,481
Nanjing	51,535	8,699	13,348	5,319	10,151	215,885
Ma'anshan	0	0	0	0	0	0
Wuhu	629	68	296	66	65	1,161
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	66	0	26	0	40	154
Jiujiang	1,180	11	434	336	52	2,738
Huangshi	270	5	0	92	76	2,129
Wuhan	7,715	192	1,364	2,802	363	34,706
Chenglingji	0	0	0	0	0	0
Shashi	96	10	68	4	0	41
Zhicheng	0	0	0	0	0	0
Yichang	22	1	0	2	16	131
Chongqing	0	0	0	0	0	0
Total	**610,203**	**74,480**	**128,913**	**90,273**	**151,784**	**3,362,685**

Source: 1994 Statistical Yearbook of the Ministry of Communications.

Table 26 1994 Container Inbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40’ box	20’ box	40’ box	20’ box	
Ningbo	60,174	8,341	17,821	8,658	8,355	252,572
Shanghai	552,488	55,384	119,325	91,529	139,337	3,211,090
Nantong	23,378	2,789	8,215	3,128	3,329	97,316
Zhangjiagang	44,518	7,281	14,344	4,109	7,394	162,118
Zhenjiang	5,027	731	2,880	211	263	5,382
Nanjing	58,790	10,092	15,517	4,925	13,239	264,973
Ma’anshan	0	0	0	0	0	0
Wuhu	738	81	339	91	55	2,159
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	0	0	0	0	0	0
Jiujiang	813	22	592	48	81	1,632
Huangshi	131	5	33	37	14	575
Wuhan	8,286	37	1,468	3,146	452	39,235
Chenglingji	0	0	0	0	0	0
Shashi	208	0	184	12	0	216
Zhicheng	0	0	0	0	0	0
Yichang	0	0	0	0	0	0
Chongqing	1,255	101	630	173	77	2,648
Total	**755,806**	**84,864**	**181,348**	**116,067**	**172,596**	**4,039,916**

Source: 1995 Statistical Yearbook of the Ministry of Communications.

Table 27 1990 Container Outbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	11,107	152	202	2,658	5,285	92,944
Shanghai	231,831	7,999	11,011	46,165	112,492	1,888,754
Nantong	5,864	147	211	898	3,563	47,455
Zhangjiagang	24,493	559	1,953	3,903	13,616	255,859
Zhenjiang	183	4	2	8	157	2,885
Nanjing	21,515	240	2,061	3,935	10,104	188,340
Ma'anshan	0	0	0	0	0	0
Wuhu	781	0	0	58	665	11,350
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	17	0	0	0	17	260
Jiujiang	103	0	0	0	103	n/a
Huangshi	0	0	0	0	0	0
Wuhan	1,606	76	129	301	723	14,392
Chenglingji	0	0	0	0	0	0
Shashi	24	2	0	2	16	199
Zhicheng	0	0	0	0	0	0
Yichang	0	0	0	0	0	0
Chongqing	0	0	0	0	0	0
Total	**297,524**	**9,179**	**15,569**	**57,928**	**146,741**	**2,502,438**

Source: 1991 Statistical Yearbook of the Ministry of Communications.

Table 28 1991 Container Outbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	17,408	144	103	4,327	8,363	150,000
Shanghai	294,902	4,379	14,998	63,798	143,550	2,468,000
Nantong	11,218	583	2,096	1,657	4,642	67,000
Zhangjiagang	30,495	363	1,340	5,980	16,469	302,000
Zhenjiang	927	0	1	159	608	10,000
Nanjing	24,785	458	1,470	5,623	11,153	218,000
Ma'anshan	0	0	0	0	0	0
Wuhu	622	6	0	51	508	8,000
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	12	3	0	0	6	n/a
Jiujiang	139	0	0	2	135	2,000
Huangshi	0	0	0	0	0	0
Wuhan	2,788	192	452	423	1,106	24,000
Chenglingji	0	0	0	0	0	0
Shashi	39	0	0	6	27	1,000
Zhicheng	0	0	0	0	0	0
Yichang	0	0	0	0	0	0
Chongqing	0	0	0	0	0	0
Total	**383,335**	**6,128**	**20,460**	**82,026**	**186,567**	**3,250,000**

Source: 1992 Statistical Yearbook of the Ministry of Communications.

Table 29 1992 Container Outbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	26,283	288	44	6,959	11,745	214,322
Shanghai	390,398	11,823	23,730	87,241	168,540	2,988,955
Nantong	18,375	2,949	1,179	2,809	5,680	94,752
Zhangjiagang	34,773	563	498	8,313	16,523	307,127
Zhenjiang	1,637	26	50	280	975	16,810
Nanjing	35,858	1,777	1,416	8,877	13,134	277,606
Ma'anshan	0	0	0	0	0	0
Wuhu	854	1	0	119	614	8,094
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	23	4	0	0	15	159
Jiujiang	284	33	2	10	196	2,273
Huangshi	172	54	21	0	43	453
Wuhan	3,021	241	291	641	966	27,902
Chenglingji	0	0	0	0	0	0
Shashi	135	6	6	13	91	1,861
Zhicheng	0	0	0	0	0	0
Yichang	109	49	11	0	0	0
Chongqing	0	0	0	0	0	0
Total	**511,922**	**17,814**	**27,248**	**115,262**	**218,522**	**3,940,314**

Source: 1993 Statistical Yearbook of the Ministry of Communications.

Table 30 1993 Container Outbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	39,362	333	55	10,681	17,279	310,623
Shanghai	485,499	6,329	12,830	122,989	214,033	3,762,269
Nantong	25,598	2,185	2,780	4,596	9,256	162,537
Zhangjiagang	42,284	350	1,036	10,527	19,494	357,745
Zhenjiang	5,688	708	979	639	2,015	37,053
Nanjing	53,392	1,167	2,896	13,665	20,832	412,757
Ma'anshan	0	0	0	0	0	0
Wuhu	848	42	32	140	452	8,721
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	120	0	59	0	61	491
Jiujiang	1,018	225	9	32	495	9,838
Huangshi	268	85	43	9	37	350
Wuhan	6,405	1,506	38	786	1,783	42,114
Chenglingji	0	0	0	0	0	0
Shashi	83	3	3	10	54	1,189
Zhicheng	0	0	0	0	0	0
Yichang	36	10	16	0	0	0
Chongqing	0	0	0	0	0	0
Total	**660,601**	**12,943**	**20,776**	**164,074**	**285,791**	**5,105,687**

Source: 1994 Statistical Yearbook of the Ministry of Communications.

Table 31 1994 Container Outbound Traffic At Major Yangtze Ports

Port name	All containers					Cargo weight (ton)
	Sum (teu)	Empty container		Heavy container		
		40' box	20' box	40' box	20' box	
Ningbo	64,961	533	194	17,942	27,817	508,063
Shanghai	640,624	12,305	15,989	162,811	274,403	4,873,720
Nantong	42,521	5,954	6,640	5,625	12,723	208,209
Zhangjiagang	50,103	932	2,326	11,440	23,033	419,304
Zhenjiang	7,431	126	254	990	4,945	90,036
Nanjing	64,922	1,173	2,851	15,889	27,947	524,708
Ma'anshan	0	0	0	0	0	0
Wuhu	597	1	3	107	378	6,946
Tongling	0	0	0	0	0	0
Chizhou	0	0	0	0	0	0
Anqing	0	0	0	0	0	0
Jiujiang	901	71	11	25	698	11,929
Huangshi	148	36	11	8	49	945
Wuhan	8,057	2,338	36	834	1,677	38,244
Chenglingji	0	0	0	0	0	0
Shashi	0	0	0	0	0	0
Zhicheng	169	0	0	1	167	2,994
Yichang	0	0	0	0	0	0
Chongqing	976	77	28	59	676	12,739
Total	**881,410**	**23,546**	**28,343**	**215,731**	**374,513**	**6,697,837**

Source: 1995 Statistical Yearbook of the Ministry of Communications.

Table 32 1990-94 Inbound Container Traffic Empty/Heavy Ratios

Port name	**Empty/Heavy Ratio**				
	1990	**1991**	**1992**	**1993**	**1994**
Ningbo	2.75	2.38	2.16	1.58	1.34
Shanghai	0.57	0.41	0.49	0.68	0.71
Nantong	3.77	2.76	1.41	1.39	1.44
Zhangjiagang	1.95	2.67	2.52	1.88	1.85
Zhenjiang	3.54	22.00	3.65	4.11	6.34
Nanjing	1.84	1.52	1.57	1.48	1.55
Ma'anshan	n/a	n/a	n/a	n/a	n/a
Wuhu	∞	∞	100.00	2.19	2.11
Tongling	n/a	n/a	n/a	n/a	n/a
Chizhou	n/a	n/a	n/a	n/a	n/a
Anqing	∞	0.30	4.25	0.65	n/a
Jiujiang	n/a	35.30	2.49	0.63	3.59
Huangshi	n/a	n/a	0.36	0.04	0.49
Wuhan	4.28	4.71	1.71	0.29	0.23
Chenglingji	n/a	n/a	n/a	n/a	n/a
Shashi	n/a	22.30	6.15	11.00	7.67
Zhicheng	n/a	n/a	n/a	n/a	n/a
Yichang	n/a	n/a	0.00	0.10	n/a
Chongqing	n/a	n/a	n/a	n/a	1.97

Table 33 1990-94 Outbound Container Traffic Empty/Heavy Ratios

Port name	Empty/Heavy Ratio				
	1990	1991	1992	1993	1994
Ningbo	0.05	0.02	0.02	0.02	0.02
Shanghai	0.13	0.09	0.14	0.06	0.07
Nantong	0.09	0.41	0.63	0.39	0.77
Zhangjiagang	0.14	0.07	0.05	0.04	0.09
Zhenjiang	0.06	0.00	0.07	0.73	0.07
Nanjing	0.14	0.11	0.16	0.11	0.08
Ma'anshan	n/a	n/a	n/a	n/a	n/a
Wuhu	0.00	0.02	0.00	0.16	0.01
Tongling	n/a	n/a	n/a	n/a	n/a
Chizhou	n/a	n/a	n/a	n/a	n/a
Anqing	n/a	n/a	n/a	0.97	n/a
Jiujiang	n/a	0.00	0.31	0.82	0.20
Huangshi	n/a	n/a	3.00	3.87	1.28
Wuhan	0.21	0.51	0.34	0.91	1.41
Chenglingji	n/a	n/a	n/a	n/a	n/a
Shashi	0.20	n/a	0.15	0.12	n/a
Zhicheng	n/a	n/a	n/a	n/a	0.00
Yichang	n/a	n/a	∞	∞	n/a
Chongqing	n/a	n/a	n/a	n/a	0.23

containers, at least the empties will be transshipped only once between the coastal arrival port at Ningbo and demanding places in Nantong, Nanjing, and Wuhan. There will be no empty container transshipment necessary in Shanghai, Nantong, Zhenjiang, and Nanjing. As a result, these direct transshipment of containers will decrease both of their transit time and cost dramatically.

4.1.2. Shipping Management In The Yangtze Valley

Currently, shipping and its related activities along the Yangtze is mainly managed by the state owned enterprise, China Changjiang[22] National Shipping Group (CCNSG). The CCNSG is a transregional, multilevel and diversified enterprise based on transportation under the Ministry of Communications. Headquartered in Wuhan, the CCNSG takes China Changjiang National Shipping Corporation as its core member. As shown in Figure 15, it owns 20 direct subordinates and two indirect subordinates, as well as many cooperative members. On freight services, the CCNSG is responsible for the state planned transport along the main stream branches of the Yangtze River, and some coastal shipping routes. In 1990, the freight volume and freight turnover volume represented 51% and 63.2% of the total made by all the shipping companies along the main stream branches of the Yangtze River, respectively.

The CCNSG owns more than 2,000 various ships with annual freight capacity over 80 million tons. The ships include barges ranging from 1,000 to 5,000 dwt, tugboats powered from 1,492 to 4,413 kilowatts (kw), oil tanks at the level of 30,000 dwt, cargo vessels from 5,000 to 40,000 dwt, container vessels, and Ro/Ro carriers. However, the upper classes of these ships are deployed in shipping routes between coastal ports and the very downstream Yangtze ports. The shipping routes from Chongqing to Shanghai are all operated by much smaller vessels. In terms of cargoes transported, the CCNSG mainly engages in the transport of raw coal, crude and product oil, iron ore, rolled steel,

---

[22] The Yangtze River in Chinese spelling.

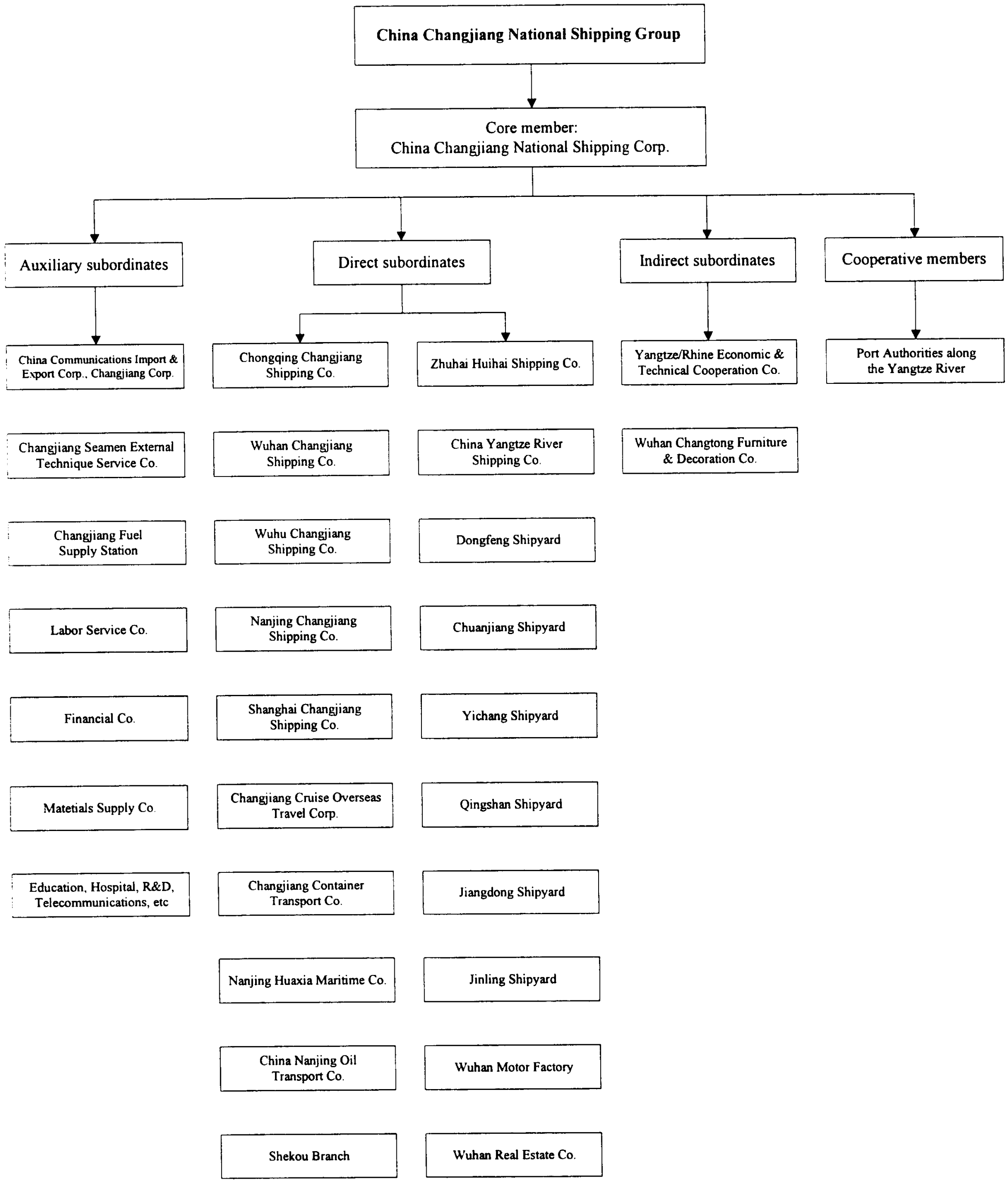


Figure 15 Organization Structure Of The CCNSG

Source: China Changjiang National Shipping Group Annual Report 1994.

and building materials, serving for along shore power plants, petrochemical refineries, and iron and steel factories. The CCNSG has a very limited engagement in moving containers along the Yangtze River. Such segregation in transporting bulk and container cargoes poses a severe management problem for the proposed integrated transport of bulks and containers by the ORATB system.

