

AN ABSTRACT OF THE DISSERTATION OF

Doug Woon Cho for the degree of Doctor of Philosophy in the  
Department of Industrial Engineering presented on July 9, 1981.

Title: Development of a Methodology for Containership Load  
Planning

Abstract approved: Redacted for Privacy  
Dr. Tom M. West

Ship turnaround time at a marine containerport is an important measure of the port's efficiency and attractiveness. Both the speed and quality of load planning affect the length of turnaround time considerably. Past attempts at computerizing the load planning function have been unsatisfactory and have not reflected the port's objective of optimally utilizing its facilities and equipment. The computerized model developed through this study generates optimum feasible load plans that minimize yard material handling of containers while meeting all ship stability and loading restrictions. The model is particularly addressed to containerports with trans- tainers as prime yard movers.

Minimization of yard handling is accomplished through formulation of the transtainer routing problem as a dynamic programming problem. The set of shipboard container-cell assignments is then checked against the constraints. Pairwise exchanges of assignments are made to meet any violated constraint so that the yard handling

increase is minimized. The model has been tested using data for five actual ship loadings at Port of Portland. It yielded feasible load plans that were better than those manually generated showing an average 3.2 percent less material handling requirements. The model can be easily implemented at containerports with medium-to-large computers with some adjustments. The success of this study not only provides a way of increasing port productivity, but also makes available an important software component for a port automation system.

© Copyright by Doug Woon Cho

July 9, 1981

All Rights Reserved

Development of a Methodology for  
Containership Load Planning

by

Doug Woon Cho

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of  
the requirements for the  
degree of

Doctor of Philosophy

Completed July 9, 1981

Commencement June 1982

APPROVED:

Redacted for Privacy

---

Associate Professor of Industrial Engineering  
in charge of major

Redacted for Privacy

---

Head, Department of Industrial Engineering

Redacted for Privacy

---

Dean of Graduate School

Date thesis is presented July 9, 1981

Typed by Donna Lee Norvell-Race for Doug Woon Cho

## ACKNOWLEDGMENTS

Many people graciously rendered support and guidance to me throughout the period of my study at Oregon State University, of which fruit is this dissertation.

I must express the greatest feeling of gratitude to my good friend and teacher, Dr. Tom West, for his patient and thoughtful guidance of this research, along with instructions in varied areas of our profession. Dr. Jim Riggs, Head of Industrial and General Engineering Department and a renowned educator, kindly provided the financial support in the form of a teaching position. Dr. Ed McDowell granted a support from a Sea Grant research project from which this study originated. Members of my doctoral committee, Drs. Dave Butler, Dave Thomas, and Terry Brown, along with Jim Riggs and Tom West, have been outstanding teachers and sources of constant encouragement.

The Port of Portland opened its facilities at its Fulton Terminal and provided much cooperation for the conduct of this study.

O.S.U. Milne Computer Center granted profuse amount of computer resources for both model development and testing. Its consulting team rendered a great service at clearing an innumerable number of problems usually associated with programs of such complexity and magnitude as of COMCL.

The Korean Navy, in which the author serves as an officer, kindly granted a three-year leave of absence to enable my study in Oregon. My loving wife Nahni's devoted support and prodding must not go unnoticed. She will be rewarded with more abundant love from me.

Finally, my heartfelt thanks go to many dear friends in Oregon who have made our time in this beautiful state both enjoyable and meaningful.

## TABLE OF CONTENTS

| <u>Chapter</u> |  | <u>Page</u> |
|----------------|--|-------------|
| 1              | INTRODUCTION .....   | 1           |
|                | 1.1 Background .....   | 1           |
|                | 1.2 Purpose and Scope of the Study .....                                   | 2           |
|                | 1.3 Method of Approach .....   | 3           |
| 2              | THE MARINE CONTAINER TRANSPORTATION SYSTEM .....                           | 6           |
|                | 2.1 The Container .....  | 6           |
|                | 2.2 Containership .....  | 9           |
|                | 2.2.1 Types and structure .....  | 9           |
|                | 2.2.2 Load restrictions .....  | 11          |
|                | 2.2.3 Ship stability .....   | 13          |
|                | 2.3 The Containerport .....  | 22          |
|                | 2.3.1 Its role and design considerations ....                              | 22          |
|                | 2.3.2 Organization .....   | 24          |
|                | 2.3.3 Material flow .....  | 28          |
|                | 2.3.4 Information system .....   | 31          |
|                | 2.3.5 Port Efficiency .....  | 36          |
| 3              | CONTAINERSHIP LOAD PLANNING PROBLEM .....                                  | 38          |
|                | 3.1 Problem Description .....  | 38          |
|                | 3.2 Present System of Load Planning .....                                  | 41          |
|                | 3.3 Desirable Properties of a CLP Program .....                            | 43          |
|                | 3.4 Existing Models and Past Attempts .....                                | 46          |
| 4              | APPROACHES TO THE PROBLEM .....  | 51          |
|                | 4.1 Formulation as an Integer Program .....                                | 51          |
|                | 4.1.1 A general model .....  | 51          |
|                | 4.1.2 A simplified problem .....   | 66          |
|                | 4.1.3 Experience with the simplified problem.                              | 66          |
|                | 4.2 Other Formulations .....   | 67          |
|                | 4.3 Dynamic Programming Formulation .....                                  | 71          |
|                | 4.3.1 Transtainer routing problem .....                                    | 71          |
|                | 4.3.2 Simplification for manageability .....                               | 74          |
| 5              | COMPUTERIZED OPTIMIZATION MODEL FOR CONTAINERSHIP<br>LOADING (COMCL) ..... | 79          |
|                | 5.1 Philosophy Behind the Model .....                                      | 79          |
|                | 5.2 General Structure of COMCL .....                                       | 80          |
|                | 5.3 Input Files .....  | 83          |

| <u>Chapter</u> |   | <u>Page</u> |
|----------------|---|-------------|
|                | 5.3.1 File CTFIL and container list .....                           | 83          |
|                | 5.3.2 File YLOUT and yard structure .....                           | 84          |
|                | 5.3.3 File FSHIP and ship structure .....                           | 87          |
|                | 5.3.4 File GPLAN structure .....                                    | 87          |
|                | 5.4 Program CLP Organization .....                                  | 89          |
|                | 5.4.1 Program CLP .....   | 89          |
|                | 5.4.2 Subroutine DYNAMIC .....                                      | 91          |
|                | 5.4.3 Subroutine DISCOMP .....                                      | 99          |
|                | 5.4.4 Subroutine ASSGN .....  | 100         |
|                | 5.4.5 Subroutine FCHECK .....                                       | 100         |
|                | 5.4.6 Subroutine LPOUT .....  | 103         |
|                | 5.4.7 Subroutine REMEDY .....                                       | 103         |
|                | 5.5 Simulation of Subroutine DYNAMIC .....                          | 107         |
|                | 5.6 Analogy with the Dual Simplex Method .....                      | 113         |
| 6              | TESTING OF COMCL .....  | 118         |
|                | 6.1 Test Environments .....   | 118         |
|                | 6.2 Test Results .....  | 122         |
|                | 6.3 Discussion of Results .....                                     | 128         |
| 7              | CONCLUSIONS AND RECOMMENDATIONS .....                               | 131         |
|                | 7.1 Usefulness and Contributions of the Model ....                  | 131         |
|                | 7.2 Restrictions in the Model .....                                 | 132         |
|                | 7.3 Possible Areas for Future Research .....                        | 134         |
|                | BIBLIOGRAPHY .....  | 138         |
|                | APPENDICES  |             |
|                | A Summary of Guidelines for Loading Con-<br>tainers Shipboard ..... | 144         |
|                | B COMCL Flowcharts .....  | 148         |
|                | C COMCL Input Data File Specifications .....                        | 172         |
|                | D COMCL Program Listing .....                                       | 180         |
|                | E A Sample Output of COMCL Run .....                                | 230         |



## LIST OF FIGURES

| <u>Figure</u> |   | <u>Page</u> |
|---------------|---|-------------|
| 2.1           | Containership with cellular structure                     | 10          |
| 2.2           | Typical lashing systems for on-deck storage               | 12          |
| 2.3           | Effect of inclining moments on ship stability             | 15          |
| 2.4           | Location of metacenter                                    | 16          |
| 2.5           | Relationship between metacenter and center of gravity     | 17          |
| 2.6           | Permissible range of containership GM                     | 18          |
| 2.7           | Envelope of permissible ship drafts                       | 21          |
| 2.8           | Containership as a link between two transport nodes       | 22          |
| 2.9           | Network representation of container transportation system | 23          |
| 2.10          | Schematic layout of a containerport                       | 26          |
| 2.11          | Major material handling alternatives at containerports    | 27          |
| 2.12          | Yardside container flow                                   | 29          |
| 2.13          | Dockside container flow                                   | 30          |
| 2.14          | Suggested computer network for West Coast ports           | 32          |
| 2.15          | Hamburg port automation system                            | 33          |
| 2.16          | Proposed computer system at a containerport               | 35          |
| 3.1           | Ship schematic diagram                                    | 39          |
| 3.2           | Manual load planning procedure                            | 42          |
| 3.3           | Bay storage plan (under-deck bay)                         | 44          |
| 3.4           | Bay storage plan (on-deck bays)                           | 45          |
| 4.1           | Yard storage pattern in simplified problem                | 59          |
| 4.2           | Hypothetical ship structure in example problem            | 61          |
| 4.3           | An example of generalized network                         | 70          |

| <u>Figure</u> |  | <u>Page</u> |
|---------------|--|-------------|
| 4.4           | A multi-stage network  | 72          |
| 4.5           | Types of systems with respect to complexity and organization | 75          |
| 5.1           | COMCL model structure  | 82          |
| 5.2           | Relative dispositions of yard sections                       | 86          |
| 5.3           | Program CLP flowchart  | 90          |
| 5.4           | Arc-node relationship in dynamic program                     | 94          |
| 5.5           | Flowchart for subroutine DYNAMIC simulation                  | 109         |
| 5.6           | Working of dual simplex algorithm                            | 114         |
| 6.1           | COMCL test procedure   | 120         |

## LIST OF TABLES

| <u>Table</u> |   | <u>Page</u> |
|--------------|---|-------------|
| 1            | Typical Container Dimensions                | 8           |
| 2            | Container Data for Simplified Problem       | 60          |
| 3            | Ship Cell Data for Simplified Problem       | 62          |
| 4            | Results of Subroutine DYNAMIC Simulation    | 111         |
| 5            | Summary of Test #1 Results                  | 123         |
| 6            | Summary of Test #2 Results                  | 124         |
| 7            | Summary of Test #3 Results                  | 125         |
| 8            | Summary of Test #4 Results                  | 126         |
| 9            | Summary of Test #5 Results                  | 127         |
| 10           | Comparison of COMCL Plan versus Manual Plan | 128         |
| 11           | MSTAGE and Number of Containers to Pick     | 128         |

## LIST OF ABBREVIATIONS

| <u>Code</u> | <u>Meaning</u>   | <u>Defined<br/>Page</u> |
|-------------|--|-------------------------|
| AR          | After-REMEDY figures   | 124                     |
| BR          | Before-REMEDY figures  | 124                     |
| CG          | Center of gravity  | 13                      |
| CLP         | Containership load planning                                  | 38                      |
| COMCL       | Computerized Optimization Model for<br>Containership Loading | 79                      |
| D           | Draft at midship at a given displacement                     | 101                     |
| Da          | Draft aft  | 102                     |
| Df          | Draft forward  | 102                     |
| D/I         | Distance/improvement ratio                                   | 104                     |
| DOV         | Degree of violation  | 103                     |
| GLP         | General loading plan   | 38                      |
| FS          | Free-surface effect contribution in<br>vertical moment       | 101                     |
| GN          | Generalized network  | 68                      |
| IAYS        | Array name for relative yard section<br>dispositions         | 84                      |
| IP          | Integer programming  | 49                      |
| KM          | Keel-to-metacenter height                                    | 16                      |
| LBP         | Length of ship between perpendiculars                        | 20                      |
| LCB         | Longitudinal center of buoyancy                              | 101                     |
| LCF         | Longitudinal center of flotation                             | 20                      |
| LM          | Longitudinal moment  | 101                     |
| LP          | Linear programming   | 67                      |
| LPAT        | Load pattern descriptor array                                | 88                      |
| MSECT       | Number of possessing yard sections                           | 108                     |
| MSTAGE      | Number of stages in dynamic program problem                  | 108                     |
| MT          | Metric ton (=1,000 Kg)                                       | 14                      |
| MTC         | Moment to change trim by 1 Cm                                | 20                      |

| <u>Code</u> | <u>Meaning</u>  | <u>Defined<br/>Page</u> |
|-------------|---|-------------------------|
| NER1        | Number of row equivalents for an adjacent section change move by transtainer    | 92                      |
| NER2        | Number of row equivalents for a non-adjacent section change move by transtainer | 92                      |
| NCTLD       | Total number of containers to load in a load set                                | 112                     |
| NRUNS       | Number of simulation runs with a parameter set                                  | 108                     |
| NTGEN       | Total number of solutions generated   | 112                     |
| NTSOL       | Total number of nodes adopted   | 112                     |
| RE          | Row equivalent  | 92                      |
| TMD         | Transtainer move distance by one transtainer                                    | 94                      |
| TTMD        | Total transtainer move distance   | 40                      |
| TR          | Trim  | 20                      |
| TRP         | Transtainer routing problem   | 71                      |
| TRUN        | CPU time per simulation run   | 112                     |
| VM          | Vertical moment   | 101                     |
| W           | Total ship displacement   | 14                      |

# Development of a Methodology for Containership Load Planning

## 1. INTRODUCTION

### 1.1 Background

Containerization has caused a revolutionary change in world shipping in that it allows a considerable increase in rapid and safe handling of cargo as units. More and more ships are exclusively built to carry containers and port facilities designed to handle containers have been built around the world to make use of this concept. The container industry is expected to continue this growth in the future as the benefits of containerization become more widely known. As containers are handled as cargo units, containerports have become major transshipment points in commodity transportation rather than distribution points as older bulk ports used to be.