Although it was established for cargo transport along the Yangtze River, the CCNSG is vigorously developing coastal and near ocean shipping. At present, it has seven shipping companies, 23 coastal cargo ships, and 13 coastal oil tankers, with a total deadweight tonnage of 372,000 dwt on coastal line services. Such coastal services include iron ore carriage from Ningbo to the ports of the Yangtze River, and crude and product oil and raw coal from other sea ports to river. It also operates foreign trade transport from the Yangtze River ports to ports in Japan, North Korea, South Korea, Hong Kong, and southeast Asia. However, these coastal and near ocean engagements are not aimed at eliminating the redundant transshipments in the lower Yangtze River.

With respect to port operations, local port authorities are responsible for the daily businesses, while the CCNSG controls state planned traffic. In other words, the CCNSG decides which port to call for the traffic of most raw materials along the Yangtze River. Needless to say, such decision making is mainly based on locations of the origin and the destination of the traffic, as well as port and berth conditions. Table 34 shows primary berth conditions of the major Yangtze River ports, including the ones in Ningbo and Shanghai for convenience.

4.1.3. Barriers For Future Shipping Development In The Yangtze Valley

As more transport demands in iron ore and containers are expected along the Yangtze River, future growth in transport services are highly hampered by the natural conditions, the current shipping practices, and its management. One of the major difficulties that are

Table 34 Berth And Yard Conditions Of Major Yangtze Ports

Port name	Berth length	No. of berths	No. of berths for 10,000 dwt	Storage capacity (tons)	Container storage capacity (teus)
Ningbo	5,075	37	17	3,523,768	7,836
Shanghai	18,285	136	67	2,761498	30,988
Nantong	3,023	31	7	712,853	n/a
Zhangjiagang	2,332	13	10	992,119	4,896
Zhenjiang	3,075	30	8	597,699	n/a
Nanjing	4,872	49	14	2,277,472	8,035
Ma'anshan	746	14	0	886,112	n/a
Wuhu	1,734	33	0	277,819	n/a
Tongling	481	8	0	213,068	n/a
Chizhou	570	9	0	196,860	n/a
Anqing	727	13	0	85,076	n/a
Jiujiang	861	12	0	150,891	n/a
Huangshi	869	13	0	133,293	n/a
Wuhan	3,288	46	0	812,395	240
Chenglingji	604	9	0	113,478	n/a
Shashi	659	15	0	55,584	1,000
Zhicheng	943	11	0	220,674	n/a
Yichang	828	16	0	128,830	n/a
Chongqing	2,441	46	0	474,205	n/a

Source: 1996 Statistical Yearbook of the Ministry of Communications.

shared among almost all the Yangtze ports is the river's shallow water. As shown in Table 35, the navigation channels of the Yangtze River have a minimum depth of 4.0 meters and an average depth of 5.5 meters in the lower reaches of the Yangtze up to Wuhan. Between April and September, water depths between Nanjing and Wuhan actually exceed 5.9 meters or 20 feet. Vessels of up to 5,000 dwt can sail up to Wuhan year round and vessels with special designs of up to 25,000 dwt can sail to Wuhan during the two-month flooding period if they are not exceeding the height restrictions of passing through the bridges.

The second major impediment to traffic is that many bridges are too low to permit ocean-going vessels to glide under. The first of these obstacles west of Shanghai is the Nanjing Bridge. Ocean-going vessels have to be redesigned with lower mast to pass under the bridge. The limitations of shallow water depths and low height clearances work in a cooperative way to prevent large vessels going up river. During the dry season, although height limitation becomes less restrictive, water depth is at its lowest. On the other hand, during the flood season, height limitation becomes more restrictive when water depth restriction is released. This results in a vast amount of cargo having to be feedered to ports upstream of Nanjing along the Yangtze River. As we showed before, all secondary transshipment was in the east of Nanjing.

The third barrier to the rapid future development is the unfavorable natural and social conditions in the Port of Shanghai which is located at the east end of the Yangtze River. The natural conditions at the Port of Shanghai are unfavorable because of its too limited water depth and too congested traffic. The maximum water depth is only -9.5 m which is by no means capable of accommodating large ocean-going vessels. In fact, the largest bulk vessels calling at Shanghai is on the order of 30,000 dwts and 2,700 teus for container vessels. In contrast, the Port of Ningbo can accommodate bulk vessels up to 200,000 dwts and container vessels up to 4,500 teus. With respect to traffic conditions, Shanghai is also much worse than Ningbo. The Port of Shanghai is basically spread

Table 35 Conditions Of Navigation Channels Along The Yangtze River

Origin to Destination	Length (km)	Depth (m)	Width (m)	Curvature radius (m)	Max. ship (dwt)	Notes
Upper reaches of the Yangtze						
Qizong to Xiluodu	643	n/a	n/a	n/a	< 30, seasonal	Total length is 1,258 km, but not continuously navigable.
Xiluodu to Xinshizhen	76	1.5	40	127	< 80, seasonal	Below are continuously navigable, but still seasonal.
Xinshizhen to Yibin	107	1.8	40	200	< 200 year round	Below are navigable year round.
Yibin to Lanjiatuo	303	1.8	40	200	< 800 year round	
Lanjiatuo to Chongqing	81	2.5	50	560	< 800 year round	
Chongqing to Yichang	660	2.9	60	750	< 1,500 year round	3,000 dwt is allowable seasonally
Middle reaches of the Yangtze						
Yichang to Linxiang	416	2.9	80	750	< 1,500 year round	
Linxiang to Wuhan	210	3.5	80	1,000	< 3,000 year round	10,000 dwt is seasonally allowable.
Lower reaches of the Yangtze						
Wuhan to Jiujiang	269	4.0	100	1,000	< 5,000 year round	30,000 dwt is seasonally allowable.
Jiujiang to Anqing	164	4.0	100	1,000	< 5,000 year round	
Anqing to Wusong	337	7.1	200	n/a	25,000 with tide	
Wusong to Coast	100	n/a	n/a	n/a	25,000 with tide	This section is for ocean going vessels.
Total	**3,609**					

Source: 1994 Statistical Yearbook of the Ministry of Communications.

along the Huangpu River, a narrow tributary of the Yangtze River. Moreover, traffic generated by smaller vessels contributed about 80% of the total traffic. On the other hand, Ningbo is located along an open coastal line with naturally deep navigation channels up to -22 m.

The social conditions are also not favorable for Shanghai to become the transshipment center for the Yangtze Valley region. Recent rapid development in urbanization posed a huge threat for any expansion of the existing port facilities mainly because of the sky rocketing real estate prices.

Should these conditions become favorable, the Port of Shanghai becomes the natural selection for being the transshipment center for cargoes going in and out the Yangtze River. However, it is not cost effective to improve the shallow water depth in the Port of Shanghai, because the vast amount of silt keeps coming from the Yangtze River. Needless to say it would be more difficult to overcome the negative aspects of the social and economic conditions.

4.2. Application Of The Proposed IAIOS Model

Based on the above analysis, we therefore propose to introduce the ORATB for the Yangtze Valley. As a matter of fact, iron ore and container barge services are becoming more and more important in river transport in the United States and Europe. It is estimated that nearly 27% of inland moves, or 475,000 containers, were moved by container barges in 1994 in Europe. According to the latest forecasts, the number will rise to some 1.2 million boxes by the year of 2010[23].

In the proposed ORATB transport system, Ningbo shall be the transshipment center for both iron ore and container transport for the Yangtze region, based on the shipping

---

[23] World Cargo News, January 1995.

demand O-D analysis presented before. Major demand nodes for iron ore transport shall include Shanghai, Ma'anshan, and Wuhan. On the other hand, major demand nodes for container transport shall be Shanghai, Nantong, Nanjing, and Wuhan.

The available navigation channel depths and width should permit the use of the ORATB system up to 20,000 dwt all the way up to Wuhan. Barges may have to sail with partial loads (say 12-15,000 tons) for two or three months during the dry season. As noted in Table 35, ports in the lower reaches of the Yangtze up to Nanjing all have berths for 10,000 dwt ocean-going vessels, while up to Wuhan all have berths for 5,000 dwt vessels. These berth conditions are generally suitable to handle 10-20,000 dwt barges with minor upgrade. The ORATB system though is not permissible beyond the lower reaches of the Yangtze River due to the channel conditions as shown in Table 35.

We devote the following sections to the application of the proposed IAIOS modeling approach to introduce the proposed ORATB transport system to the Yangtze Valley.

4.2.1. Optimization Model Application: Results And Evaluation

The objective of this mixed integer model is to minimize the total cost, *i.e.*, the *TAAC*, of the ORATB transport system. The *TAAC* consists of all the investment and operating costs of the tugs and barges, as well as the berthing and loading and unloading facilities required by the ORATB transport. The decision variables are the number of tugs and barges in different sizes. The constraints are to meet all transport demands on time at all locations, and the interdependent relations between the tugs and barges.

The GAMS input statements are as shown in Table 4. The output summary report for this ORATB application is given in Table 36. As can be seen from the output statistics, the optimum value of *TAAC* is \$3,483.6 million. In this amount, \$1,800.0 million (or 51.7% of the *TAAC*) is for the berthing facilities at Ningbo, and \$540.0 million (or 15.5%) for

Table 36 The ORATB Optimization Model Output Summary Report

```
************************************************************************
GAMS 2.25.059  386/486 DOS                              12/03/96 03:13:46
```

General Algebraic Modeling System Compilation

```
**************************************************
**      ORATB Transport Optimization Model      **
**               M. Qi  1996  MIT               **
**************************************************

SETS
  I origination nodes   /O1/
  K destination nodes   /D1,D2,D3,D4,D5/
  T time periods        /T1,T2,T3,T4,T5,T6,T7,T8,T9,T10,T11,T12/
  L barge sizes (kdwt)  /BG1,BG2,BG3/
  S tug types (kkw)     /TG1,TG2,TG3/;
TABLE
  BKDM(I,K,T)      iron ore demand at k from i for period t (kton)
             T1  T2  T3  T4  T5  T6  T7  T8  T9  T10  T11  T12
      O1.D1 14  14  15  16  15  16  15  14  13  14   15   15
      O1.D2 4   5   5   6   6   5   4   3   4   6    6    5
      O1.D3 5   5   4   4   3   6   5   4   4   5    6    6
      O1.D4 4   4   5   4   6   5   5   5   4   5    6    5
      O1.D5 9   9   10  11  10  10   9   9   9   10   11  11;

TABLE
  GAMA(S,L)        match-up relations between tug and barge
             BG1   BG2   BG3
      TG1    1     0     0
      TG2    2     1     0
      TG3    4     2     1;

PARAMETERS
  BKBHO(I)   AAC of berthing facility at origination port i ($m)
      /O1    30/
  BKBHD(K)   AAC of berthing facility for l-size barge at
             destination port k ($m)
      /D1    20
       D2    5
       D3    5
       D4    5
       D5    10/

  BKBEO(I)   AAC of berth equipment for l-size barge at origination
             port i ($m)
      /O1    8/
  BKBED(K)   AAC of berth equipment for l-size barge at destination
             port k ($m)
```

Table 36 (continued)

```
      /D1   6
       D2   4
       D3   4
       D4   4
       D5   5/

  TGF(S)    AAC fixed of s-type tug ($m)
      /TG1  4
       TG2  6
       TG3  8/

  TGV(S)    AAC variable of s-type tug ($m)
      /TG1  1.0
       TG2  1.1
       TG3  1.2/

  BGF(L)    AAC fixed of l-size barge ($m)
      /BG1  1.0
       BG2  1.5
       BG3  2.0/

  BGV(L)    AAC variable of l-size barge ($m)
      /BG1  0.1
       BG2  0.2
       BG3  0.3/

  SZ(L)     capacity of the l-size barge
      /BG1  5
       BG2  10
       BG3  15/;

VARIABLES
  Y(I,K,T)          0-1 DV to select routing for period t
  M(I,K,S,T)        0-1 DV to select tug type for period t
  N(I,K,L,T)        0-1 DV to select barge size for period t
  MM(I,K,S,T)       integer DV to determine # of tugs for period t
  NN(I,K,L,T)       integer DV to determine # of barges for period t
  TAAC              total annual average costs
  TAAC11            AAC of berthing facility at origination port i
  TAAC12            AAC of berthing facility at destination port k
  TAAC21            AAC of berth equipment at origination port i
  TAAC22            AAC of berth equipment at destination port k
  TAAC31            AAC of tug related
  TAAC41            AAC of barge related;

INTEGER VARIABLES
  MM(I,K,S,T)       integer DV to determine # of tugs for period t
  NN(I,K,L,T)       integer DV to determine # of barges for period
                    t;