Containerports, however, require large capital investments for initial construction. Their efficient operation is necessary for that reason alone. Looking at the entire picture of world trade, however, efficient operation of ports is essential for a number of reasons. Over 96 percent of international cargo moves through ports of the world. According to Frankel (1974), port costs total in excess of all costs of international transport from port of entry to port of exit. Also, over 55 percent of all port-related costs is the result of time losses in ship turnaround. If cost of stevedoring, usually paid by shipping lines, is included, total port-related costs become even more significant. This total amounted to more than \$28 billion in 1975 according to Frankel (1974). These figures provide an incentive for shipping lines to use efficient ports and avoid less efficient ports.

Ship turnaround time consists primarily of the following elements:

- Waiting time at anchor.
- Quay mooring and demoorings time.
- Document transfer.
- Unload/load planning activities.
- Unloading and loading operations.

Operations at a containerport can be modeled using techniques of operations research. Although the entire port operation is an extremely complex one, its constituent parts can be separately modeled and optimization of each may be sought. Then, a system synthesis can be done to integrate various models towards overall system optimization.

This study is particularly addressed to containership load planning activity. Unload planning, although it has its complexities, is a far simpler task compared to load planning. Thus, we seek to reduce the last two components of ship turnaround time. Total impact of successfully completing this study is difficult to quantify at this time. However, it is not difficult to see that it could have serious economic impact on world commerce.

## 1.2 Purpose and Scope of the Study

General guidelines for load planning are generally provided by a shipping line representative in the form of a color-coded load schematic. This basically indicates how many of each container group is to be loaded in each bay. This information, along with the container distribution pattern of the yard, is used to construct a work sequence which specifies the bay sequence of loading.

Load schematic, work sequence, and container yard distribution pattern, are taken as input for the model developed through this study. The load planning model is intended to produce optimal feasible load plans; optimal in the sense of achieving the minimum transtainer move distance, and feasible in the sense that all load restrictions as well as ship stability constraints are satisfied. Existing models only attempt to obtain feasible plans. In this regard, the present model embraces the new concept of minimizing the yard material handling. This would provide an important contribution to ship turnaround time minimization.

The model is addressed only to containerports with "overhead crane and yard truck" type of yard handling. However, concepts developed in the model may be used to develop models for other yard configurations.

The developed model is intended to fill the basic requirements of an important component in a port automation system. Here, port automation must be viewed with the objective of reducing ship turnaround times as well as efficient utilization of port facilities. This study suggests a possible configuration of an automated port system and explains where the developed model fits in such a system.

The model, titled COMCL (Computerized Optimization Model for Containership Loading), is aimed at practicality and applicability. A model of great complexity and academic sophistication would serve no purpose if it cannot be used by most ports concerned. Ease of final adjustments on a produced load plan is another consideration included in the modelling effort.

### 1.3 Method of Approach

Past attempts at modelling containership load planning and



existing models have been surveyed. The current practice of manual load planning was also examined. This survey revealed the apparent need for an efficient computerized model. Since container-ship cell assignments constitute 0-1 type decision variables, integer-program-related techniques were first investigated for use as possible model structures. Inherent limitations in the operational size of such models make them impractical as day-to-day solution techniques for load planning.

The problem of obtaining an optimal routing plan for yard transtainers was then constructed as a dynamic programming problem. After some useful aggregation process, a structure of a manageable size was formulated. Thus, a two-stage model incorporating this dynamic program structure was written in FORTRAN IV. An optimal transtainer routing plan with a minimum total transtainer move distance produces a load plan which may or may not be feasible. An efficient heuristic was, then, constructed in order to attain feasibility in any violated constraints in a manner that the sacrifice in material handling is kept at a minimum. The model conducts feasibility checks and stability computations which are currently done by hand at most ports.

The model is divided into two separate programs: The first part prepares raw data into a useful format for the second part. This is necessary because raw container data cannot readily be used by the main program.

It is observed that the structure of COMCL embodies the idea behind Lemke's dual simplex method. Thus, it initially obtains a superoptimal load plan. If it is feasible, the algorithm terminates. Otherwise, iterations involving pairwise exchanges of assignments are conducted towards feasibility. This approach was adopted from the observation that large modern containerhips provide such stable loading platforms that stability constraints may not be critical in most loading situations.

Testing of the model was conducted using actual loading situations of ships that had visited Fulton Terminal, Port of Portland, Oregon. Total transtainer move distances of computer-generated plans and manually generated plans were computed for comparison.

## 2. THE MARINE CONTAINER TRANSPORTATION SYSTEM

The marine container transportation system consists of the following subsystems: container, containership and container-port. The container provides a medium of goods transportation while the containership is identified as the vehicle of container transfer among containerports which are major transshipment points in container flow. Understanding of the marine container transportation system is essential in the analysis of containership load planning.

### 2.1 The Container

The container is the key element in marine container transportation systems. The freight container takes many different forms and a number of shapes. In general, all have in common a rectangular outer shape, a relatively weatherproof outer shell over a strong inner structure that protects the load, and the fact that the contents are handled as a unit load, being transshipped without repacking.

Containers are categorized according to their contents during a phase of their life in the following three basic types:

- a. Dry cargoes,
- b. Liquid cargoes and bulk commodities,
- c. Special cargoes requiring protection from the environment.

Special cargoes can be broken down into two major subcategories: refrigerated cargoes (commonly called "reefers") and dangerous cargoes. Refrigerated containers must be loaded in special areas on a ship where electric outlets are provided to power

refrigeration machinery attached on them. Basically, there are eight major types of dangerous cargoes, each identified by a specific type of label:

- a. Explosives (labeled as such),
- b. Inflammable compressed gas (Red Gas),
- c. Inflammable liquids (Red Label),
- d. Inflammable solids, oxidizing materials (Yellow Label),
- e. Corrosive liquids (White Label),
- f. Non-flammable compressed gas (Green Label),
- g. Poisonous articles (Blue Label),
- h. Hazardous articles (labeled according to contents).

Coast Guard regulations require certain storage restrictions to be applied on dangerous cargoes. Beliech (1974) describes the specific details of cargo types.

Material used to construct containers may be used to categorize containers as follows:

- a. Aluminum containers,
- b. Steel containers,
- c. Plywood/fiberglass containers.

Over 95 percent of existing containers are of the general purpose, box-type containers. Rath (1973) includes a good coverage on container structure details. Variations from these include the following:

- a. Open top containers,
- b. Racks and half-height containers,
- c. Tank-type containers.

Due to lack of container shape/size standards, there are numerous different sizes that are currently used. Typical sizes which dominate the industry are shown in Table 1.

TABLE 1. TYPICAL CONTAINER DIMENSIONS  
(Source: Rath)

| Outside Dimensions       | 40-ft.           | 40-ft.           | 40-ft.           | 40-ft.                | 40-ft.           | 40-ft.                | (AL)                | (FRP)               |
|--------------------------|------------------|------------------|------------------|-----------------------|------------------|-----------------------|---------------------|---------------------|
|                          | Dry Cargo        | Dry Cargo        | Reefer           | Tank                  | Open Top         | Half-High<br>Open Top | 20-ft.<br>Dry Cargo | 20-ft.<br>Dry Cargo |
| Length                   | 40'              | 40'              | 40'              | 40'                   | 40'              | 40'                   | 20'                 | 20'                 |
| Width                    | 8'               | 8'               | 8'               | 8'                    | 8'               | 8'                    | 8'                  | 8'                  |
| Height                   | 8'               | 8'6"             | 8'6"             | 4'3"                  | 8'6"             | 4'3"                  | 8'                  | 8'                  |
| <u>Inside Dimensions</u> |                  |                  |                  |                       |                  |                       |                     |                     |
| Length                   | 39'7"            | 39'6"            | 37'10"           |                       | 39'6"            | 39'6"                 | 19'6"               | 19'7"               |
| Width                    | 7'9"             | 7'9"             | 7'4"             |                       | 6'8"             | 7'9"                  | 7'8"                | 7'10"               |
| Height                   | 7'4"             | 7'10"            | 7'1/2"           |                       | 7'6"             | 3'4"                  | 7'4"                | 7'6"                |
| <u>Door Opening</u>      |                  |                  |                  |                       |                  |                       |                     |                     |
| Width                    | 7'5-3/4"         | 7'6"             | 7'6"             |                       | 7'6"             | 7'6"                  | 7'5-3/4"            | 7'5-3/4"            |
| Height                   | 6'11-3/4"        | 7'5-1/2"         | 7'5-3/4"         |                       | 6'11"            | 3'4"                  | 6'11-3/4"           | 6'11-3/4"           |
| Construction             | aluminum         | aluminum         | aluminum         | st. steel<br>AISI 304 | aluminum         | aluminum              | aluminum            | plastic<br>plywood  |
| Internal cu. cap.        | 2,250<br>cu. ft. | 2,398<br>cu. ft. | 1,988<br>cu. ft. | 6,020<br>gals.        | 2,296<br>cu. ft. | 1,020<br>cu. ft.      | 1,096<br>cu. ft.    | 1,151<br>cu. ft.    |
| Max. load cap.           | 60,260<br>lbs.   | 60,950<br>lbs.   | 50,000<br>lbs.   | 51,600<br>lbs.        | 59,280<br>lbs.   | 45,000<br>lbs.        | 44,800<br>lbs.      | 40,650<br>lbs.      |
| Tare weight              | 6,400<br>lbs.    | 6,940<br>lbs.    | 10,700<br>lbs.   | 9,260<br>lbs.         | 7,750<br>lbs.    | 6,660<br>lbs.         | 4,500<br>lbs.       | 4,500<br>lbs.       |

A great majority is either of 40-foot or 20-foot length. Matson and Sea-Land have used 23.5-foot and 34.6-foot lengths, respectively. But the current trend is one of convergence to these two prevailing sizes. Establishment of world-wise standards is necessary as container industry grows and more ships are built to carry containers.

## 2.2 The Containership

### 2.2.1 Types and structure

Many different types of ships carry containers on board. They may be classified into the following general categories:

- a. Cellular containerships,
- b. Roll-on, roll-off ships,
- c. Barge carriers,
- d. Combination roll-on containerships,
- e. Conventional type, semi-containerships.

In this study, the primary interest is with the first category of cellular containerships. These ships are designed from keel up solely for container transportation. They have fixed cellular construction and the lift-on, lift-off method of handling is used. Some ships offer the flexibility of modifying a cellular structure to a limited degree to accommodate container size changes.

Container cells are grouped into "bays." A bay is either "on-deck" or "under-deck." "Cell guides" facilitate lowering of containers to their proper storage positions and resist horizontal loads exerted by containers as a ship rolls sideways during its voyage. Figure 2.1 illustrates some examples of



cellular containership structures. These can accommodate both 20-foot and 40-foot lengths.

### 2.2.2 Load Restrictions

Containers are loaded from bottom to top of a ship, tier by tier. This creates the requirement that containers going to later ports of call should be stowed under those going to earlier ones. Violation of this condition is referred to as "over-stowage."

Under-deck bays are usually dedicated to a specific container size, either 40 feet or 20 feet in most cases. That is not true with on-deck bays. Most on-deck bays allow mixes of different sizes. In practice, however, a given stack of an on-deck bay has only one-size containers in order to minimize technical problems associated with loading as well as securing containers of different sizes. Reefers can only be stowed in areas of ship with electric outlets, either on-deck or under-deck.

All on-deck containers must be properly secured to the deck before a voyage begins. This activity should be done so that container movements are minimized under ship movements. Figure 2.2 illustrates typical lashing systems used for securing containers on deck.

Total container weight in a stack must not exceed a certain deck strength limit. This restriction often forces some light-weight containers to be selected for on-deck stowage.

Guidelines for loading containers shipboard were developed by the Maritime Administration with the help of the Technical and Research Panel 0-31 of the Society of Naval Architects and Marine Engineers. A summary of the guidelines is listed in Appendix A.



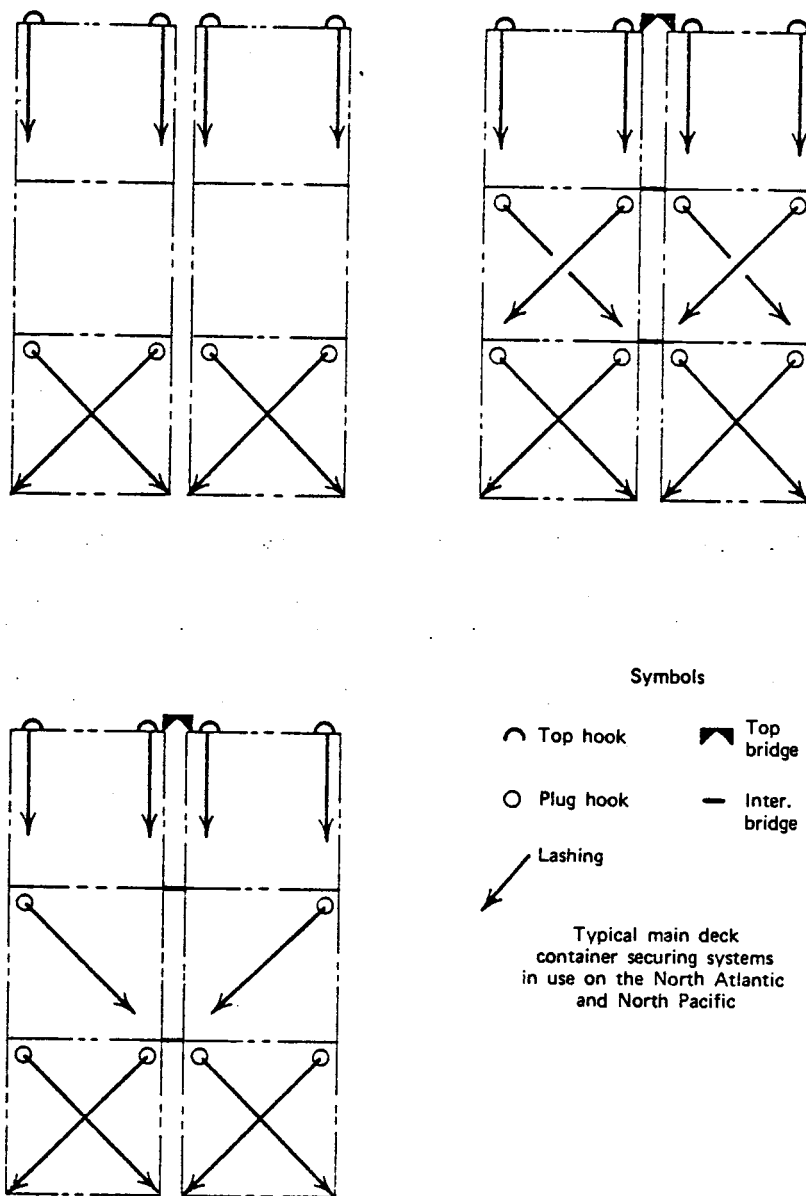


Figure 2.2 Typical lashing systems for on-deck storage  
(Source: Rath)

### 2.2.3 Ship stability

Ship stability theory is a large subject involving hydrodynamics and material science. This section presents a summarized excerpt of ship stability theory only as related to containership load planning. Much of the material in this section came from Bureau of Naval Personnel (1970) and Rawson and Tupper (1976).

#### A. Principles of ship stability

When a solid body is immersed in water, it experiences an upward thrust equal to the weight of the water it displaces. This fact is commonly referred to as the Archimedes' Principle. The weight, or displacement, of a containership consists of a ship's light condition weight, total weight of loaded containers, constant loads and temporary structure, and all liquid stores. This total weight represents the effect of a gravitational force. The ship rests at a waterline level at which the weight of the displaced water is equal to the weight of the ship. At this point, the algebraic sum of all forces acting on the ship is zero. When such a condition exists, we say the ship is "in equilibrium."

We may regard the total force of gravity as a single composite force acting vertically downward through the ship's center of gravity (G). Similarly, the force of buoyancy acts through the center of buoyancy (B), located at the geometric center of the ship's underwater body. When a ship is in equilibrium, G and B lie on the same vertical line.

Since the weight of a ship is equal to the weight of the displaced water, a linear relationship exists between the ship displacement and the volume of the displaced water. Therefore,

it is possible to obtain a plot of draft (waterline levels) versus ship displacement by computing the volume of a ship's underwater body at different waterline levels. This curve is called a "displacement curve."

As draft increases, center of buoyancy rises with respect to the keel (K). Values of  $\overline{KB}$  corresponding to possible values of draft and hence, displacement, form a part of a ship's hydrostatic data. The volume of the watertight portion of a ship above a waterline is known as the ship's "reserve buoyancy." Reserve buoyancy is an important factor in a ship's ability to survive flooding due to a damage. It also contributes to the seaworthiness of a ship in adverse weather conditions.

When a disturbing force such as caused by taking a load on-board exerts an inclining moment on a ship, the shape of the ship's underwater body changes, and this in turn produces a relocation of B. Now, B and G are no longer equal and opposite forces, but they form a couple. This newly formed couple produces either a righting moment or an upsetting moment, depending on the relative locations of B and G, as seen in Figure 2.3. The resulting righting moment has the following magnitude:

$$RM = W \times GZ \quad (2-1)$$

where RM: Righting moment (M-MT)

W: Displacement (MT)

GZ: Righting arm (M)

A ship's metacenter (M) is the intersection point of two successive lines of action of the force of buoyancy as the ship heels through a very small angle. Figure 2.4 depicts this point M where the line of buoyant force at a small angle of keel intersects the original line of buoyant force at rest. The distance from B to M when a ship is on an even keel is called "metacentric

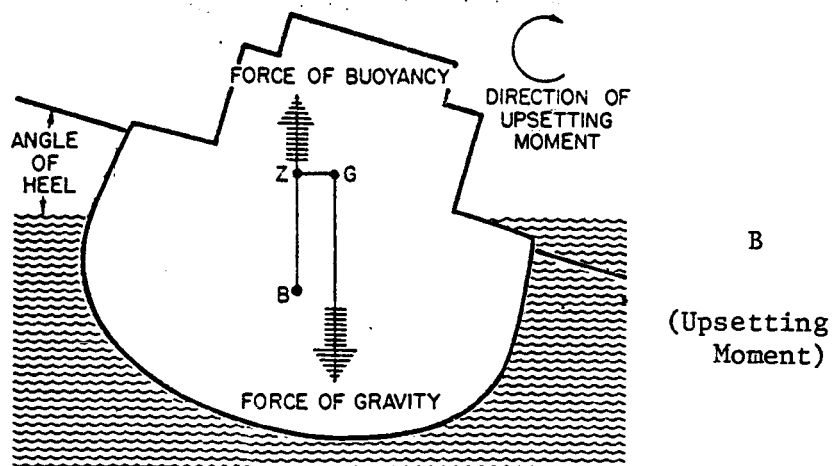
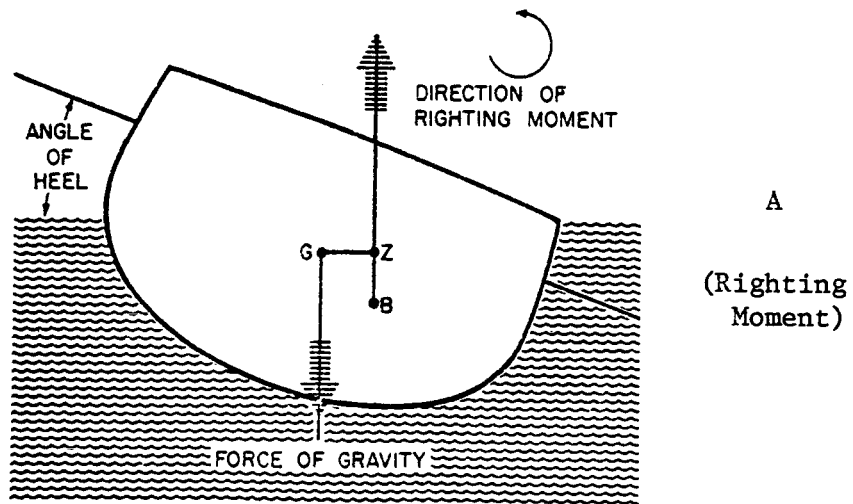


Figure 2.3 Effect of inclining moments on ship stability  
(Source: U.S. Navy NAVPERS 1-788-B)

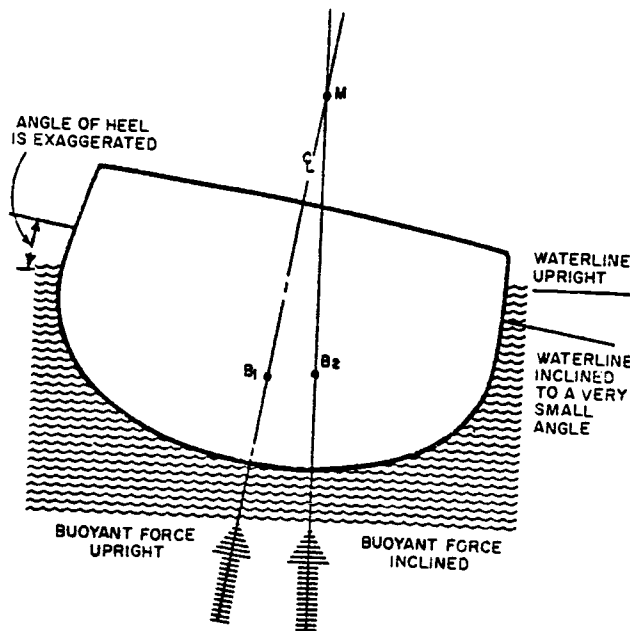


Figure 2.4 Location of metacenter  
(Source: U.S. Navy, NAVPERS 1.788-B)

radius." Also, the distance from  $G$  to  $M$  is known as a ship's "metacentric height ( $GM$ ).". This quantity,  $GM$ , is a very important measure of a ship's righting ability. Figure 2.5 shows a definite relationship between  $GZ$  and  $GM$ , expressed mathematically as follows:

$$GZ = GM \cdot \sin \theta \quad (2-2)$$

where  $\theta$ : angle of heel.

Thus,  $GM$  is a direct measure of  $GZ$ .

If  $GM$  is positive, i.e.,  $M$  is above  $G$ , then righting moments are developed and a ship returns to more stable condition. But, if  $M$  is below  $G$ , upsetting moments are developed and the

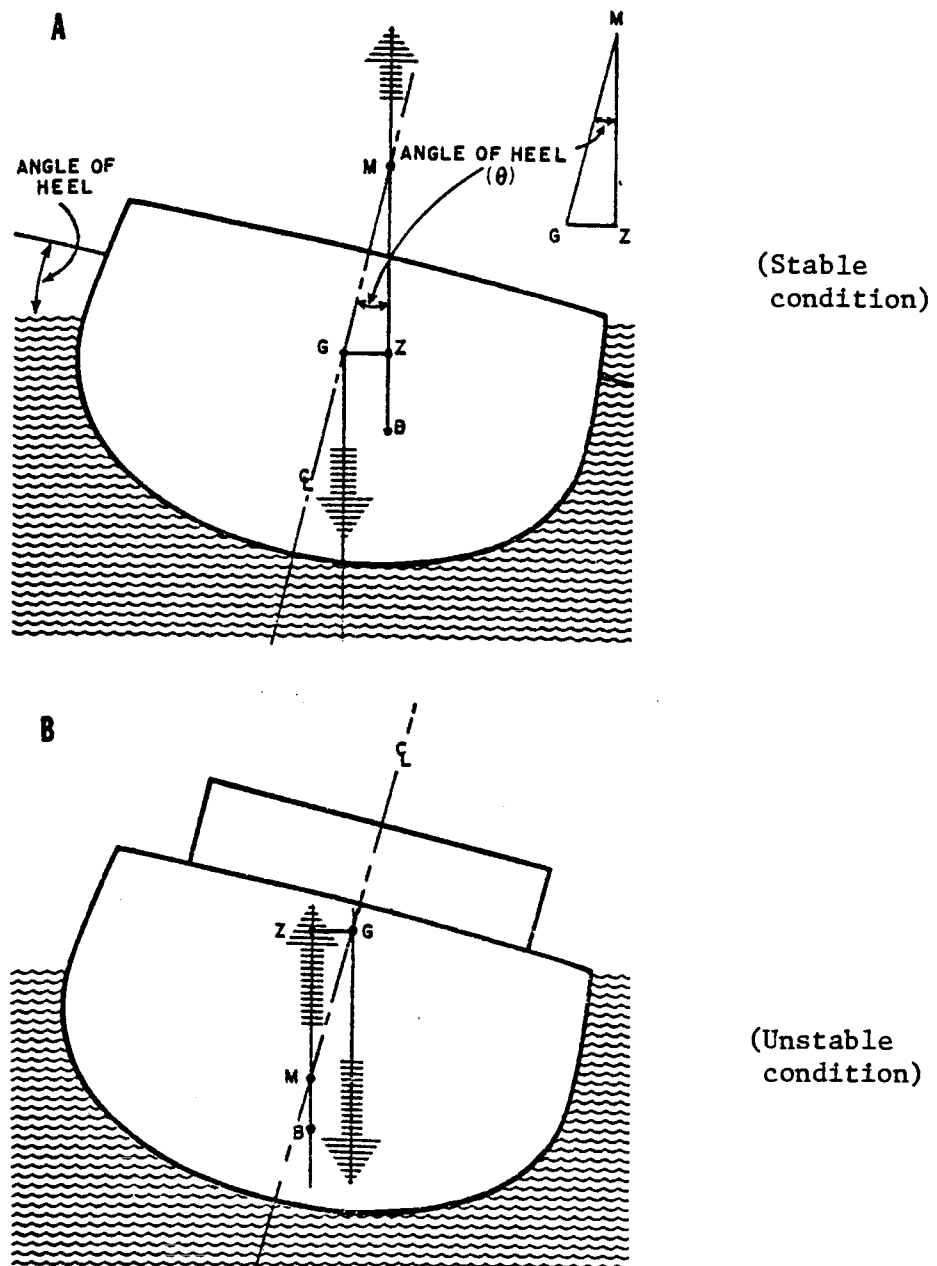


Figure 2.5 Relationship between metacenter and center of gravity  
(Source: U.S. Navy, NAVPERS 1-788-B)

ship is unstable. When GM is large, a ship resists rolling and is said to be "stiff." When GM is small, a ship rolls slowly and is said to be "tender." Large GM is desirable for resistance to flooding effects of a damage. A small GM, however, is sometimes desirable for ability to ride waves easily. A typical range of GM for a containership developed through inclining experiments by design engineers would look like Figure 2.6.