```

Table 36 (continued)

```
90    BINARY VARIABLES
91      Y(I,K,T)         0-1 DV to select routing for period t
92      M(I,K,S,T)       0-1 DV to select tug type for period t
93      N(I,K,L,T)       0-1 DV to select barge size for period t;
94
95    EQUATIONS
96      TAACDEF                as eq.2
97      TAAC11DEF              as eq.2
98      TAAC12DEF              as eq.2
99      TAAC21DEF              as eq.2
100     TAAC22DEF              as eq.2
101     TAAC31DEF              as eq.2
102     TAAC41DEF              as eq.2
103     BKDEMAND(I,K,T)        as eq.4
104     BALANCE1(I,K,T)        as eq.5
105     BALANCE2(I,K,T)        as eq.6
106     BALANCE3(I,K,S,T)      as eq.14
107     BALANCE4(I,K,L,T)      as eq.16;
108*    BALANCE5(I,K,S,L,T)    as eq.17
109*    BALANCE6(I,K,S,T)      as eq.19
110*    BALANCE7(I,K,L,T)      as eq.20;
111
112     TAACDEF..        TAAC =E= TAAC11+TAAC12+TAAC21+TAAC22+TAAC31
                                  +TAAC41;
113     TAAC11DEF..      TAAC11 =E= SUM((I,K),BKBHO(I)*SUM(T,Y(I,K,T)));
114     TAAC12DEF..      TAAC12 =E= SUM((I,K),BKBHD(K)*SUM(T,Y(I,K,T)));
115     TAAC21DEF..      TAAC21 =E= SUM((I,K),BKBEO(I)*SUM(T,Y(I,K,T)));
116     TAAC22DEF..      TAAC22 =E= SUM((I,K),BKBED(K)*SUM(T,Y(I,K,T)));
117     TAAC31DEF..      TAAC31 =E= SUM((I,K,S),(TGF(S)+TGV(S))*SUM(T,M
                                    (I,K,S,T)));
118     TAAC41DEF..      TAAC41 =E= SUM((I,K,L),(BGF(L)+BGV(L))*SUM(T,N
                                    (I,K,L,T)));
119     BKDEMAND(I,K,T)..      SUM(L,N(I,K,L,T)*SZ(L)) =G= BKDM(I,K,T);
120     BALANCE1(I,K,T)..      SUM(S,M(I,K,S,T)) =E= 1;
121     BALANCE2(I,K,T)..      SUM(L,N(I,K,L,T)) =E= 1;
122     BALANCE3(I,K,S,T)..    M(I,K,S,T) =L= Y(I,K,T);
123     BALANCE4(I,K,L,T)..    N(I,K,L,T) =L= Y(I,K,T);
124*    BALANCE5(I,K,S,L,T).. MM(I,K,S,T) =E= GAMA(S,L)*NN(I,K,L,T);
125*    BALANCE6(I,K,S,T)..    MM(I,K,S,T)-100*M(I,K,S,T) =L= 0;
126*    BALANCE7(I,K,L,T)..    NN(I,K,L,T)-100*N(I,K,L,T) =L= 0;
127
128    OPTION ITERLIM = 5000
129
130    OPTION OPTCR = 0.005
131
132    OPTION RESLIM = 10000
133
134    MODEL ORATB /ALL/;
135
136    SOLVE ORATB USING MIP MINIMIZING TAAC;
```

Table 36 (continued)

```
137
138   DISPLAY
139   Y.L, M.L, N.L;

General Algebraic Modeling System Symbol Listing

SYMBOL      TYPE   REFERENCES
BALANCE1    EQU     DECLARED      104  DEFINED      120 IMPL-ASN      136
                         REF      134
BALANCE2    EQU     DECLARED      105  DEFINED      121 IMPL-ASN      136
                         REF      134
BALANCE3    EQU     DECLARED      106  DEFINED      122 IMPL-ASN      136
                         REF      134
BALANCE4    EQU     DECLARED      107  DEFINED      123 IMPL-ASN      136
                         REF      134
BGF         PARAM   DECLARED       57  DEFINED       58      REF      118
BGV         PARAM   DECLARED       62  DEFINED       63      REF      118
BKBED       PARAM   DECLARED       40  DEFINED       41      REF      116
BKBEO       PARAM   DECLARED       38  DEFINED       39      REF      115
BKBHD       PARAM   DECLARED       31  DEFINED       32      REF      114
BKBHO       PARAM   DECLARED       29  DEFINED       30      REF      113
BKDEMAND    EQU     DECLARED      103  DEFINED      119 IMPL-ASN      136
                         REF      134
BKDM        PARAM   DECLARED       13  DEFINED       13      REF      119
GAMA        PARAM   DECLARED       22  DEFINED       22
I           SET     DECLARED        7  DEFINED        7      REF       13
                          29       38       73       74       75       76
                          77       87       88       91       92       93
                         103      104      105      106      107    2*113
                         114    2*115      116      117      118    2*119
                         120      121    2*122    2*123  CONTROL      113
                         114      115      116      117      118      119
                         120      121      122      123
K           SET     DECLARED        8  DEFINED        8      REF       13
                          31       40       73       74       75       76
                          77       87       88       91       92       93
                         103      104      105      106      107      113
                       2*114      115    2*116      117      118    2*119
                         120      121    2*122    2*123  CONTROL      113
                         114      115      116      117      118      119
                         120      121      122      123
L           SET     DECLARED       10  DEFINED       10      REF       22
                          57       62       67       75       77       88
                          93      107    3*118    2*119      121      123
                     CONTROL      118      119      121      123
M           VAR     DECLARED       74       92 IMPL-ASN      136
                         REF      117      120      122      139
MM          VAR     DECLARED       76       87
N           VAR     DECLARED       75       93 IMPL-ASN      136
                         REF      118      119      121      123      139
```

Table 36 (continued)

```
SYMBOL      TYPE   REFERENCES
NN          VAR    DECLARED       77       88
ORATB       MODEL  DECLARED      134  DEFINED      134 IMPL-ASN      136
                        REF      136
S           SET    DECLARED       11  DEFINED       11      REF       22
                         47       52       74       76       87       92
                        106    3*117      120      122  CONTROL      117
                        120      122
SZ          PARAM  DECLARED       67  DEFINED       68      REF      119
T           SET    DECLARED        9  DEFINED        9      REF       13
                         73       74       75       76       77       87
                         88       91       92       93      103      104
                        105      106      107      113      114      115
                        116      117      118    2*119      120      121
                      2*122    2*123  CONTROL      113      114      115
                        116      117      118      119      120      121
                        122      123
TAAC        VAR    DECLARED       78 IMPL-ASN      136      REF      112
                        136
TAAC11      VAR    DECLARED       79 IMPL-ASN      136      REF      112
                        113
TAAC11DEF   EQU    DECLARED       97  DEFINED      113 IMPL-ASN      136
                        REF      134
TAAC12      VAR    DECLARED       80 IMPL-ASN      136      REF      112
                        114
TAAC12DEF   EQU    DECLARED       98  DEFINED      114 IMPL-ASN      136
                        REF      134
TAAC21      VAR    DECLARED       81 IMPL-ASN      136      REF      112
                        115
TAAC21DEF   EQU    DECLARED       99  DEFINED      115 IMPL-ASN      136
                        REF      134
TAAC22      VAR    DECLARED       82 IMPL-ASN      136      REF      112
                        116
TAAC22DEF   EQU    DECLARED      100  DEFINED      116 IMPL-ASN      136
                        REF      134
TAAC31      VAR    DECLARED       83 IMPL-ASN      136      REF      112
                        117
TAAC31DEF   EQU    DECLARED      101  DEFINED      117 IMPL-ASN      136
                        REF      134
TAAC41      VAR    DECLARED       84 IMPL-ASN      136      REF      112
                        118
TAAC41DEF   EQU    DECLARED      102  DEFINED      118 IMPL-ASN      136
                        REF      134
TAACDEF     EQU    DECLARED       96  DEFINED      112 IMPL-ASN      136
                        REF      134
TGF         PARAM  DECLARED       47  DEFINED       48      REF      117
TGV         PARAM  DECLARED       52  DEFINED       53      REF      117
Y           VAR    DECLARED       73       91 IMPL-ASN      136
                        REF      113      114      115      116      122
                        123      139
```

Table 36 (continued)

SETS

I	origination nodes
K	destination nodes
L	barge sizes (kdwt)
S	tug types (kkw)
T	time periods

PARAMETERS

BGF	AAC fixed of l-size barge ($m)
BGV	AAC variable of l-size barge ($m)
BKBED	AAC of berth equipment for l-size barge at destination port k ($m)
BKBEO	AAC of berth equipment for l-size barge at origination port i ($m)
BKBHD	AAC of berthing facility for l-size barge at destination port k ($m)
BKBHO	AAC of berthing facility at origination port i ($m)
BKDM	iron ore demand at k from i for period t (kton)
GAMA	match-up relations between tug and barge
SZ	capacity of l-size barge
TGF	AAC fixed of s-type tug ($m)
TGV	AAC variable of s-type tug ($m)

VARIABLES

M	0-1 DV to select tug type for period t
MM	integer DV to determine # of tugs for period t
N	0-1 DV to select barge size for period t
NN	integer DV to determine # of barges for period t
TAAC	total annual average costs
TAAC11	AAC of berthing facility at origination port i
TAAC12	AAC of berthing facility at destination port k
TAAC21	AAC of berth equipment at origination port i
TAAC22	AAC of berth equipment at destination port k
TAAC31	AAC of tug related
TAAC41	AAC of barge related
Y	0-1 DV to select routing for period t

EQUATIONS

BALANCE1	as eq.5
BALANCE2	as eq.6
BALANCE3	as eq.14
BALANCE4	as eq.16
BKDEMAND	as eq.4
TAAC11DEF	as eq.2
TAAC12DEF	as eq.2
TAAC21DEF	as eq.2
TAAC22DEF	as eq.2
TAAC31DEF	as eq.2
TAAC41DEF	as eq.2
TAACDEF	as eq.2

Table 36 (continued)

```
MODELS

ORATB

COMPILATION TIME = 0.110 SECONDS.  VERID MW2-00-059

General Algebraic Modeling System Equation Listing

---- TAACDEF =E=  as eq.2
TAACDEF..  TAAC-TAAC11-TAAC12-TAAC21-TAAC22-TAAC31-TAAC41 =E= 0;

---- TAAC11DEF   =E=  as eq.2
TAAC11DEF..  -30*Y(O1,D1,T1)-30*Y(O1,D1,T2)-30*Y(O1,D1,T3)
-30*Y(O1,D1,T4)-30*Y(O1,D1,T5)-30*Y(O1,D1,T6)-30*Y(O1,D1,T7)
-30*Y(O1,D1,T8)-30*Y(O1,D1,T9)-30*Y(O1,D1,T10)-30*Y(O1,D1,T11)
-30*Y(O1,D1,T12)-30*Y(O1,D2,T1)-30*Y(O1,D2,T2)-30*Y(O1,D2,T3)
-30*Y(O1,D2,T4)-30*Y(O1,D2,T5)-30*Y(O1,D2,T6)-30*Y(O1,D2,T7)
-30*Y(O1,D2,T8)-30*Y(O1,D2,T9)-30*Y(O1,D2,T10)-30*Y(O1,D2,T11)
-30*Y(O1,D2,T12)-30*Y(O1,D3,T1)-30*Y(O1,D3,T2)-30*Y(O1,D3,T3)
-30*Y(O1,D3,T4)-30*Y(O1,D3,T5)-30*Y(O1,D3,T6)-30*Y(O1,D3,T7)
-30*Y(O1,D3,T8)-30*Y(O1,D3,T9)-30*Y(O1,D3,T10)-30*Y(O1,D3,T11)
-30*Y(O1,D3,T12)-30*Y(O1,D4,T1)-30*Y(O1,D4,T2)-30*Y(O1,D4,T3)
-30*Y(O1,D4,T4)-30*Y(O1,D4,T5)-30*Y(O1,D4,T6)-30*Y(O1,D4,T7)
-30*Y(O1,D4,T8)-30*Y(O1,D4,T9)-30*Y(O1,D4,T10)-30*Y(O1,D4,T11)
-30*Y(O1,D4,T12)-30*Y(O1,D5,T1)-30*Y(O1,D5,T2)-30*Y(O1,D5,T3)
-30*Y(O1,D5,T4)-30*Y(O1,D5,T5)-30*Y(O1,D5,T6)-30*Y(O1,D5,T7)
-30*Y(O1,D5,T8)-30*Y(O1,D5,T9)-30*Y(O1,D5,T10)-30*Y(O1,D5,T11)
-30*Y(O1,D5,T12)+TAAC11 =E= 0;

---- TAAC12DEF   =E=  as eq.2
TAAC12DEF..  -20*Y(O1,D1,T1)-20*Y(O1,D1,T2)-20*Y(O1,D1,T3)
-20*Y(O1,D1,T4)-20*Y(O1,D1,T5)-20*Y(O1,D1,T6)-20*Y(O1,D1,T7)
-20*Y(O1,D1,T8)-20*Y(O1,D1,T9)-20*Y(O1,D1,T10)-20*Y(O1,D1,T11)
-20*Y(O1,D1,T12)-5*Y(O1,D2,T1)-5*Y(O1,D2,T2)-5*Y(O1,D2,T3)
-5*Y(O1,D2,T4)-5*Y(O1,D2,T5)-5*Y(O1,D2,T6)-5*Y(O1,D2,T7)-5*Y(O1,D2,T8)
-5*Y(O1,D2,T9)-5*Y(O1,D2,T10)-5*Y(O1,D2,T11)-5*Y(O1,D2,T12)
-5*Y(O1,D3,T1)-5*Y(O1,D3,T2)-5*Y(O1,D3,T3)-5*Y(O1,D3,T4)-5*Y(O1,D3,T5)
-5*Y(O1,D3,T6)-5*Y(O1,D3,T7)-5*Y(O1,D3,T8)-5*Y(O1,D3,T9)-5*Y(O1,D3,T10)
-5*Y(O1,D3,T11)-5*Y(O1,D3,T12)-5*Y(O1,D4,T1)-5*Y(O1,D4,T2)
-5*Y(O1,D4,T3)-5*Y(O1,D4,T4)-5*Y(O1,D4,T5)-5*Y(O1,D4,T6)-5*Y(O1,D4,T7)
-5*Y(O1,D4,T8)-5*Y(O1,D4,T9)-5*Y(O1,D4,T10)-5*Y(O1,D4,T11)
-5*Y(O1,D4,T12)-10*Y(O1,D5,T1)-10*Y(O1,D5,T2)-10*Y(O1,D5,T3)
-10*Y(O1,D5,T4)-10*Y(O1,D5,T5)-10*Y(O1,D5,T6)-10*Y(O1,D5,T7)
-10*Y(O1,D5,T8)-10*Y(O1,D5,T9)-10*Y(O1,D5,T10)-10*Y(O1,D5,T11)
-10*Y(O1,D5,T12)+TAAC12 =E= 0;

---- TAAC21DEF   =E=  as eq.2
TAAC21DEF..  -8*Y(O1,D1,T1)-8*Y(O1,D1,T2)-8*Y(O1,D1,T3)-8*Y(O1,D1,T4)
-8*Y(O1,D1,T5)-8*Y(O1,D1,T6)-8*Y(O1,D1,T7)-8*Y(O1,D1,T8)-8*Y(O1,D1,T9)
```

Table 36 (continued)

```
-8*Y(O1,D1,T10)-8*Y(O1,D1,T11)-8*Y(O1,D1,T12)-8*Y(O1,D2,T1)
-8*Y(O1,D2,T2)-8*Y(O1,D2,T3)-8*Y(O1,D2,T4)-8*Y(O1,D2,T5)-8*Y(O1,D2,T6)
-8*Y(O1,D2,T7)-8*Y(O1,D2,T8)-8*Y(O1,D2,T9)-8*Y(O1,D2,T10)-8*Y(O1,D2,T11)
-8*Y(O1,D2,T12)-8*Y(O1,D3,T1)-8*Y(O1,D3,T2)-8*Y(O1,D3,T3)-8*Y(O1,D3,T4)
-8*Y(O1,D3,T5)-8*Y(O1,D3,T6)-8*Y(O1,D3,T7)-8*Y(O1,D3,T8)-8*Y(O1,D3,T9)
-8*Y(O1,D3,T10)-8*Y(O1,D3,T11)-8*Y(O1,D3,T12)-8*Y(O1,D4,T1)
-8*Y(O1,D4,T2)-8*Y(O1,D4,T3)-8*Y(O1,D4,T4)-8*Y(O1,D4,T5)-8*Y(O1,D4,T6)
-8*Y(O1,D4,T7)-8*Y(O1,D4,T8)-8*Y(O1,D4,T9)-8*Y(O1,D4,T10)-8*Y(O1,D4,T11)
-8*Y(O1,D4,T12)-8*Y(O1,D5,T1)-8*Y(O1,D5,T2)-8*Y(O1,D5,T3)-8*Y(O1,D5,T4)
-8*Y(O1,D5,T5)-8*Y(O1,D5,T6)-8*Y(O1,D5,T7)-8*Y(O1,D5,T8)-8*Y(O1,D5,T9)
-8*Y(O1,D5,T10)-8*Y(O1,D5,T11)-8*Y(O1,D5,T12)+TAAC21 =E= 0;

---- TAAC22DEF   =E=  as eq.2
TAAC22DEF..  -6*Y(O1,D1,T1)-6*Y(O1,D1,T2)-6*Y(O1,D1,T3)-6*Y(O1,D1,T4)
-6*Y(O1,D1,T5)-6*Y(O1,D1,T6)-6*Y(O1,D1,T7)-6*Y(O1,D1,T8)-6*Y(O1,D1,T9)
-6*Y(O1,D1,T10)-6*Y(O1,D1,T11)-6*Y(O1,D1,T12)-4*Y(O1,D2,T1)
-4*Y(O1,D2,T2)-4*Y(O1,D2,T3)-4*Y(O1,D2,T4)-4*Y(O1,D2,T5)-4*Y(O1,D2,T6)
-4*Y(O1,D2,T7)-4*Y(O1,D2,T8)-4*Y(O1,D2,T9)-4*Y(O1,D2,T10)-4*Y(O1,D2,T11)
-4*Y(O1,D2,T12)-4*Y(O1,D3,T1)-4*Y(O1,D3,T2)-4*Y(O1,D3,T3)-4*Y(O1,D3,T4)
-4*Y(O1,D3,T5)-4*Y(O1,D3,T6)-4*Y(O1,D3,T7)-4*Y(O1,D3,T8)-4*Y(O1,D3,T9)
-4*Y(O1,D3,T10)-4*Y(O1,D3,T11)-4*Y(O1,D3,T12)-4*Y(O1,D4,T1)
-4*Y(O1,D4,T2)-4*Y(O1,D4,T3)-4*Y(O1,D4,T4)-4*Y(O1,D4,T5)-4*Y(O1,D4,T6)
-4*Y(O1,D4,T7)-4*Y(O1,D4,T8)-4*Y(O1,D4,T9)-4*Y(O1,D4,T10)-4*Y(O1,D4,T11)
-4*Y(O1,D4,T12)-5*Y(O1,D5,T1)-5*Y(O1,D5,T2)-5*Y(O1,D5,T3)-5*Y(O1,D5,T4)
-5*Y(O1,D5,T5)-5*Y(O1,D5,T6)-5*Y(O1,D5,T7)-5*Y(O1,D5,T8)-5*Y(O1,D5,T9)
-5*Y(O1,D5,T10)-5*Y(O1,D5,T11)-5*Y(O1,D5,T12)+TAAC22 =E= 0;

---- TAAC31DEF   =E=  as eq.2
TAAC31DEF..  -5*M(O1,D1,TG1,T1)-5*M(O1,D1,TG1,T2)-5*M(O1,D1,TG1,T3)
-5*M(O1,D1,TG1,T4)-5*M(O1,D1,TG1,T5)-5*M(O1,D1,TG1,T6)-5*M(O1,D1,TG1,T7)
-5*M(O1,D1,TG1,T8)-5*M(O1,D1,TG1,T9)-5*M(O1,D1,TG1,T10)
-5*M(O1,D1,TG1,T11)-5*M(O1,D1,TG1,T12)-7.1*M(O1,D1,TG2,T1)
-7.1*M(O1,D1,TG2,T2)-7.1*M(O1,D1,TG2,T3)-7.1*M(O1,D1,TG2,T4)
-7.1*M(O1,D1,TG2,T5)-7.1*M(O1,D1,TG2,T6)-7.1*M(O1,D1,TG2,T7)
-7.1*M(O1,D1,TG2,T8)-7.1*M(O1,D1,TG2,T9)-7.1*M(O1,D1,TG2,T10)
-7.1*M(O1,D1,TG2,T11)-7.1*M(O1,D1,TG2,T12)-9.2*M(O1,D1,TG3,T1)
-9.2*M(O1,D1,TG3,T2)-9.2*M(O1,D1,TG3,T3)-9.2*M(O1,D1,TG3,T4)
-9.2*M(O1,D1,TG3,T5)-9.2*M(O1,D1,TG3,T6)-9.2*M(O1,D1,TG3,T7)
-9.2*M(O1,D1,TG3,T8)-9.2*M(O1,D1,TG3,T9)-9.2*M(O1,D1,TG3,T10)
-9.2*M(O1,D1,TG3,T11)-9.2*M(O1,D1,TG3,T12)-5*M(O1,D2,TG1,T1)
-5*M(O1,D2,TG1,T2)-5*M(O1,D2,TG1,T3)-5*M(O1,D2,TG1,T4)
-5*M(O1,D2,TG1,T5)-5*M(O1,D2,TG1,T6)-5*M(O1,D2,TG1,T7)
-5*M(O1,D2,TG1,T8)-5*M(O1,D2,TG1,T9)-5*M(O1,D2,TG1,T10)
-5*M(O1,D2,TG1,T11)-5*M(O1,D2,TG1,T12)-7.1*M(O1,D2,TG2,T1)
-7.1*M(O1,D2,TG2,T2)-7.1*M(O1,D2,TG2,T3)-7.1*M(O1,D2,TG2,T4)
-7.1*M(O1,D2,TG2,T5)-7.1*M(O1,D2,TG2,T6)-7.1*M(O1,D2,TG2,T7)
-7.1*M(O1,D2,TG2,T8)-7.1*M(O1,D2,TG2,T9)-7.1*M(O1,D2,TG2,T10)
-7.1*M(O1,D2,TG2,T11)-7.1*M(O1,D2,TG2,T12)-9.2*M(O1,D2,TG3,T1)
-9.2*M(O1,D2,TG3,T2)-9.2*M(O1,D2,TG3,T3)-9.2*M(O1,D2,TG3,T4)
-9.2*M(O1,D2,TG3,T5)-9.2*M(O1,D2,TG3,T6)-9.2*M(O1,D2,TG3,T7)
```

Table 36 (continued)

```
-9.2*M(O1,D2,TG3,T8)-9.2*M(O1,D2,TG3,T9)-9.2*M(O1,D2,TG3,T10)
-9.2*M(O1,D2,TG3,T11)-9.2*M(O1,D2,TG3,T12)-5*M(O1,D3,TG1,T1)
-5*M(O1,D3,TG1,T2)-5*M(O1,D3,TG1,T3)-5*M(O1,D3,TG1,T4)
-5*M(O1,D3,TG1,T5)-5*M(O1,D3,TG1,T6)-5*M(O1,D3,TG1,T7)
-5*M(O1,D3,TG1,T8)-5*M(O1,D3,TG1,T9)-5*M(O1,D3,TG1,T10)
-5*M(O1,D3,TG1,T11)-5*M(O1,D3,TG1,T12)-7.1*M(O1,D3,TG2,T1)
-7.1*M(O1,D3,TG2,T2)-7.1*M(O1,D3,TG2,T3)-7.1*M(O1,D3,TG2,T4)
-7.1*M(O1,D3,TG2,T5)-7.1*M(O1,D3,TG2,T6)-7.1*M(O1,D3,TG2,T7)
-7.1*M(O1,D3,TG2,T8)-7.1*M(O1,D3,TG2,T9)-7.1*M(O1,D3,TG2,T10)
-7.1*M(O1,D3,TG2,T11)-7.1*M(O1,D3,TG2,T12)-9.2*M(O1,D3,TG3,T1)
-9.2*M(O1,D3,TG3,T2)-9.2*M(O1,D3,TG3,T3)-9.2*M(O1,D3,TG3,T4)
-9.2*M(O1,D3,TG3,T5)-9.2*M(O1,D3,TG3,T6)-9.2*M(O1,D3,TG3,T7)
-9.2*M(O1,D3,TG3,T8)-9.2*M(O1,D3,TG3,T9)-9.2*M(O1,D3,TG3,T10)
-9.2*M(O1,D3,TG3,T11)-9.2*M(O1,D3,TG3,T12)-5*M(O1,D4,TG1,T1)
-5*M(O1,D4,TG1,T2)-5*M(O1,D4,TG1,T3)-5*M(O1,D4,TG1,T4)
-5*M(O1,D4,TG1,T5)-5*M(O1,D4,TG1,T6)-5*M(O1,D4,TG1,T7)
-5*M(O1,D4,TG1,T8)-5*M(O1,D4,TG1,T9)-5*M(O1,D4,TG1,T10)
-5*M(O1,D4,TG1,T11)-5*M(O1,D4,TG1,T12)-7.1*M(O1,D4,TG2,T1)
-7.1*M(O1,D4,TG2,T2)-7.1*M(O1,D4,TG2,T3)-7.1*M(O1,D4,TG2,T4)
-7.1*M(O1,D4,TG2,T5)-7.1*M(O1,D4,TG2,T6)-7.1*M(O1,D4,TG2,T7)
-7.1*M(O1,D4,TG2,T8)-7.1*M(O1,D4,TG2,T9)-7.1*M(O1,D4,TG2,T10)
-7.1*M(O1,D4,TG2,T11)-7.1*M(O1,D4,TG2,T12)-9.2*M(O1,D4,TG3,T1)
-9.2*M(O1,D4,TG3,T2)-9.2*M(O1,D4,TG3,T3)-9.2*M(O1,D4,TG3,T4)
-9.2*M(O1,D4,TG3,T5)-9.2*M(O1,D4,TG3,T6)-9.2*M(O1,D4,TG3,T7)
-9.2*M(O1,D4,TG3,T8)-9.2*M(O1,D4,TG3,T9)-9.2*M(O1,D4,TG3,T10)
-9.2*M(O1,D4,TG3,T11)-9.2*M(O1,D4,TG3,T12)-5*M(O1,D5,TG1,T1)
-5*M(O1,D5,TG1,T2)-5*M(O1,D5,TG1,T3)-5*M(O1,D5,TG1,T4)
-5*M(O1,D5,TG1,T5)-5*M(O1,D5,TG1,T6)-5*M(O1,D5,TG1,T7)
-5*M(O1,D5,TG1,T8)-5*M(O1,D5,TG1,T9)-5*M(O1,D5,TG1,T10)
-5*M(O1,D5,TG1,T11)-5*M(O1,D5,TG1,T12)-7.1*M(O1,D5,TG2,T1)
-7.1*M(O1,D5,TG2,T2)-7.1*M(O1,D5,TG2,T3)-7.1*M(O1,D5,TG2,T4)
-7.1*M(O1,D5,TG2,T5)-7.1*M(O1,D5,TG2,T6)-7.1*M(O1,D5,TG2,T7)
-7.1*M(O1,D5,TG2,T8)-7.1*M(O1,D5,TG2,T9)-7.1*M(O1,D5,TG2,T10)
-7.1*M(O1,D5,TG2,T11)-7.1*M(O1,D5,TG2,T12)-9.2*M(O1,D5,TG3,T1)
-9.2*M(O1,D5,TG3,T2)-9.2*M(O1,D5,TG3,T3)-9.2*M(O1,D5,TG3,T4)
-9.2*M(O1,D5,TG3,T5)-9.2*M(O1,D5,TG3,T6)-9.2*M(O1,D5,TG3,T7)
-9.2*M(O1,D5,TG3,T8)-9.2*M(O1,D5,TG3,T9)-9.2*M(O1,D5,TG3,T10)
-9.2*M(O1,D5,TG3,T11)-9.2*M(O1,D5,TG3,T12)+TAAC31 =E= 0;

---- TAAC41DEF   =E=  as eq.2
TAAC41DEF..  -1.1*N(O1,D1,BG1,T1)-1.1*N(O1,D1,BG1,T2)
-1.1*N(O1,D1,BG1,T3)-1.1*N(O1,D1,BG1,T4)-1.1*N(O1,D1,BG1,T5)
-1.1*N(O1,D1,BG1,T6)-1.1*N(O1,D1,BG1,T7)-1.1*N(O1,D1,BG1,T8)
-1.1*N(O1,D1,BG1,T9)-1.1*N(O1,D1,BG1,T10)-1.1*N(O1,D1,BG1,T11)
-1.1*N(O1,D1,BG1,T12)-1.7*N(O1,D1,BG2,T1)-1.7*N(O1,D1,BG2,T2)
-1.7*N(O1,D1,BG2,T3)-1.7*N(O1,D1,BG2,T4)-1.7*N(O1,D1,BG2,T5)
-1.7*N(O1,D1,BG2,T6)-1.7*N(O1,D1,BG2,T7)-1.7*N(O1,D1,BG2,T8)
-1.7*N(O1,D1,BG2,T9)-1.7*N(O1,D1,BG2,T10)-1.7*N(O1,D1,BG2,T11)
-1.7*N(O1,D1,BG2,T12)-2.3*N(O1,D1,BG3,T1)-2.3*N(O1,D1,BG3,T2)
-2.3*N(O1,D1,BG3,T3)-2.3*N(O1,D1,BG3,T4)-2.3*N(O1,D1,BG3,T5)
```