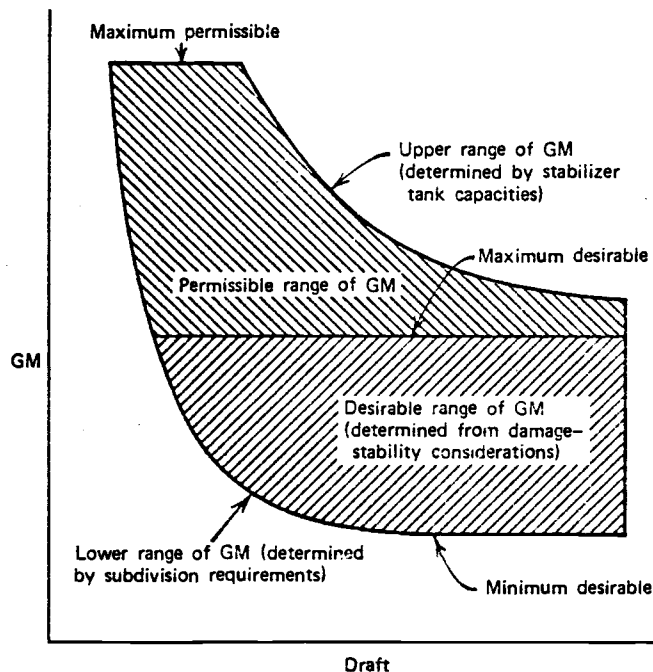


Figure 2.6 Permissible range of container-ship GM  
(Source: Hydronautics, Inc.)

#### B. Effects of Load Changes

If a weight,  $w$ , was added to a ship exactly above  $G$ , then its displacement  $W'$  would be:

$$W' = W + w \quad (2-3)$$

The resulting vertical shift of G can be found by the following:

$$GG_1 = \frac{wz}{W'} \quad (2-4)$$

where  $GG_1$ : Shift of G up or down (M)

$z$ : Vertical distance  $w$  is added above or below G.

Then, the new metacentric height can be found by the following:

$$G_1M_1 = KM_1 - KG_1 \quad (2-5)$$

where  $KM_1$ : New KM

$KG_1$ : New KG.

The horizontal shift in G due to a weight addition can be similarly determined.

#### D. Free surface effect of loose liquid stores

Liquid stores aboard container ships cause vertical and horizontal shift of their metacentric heights. When a liquid body is free to move in a compartment, it creates "free surface effect." Free surface in a ship always causes a reduction in GM with a consequent reduction in stability. The vertical rise of G due to a body of loose water regardless of its stored location due to its free surface effect can be determined by the following:

$$GG_2 = \frac{i}{V} = \frac{b^3 l}{12V} \quad (2-6)$$

where  $i$ : Moment of inertia of the surface of liquid in a compartment about a longitudinal axis through the area of that surface,



- V: Existing volume of displacement of the ship,
- b: Athwartship breadth of the free surface (M),
- l: Fore-and-aft length of the free surface (M).

#### D. Longitudinal stability and effects of trim

When a ship trims, it inclines about an axis through the geometric center of the waterline plane. This point is known as the "center of flotation (LCF)."

Trim (TR) is defined as the difference in draft forward ( $D_f$ ) and draft aft ( $D_a$ ). If the draft forward becomes greater than draft aft, the change in trim is said to be "by the bow," and conversely, the opposite case is "by the stern."

A ship's ability to resist a change in trim is referred to as "longitudinal stability." It is expressed by the moment to change trim by one unit, such as 1 Cm. Longitudinal metacentric height ( $GM'$ ) can be found by the following relationship:

$$GM' = KB + BM' - KG \quad (2-7)$$

where  $BM'$ : Longitudinal metacentric radius.

Moment to change trim by one centimeter (MTC) is now obtained by the following equation:

$$MTC = \frac{GM' \times W}{100 \text{ LBP}} \quad (2-8)$$

where LBP: Length of ship between two perpendiculars.

The change in trim due to relocation of a weight  $w$  by a distance  $t$  can be obtained as follows:

$$\Delta TR = \frac{wt}{MTC} \quad (2-9)$$

The direction of change in trim is the same as that of the weight movement.

Each ship has a permissible range of drafts. Usually trim is restricted to be some positive value. An example of such a range is illustrated in Figure 2.7.

Ship stability limits are some of the important constraints in containership load planning. The knowledge presented in this section provides the basis for the routines for conducting feasibility checks and for making remedial actions to attain feasibility within the computerized model. A ship's hydrostatic data as well as current load conditions constitute an important input data block.

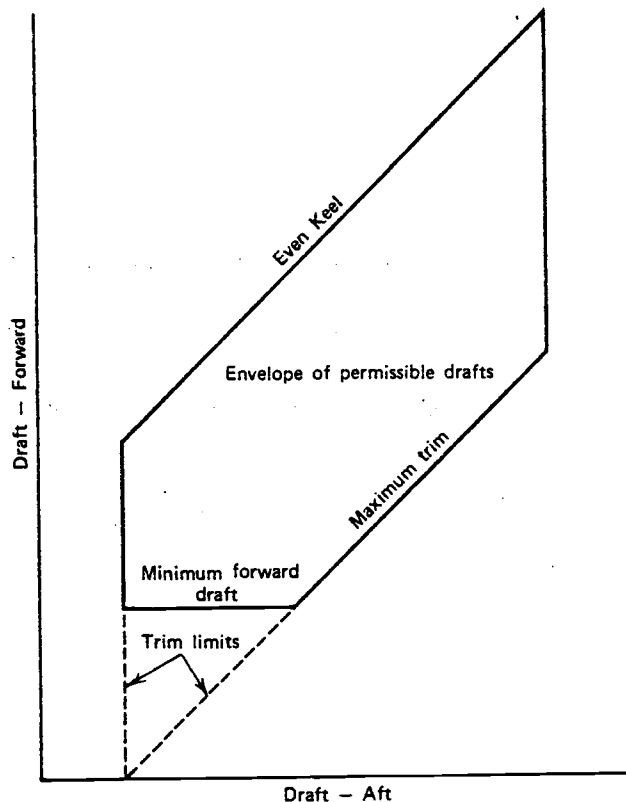


Figure 2.7 Envelope of permissible ship drafts  
(Source: Hydronautics, Inc.)

## 2.3 The Containerport

### 2.3.1 Its role and design considerations

A containerport is sometimes referred to as container "terminal." A containerport is a primary node where transshipment of containers occur between land transport and sea transport, as seen in Figure 2.8.

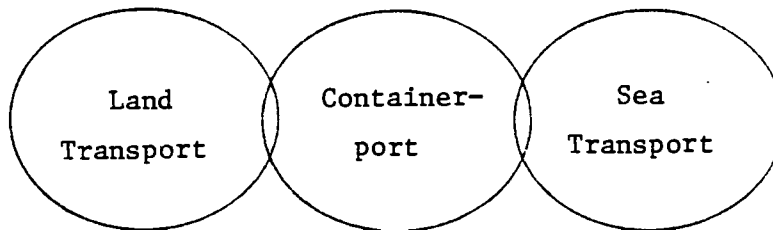


Figure 2.8 Containerport as a link between two transport modes

The role of a containerport is better illustrated by a network type diagram which shows points of transshipment and transport nodes connecting them. The design of a containerport should be based on a broad examination of container flow throughout a network, such as that shown in Figure 2.9. In Frankel (ed.) (1974), a containerport is viewed as a link in a regional trade network which incorporates a set of hinterland demand centers and foreign trade regions. Then an optimization algorithm is employed to determine the set of flows through all arcs which will satisfy the hinterland demands and minimize the total transportation cost to meet these demands. In Figure 2.9

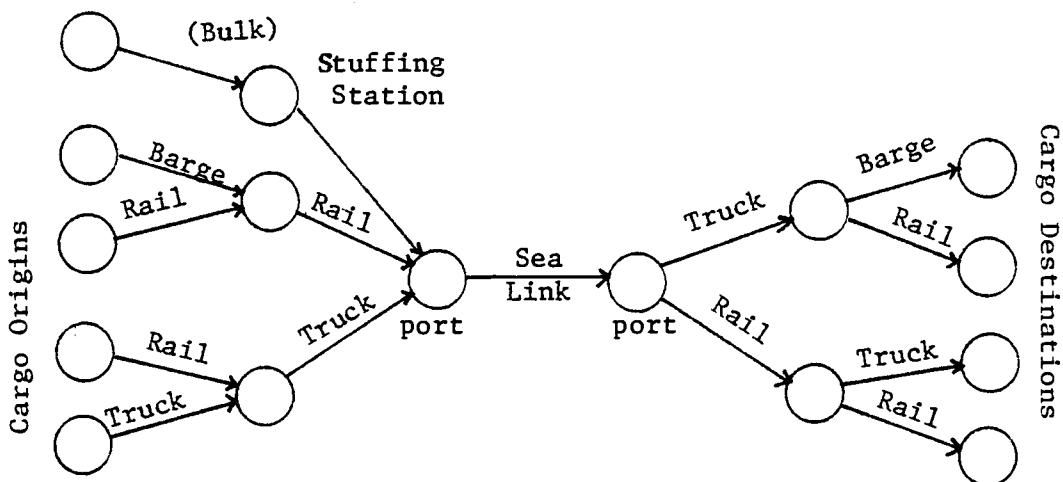


Figure 2.9 Network representation of container transportation system

the air transportation of containers is purposefully left out, as its traffic volume is insignificant compared to that handled through the marine system.

Computer simulation techniques and queueing theory have been employed to determine optimum values of port design parameters (see Borovits and Ein-Dor, 1976; Hansen, 1972; Mytton and Walker, 1979; Edmond, 1975; Frankel, 1974; Jimenez, 1976; Edmond and Maggs, 1978; and Hwang, 1978). In these models, a container-port is viewed as a queueing system composed of container handling equipment and stations as servers and containers as entities requiring various types of service. Considerable savings can result from judicious information obtained from such simulations. The Port of Portland reports a multi-million dollar savings, which arose from the difference between the cost of the original proposal before simulation and that of an optimum design through simulation work. The key design parameters for containerports are:

- a. Berth lengths,

- b. Number of berths,
- c. Equipment requirements and usage,
- d. Yard layout and entry gate design.

The biggest part of the reported savings in the case of the Port of Portland development came from not having to purchase some of the multi-million dollar equipment such as shipside cranes and transtainers (gantry cranes that operate in the yard to pick up and stow containers), which would have been bought had simulation not been adopted as a method of optimum port design.

Monetary savings in investment is only one of the several types of benefits that can be garnered from using simulation as a design aid. Understanding of a containerport as a system and management confidence in its investment decision are some of the other benefits.

While the Port of Portland model is a highly complex one written in FORTRAN IV language and requiring a large computer core memory to run, the model constructed at MIT in GPSS (General Purpose System Simulator) can be utilized by any containerport planner with a relative ease in adaptation. GPSS is a block-oriented simulation language. For this reason, the MIT model is highly aggregated and does not offer the level of detail which the Port of Portland model possesses. Given only an estimate of future arrival patterns of ships and trucks, the resulting simulation output is likely to yield no better validity than that of the input data. More emphasis should be placed on correct estimation of input data than obtaining a highly detailed model, in this regard.

### 2.3.2 Organization

A containerport usually consists of the following components:

- a. Dock area,
- b. Yard storage area,
- c. Gate processing area,
- d. Container freight station (CFS),
- e. Control/administration area,
- f. Maintenance area.

Dock area covers the ship berths and their immediate waterfront area where guay cranes move on fixed rail tracks. Yard storage area is used to stow both inbound and outbound containers temporarily before they are shipped out to their next destinations. Gate area is where incoming and outgoing trucks are processed with necessary paperwork and weighed as necessary. Yard storage spots are designated here for incoming trucks with load. Container consolidation activities are performed at a CFS. Some outbound containers originate from a CFS. Functions of load planning, information processing and operational control are performed within the control/administration area. Figure 2.10 shows the schematic diagram of the Fulton Terminal (T-6) at Port of Portland.

Figure 2.11 shows major alternative configurations of material handling systems at containerports. Comparison of their relative merits is outside of the scope of this work. It must be pointed out, however, frequent troubles with systems using straddle carriers as yard movers have been reported. Our primary interest in this study is restricted to the overhead crane system which employs both yard trucks with chassis and transtainers as prime yardside movers.

At the Fulton Terminal, yard sections are numbered from 41 to 57. A yard section consists of 73 row positions (numbered 1 to 73), six columns (labeled A to F) and containers can be stacked up to four tiers high (labeled Bottom, Center, Top, and

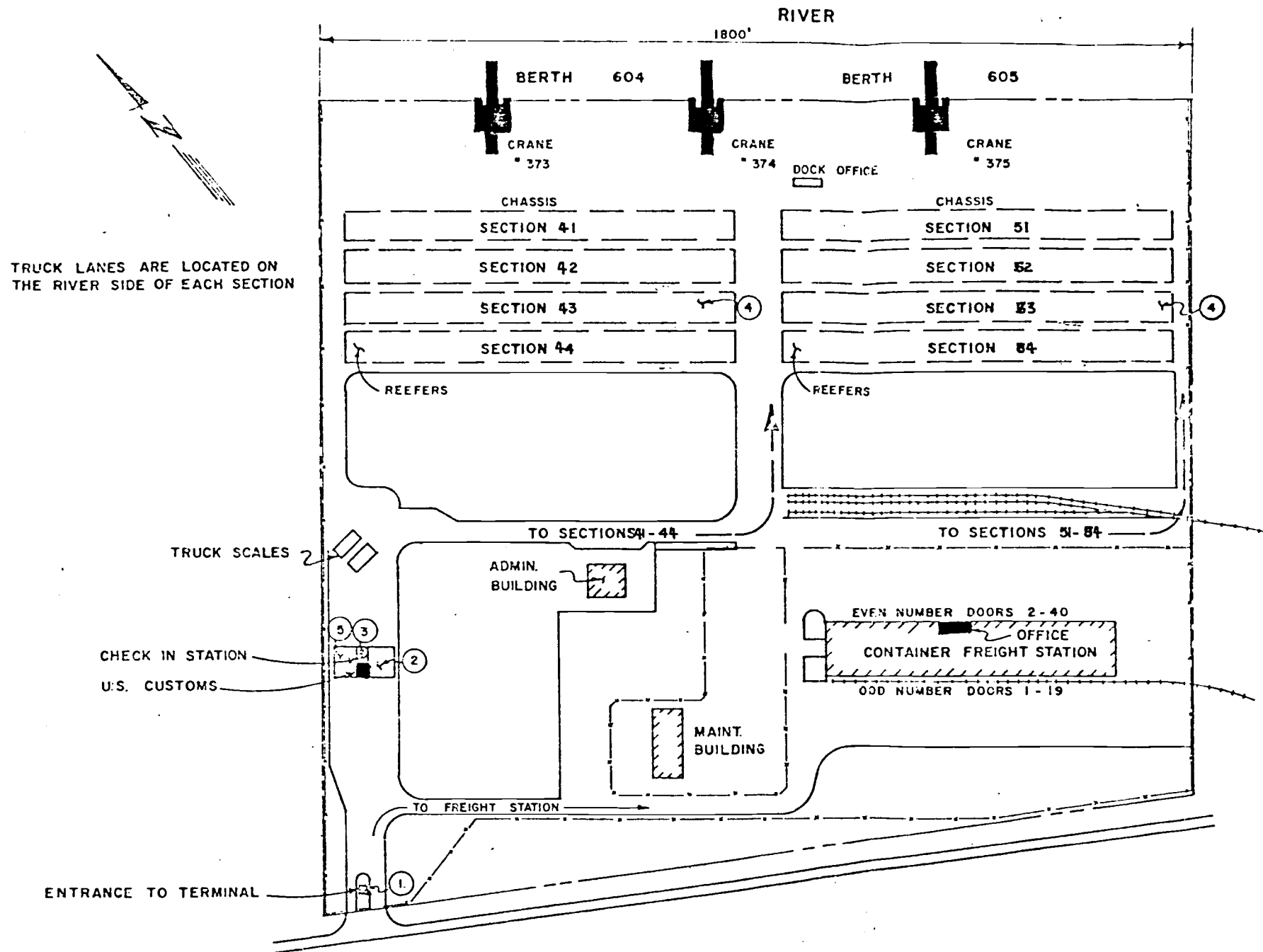


Figure 2.10 Schematic layout of a containerport  
(Courtesy of Port of Portland)

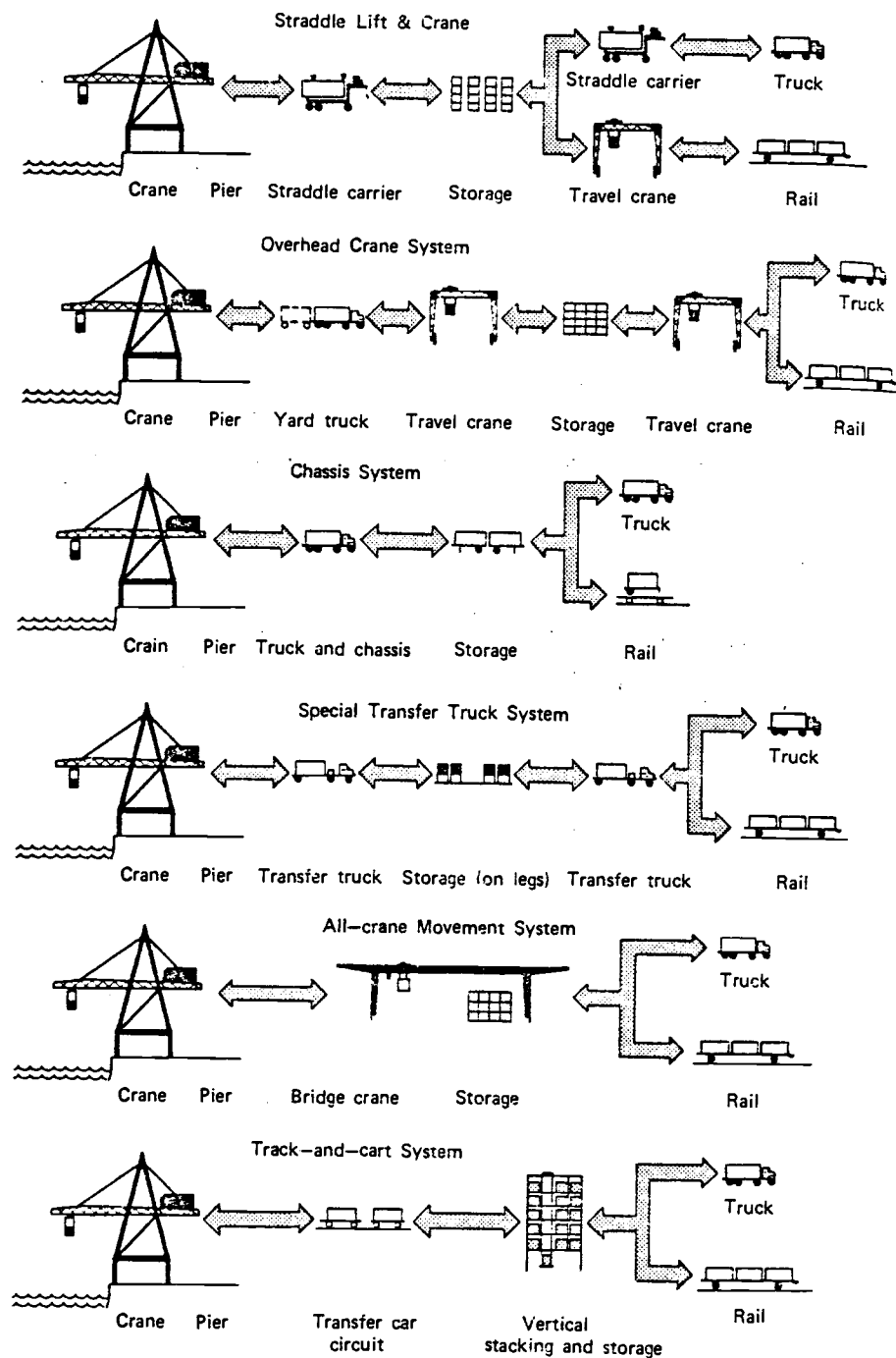


Figure 2.11 Major material handling alternatives at containerports  
(Source: Rath)



Zulu). Row numbers start at the right end of each section and increase toward its left end. Sections four and eight have electric outlets provided for reefers. Sections 46 and 47 are primarily used to store empty containers. Side loaders are used to move these instead of transtainers.

Since two row positions take up one 20-foot container length, one yard section has the length of 37 of 20-foot containers. Odd numbers are used to designate 20-foot container row positions and even numbers for 40-foot containers. Thus, row position 5427 is a 20-foot container location in section 54.

Transtainers move along the side of a section and can turn their wheels to move to a different section not adjacent to the current one laterally. Transtainer movement between sections must be done via the middle two-way path between two columns of sections. This is primarily for safety reasons. Since turning of transtainer wheels is time-consuming, the number of such section changes should be minimized if a smooth flow of containers from yard to the loading dock area is to be maintained during a loading operation.

### 2.3.3 Material Flow

Container flow at a containerport can be divided into two different types for all practical purposes: yardside flow and dockside flow. The yardside flow refers to transfer of containers between arriving trucks and yard storage area. The dockside flow is between the ship and the yard. Figure 2.12 schematically shows the yardside flow for full as well as empty containers. Figure 2.13 shows the dockside flow for both inbound and outbound containers.

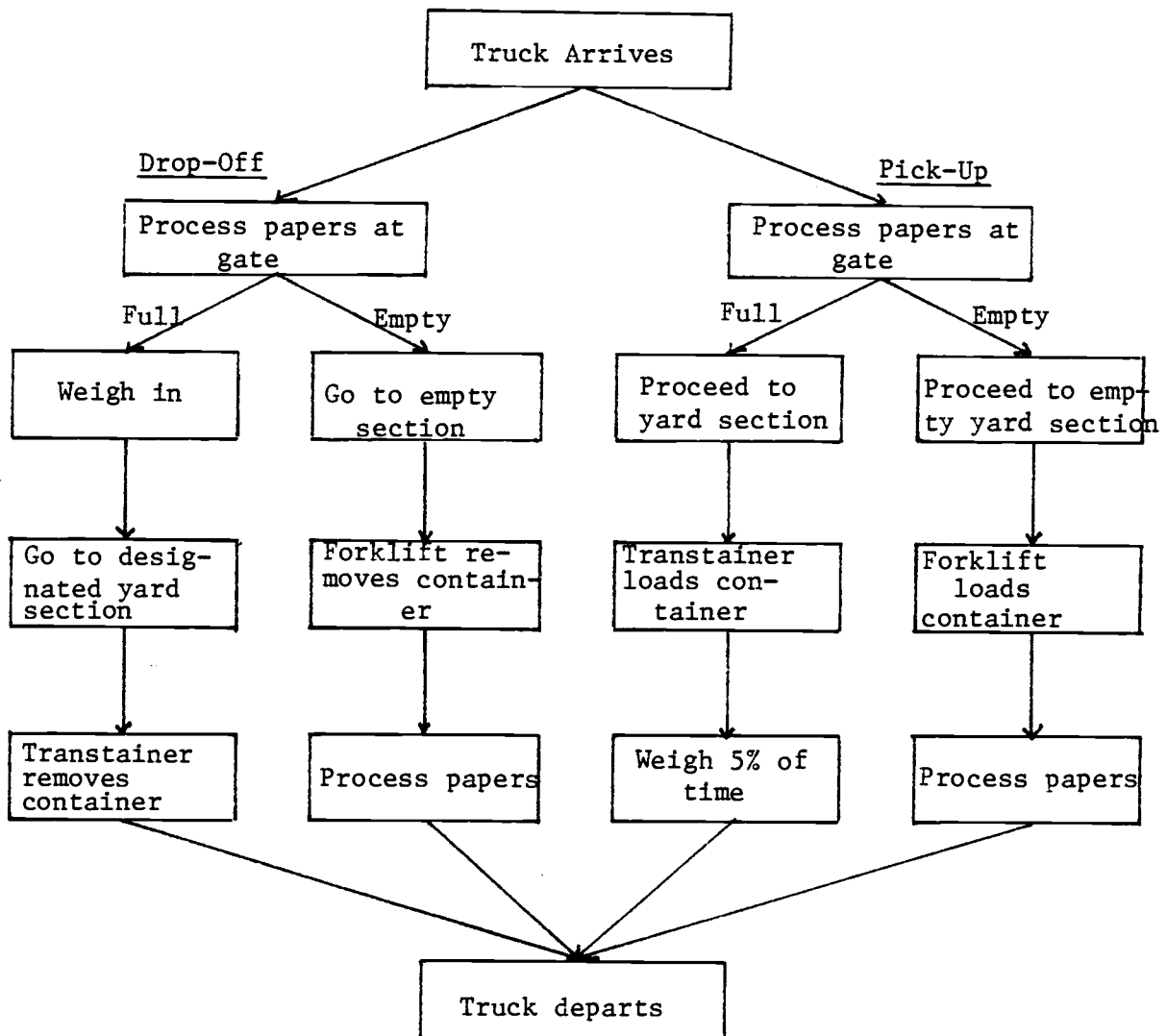


Figure 2.12 Yardside container flow

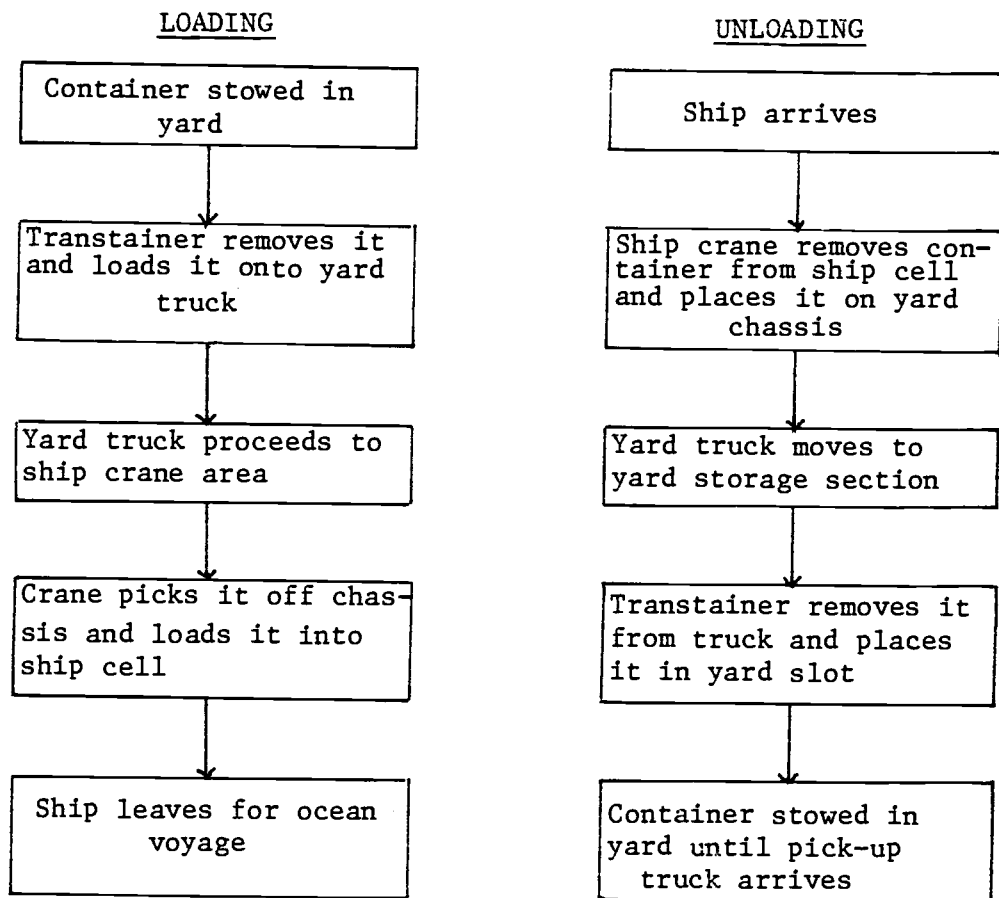


Figure 2.13 Dockside container flow

#### 2.3.4 Information system

Information flow with regards to containerports can be divided into the following two categories: The first is between containerports. The second is within a containerport. The following descriptive information is associated with a single container.