Table 36 (continued)

```
-2.3*N(O1,D1,BG3,T6)-2.3*N(O1,D1,BG3,T7)-2.3*N(O1,D1,BG3,T8)
-2.3*N(O1,D1,BG3,T9)-2.3*N(O1,D1,BG3,T10)-2.3*N(O1,D1,BG3,T11)
-2.3*N(O1,D1,BG3,T12)-1.1*N(O1,D2,BG1,T1)-1.1*N(O1,D2,BG1,T2)
-1.1*N(O1,D2,BG1,T3)-1.1*N(O1,D2,BG1,T4)-1.1*N(O1,D2,BG1,T5)
-1.1*N(O1,D2,BG1,T6)-1.1*N(O1,D2,BG1,T7)-1.1*N(O1,D2,BG1,T8)
-1.1*N(O1,D2,BG1,T9)-1.1*N(O1,D2,BG1,T10)-1.1*N(O1,D2,BG1,T11)
-1.1*N(O1,D2,BG1,T12)-1.7*N(O1,D2,BG2,T1)-1.7*N(O1,D2,BG2,T2)
-1.7*N(O1,D2,BG2,T3)-1.7*N(O1,D2,BG2,T4)-1.7*N(O1,D2,BG2,T5)
-1.7*N(O1,D2,BG2,T6)-1.7*N(O1,D2,BG2,T7)-1.7*N(O1,D2,BG2,T8)
-1.7*N(O1,D2,BG2,T9)-1.7*N(O1,D2,BG2,T10)-1.7*N(O1,D2,BG2,T11)
-1.7*N(O1,D2,BG2,T12)-2.3*N(O1,D2,BG3,T1)-2.3*N(O1,D2,BG3,T2)
-2.3*N(O1,D2,BG3,T3)-2.3*N(O1,D2,BG3,T4)-2.3*N(O1,D2,BG3,T5)
-2.3*N(O1,D2,BG3,T6)-2.3*N(O1,D2,BG3,T7)-2.3*N(O1,D2,BG3,T8)
-2.3*N(O1,D2,BG3,T9)-2.3*N(O1,D2,BG3,T10)-2.3*N(O1,D2,BG3,T11)
-2.3*N(O1,D2,BG3,T12)-1.1*N(O1,D3,BG1,T1)-1.1*N(O1,D3,BG1,T2)
-1.1*N(O1,D3,BG1,T3)-1.1*N(O1,D3,BG1,T4)-1.1*N(O1,D3,BG1,T5)
-1.1*N(O1,D3,BG1,T6)-1.1*N(O1,D3,BG1,T7)-1.1*N(O1,D3,BG1,T8)
-1.1*N(O1,D3,BG1,T9)-1.1*N(O1,D3,BG1,T10)-1.1*N(O1,D3,BG1,T11)
-1.1*N(O1,D3,BG1,T12)-1.7*N(O1,D3,BG2,T1)-1.7*N(O1,D3,BG2,T2)
-1.7*N(O1,D3,BG2,T3)-1.7*N(O1,D3,BG2,T4)-1.7*N(O1,D3,BG2,T5)
-1.7*N(O1,D3,BG2,T6)-1.7*N(O1,D3,BG2,T7)-1.7*N(O1,D3,BG2,T8)
-1.7*N(O1,D3,BG2,T9)-1.7*N(O1,D3,BG2,T10)-1.7*N(O1,D3,BG2,T11)
-1.7*N(O1,D3,BG2,T12)-2.3*N(O1,D3,BG3,T1)-2.3*N(O1,D3,BG3,T2)
-2.3*N(O1,D3,BG3,T3)-2.3*N(O1,D3,BG3,T4)-2.3*N(O1,D3,BG3,T5)
-2.3*N(O1,D3,BG3,T6)-2.3*N(O1,D3,BG3,T7)-2.3*N(O1,D3,BG3,T8)
-2.3*N(O1,D3,BG3,T9)-2.3*N(O1,D3,BG3,T10)-2.3*N(O1,D3,BG3,T11)
-2.3*N(O1,D3,BG3,T12)-1.1*N(O1,D4,BG1,T1)-1.1*N(O1,D4,BG1,T2)
-1.1*N(O1,D4,BG1,T3)-1.1*N(O1,D4,BG1,T4)-1.1*N(O1,D4,BG1,T5)
-1.1*N(O1,D4,BG1,T6)-1.1*N(O1,D4,BG1,T7)-1.1*N(O1,D4,BG1,T8)
-1.1*N(O1,D4,BG1,T9)-1.1*N(O1,D4,BG1,T10)-1.1*N(O1,D4,BG1,T11)
-1.1*N(O1,D4,BG1,T12)-1.7*N(O1,D4,BG2,T1)-1.7*N(O1,D4,BG2,T2)
-1.7*N(O1,D4,BG2,T3)-1.7*N(O1,D4,BG2,T4)-1.7*N(O1,D4,BG2,T5)
-1.7*N(O1,D4,BG2,T6)-1.7*N(O1,D4,BG2,T7)-1.7*N(O1,D4,BG2,T8)
-1.7*N(O1,D4,BG2,T9)-1.7*N(O1,D4,BG2,T10)-1.7*N(O1,D4,BG2,T11)
-1.7*N(O1,D4,BG2,T12)-2.3*N(O1,D4,BG3,T1)-2.3*N(O1,D4,BG3,T2)
-2.3*N(O1,D4,BG3,T3)-2.3*N(O1,D4,BG3,T4)-2.3*N(O1,D4,BG3,T5)
-2.3*N(O1,D4,BG3,T6)-2.3*N(O1,D4,BG3,T7)-2.3*N(O1,D4,BG3,T8)
-2.3*N(O1,D4,BG3,T9)-2.3*N(O1,D4,BG3,T10)-2.3*N(O1,D4,BG3,T11)
-2.3*N(O1,D4,BG3,T12)-1.1*N(O1,D5,BG1,T1)-1.1*N(O1,D5,BG1,T2)
-1.1*N(O1,D5,BG1,T3)-1.1*N(O1,D5,BG1,T4)-1.1*N(O1,D5,BG1,T5)
-1.1*N(O1,D5,BG1,T6)-1.1*N(O1,D5,BG1,T7)-1.1*N(O1,D5,BG1,T8)
-1.1*N(O1,D5,BG1,T9)-1.1*N(O1,D5,BG1,T10)-1.1*N(O1,D5,BG1,T11)
-1.1*N(O1,D5,BG1,T12)-1.7*N(O1,D5,BG2,T1)-1.7*N(O1,D5,BG2,T2)
-1.7*N(O1,D5,BG2,T3)-1.7*N(O1,D5,BG2,T4)-1.7*N(O1,D5,BG2,T5)
-1.7*N(O1,D5,BG2,T6)-1.7*N(O1,D5,BG2,T7)-1.7*N(O1,D5,BG2,T8)
-1.7*N(O1,D5,BG2,T9)-1.7*N(O1,D5,BG2,T10)-1.7*N(O1,D5,BG2,T11)
-1.7*N(O1,D5,BG2,T12)-2.3*N(O1,D5,BG3,T1)-2.3*N(O1,D5,BG3,T2)
-2.3*N(O1,D5,BG3,T3)-2.3*N(O1,D5,BG3,T4)-2.3*N(O1,D5,BG3,T5)
-2.3*N(O1,D5,BG3,T6)-2.3*N(O1,D5,BG3,T7)-2.3*N(O1,D5,BG3,T8)
-2.3*N(O1,D5,BG3,T9)-2.3*N(O1,D5,BG3,T10)-2.3*N(O1,D5,BG3,T11)
```

Table 36 (continued)

```
-2.3*N(O1,D5,BG3,T12)+TAAC41 =E= 0;

---- BKDEMAND    =G=  as eq.4
BKDEMAND(O1,D1,T1)..
5*N(O1,D1,BG1,T1)+10*N(O1,D1,BG2,T1)+15*N(O1,D1,BG3,T1) =G= 14;
BKDEMAND(O1,D1,T2)..
5*N(O1,D1,BG1,T2)+10*N(O1,D1,BG2,T2)+15*N(O1,D1,BG3,T2) =G= 14;
BKDEMAND(O1,D1,T3)..
5*N(O1,D1,BG1,T3)+10*N(O1,D1,BG2,T3)+15*N(O1,D1,BG3,T3) =G= 15;
REMAINING 57 ENTRIES SKIPPED

---- BALANCE1    =E=  as eq.5
BALANCE1(O1,D1,T1)..  M(O1,D1,TG1,T1)+M(O1,D1,TG2,T1)+M(O1,D1,TG3,T1)
 =E= 1;
BALANCE1(O1,D1,T2)..  M(O1,D1,TG1,T2)+M(O1,D1,TG2,T2)+M(O1,D1,TG3,T2)
 =E= 1;
BALANCE1(O1,D1,T3)..  M(O1,D1,TG1,T3)+M(O1,D1,TG2,T3)+M(O1,D1,TG3,T3)
 =E= 1;
REMAINING 57 ENTRIES SKIPPED

---- BALANCE2    =E=  as eq.6
BALANCE2(O1,D1,T1)..  N(O1,D1,BG1,T1)+N(O1,D1,BG2,T1)+N(O1,D1,BG3,T1)
 =E= 1;
BALANCE2(O1,D1,T2)..  N(O1,D1,BG1,T2)+N(O1,D1,BG2,T2)+N(O1,D1,BG3,T2)
 =E= 1;
BALANCE2    =E=  as eq.6
BALANCE2(O1,D1,T3)..  N(O1,D1,BG1,T3)+N(O1,D1,BG2,T3)+N(O1,D1,BG3,T3)
 =E= 1;
REMAINING 57 ENTRIES SKIPPED

---- BALANCE3    =L=  as eq.14
BALANCE3(O1,D1,TG1,T1)..  -Y(O1,D1,T1)+M(O1,D1,TG1,T1) =L= 0;
BALANCE3(O1,D1,TG1,T2)..  -Y(O1,D1,T2)+M(O1,D1,TG1,T2) =L= 0;
BALANCE3(O1,D1,TG1,T3)..  -Y(O1,D1,T3)+M(O1,D1,TG1,T3) =L= 0;
REMAINING 177 ENTRIES SKIPPED

---- BALANCE4    =L=  as eq.16
BALANCE4(O1,D1,BG1,T1)..  -Y(O1,D1,T1)+N(O1,D1,BG1,T1) =L= 0;
BALANCE4(O1,D1,BG1,T2)..  -Y(O1,D1,T2)+N(O1,D1,BG1,T2) =L= 0;
BALANCE4(O1,D1,BG1,T3)..  -Y(O1,D1,T3)+N(O1,D1,BG1,T3) =L= 0;
REMAINING 177 ENTRIES SKIPPED

---- Y0-1 DV to select routing for period t
Y(O1,D1,T1) (.LO, .L, .UP = 0, 0, 1)
 -30  TAAC11DEF
 -20  TAAC12DEF
  -8  TAAC21DEF
  -6  TAAC22DEF
  -1  BALANCE3(O1,D1,TG1,T1)
  -1  BALANCE3(O1,D1,TG2,T1)
```

Table 36 (continued)

```
  -1  BALANCE3(O1,D1,TG3,T1)
  -1  BALANCE4(O1,D1,BG1,T1)
  -1  BALANCE4(O1,D1,BG2,T1)
  -1  BALANCE4(O1,D1,BG3,T1)
Y(O1,D1,T2) (.LO, .L, .UP = 0, 0, 1)
 -30  TAAC11DEF
 -20  TAAC12DEF
  -8  TAAC21DEF
  -6  TAAC22DEF
  -1  BALANCE3(O1,D1,TG1,T2)
  -1  BALANCE3(O1,D1,TG2,T2)
  -1  BALANCE3(O1,D1,TG3,T2)
  -1  BALANCE4(O1,D1,BG1,T2)
  -1  BALANCE4(O1,D1,BG2,T2)
  -1  BALANCE4(O1,D1,BG3,T2)
Y(O1,D1,T3) (.LO, .L, .UP = 0, 0, 1)
 -30  TAAC11DEF
 -20  TAAC12DEF
  -8  TAAC21DEF
  -6  TAAC22DEF
  -1  BALANCE3(O1,D1,TG1,T3)
  -1  BALANCE3(O1,D1,TG2,T3)
  -1  BALANCE3(O1,D1,TG3,T3)
  -1  BALANCE4(O1,D1,BG1,T3)
  -1  BALANCE4(O1,D1,BG2,T3)
  -1  BALANCE4(O1,D1,BG3,T3)
REMAINING 57 ENTRIES SKIPPED

---- M0-1 DV to select tug type for period t
M(O1,D1,TG1,T1) (.LO, .L, .UP = 0, 0, 1)
  -5  TAAC31DEF
   1  BALANCE1(O1,D1,T1)
   1  BALANCE3(O1,D1,TG1,T1)
M(O1,D1,TG1,T2) (.LO, .L, .UP = 0, 0, 1)
  -5  TAAC31DEF
   1  BALANCE1(O1,D1,T2)
   1  BALANCE3(O1,D1,TG1,T2)
M(O1,D1,TG1,T3)
 (.LO, .L, .UP = 0, 0, 1)
  -5  TAAC31DEF
   1  BALANCE1(O1,D1,T3)
   1  BALANCE3(O1,D1,TG1,T3)
REMAINING 177 ENTRIES SKIPPED

---- N0-1 DV to select barge size for period t
N(O1,D1,BG1,T1) (.LO, .L, .UP = 0, 0, 1)
  -1.1TAAC41DEF
   5  BKDEMAND(O1,D1,T1)
   1  BALANCE2(O1,D1,T1)
   1  BALANCE4(O1,D1,BG1,T1)
```

Table 36 (continued)

```
N(O1,D1,BG1,T2) (.LO, .L, .UP = 0, 0, 1)
  -1.1TAAC41DEF
   5  BKDEMAND(O1,D1,T2)
   1  BALANCE2(O1,D1,T2)
   1  BALANCE4(O1,D1,BG1,T2)
N(O1,D1,BG1,T3) (.LO, .L, .UP = 0, 0, 1)
  -1.1TAAC41DEF
   5  BKDEMAND(O1,D1,T3)
   1  BALANCE2(O1,D1,T3)
   1  BALANCE4(O1,D1,BG1,T3)
REMAINING 177 ENTRIES SKIPPED

---- TAAC  total annual average costs
TAAC (.LO, .L, .UP = -INF, 0, +INF)
   1  TAACDEF

---- TAAC11AAC of berthing facility at origination port i
TAAC11 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC11DEF

---- TAAC12AAC of berthing facility at destination port k
TAAC12 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC12DEF

---- TAAC21AAC of berth equipment at origination port i
TAAC21 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC21DEF

---- TAAC22AAC of berth equipment at destination port k
TAAC22 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC22DEF

---- TAAC31AAC of tug related
TAAC31 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC31DEF

---- TAAC41AAC of barge related
TAAC41 (.LO, .L, .UP = -INF, 0, +INF)
  -1  TAACDEF
   1  TAAC41DEF

General Algebraic Modeling System Model Statistics

BLOCKS OF EQUATIONS 12 SINGLE EQUATIONS: 547
BLOCKS OF VARIABLES 10 SINGLE VARIABLES: 427
```