- Port of destination
- Size
- Type of commodity
- Structure
- Identification number
- Shipping line which carries it
- Height
- Total weight (in metric tons)
- Yard location
- Source of origin

Load planners often have to make do with imperfect set of information. Containers that are scheduled to be loaded to a certain voyage and have not arrived at the port usually carry only partial information. They include the ship's displacement, moments and weight distribution by bay, and foremost of all, what weight is loaded in each cell. Despite the rapid growth of the computer industry, data transfer from a port to the next port of call on a ship's load condition is currently only as fast as the speed of the ship.

A computer network connecting ports of a geographic region can greatly facilitate this data transfer. For example, a network connecting Vancouver, Seattle, Portland, Oakland/San Francisco, Long Beach/Los Angeles, and San Diego could be set

up for the benefit of all ports. Such a network is illustrated in Figure 2.14. A central computer with necessary software components for data management and planning could be installed at one port with all other ports connected to it through batch terminals which can access the system on a time-sharing basis.

Absence of such a computer network is deemed to stem from the lack of full comprehension of its potential benefits within the container industry and the lack of federal research support in this area. The federal government, however, can only provide help and guidance. Related ports in a region must exercise co-operation towards establishment of such a network for the benefit of all.

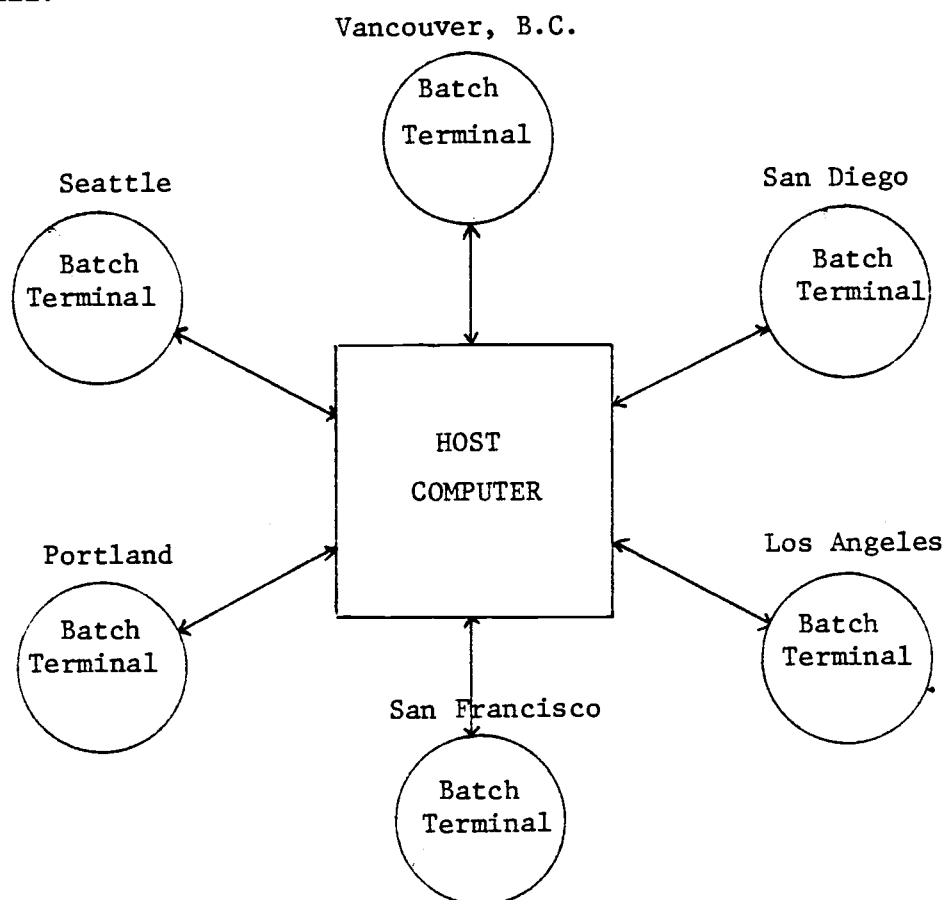


Figure 2.14 Suggested computer network for West Coast ports

Geisler and Trautnitz (1977) report a considerable increase in container throughput through partial automation of the EUROKAI container terminal in Hamburg. The automation consists of four levels: planning level, central control level, individual control level, and drive level. Only stacking cranes (transtainers) are included in this scheme. Shippside cranes are excluded. The three stacking cranes all operate on a single large yard section. The Hamburg system is depicted in Figure 2.15.

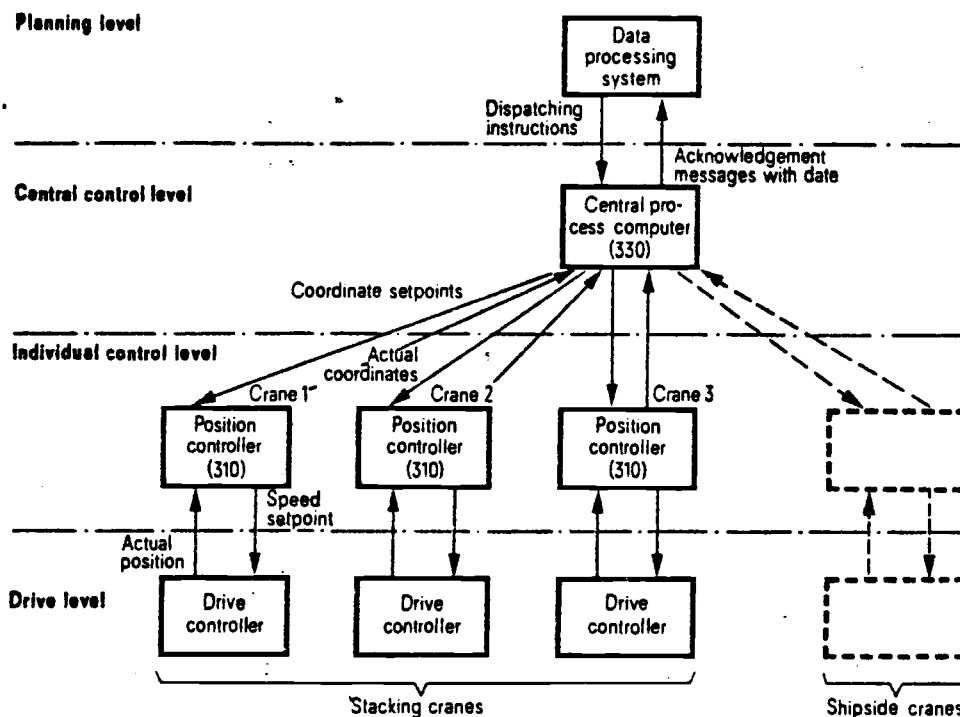


Figure 2.15 Hamburg port automation system  
(Source: Geisler and Trautnitz, 1977:349)

For containerports with many yard sections and diesel-driven transtainers operating among them, a different organization must be employed for automation. A proposed system which is more comprehensive than that of Hamburg is depicted in Figure 2.16. In this system, the yard location of an incoming container is displayed by the central computer to a display unit at the gate. This, together with the rest of the container information shown on its shipping document, is fed into the central memory from the terminal device at the gate. Yard container file thus created can be later verified by DP personnel as a copy of the document is delivered to the administrative area of the port. The container manifest for a voyage then can be entered to the central computer which goes into the load/unload planning using the data files which are constantly being updated. In order to work with a fixed set of containers, a deadline must be applied. Any container which has not arrived by this time is excluded from the voyage.

The load/unload plans generated are then converted to sequential movement orders to individual ship cranes and transtainers. A movement order would consist of yard locations, container identification number, size-type, and ship-cell coordinates. Process computers could be used to control stacking or picking of containers by ship cranes and transtainers. Manual operation must be allowed to effect the interface between transtainers and yard trucks and between ship cranes and yard trucks. As containers are moved/stacked in the yard, a container file is updated continuously within the central memory without having to manually enter their movements. Such an automation system would require the following major software components besides the usual file manipulation routines:

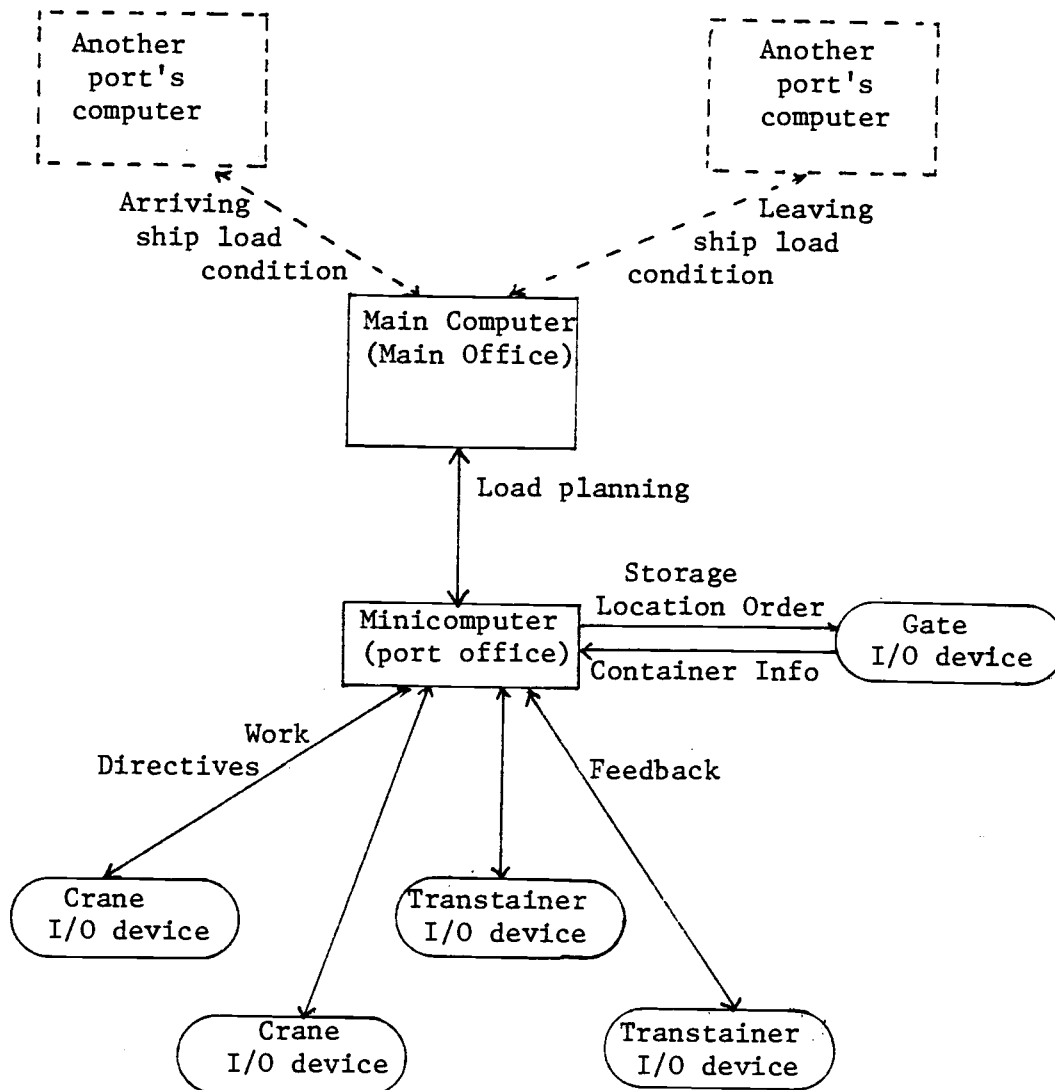


Figure 2.16 Proposed computer system at a containerport



1. Yard Location Designator Program
2. Unload Plan Program
3. Load Plan Program
4. Transtainer Control Program
5. Crane Control Program

These programs all interact through a master program which controls entire port activities. Such an automation system would greatly increase a port's throughput. It would require, however, a large amount of capital investment. Its potential benefits are becoming more and more attractive to ports of large sizes as labor costs skyrocket and control over labor practices have become increasingly difficult.

#### 2.3.5 Port Efficiency

Frankel et al. (1974) list the following measures of port performance as possible criteria in port analysis:

1. Maximum throughput (per unit time),
2. Minimum lost/unit throughput,
3. Maximum cost/unit time,
4. Maximum profit,
5. Maximum return on investment,
6. Maximum productivity (output per unit input),
7. Maximum efficiency (in the use of resources),
8. Least total integrated transportation costs,
9. Maximum resource utilization,
10. Maximum regional income distribution,
11. Maximum balance of payments effect,
12. Other.

They are all very much interrelated. Since our assumption

is that the present level of investments are to continue for the immediate future, criterion (1) provides a reasonable measure of effectiveness. Specifically, ship turnaround times must be minimized in order to maximize a port's throughput.