Table 36 (continued)

```
NON ZERO ELEMENTS 1873 DISCRETE VARIABLES: 420
GENERATION TIME = 0.770 SECONDS
EXECUTION TIME = 1.710 SECONDS.  VERID MW2-00-059

General Algebraic Modeling System Solution Report

MODEL ORATBOBJECTIVE: TAAC
TYPE MIP DIRECTION: MINIMIZE
SOLVER OSL FROM LINE 136
OBJECTIVE VALUE: 3483.6
RESOURCE USAGE: 10000
ITERATION COUNT: 5000
Work space allocated: 0.71 Mb
Objective value of this solution: 3483.6
Optcr: 0.005
Optca: 0.0
The solution satisfies the termination tolerances

                  LOWER      LEVEL       UPPER       MARGINAL
---- VAR TAAC     -INF       3483.6      +INF        0
---- VAR TAAC11   -INF       1800.0      +INF        0
---- VAR TAAC12   -INF       540.0       +INF        0
---- VAR TAAC21   -INF       480.0       +INF        0
---- VAR TAAC22   -INF       276.0       +INF        0
---- VAR TAAC31   -INF       300.0       +INF        0
---- VAR TAAC41   -INF       87.6        +INF        0

TAAC        total annual average costs
TAAC11      AAC of berthing facility at origination port i
TAAC12      AAC of berthing facility at destination port k
TAAC21      AAC of berth equipment at origination port i
TAAC22      AAC of berth equipment at destination port k
TAAC31      AAC of tug related
TAAC41      AAC of barge related

REPORT SUMMARY: 0 INFEASIBLE; 0 UNBOUNDED

---- Y.L    0-1 DV to select routing for period t

            T1    T2    T3    T4    T5    T6
O1.D1       1     1     1     1     1     1
O1.D2       1     1     1     1     1     1
O1.D3       1     1     1     1     1     1
O1.D4       1     1     1     1     1     1
O1.D5       1     1     1     1     1     1

+           T7    T8    T9    T10   T11   T12
O1.D1       1     1     1     1     1     1
O1.D2       1     1     1     1     1     1
O1.D3       1     1     1     1     1     1
```

Table 36 (continued)

```
O1.D4        1     1     1     1     1     1
O1.D5        1     1     1     1     1     1

---- M.L 0-1 DV to select tug type for period t

             T1    T2    T3    T4    T5    T6
D1.TG3       1     1     1     1     1     1
D2.TG1       1     1     1     1     1     1
D3.TG1       1     1     1     1     1     1
D4.TG1       1     1     1     1     1     1
D5.TG2       1     1     1     1     1     1

+            T7    T8    T9    T10   T11   T12
D1.TG3       1     1     1     1     1     1
D2.TG1       1     1     1     1     1     1
D3.TG1       1     1     1     1     1     1
D4.TG1       1     1     1     1     1     1
D5.TG2       1     1     1     1     1     1

---- N.L 0-1 DV to select barge size for period t

             T1    T2    T3    T4    T5    T6
D1.BG3       1     1     1     1     1     1
D2.BG1       1     1     1     1     1     1
D3.BG1       1     1     1     1     1     1
D4.BG1       1     1     1     1     1     1
D5.BG2       1     1     1     1     1     1

+            T7    T8    T9    T10   T11   T12
D1.BG3       1     1     1     1     1     1
D2.BG1       1     1     1     1     1     1
D3.BG1       1     1     1     1     1     1
D4.BG1       1     1     1     1     1     1
D5.BG2       1     1     1     1     1     1
************************************************************************
```

the berthing facilities at the other five ports. With respect to costs for loading and unloading facilities, \$480.0 million (or 13.8%) are for Ningbo, and \$276.0 million (or 7.9%) for the other five. Total investment and operating costs of tugs required represent 8.6% (or \$300.0 million) of the *TAAC*, while barges only 2.5% (or \$87.6 million).

It is necessary to point out that in this sample run, we only considered the iron ore transport for the time being. Container transport components of the model can be easily added. The main reason for performing this simplification is that we lack some basic data on the container transport.

The second notice about this sample run is that the results from this optimization model are subject to change after the performance of this recommended system is evaluated in the simulation model. A large improvement on this recommended resource acquisition strategy is anticipated once the resource utilization is assessed.

4.2.2. Simulation Model Application: Results And Evaluation

The objective of this simulation model is to evaluate the performance of the resource acquisition strategy recommended by the optimization model under a set of local specific conditions. The local specific conditions should include port infrastructure conditions, loading and unloading facilities, storage and yard facilities, and navigation channel limitations at each specific demand nodes. In this simulation sample run, however, we take the conditions of the Port of Wuhan as an example.

The SLAM input statements are as shown in Table 7 in Section 3.4.5. The output summary report for this ORATB application is given in Table 37. As can be seen from the output statistics, the mill is operated 100% of the time from day 165 to day 365. This high percentage of mill utilization occurs at the expenses of the barges which wait on an average of 1.508 days for the unloading dock. This is further illustrated by the file

Table 37 The ORATB Simulation Model Output Summary Report

```
****************************************************************************
               SIMULATION PROJECT  ORATB FLEET             BY  MING QI
               DATE  11/18/1996                            RUN NUMBER  1 OF 1

               CURRENT TIME  .3650E+03
               STATISTICAL ARRAYS CLEARED AT TIME  .1650E+03


                       **STATISTICS FOR VARIABLES BASED ON OBSERVATION**

                   MEAN      STANDARD     COEFF. OF    MINIMUM     MAXIMUM    NUMBER OF
                   VALUE     DEVIATION    VARIATION    VALUE       VALUE      OBSERVATIONS
TRIP TIME        .1436E+02   .2045E+01    .1425E+00    .9574E+01   .1999E+02     211


                       **STATISTICS FOR TIME-PERSISTENT VARIABLES**

                    MEAN      STANDARD     MINIMUM     MAXIMUM     TIME        CURRENT
                    VALUE     DEVIATION    VALUE       VALUE       INTERVAL    VALUE
STML INPUT AVAIL .1000E+01   .0000E+00    .1000E+01    .1000E+01   .2000E+03   .1000E+01
YARD LEVEL       .1807E+04   .1085E+03    .1592E+04    .2004E+04   .2000E+03   .1661E+04


                                      **FILE STATISTICS**

FILE              ASSOC NODE    AVERAGE     STANDARD     MAXIMUM    CURRENT    AVERAGE
NUMBER            LABEL/TYPE    LENGTH      DEVIATION    LENGTH     LENGTH     WAITING TIME
  1               AWAIT          1.5759      1.2173         5          2         1.5080
  2               CALENDAR      14.2213      1.3584        18         13         1.2197


                              **REGULAR ACTIVITY STATISTICS**

ACTIVITY         AVERAGE        STANDARD      MAXIMUM        CURRENT        ENTITY
INDEX/LABEL      UTILIZATION    DEVIATION     UTILIZATION    UTILIZATION    COUNT
1 LOADING           3.1566        1.5206          7              5           208
2 TO WUHAN          5.1803        1.4911         10              4           208
3 UNLOADING         0.9529        0.2119          1              1           207
4 RETURN TRIP       4.1344        1.5377          8              3           211
```

Table 37 (continued)

```
                                **RESOURCE STATISTICS**

RESOURCE    RESOURCE    CURRENT     AVERAGE       STANDARD     MAXIMUM       CURRENT
NUMBER      LABEL       CAPACITY    UTILIZATION   DEVIATION    UTILIZATION   UTILIZATION

   1        DOCK           1         0.9529        0.2119          1             1

RESOURCE    RESOURCE    CURRENT     AVERAGE       MINIMUM      MAXIMUM
NUMBER      LABEL       AVAILABLE   AVAILABLE     AVAILABLE    AVAILABLE

   1        DOCK           0         0.0471          0             1

                                **STATE AND DERIVATIVE VARIABLES**

                                 (I)         SS(I)           DD(I)

                                  1        .1378E+02       .0000E+00

                                  2        .1661E+04       .0000E+00
************************************************************************
```

statistics which indicate that the average number waiting for the unloading dock is 1.58, and that as many as five barges were waiting at one time. The resource statistics indicate that there was a barge in the unloading dock 95.29% of the time. This statistic can also be obtained from the statistics for $ACTIVITY3$. Also available from the activity statistics is the average number of barges being loaded as this quantity is the average utilization of $ACTIVITY1$. From the output, it is seen that approximately 3.16 barges are being loaded and the maximum number of barges loaded concurrently is seven.

Now, let us propose to build floating docks for unloading operations at Wuhan for the ORATB system. Three such docks are proposed, each of which can process at a rate two-thirds that of the current dock. We can revise the simulation model to compare system operation between the three floating unloading docks operating on only one shift versus the current unloading dock operating on a three-shift basis. The easiest way to modify the original problem is to set *XX(1)* = 1 in an *INTLC* statement. The start-up/shut-down sub-process as shown in Figure 8 should be first deleted from the network statements. The revised model for the three floating docks is shown in Figure 16. The corresponding network statements for this modification are shown in Table 38 and the summary report Table 39.

A comparison of the results shows that the three-dock operation is more efficient with a significant reduction in waiting time. The use of three docks on a one shift basis does not decrease the average round-trip time. The reason for this lack of decrease is that the operation of the unloading dock for only one shift a day causes barges to spend at least two separate portions of a day at the unloading dock because it takes at least a half a day to unload a barge. Thus, the proposed redesign enforces at least a two-thirds of a day wait in the dock for each barge. For some barges that arrive near the end of a working shift, an additional two-thirds of a day is spent in the dock.

The output shows that barges on the average wait 0.4271 days to gain access to one of the unloading docks. They spend 0.77 days being unloaded. Subtracting these values and

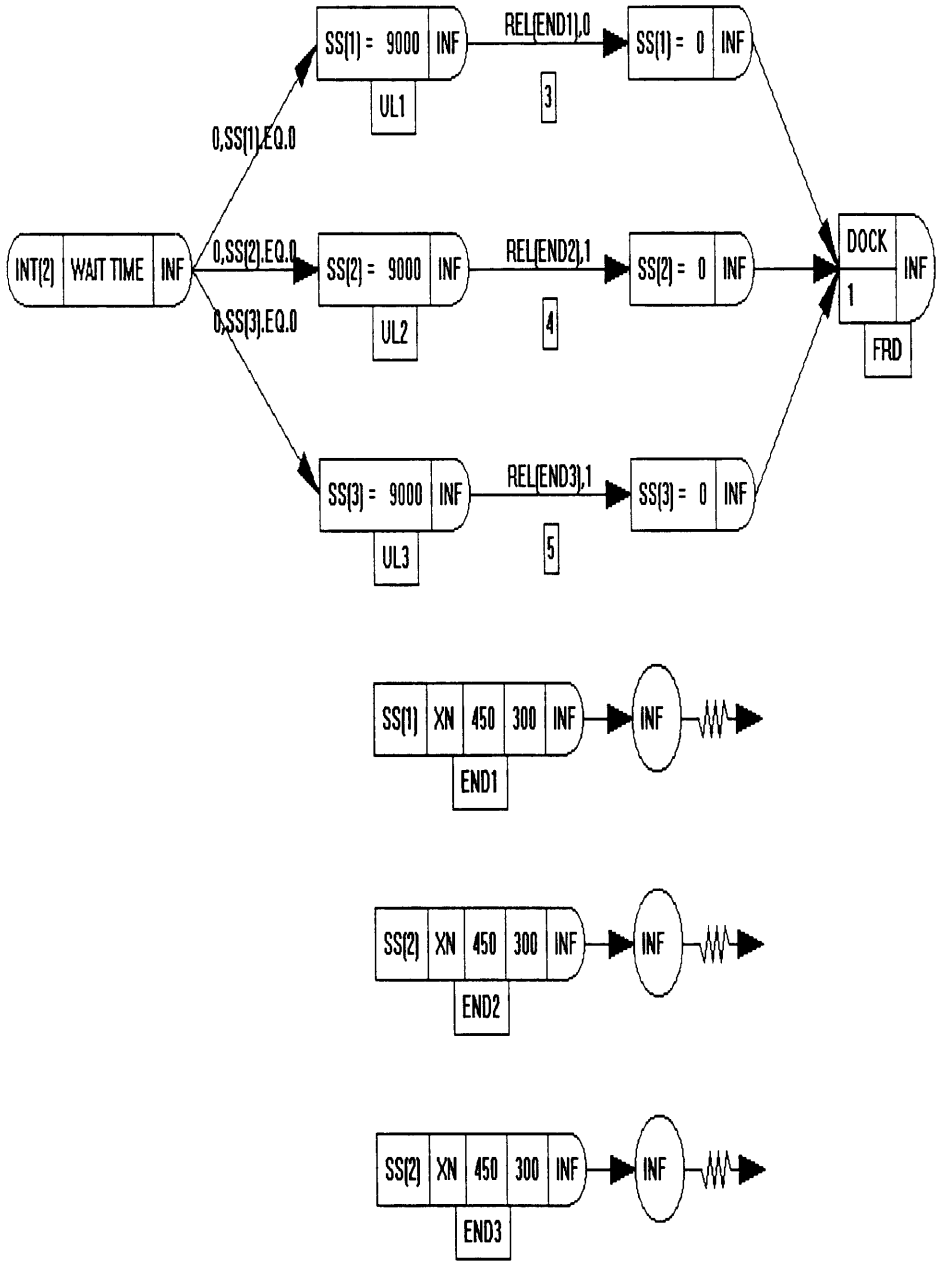


Figure 16 The ORATB Simulation Model - Floating Dock Operation

Table 38 The ORATB Simulation Model Input Statements - Floating Dock Operation

```
**********************************************************************
GEN,MINGQI,ORATB FLDK,11/18/96,1;
LIMITS,1,2,100;
TIMST,XX(3),STML INPUT AVAIL;
CONT,0,4,.0025,.25,.25;
RECORD,TNOW,DAYS,O,P,.25,200,250;
VAR,SS(1),1,BARGE 1 LEVEL,0,9000;
VAR,SS(2),1,BARGE 2 LEVEL,0,9000;
VAR,SS(3),1,BARGE 3 LEVEL,0,9000;
VAR,SS(4),S,YARD LEVEL,0,120000;
TIMST,SS(4),YARD LEVEL;
INTLC,SS(4)=60000,XX(2)=1,XX(3)=1,XX(1)=0;
NETWORK;
;
;BARGE FLOW SUBPROCESS
;---------------------
      RESOURCE/DOCK(3),1;
      CREATE,.5,0,,15;
NB    ASSIGN,ATRIB(1)=TNOW;
      ACT/1,UNFRM(2.9,3.1);
      GOON;
      ACT/2,RNORM(5.,1.5);
      ASSIGN,ATRIB(2)=TNOW;
      AWAIT,DOCK;
      COLCT,INT(2),WAITING TIME,,1;
      ACT,,SS(1).EQ.0,UL1;
      ACT,,SS(2).EQ.0,UL2;
      ACT,,SS(3).EQ.0,UL3;
UL1   ASSIGN,SS(1)=9000;
      ACT/3,REL(END1);
      ASSIGN,SS(1)=0;
```

Table 38 (continued)

```
      ACT,,,FRD;
UL2  ASSIGN,SS(2)=9000;
     ACT/4,REL(END2);
     ASSIGN,SS(2)=0;
     ACT,,,FRD;
UL3  ASSIGN,SS(3)=9000;
     ACT/5,REL(END3);
     ASSIGN,SS(1)=0;
     ACT,,,FRD;
FRD  FREE,DOCK;
     ACT/6,RNORM(4.,1.);
     COLCT,INT(1),TRIP TIME;
     ACT,,,NB;
;
;SHIFT START UP/SHUT DOWN SUBPROCESS
;-----------------------------------
     CREATE,1,.25;
     ASSIGN,XX(1)=1;
     ACT,.3333;
     ASSIGN,XX(1)=0;
     TERM;
;
;STATE EVENT SUBPROCESSES
;------------------------
END1 DETECT,SS(1),XN,450,300;
     TERM;
END2 DETECT,SS(2),XN,450,300;
     TERM;
END3 DETECT,SS(3),XN,450,300;
     TERM;
     DETECT,SS(4),XN,300,300;
     ASSIGN,XX(3)=0;
     TERM;
```

Table 38 (continued)

```
         DETECT,SS(4),XP,3000,300;
         ASSIGN,XX(3)=1;
         TERM;
         DETECT,SS(4),XP,120000,300;
         ASSIGN,XX(2)=0;
         TERM;
         DETECT,SS(4),XN,96000,300;
         ASSIGN,XX(2)=1;
         TERM;
         END;
;
INITIALIZE,0,365;
MONTR,CLEAR,65;
MONTR,SUMRY,165;
MONTR,CLEAR,165;
FIN;
************************************************************************
```

Table 39 The ORATB Simulation Output Summary Report - Floating Dock Operation

```
**********************************************************************

              SIMULATION PROJECT  ORATB FLDK              BY  MING QI

              DATE  11/18/1996                            RUN NUMBER  1 OF 1

              CURRENT TIME  .3650E+03

              STATISTICAL ARRAYS CLEARED AT TIME  .1650E+03


                   **STATISTICS FOR VARIABLES BASED ON OBSERVATION**

                    MEAN      STANDARD     COEFF. OF    MINIMUM     MAXIMUM    NUMBER OF
                    VALUE     DEVIATION    VARIATION    VALUE       VALUE      OBSERVATIONS

WAITING TIME      .4271E+00   .6314E+00    .1479E+01   .0000E+00   .3063E+01     209
TRIP TIME         .1438E+02   .1872E+01    .1302E+00   .9582E+01   .1947E+02     208


                   **STATISTICS FOR TIME-PERSISTENT VARIABLES**

                     MEAN      STANDARD     MINIMUM     MAXIMUM     TIME        CURRENT
                     VALUE     DEVIATION    VALUE       VALUE       INTERVAL    VALUE

STML INPUT AVAIL .9977E+00   .4743E-01    .0000E+00   .1000E+01   .2000E+03   .1000E+01
YARD LEVEL        .3620E+03   .1185E+03    .4963E+01   .6757E+03   .2000E+03   .2244E+03


                               **FILE STATISTICS**

FILE              ASSOC NODE   AVERAGE    STANDARD     MAXIMUM    CURRENT    AVERAGE
NUMBER            LABEL/TYPE   LENGTH     DEVIATION    LENGTH     LENGTH     WAITING TIME

  1               AWAIT           .4463      .7963        4          0          .4271
  2               CALENDAR      13.5845     1.5007       19         15          .9985


                          **REGULAR ACTIVITY STATISTICS**

ACTIVITY          AVERAGE       STANDARD     MAXIMUM        CURRENT        ENTITY
INDEX/LABEL       UTILIZATION   DEVIATION    UTILIZATION    UTILIZATION    COUNT
1 LOADING           3.1274        1.2623         7              3            209
2 TO WUHAN          5.0379        1.5035        10              7            209
3 UNLOADING         0.0873        0.3944         1              0             74
4 UNLOADING         0.7987        0.4009         1              0             72
```

Table 39 (continued)

```
5 UNLOADING         0.6964        0.4598          1              1            64

6 RETURN TRIP       4.0860        1.3325          8              4           208


                              **RESOURCE STATISTICS**


RESOURCE     RESOURCE    CURRENT    AVERAGE        STANDARD     MAXIMUM        CURRENT
NUMBER       LABEL       CAPACITY   UTILIZATION    DEVIATION    UTILIZATION    UTILIZATION

   1         DOCK           3        2.3025         0.8358          3              1


RESOURCE     RESOURCE    CURRENT      AVERAGE      MINIMUM      MAXIMUM
NUMBER       LABEL       AVAILABLE    AVAILABLE    AVAILABLE    AVAILABLE

   1         DOCK           2          0.6975         0            3


                              **STATE AND DERIVATIVE VARIABLES**


                               (I)          SS(I)           DD(I)

                                1         .0000E+00       .0000E+00

                                2         .0000E+00       .0000E+00

                                3         .1667E+02       .0000E+00

                                4         .2244E+03       .0000E+00
*************************************************************************
```

the average three days for loading and nine days for traveling from the total trip time yields an expected time spent in the dock of 1.61 days. Thus, unloading over a two-day period should be anticipated. If this is the case, then we only need ten tugs at the maximum for the 15 ORATB systems. In other words, we can use barges waiting to unload as temporary storage facilities. This means that we can save the investment and operating costs of five tugs, the investment costs to build a permanent dock, as well as the investment costs to build a large storage yard at the costs of building three floating docks and a small storage yard.

Additional analyses with respect to trade-off between costs and savings, similar to the above one, can be easily achieved from the minor changes of this established ORATB simulation model.

The whole purpose of this evaluation process is to provide feedback to the optimization model, so that acquisition strategy can be improved. At the same time, the simulation model is used for sensitivity analysis by testing alternative scenarios. These two important applications of this simulation model are discussed in detail in the following section.

4.2.3. Integrated Model Application: Results And Evaluation

The essence of the integrated optimization model is the adaptive iterations. The adaptation is always based on the performance evaluation in the simulation model. As we pointed out before, the simulation model is established with local specific constraints. In general, local constraints include detailed site-specific conditions about berthing facilities, loading and unloading facilities, and navigation channels.

An iteration occurs if any inferior performance is identified by the simulation model. For example, we may find that the navigation channel is too shallow for barge type BG3 at the Port of Ma'anshan. In that case, we can prevent the access of barge BG3 to the port by adding a constraint of $\sum_{t=1}^{T} N^{t}_{i,Ma'anshan,BG3} = 0$ to the optimization model. Then, we run the revised optimization and obtain a new set of tug/barge acquisition strategy with a new optimum objective value of $TAAC_{new}$. In most cases, this new value should be higher than the old one, $TAAC_{old}$. Then, we can use the difference

$$DD = TAAC_{new} - TAAC_{old},$$

as the investment guideline to dredge navigation channels at the Port of Ma'anshan. If *DD* is larger than the cost of dredging, we should go ahead for the dredging project. Otherwise, we should adopt the new acquisition strategy. However, we must be very cautious for the cost comparison, because the new acquisition strategy has a different impact on operation not only in Ma'anshan but also in all other ports.

An iteration can also occur if we find some resource utilization levels are too low or too high. For example, in order to prevent excessive undertime for tugs of size TG3, the following constraint might be added to the optimization model:

$$\sum_{t=1}^{T} M^{t}_{i,k,TG3} - T(0.75)\, M^{*}_{i,k,TG3} \geq 0,$$

where $M^{*}_{i,k,TG3}$ is the number of tugs recommended by the previous run of the optimization model. This constraint would then require the average utilization of tugs of size TG3 to be at least 75% over the whole T time period. Again, a new set of acquisition strategy is obtained from the optimization model.

An iteration may occur when cost parameters are modified through the evaluation of local conditions in the simulation model. For example, we may find that the existing berthing facilities at Ningbo are capable of handling all the traffic generated by the recommended acquisition strategy. In this case, we can change the cost parameters associated with berthing facilities in Ningbo to be zeroes, and run the optimization model again. Such modification can also be in the opposite direction, when we find that berthing facilities at Nanjing are not sufficient for the traffic. Then we increase the cost parameters and run the optimization model again. By doing so, we can compare the costs and benefits of adding more berthing facilities at the Port of Nanjing.

Adding new constraints and changing cost parameters to the problem allows decision makers to explore the cost sensitivity to the proposed changes. The adaptive iterations of optimization and simulation stop when there is no proposed change that can improve the acquisition strategy.

In a linear programming model, most of this information is provided directly by the shadow prices associated with the original model constraints. Our optimization model, however, is of a linear mixed integer programming type, which does not generate similar shadow price information. On the other hand, although there is no integer variables involved, we cannot use the shadow price for the sensitivity analysis for the IAIOS model. This is because once constraints are changed and/or added, the optimization model becomes different from the original model. Therefore, we conduct the sensitivity analysis for the IAIOS model through simulation modeling.

4.3. Parametric Cost Analysis Between The Current Iron Ore Transport And The ORATB System

In this section, we present a simple parametric cost analysis between the current iron ore transport and the ORATB system. The purpose of this analysis is to show the potential cost savings to transport iron ore for the Wuhan Iron and Steel Mill (WISM) using the proposed ORATB transport system.

In 1993, the WISM imported roughly 5.5 million tons of iron ore from Australia by using the following two schemes to transport 3.0 and 2.5 million tons, respectively.

- Scheme A: Australia to Ningbo using 100,000 dwt class ore carrier; Ningbo to Nantong using 20,000 dwt class ships; and Nantong to Wuhan using 5,000 dwt class barges.
- Scheme B: Australia to Nantong using 20,000 dwt class ships; and Nantong to Wuhan using 5,000 dwt class barges.

The breakdown of unit transport costs in 1993 by these two alternative schemes is as follows.

Activity	Unit cost ($/ton)
Scheme A:	
Australia to Beilun transport costs	6.50
Beilun port handling costs	2.66
Beilun to Nantong transport costs	1.95
Nantong port handling costs	1.13
Nantong to Wuhan transport costs	2.65
Total costs of Scheme A	***14.89***
Scheme B:	
Australia to Nantong transport costs	11.30
Nantong port handling costs	1.13
Nantong to Wuhan transport costs	2.65
Total costs of Scheme B	***15.08***

Although economies of scale in ocean transport between the use of 100,000 dwt class and 20,000 dwt class ships are evident, $6.50/ton *vs.* $11.30/ton, the overall savings in transport costs were only $0.19/ton, representing only 1.28% of the total costs in Scheme A. The main reasons for this rather insignificant savings were the double handling of iron ore at Nantong, the high handling costs at Beilun, and the transport costs from Beilun to Nantong. The overall transshipment costs in Scheme A, after iron ore arrived at Beilun was up to $8.39/ton, accounting for 56.35% of the total transport costs. In Scheme B, however, the total transshipment costs, after iron ore arrived at Nantong, was $3.78, which was only 25.07% of the total transport costs.

Using a 5,000 dwt ORATB system from Beilun to Wuhan directly without transshipment at Nantong provides one alternative, Scheme C. Assuming that the 5,000 dwt ORATB transport system costs the same as the sum of the Beilun-Nantong, Nantong-Wuhan transport costs by the ship and the barge as shown in Scheme A. Then we have the breakdown of the unit costs like the following:

Scheme C:

Australia to Beilun transport costs	$6.50
Beilun port handling costs	$2.66
Beilun to Wuhan transport costs	$1.95+2.65 = $4.60
Total costs of Scheme A	***$13.76***

Thus, the cost savings between Schemes C and A are: $14.89-13.76 = $1.13 per ton, which is the Nantong port handling costs in Scheme A.

Consider improving Scheme C by using 20,000 dwt ORATB system as Scheme D. If we assume that the ORATB transporting costs are 70% of the 20,000 dwt ship's costs from Beilun to Nantong, and 50% of the 5,000 dwt barge's costs from Nantong to Wuhan, then we have the breakdown of the unit costs like the following:

Scheme D:

Australia to Beilun transport costs	$6.50
Beilun port handling costs	$2.66
Beilun to Wuhan transport costs	$1.95x0.7+2.65x0.5 = $2.69
Total costs of Scheme D	***$11.85***

Thus, the cost savings between Schemes D and A are: \$14.89-11.85 = \$3.04 per ton, which represents 20.42% decrease of the total unit costs in Scheme A. With 5.5 million tons per year, the grant total savings using the 20,000 dwt ORATB system sailing directly between Beilun and Wuhan would be \$3.04/ton x 5,500,000 tons/year = \$16.72 million/year. This represents a 20.30% reduction over the total costs of the current transport schemes, which is quite significant and attractive.

4.4. Chapter Summary

In this Chapter, we conducted a case study to introduce the proposed ORATB transport system to the Yangtze region through the IAIOS modeling approach. It starts by reviewing current shipping practices, including shipping demands and management structures along the Yangtze River. Based on the analysis of traffic O-D patterns and the identification of barriers for future development, we introduced the proposed ORATB transport system. The proposed system is aimed at transporting iron ore upstream from the coastal deep sea port located in Ningbo to the Yangtze River ports, and containers downstream from the river ports to Ningbo.

The proposed ORATB transport system establishes a direct link between deep coastal port and shallow river ports. It is believed to be superior to the current shipping practices along the Yangtze River because it eliminates intermediate transshipment by using a low draft and low height clearance system. By eliminating redundant transshipment activities, the ORATB system becomes more cost effective and time efficient. By introducing the ORATB system, shippers spend less for their cargoes for handling at berths and for idling at storage yards, and therefore save time and cost.