### 3. CONTAINERSHIP LOAD PLANNING PROBLEM

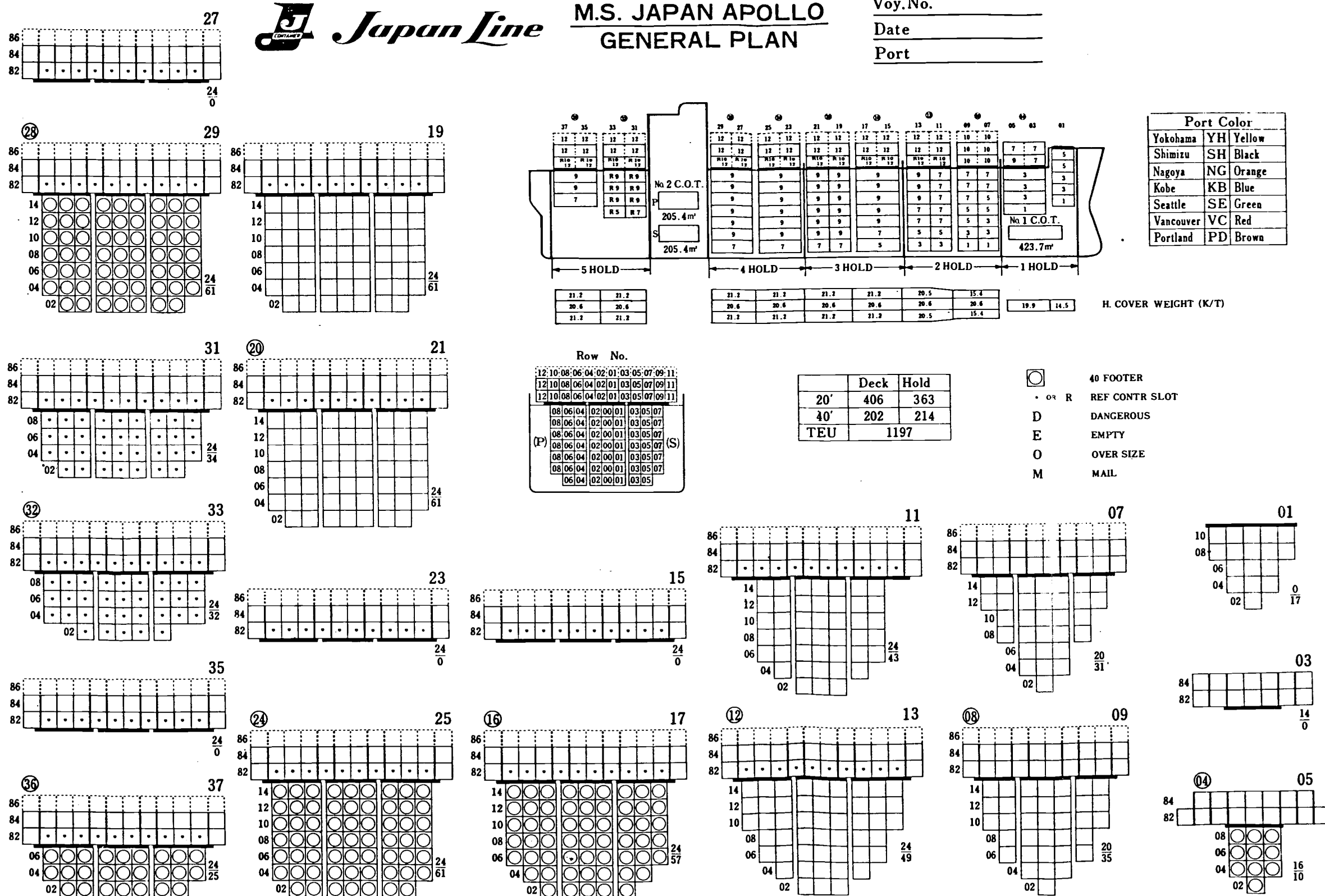
#### 3.1 Problem Description

The problem of containership load planning (CLP) is to generate a set of container-to-cell assignments which meet the general loading requirements set by shipping line, ship stability requirements, and minimize total material handling. It is an extremely complex task because the interrelationships among influencing factors are difficult to identify and various parameters effect operating constraints in both linear and nonlinear manners.

The shipping line's load requirements provide the main basis of a load planning work. They are color-coded on the ship's schematic plan sheet such as shown in Figure 3.1, and given to the load planning personnel. As far as a port is concerned, this sheet effectively sets the boundary within which planning work is done while searching for a feasible plan. In other words, the optimality in load planning is constrained within the demands of the shipping company. This is naturally so because the shipping company owns the ship and they have every right to utilize the ship capacity in whatever manner is desired. This set of requirements will be referred to as the general load plan (GLP) from this point on.

A GLP defines the following items:

- Number and destinations of containers to be loaded into each bay,
- Type and size of containers for each bay,
- Special loading instructions such as reefer machine positions,



- Total number of bays to load,
- Total number of containers to load.

Ship stability and attitude requirements impose another set of critical constraints on load planning. A permissible range of metacentric height (GM) is the explicit form of ship stability constraint, while draft forward, draft aft and trim restrictions pertain to ship attitude. Trim range is affected by seasonal variation of the sea state expected. During stormy periods a tight limit is required. Heel angle during a load operation must be close to zero. This requirement scarcely poses any problem if GLP allows even distribution of load between starboard and port sections of bays.

A number of alternative criteria can be formulated as the objective of containership load planning. Some examples are as follows:

- Minimize total port system cost per container,
- Minimize turnaround time,
- Minimize total transtainer move distance (TTMD),
- Minimize total crane operating time,
- Maximize throughput per unit time.

They are all relevant criteria for the interests of the port and consequently, the ship. The objective should be formulated within the set of facilities a port currently possesses. Selection of an appropriate objective function depends on ease of formulation and tractability of an effectiveness measure. The first one in the example includes many variables that interact and, hence, it could pose a problem of complexity before a successful translation, whereas other criteria tend to be more well-defined.

### 3.2 Present System of Load Planning

Containership load planning is done manually at most ports, with computer assistance in some cases. At some ports, computer-generated tags bearing container information are used for manual planning work by operations personnel. The planning activity is conducted as shown in the flowchart of Figure 3.2. The following explain the steps in this planning process:

- a. Yard locations assigned to individual containers as they are processed through the gate are entered to their shipping documents.
- b. Container release document from the steamship line is verified against individual container shipping documents. Appropriate adjustments are made for any discrepancy.
- c. Data processing takes release information to generate container tags. (Some tags as attached on a bay storage plan are shown in Figure 3.3.)
- d. Planners (operations personnel) check the tags against release information to correct any errors.
- e. Extra tags are written out by hand for containers added to the release list after tags printout.
- f. Planners prepare "Operators Recapitulation Sheet" which breaks down the number of released containers according to port, size-type, and line.
- g. The shipping line representative delivers the general load plan to the planner.
- h. Planners prepare "Ship Work Sequence" sheets. These include bay-by-bay load/unload work plan in sequence.

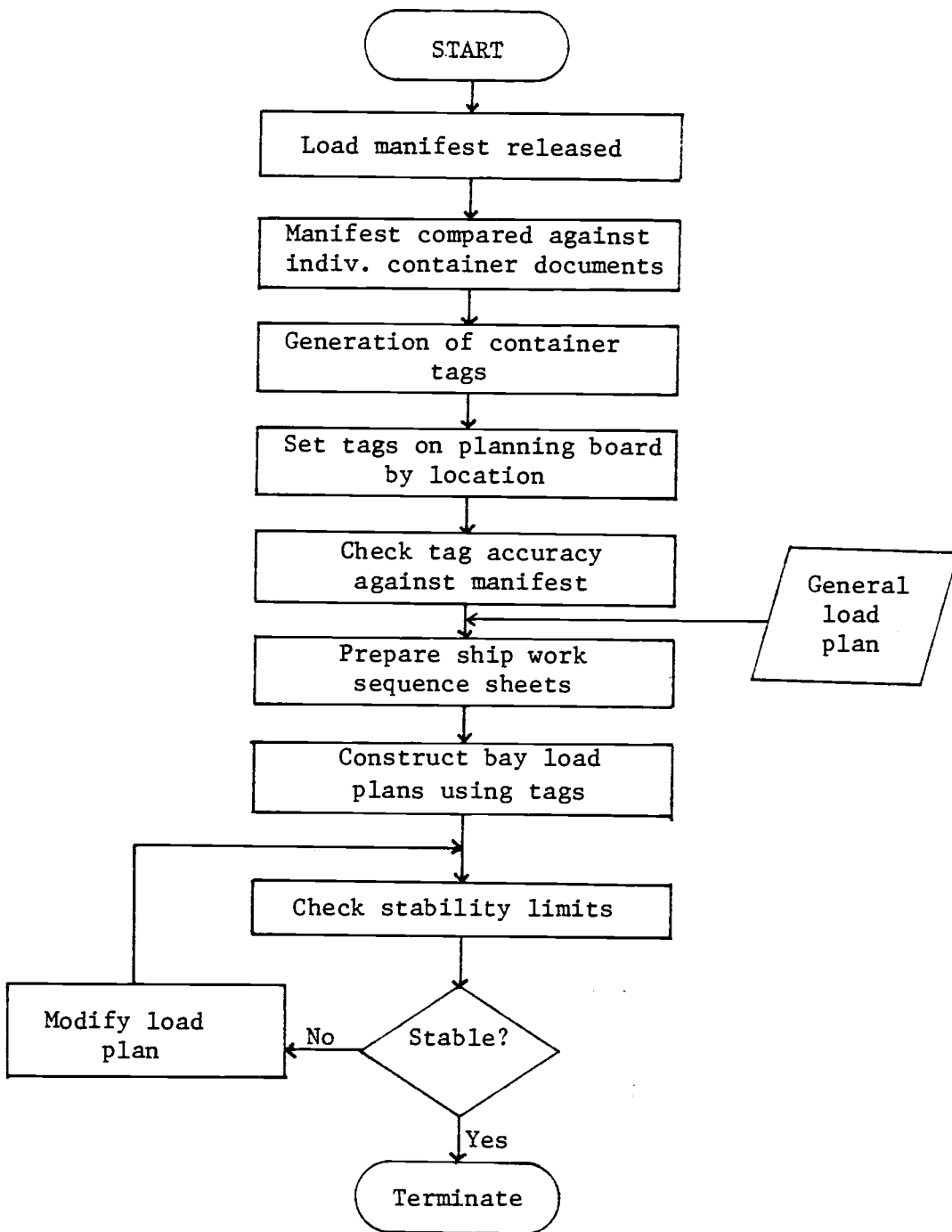


Figure 3.2 Manual load planning procedure

These are prepared in a way such that the work restrictions between two cranes and between two transtainers are satisfied, and that the work load is balanced between two gangs.

- i. Planners pull tags from the board and place them on bay storage plan sheets such as in Figures 3.3 and 3.4. This activity is conducted with all loading restrictions in mind as well as to minimize total transtainer move distance.
- j. Stability check is made for the load plan generated. Adjustments are made if either GM or trim condition is violated.

### 3.3 Desirable Properties of a CLP Program

The following attributes were considered essential for a CLP model:

- a. It must yield load plans that are equivalent to or better than those generated manually.
- b. Ease of implementation at most ports.
- c. General enough to be used by most ports with overhead crane system.
- d. Minimize yard equipment and crane moves.
- e. Maximize the number of loaded containers.
- f. Incorporate an interactive routine to enable minor changes.
- g. Output format should be generally acceptable.



**BAY NO. 16**  
 UNDER DECK

PAGE 1730

|   |   |   |   |   |  |   |   |   |
|---|---|---|---|---|--|---|---|---|
| 16 08 12  | 16 06 12  | 16 04 12  | 16 02 12  | 16 00 12  | 16 01 12                                       | 16 03 12  | 16 05 12  | 16 07 12  |
| SLTF4864019<br>SLTF4860016<br>SLTF4860001<br>27.4 21-23 | SLTF4860016<br>SLTF4860001<br>SLTF4860001<br>27.0 05-07 | SLTF4860001<br>SLTF4860001<br>SLTF4860001<br>27.0 09-11 | SHIMXPL<br>KLU201296<br>K DRY<br>24.9 5260EC    | SHIMXPL<br>ITSU202581<br>K DRY<br>24.1 5260EC   | SHIMXPL<br>ETIU4870415<br>K DRY<br>22.3 5260EC | SLTF4860002<br>SLTF4860002<br>SLTF4860002<br>27.3 13-15 | SLTF4860008<br>SLTF4860008<br>SLTF4860008<br>27.0 17-19 | SLTF4860077<br>SLTF4860077<br>SLTF4860077<br>29.1 27-29 |
| 16 08 10  | 16 06 10  | 16 04 10  | 16 02 10  | 16 00 10  | 16 01 10                                       | 16 03 10  | 16 05 10  | 16 07 10  |
| TXVUC<br>MOLU2041072<br>MO 009<br>26.1 25-27            | TXVUC<br>CTIU4539768<br>MO 062<br>27.1 04-06            | TXVUC<br>MOLU2004356<br>MO 062<br>25.2 03-05            | YONOXPL<br>JLU202219<br>JPL DRY<br>27.8 4216EC  | SHIMXPL<br>YSU203257<br>YS DRY<br>23.0 5260EC   | YONOXPL<br>JLU206162<br>JPL DRY<br>22.7 4216EC | TXVUC<br>MOLU2028785<br>MO 062<br>28.7 08-10            | TXVUC<br>CTIU4252314<br>MO 062<br>27.8 07-09            | TXVUC<br>JLU302043<br>JL FR 062<br>29.7 28-30           |
| 16 08 08  | 16 06 08  | 16 04 08  | 16 02 08  | 16 00 08  | 16 01 08                                       | 16 03 08  | 16 05 08  | 16 07 08  |
| TXVUC<br>JLU306099<br>JL FR 082<br>29.7 19-21           | TXVUC<br>CTIU4900422<br>MO 009<br>29.3 09-11            | TXVUC<br>JLU306124<br>JL FR 082<br>28.8 24-26           | YONOXPL<br>SSIU202305<br>JPL DRY<br>21.5 4216EC | YONOXPL<br>CTIU429293<br>JPL DRY<br>26.3 4216EC | YONOXPL<br>JLU201074<br>JPL DRY<br>22.4 4216EC | TXVUC<br>CTIU4900361<br>MO 009<br>28.8 10-12            | TXVUC<br>MOLU2063045<br>MO 009<br>25.1 26-28            | TXVUC<br>JLU306120<br>JL FR 082<br>29.7 20-22           |
| 16 06 06  | 16 04 06  | 16 02 06  | 16 00 06  | 16 01 06  | 16 03 06                                       | 16 05 06  | 16 07 06  | 16 09 06  |
| TXVUC<br>JLU302015<br>JL FR 082<br>29.5 27-29           | TXVUC<br>JLU306124<br>JL FR 082<br>28.8 24-26           | TXVUC<br>CTIU4819481<br>MO DRY 081<br>29.4 19-21        | YONOXPL<br>YSAA207213<br>YS DRY<br>27.6 4212EC  | YONOXPL<br>YSU206120<br>YS DRY<br>24.5 4212EC   | YONOXPL<br>FECU340049<br>YS DRY<br>23.5 4212EC | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           |
| 16 04 04  | 16 02 04  | 16 00 04  | 16 01 04  | 16 03 04  | 16 05 04                                       | 16 07 04  | 16 09 04  | 16 11 04  |
| TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30   | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30   | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30  | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           | TXVUC<br>JLU306118<br>JL FR 082<br>28.8 28-30           |