The IAIOS model is then applied to this case to provide numerical results on how many different sizes of tugs and barges should be deployed and what are the utilization and performance levels of these resources to satisfy the iron ore shipping demands along the

Yangtze River. During the IAIOS application, the nature of the adaptive iterations is illustrated through several examples. Sensitivity analysis of the IAIOS model is also performed to analyze alternative shipping investment strategies.

Through this application, the IAIOS modeling approach is proved to be an effective tool for decision makers to test the performance of the ORATB operations under a wide range of anticipated conditions, and thus ensures a satisfactory deployment of the transport system.

Finally, in Section 4.3, we presented a simple parametric cost comparison between the current practice and the proposed ORATB system for transporting iron ore. We concluded that there would be a significant savings in overall transport costs by using the propose ORATB system.

Chapter Five

5. SUMMARY AND CONCLUSION

5.1. Summary Of Major Findings

This thesis research develops the IAIOS approach for modeling complex systems subject to resource constraints as well as alternative acquisition and utilization. The IAIOS model is an integration of an optimization model and a simulation model through sequential adaptive iterations. The proposed IAIOS modeling approach is applied to solve the acquisition and utilization problems in designing the proposed ORATB system for transporting iron ore and containers between a coastal port and the Yangtze River ports in China.

The IAIOS modeling approach illustrates how to complement the strengths of the two important modeling techniques: mathematical programming and simulation. In the proposed IAIOS modeling approach, the optimization model is designed for the strategic planning on resource acquisition. The simulation model, on the other hand, is dedicated for the tactical planning on resource utilization. The integrated modeling approach is adaptive and iterative in the sense that the optimization model first suggests the acquisition strategy, which is then tested and evaluated in the simulation model. If the performance of the acquisition strategy is unacceptable, constraints and cost parameters are added and/or modified in the optimization model and the procedure is iterated until no significant improvement in performance can be achieved. Such an adaptive iteration process can also be used to generate recommendations to improve conditions that affect the performance of the system, particularly those site-specific conditions or local constraints.

The IAIOS approach provides a viable way of eliminating the weaknesses inherent in both the optimization and simulation modeling approaches. On one hand, the optimization model cannot incorporate all the detailed local constraints, sequential relationships, as well as uncertainties explicitly, without becoming too large and too complicated to solve. On the other hand, the simulation model does not generate any alternative acquisition schedule, but merely evaluates those presented to it.

One of the distinctions of this IAIOS model is that it deals with system optimization. There is a significant difference between system optimization and ship optimization or fleet optimization. Ship optimization is usually done by the naval architect who varies the characteristics of a fairly well defined ship design to minimize cost or maximize profit in one or a few well defined trade routes. Fleet optimization is more likely done by a ship operator who optimizes the allocation of a number of vessels of various types of designs to various trade routes or missions. The result may be a plan of allocation or an evaluation of a mission or contract. The system optimization in this study embraces a much larger scope. The scope of this system optimization includes not only the vessel and the fleet, but also port operations. This system optimization model is designed for planners who are responsible not only for the performance of a ship, a fleet or a port, but for the performance of a transport system as a whole.

The other distinction of this proposed IAIOS approach is that it solves resource acquisition and resource utilization simultaneously. There are so many lessons to be learned from doing resource acquisition without consideration of its utilization, or *vice verse*. By solving resource acquisition and utilization through the adaptive iterations, the IAIOS approach provides a mechanism for decision makers to interact with the model at different levels during the whole decision making process.

This modeling approach is also distinctive in terms of simplicity and ease of implementation. It simplifies the complex system optimization decision making because it starts from a very simple optimization model. The model becomes comprehensive through a step-by-step adaptive iteration process. With respect to implementation, it is superior because the model takes utilization and performance evaluation to generate adaptive actions.

5.2. Conclusions And Future Research

This proposed IAIOS modeling approach is tested by an application to a real world ORATB transport system. It is applied to solving the acquisition and utilization problems in designing the ORATB system for transporting iron ore and containers between a coastal port and the Yangtze River ports in China. Through this application, the IAIOS modeling approach is proved to be an effective tool for decision makers to test the performance of the ORATB operations under a wide range of anticipated conditions, and thus ensures a satisfactory deployment of the transport system.

Future research efforts to improve this proposed IAIOS modeling approach should be very challenging and rewarding. On of the challenges is to further test the model's ability and efficiency in a more complex and larger system with more detailed real world data. The first step planned is to apply this model comprehensively to the deployment of the ORATB system in the Yangtze River. The case study conducted in this research only serves as a demonstration of the application procedure of the IAIOS modeling approach. In a real world application, more simulation models should be established, and more evaluations and iterations should be required.

The second step planned in improving this modeling approach is to make the adaptive iterations of the optimization model and the simulation model through computer programming. The current adaptive iterations are performed in a heuristic way which may cause lengthy time to complete. Obviously, to program the steps of adaptive iterations, we need more experiments to run the model which in turn require more reliable input information operational data.

Other steps in expanding the model include the integration of a more sophisticated optimization model, such as non-linear and dynamic mathematical programming. However, such effort may derail one of the fundamental objectives of this whole study which is to simplify decision making process. Thus, a sophisticated programming effort shall only be incorporated when it becomes absolutely necessary.

In summary, all future improvement of the IAIOS model are dependent on more detailed and reliable input information on constraints as well as operational parameters. We believe that once this model is built, it can be improved to solve resource acquisition and resource utilization problems of large systems, because the model is based on adaptive iterations.

References

[1] Alexis, G. A. (1982) *A Survey Of Routing And Scheduling Models In Ocean Transportation.* Master Thesis. Massachusetts Institute of Technology.

[2] Almogy, Y. and O. Levin (1970) Parametric Analysis Of A Multi-stage Stochastic Shipping Problem. *Proceedings of the Fifth International Conference on Operations Research.* Tavistock, London. 359-370.

[3] Applegren, L. H. (1969) A Column Generation Algorithm For A Ship Scheduling Problem. *Transportation Science.* 3: 63-68.

[4] Applegren, L. H. (1971) Integer Programming Methods For A Vessel Scheduling Problem. *Transportation Science.* 5: 64-78.

[5] Armstrong, R. J. and A. C. Hax (1974) *A Hierarchical Approach For A Naval Tender Job Shop Design.* Technical Report, No. 101. Operations Research Center, Massachusetts Institute of Technology.

[6] Baker, T. E. (1981) *Interactive Vessel Scheduling At Exxon.* Joint National CORES, TIMS and ORSA paper. Toronto, Canada.

[7] Beaujon, G. J. and M. A. Turnquist (1991) A Model For Fleet Sizing And Vehicle Allocation. *Transportation Science.* 25(1): 19-45.

[8] Bellmore, M. (1968) A Maximum Utility Solution To A Vehicle Constrained Tanker Scheduling Problem. *Naval Research Logistics Quarterly.* 15: 403-411.

[9] Benford, H. (1981) A Simple Approach To Fleet Deployment. *Maritime Policy And Management.* 8(4): 223-228.

[10] Boffey, T. B. *et al* (1979) Two Approaches To Scheduling Container Ships With An Application To The North Atlantic Route. *Operation Research.* 30:413-425.

[11] Bremer, W. M. and A. N. Perakis (1992) An Operational Tanker Scheduling Optimization System: Model Implementation, Results And Possible Extensions. *Maritime Policy And Management.* 19(3): 189-199.

[12] Briskin, L. E. (1965) Selecting Delivery Dates In The Tanker Scheduling Problem. *Management Science.* 12(B): 224-235.

[13] Brooke, A., D. Kendrick, and A. Meeraus (1992) *GAMS: A User's Manual, Release 2.25.* The Scientific Press. San Francisco, CA. USA.

[14] Chang, S. (1988) Estimating Short-Run Costs Of A Barge Company. *Maritime Policy And Management.* 15(1): 67-76.

[15] Claessens, E. M. (1987) Optimization Procedures In Maritime Fleet Management. *Maritime Policy And Management.* 14(1): 27-48.

[16] Dantzig, G. B. and D. R. Fulkerson (1954) Minimizing The Number Of Tankers To Meet A Fixed Schedule. *Naval Research Logistics Quarterly.* 1: 217-222.

[17] Everett, J. L., A. C. Hax, V. A. Lewinson, and D. Nudds (1972) Optimization Of A Fleet Of Large Tankers And Bulkers: A Linear Programming Approach. *Marine Technology.* 9: 430-438.

[18] Fisher, M. L. and M. B. Rosenwein (1985) *An Interactive Optimization System For Bulk Cargo Ship Scheduling.* The Wharton School, University of Pennsylvania.

[19] Flood, M. F. (1954) Application Of Transportation Theory To Scheduling A Military Tanker Fleet. *Operation Research.* 2: 150-162.

[20] Frankel, E. G. (1989) Strategic Planning Applied To Shipping And Ports. *Maritime Policy And Management.* 16(2): 123-132.

[21] Glendinning, I. (1974) *A Mixed Integer Linear Programming Formulation For The Port Linkage Problem.* FTL Memorandum M74-10. Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.

[22] Jaramillo, D. I. and A. N. Perakis (1991) Fleet Deployment Optimization For Liner Shipping Part 2: Implementation And Results. *Marine Policy And Management.* 18(4): 235-262.

[23] Kaskin, J. D. (1979) *Applications Of Ocean-Going Tug/Barges To Military Operations.* MIT Report, No. 80-1. Department of Ocean Engineering, Massachusetts Institute of Technology.

[24] Kydland, F. (1969) *Simulation Of Liner Operations.* Institute for Shipping Research. Bergen.

[25] Laderman, J., L. Gleiberman and J. F. Egan (1965) Vessel Allocation By Linear Programming. *Naval Research Logistics Quarterly.* 12: 315-320.

[26] Li, Z. and Y. Yang (1989) System Analysis Of River-Sea Container Transportation For Overseas Trade In The Yangtze Valley. *Marine Technology.* 26(4): 282-288.

[27] McKay, M. D. and H. O. Hartley (1974) Computerized Scheduling Of Seagoing Tankers. *Naval Research Logistics Quarterly.* 21: 255-264.

[28] Magnanti, T. L. (1981) Combinatorial Optimization And Vehicle Fleet Planning: Perspectives And Prospects. *Networks.* 11: 179-214.

[29] Mousa, R. M., N. M. Rouphail, and F. Azadivar (1990) Integrating Microscopic Simulation And Optimization: Application To Freeway Work Zone Traffic Control. *Transportation Research Record.* 1254: 14-25.

[30] Naslund, B. (1976) Combined Sea And Land Transportation. *Operational Research Quarterly.* 21(1): 47-59.

[31] Nemhauser, G. L. and P. L. Yu (1972) A Problem In Bulk Service Scheduling. *Operations Research.* 20: 813-819.

[32] Oberle, J. (1989) *Oil Terminal Design By Network Simulation.* Master Thesis. Department of Ocean Engineering, Massachusetts Institute of Technology.

[33] O'brien G. G. and R. R. Crane (1959) The Scheduling Of A Barge Line. *Operations Research.* 7: 561-570.

[34] Olson, C. A. *et al* (1969) Medium Range Scheduling For Freighter Fleet. *Operations Research.* 17: 565-582.

[35] Perakis, A. N. and W. M. Bremer (1992) An Operational Tanker Scheduling Optimization System: Background, Current Practice And Model Formulation. *Maritime Policy And Management.* 19(3): 177-187.

[36] Perakis A. N. and D. I. Jaramillo (1991) Fleet Deployment Optimization For Liner Shipping Part 1: Background, Problem Formulation And Solution Approaches. *Maritime Policy And Management*. 18(3): 183-200.

[37] Perakis, A. N. and N. Papadakis (1987) Fleet Deployment Optimization Models: Part 1. *Maritime Policy And Management*. 14(2): 127-144.

[38] Perakis, A. N. (1985) A Second Look At Fleet Deployment. *Maritime Policy And Management*. 12(3): 209-214.

[39] Pritsker, A. A. B. (1990) *Papers, Experiences, Perspectives*. System Publishing Corporation. West Lafayette, IN. USA.

[40] Pritsker, A. A. B., C. E. Sigal, and R. D. J. Hammesfahr (1989) *SLAM II: Network Models For Decision Support*. Prentice-Hall, Inc. Englewood, NJ. USA.

[41] Pritsker, A. A. B. (1986) *Introduction To Simulation And SLAM II*. System Publishing Corporation. West Lafayette, IN. USA.

[42] Rana, K. (1985) *Routing And Scheduling Container Ships Using Lagrangean Relaxation And Decomposition*. Ph.D. Dissertation, University of Waterloo.

[43] Rana, K. and R. G. Vickson (1988) A Model And Solution Algorithm For Optimal Routing Of A Time-Chartered Containership. *Transportation Science*. 22(2): 83-95.

[44] Rao, M. R. and S. Zioints (1968) Allocation Of Transportation Units To Alternative Trips - A Column Generation Scheme With Out-Of-Kilter Sub-problems. *Operations Research*. 16: 52-63.

[45] Ronen, D (1979) *Scheduling Of Vessels For Shipment Of Bulk And Semi-bulk Commodities Originating In A Single Area*. Ph.D. Dissertation. The Ohio State University.

[46] Ronen, D. (1982) A Review Of Cargo Ship Routing And Scheduling Models. *Proceedings of a Symposium on a Cargo Ship Routing and Scheduling*. Washington, DC.

[47] Ronen, D. (1986) Short-Term Scheduling Of Vessels For Shipping Bulk Or Semi-Bulk Commodities Originating In A Single Area. *Operations Research*. 34(1): 164-173.

[48] Rosenblatt, M. J., Y. Roll and V. Zyser (1993) A Combined Optimization And Simulation Approach For Designing Automated Storage/Retrieval Systems. *IIE Transactions*. 25(1): 40-50.

[49] Safizadeh, M. H. (1990) Optimization In Simulation: Current Issues And The Future Outlook. *Naval Research Logistics*. 37: 807-825.

[50] Scott, K. L. and B. W. Douglas (1981) *A Model-Based Decision Support System For Planning Scheduling Oceanborne Transportation.* Joint National CORES, TIMS and ORSA paper. Toronto, Canada.

[51] Winston, W. L. (1990) *Operations Research: Applications And Algorithms.* PWS-Kent Publishing Company. Boston, MA. USA.

[52] Worzel, K. J., C. Vassiadou-Zeniou, and S. A. Zenios (1994) Integrated Simulation And Optimization Models For Tracking Indices Of Fixed-Income Securities. *Operations Research.* 42(2): 223-233.

[53] The American Waterways Operations, Inc. (1973) *Big Load Afloat: US Domestic Water Transportation Resources.* The American Waterways Operations, Inc. Washington, DC. USA.

[54] Yangtze Economic Zone Comprehensive Transport Study Group. (1992) *Yangtze Economic Zone Transport Study.* The World Bank. Washington, DC. USA.