| WEIGHT<br>(K) | KG<br>(M) | V. MOMENT<br>(M) |
|---------------|-----------|------------------|
| 12            | 71.3      | 1154             |
| 17.5          | TK        | 283              |
| 10            | 70.4      | 959              |
| 135.7         | TK        | 1844             |
| 29.1          | TK        | 375              |
| 08            | 23.0      | 253              |
| 245.1         | TK        | 2366             |
| 06            | 181.1     | 1521             |
| 200.8         | TK        | 1119             |
| 02            | 133.3     | 432              |
| 164.9         | TK        | 2346             |
| 135.7         | TK        | 1844             |
| 772.4         | TK        | 6166             |
| 1078.0        | TOT       | 10376            |

|                 |                 |                 |                 |                 |                 |                 |                 |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| (08) 11.47      | (06) 8.81       | (04) 8.15       | (02) 2.66       | (01) 2.66       | (03) 6.15       | (05) 8.81       | (07) 11.47      |
| W.T             | W.T             | W.T             | W.T             | W.T             | W.T             | W.T             | W.T             |
| M.T             | M.T             | M.T             | M.T             | M.T             | M.T             | M.T             | M.T             |
| PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL |
| PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL |
| PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL | PORT SIDE TOTAL |

**BAY NO.**  
**16**

D. 160

Figure 3.3 Bay storage plan (under-deck bay)


**Mitsui O.S.K. Lines**

M.S. ALASKA MARU

**BAY NO. 19**

ON DECK

PAGE 18 / 30

| WEIGHT (KT) | KG (M) | V-MOMENT (KT-M) |
|-------------|--------|-----------------|
| 86          | 2.7    | KB              |
|             |        | 24.68           |
| 84          | 3.5    | KB              |
|             |        | 22.23           |
| 82          | 19.9   | KB              |
|             |        | 19.78           |
|             |        | 539             |

|                 |              |                  |
|-----------------|--------------|------------------|
| PORT SIDE TOTAL | 8 SIDE TOTAL | TORSIONAL MOMENT |
|-----------------|--------------|------------------|

64.2 20' KB TOTAL 1350

**BAY NO. 21 (20)**

ON DECK

| WEIGHT (KT) | KG (M) | V-MOMENT (KT-M) |
|-------------|--------|-----------------|
| 86          | 2.5    | KB 20'          |
|             |        | 63              |
|             |        | 160.3           |
|             |        | 47.1            |
|             |        | 24.68           |
|             |        | (25.06)         |
| 84          | 15.7   | KB 20'          |
|             |        | 353             |
|             |        | 152.5           |
|             |        | KB              |
|             |        | 3515            |
|             |        | 51.8            |
|             |        | 22.23           |
|             |        | (22.46)         |
| 82          | 19.9   | KB 20'          |
|             |        | 395             |
|             |        | 160.1           |
|             |        | KB              |
|             |        | 3180            |
|             |        | 53.3            |
|             |        | 19.78           |
|             |        | (19.86)         |
|             |        | 38.1            |
|             |        | 476.9           |
|             |        | 153.2           |
|             |        | 628.1           |
|             |        | 14114           |

|                 |              |                  |
|-----------------|--------------|------------------|
| PORT SIDE TOTAL | 8 SIDE TOTAL | TORSIONAL MOMENT |
|-----------------|--------------|------------------|

BAY NO.  
19, 21 (20)

D. 160

Figure 3.4 Bay storage plan (on-deck bays)

Throughout this study, these attribute goals were kept in mind and efforts were exerted to achieve them. Two or more CLP models can be compared using the above criteria.

### 3.4 Existing Models and Past Attempts

The only working computer model for CLP that exists today besides the model introduced here has been developed by Hydronautics, Inc. of Maryland, as reported in Webster et al. (1969) and Cojeen and Van Dyke (1976). The Hydronautics' model develops a ship-loading plan which achieves desired stability parameters through consideration of both detailed current-port information and complete cycle estimates of expected trade. Practical constraints include: (1) consideration of exceptions and modifications to cargo; (2) specific locations for given cargo; (3) time dependence of cargo availability; (4) crane movement restrictions; and (5) achievement of desirable ship trim and stability characteristics. The input and output formats are excellent in that they are of readily available form and suitable for immediate use by ship and dock operations personnel. This work was done under the Ship Operations Informations System (SOIS) program, funded by the Maritime Administration and certain U.S. Merchant Marine operators. United States Lines, specifically, provided the terminal and stevedoring operating environment for testing their model.

The program was originally developed for a batch mode of operation and was designed for execution on a central computer facility. It is now available through the General Electric Mark III timesharing system which allows terminals in an international network to have access to SOIS developed systems.

In 1977, Matson Terminals, Inc. entered into a one-year contract with the Maritime Administration to perform the test of the

Hydronautics model using the G.E. Mark III system. Matson's charter, under the contract, was to perform a test of the model under operational conditions and make a comparison of automated versus manual load planning. Hydronautics customized their program to reflect Matson's particular ship and yard procedures.

The test resulted in a negative GM. This was an unacceptable result. In addition, the program's inability to consider vessel deck strength, lashing strength, and container racking strength limitations was identified. Also, the program did not ensure that a container placed aboard is supported by either the vessel structure or another container. Lastly, it did not properly constrain placement of hazardous containers in accordance with U.S. Coast Guard requirements.

The model's failure to produce a stable load plan stems from its assumption about container weight distribution. The model assumes uniform distribution of container weights and picks a container whose weight most closely fits the target weight for a given ship cell as generated towards meeting the stability requirements. The container weights of Matson's operating environments had a distribution of a bath tub shape, i.e., both ends were heavily distributed. After containers with "attractive" weights were run out, each selection tended to contribute negatively toward ship stability.

Another important aspect of the Hydronautics' model is that it was developed for loading using a multitude of chasses in which case random access of desirable containers is possible. This model would be inadequate for a port with transtainer operation in the yard. The key goal of this model was to meet stability, not to minimize yard handling of containers.

Beliech (1974) of the U.S. Navy wrote a heuristic for CLP, but coding and application of his logic have yet to appear. Also, his work gave no regard to minimizing yard-side material

handling but a heavy emphasis on load restrictions.

Scott and Chen (1978) of Matson Navigation Co. proposed a heuristic algorithm for CLP. The objective of this model is to maximize the number of containers which can be placed on the ship and to minimize the ballasting requirements. This form of objective stems from the unique Matson operating condition of bare loading a ship as fully as possible and of engaging in a single leg of voyage between Hawaii and Oakland, California. For most ports, this mode of strip-down, bare-loading operation is an unusual one. The algorithm, as proposed, would proceed as follows:

Stage 0 (Preparation):

Containers are classified into groups by type, size, destination and weight.

Stage 1:

Containers of other than the dry standard group are assigned to individual slots on the ship by destination from lower tiers and on up so that heavier containers are placed on lower tiers and closer to the midship.

Stage 2:

An integer programming problem is solved to determine the number of containers of each weight bracket of the dry standard type to be assigned to each row of a bay so that the total number of containers is maximized while meeting the requirements of trim and stability.

Stage 3:

Individual container-cell assignments of the dry standard type are made within each row so that stability is preserved.

Stage 4:

Any violation of load restrictions is smoothed out through the most favorable exchanges to resolve it.

Application of this heuristic has not been reported as yet. Some examination of this approach was conducted during the period of this research in order to determine its worthwhileness. The following summarizes the findings from the work:

First, the objective function is of an inadequate form for most ports concerned. This problem was discussed previously.

Second, material handling in the yard is not given any consideration. It is probably because Matson's Oakland yard employs a marshalling scheme with straddle carriers. These 'strads' can move only in one direction, making it indifferent where the container is located. Again, this is a unique situation which is not shared by ports such as Portland, Seattle and Vancouver where transtainers are the major movers in the yard.

Third, formulation and solution of the integer programming (IP) problem could create difficulty of software compatibility. Generation of a program data to fit an existing IP package is by no means easy. Also, there is no guarantee that an IP routine would yield an optimal feasible solution at all times. Experience with IP routines in the MPOS package developed by the Northwestern University justifies this concern. Gomory's cutting plane algorithm, in particular, sometimes gave a "presumed infeasible" response. It is difficult to determine whether the problem is really infeasible or not. If it was infeasible, then the heuristic fails to generate a load plan. If it was feasible, but the IP failed to find a feasible solution, an erroneous result is obtained.

The literature survey revealed the definite need for a good CLP model. Considering the significance of container industry

in world trade, not enough attention is being placed in this area. It was the author's hope that this work could spark a flood of research activity in useful CLP development even if it fails to achieve the aim.

#### 4. APPROACHES TO THE PROBLEM

The mathematical nature of the containership load planning problem is explored in this chapter. Various approaches of model formulation are investigated especially with regard to feasibility in implementation.

##### 4.1 Formulation as An Integer Program

Container-to-ship cell assignments are of one-to-one type. This nature easily leads to the 0-1 integer program as a possible model structure for solving the containership load planning problem. Thoolen et al. (1978) and Scott and Chen (1978) provide much of the basis of the following general integer program structure of the CLP problem.

###### 4.1.1 A General Model

The containership loading problem consists of the following objectives and constraints:

###### Objectives:

- 0-1: Maximize number of containers to be loaded.
- 0-2: Minimize total equipment move distance.

###### Constraints:

- C-1: Physical constraints (only one container is loaded into a cell on board).
- C-2: Stability limits.
- C-3: Container support constraints (no empty cell must exist between two containers vertically or between a container and deck).
- C-4: Deck strength limits.



- C-5: Container racking strength limits.
- C-6: Stack height limits for under-deck bays.
- C-7: Overstow limits.
- C-8: Hazardous cargo placement restrictions.
- C-9: Reefer placement restrictions.

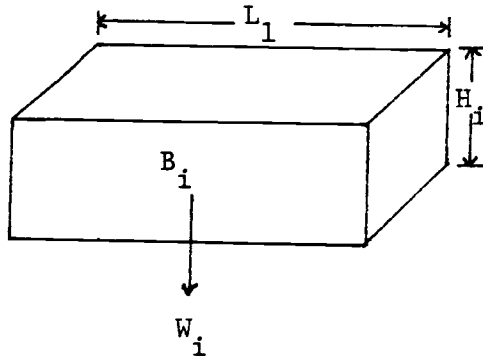
A 0-1 integer assignment problem can be formulated to represent this problem. We define decision variables as follows:

|          | $S_1$    | ... | $S_j$    | ... | $S_N$    |              |
|----------|----------|-----|----------|-----|----------|--------------|
| $B_1$    | $X_{11}$ | ... | $X_{1j}$ | ... | $X_{1N}$ | M containers |
| $\vdots$ | $\vdots$ |     | $\vdots$ |     | $\vdots$ |              |
| $B_i$    | $X_{i1}$ | ... | $X_{ij}$ | ... | $X_{iN}$ |              |
| $\vdots$ | $\vdots$ |     | $\vdots$ |     | $\vdots$ | N ship cells |
| $B_M$    | $X_{M1}$ | ... | $X_{Mj}$ | ... | $X_{MN}$ |              |

Thus,  $X_{ij} = 1$ , if container  $i$  is assigned to cell  $j$ ,  
 $= 0$ , otherwise.

$$(i=1, \dots, M; j=1, \dots, N)$$

Container characteristics are defined as follows:

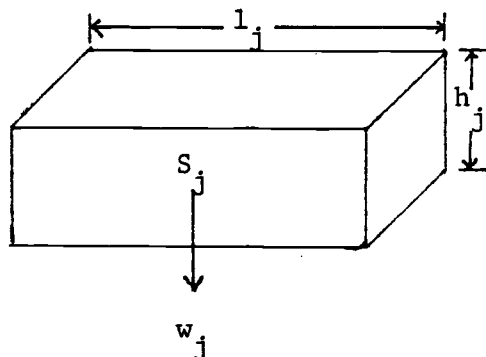


Here,

$D_i$ : Destination port number of container  $i$ ,

$L_i$ : Length of container  $i$ ,  
 $H_i$ : Height of container  $i$ ,  
 $W_i$ : Weight of container  $i$ ,  
 $S_i$ : Racking strength of container  $i$ ,  
 $Y_i$ : Yard location of container  $i$ .

Characteristics of a container assigned to a ship cell may be defined as follows:



Here,

$$d_j \text{ (Destination no. of container in cell } j) = \sum_{i=1}^M D_i X_{ij}$$

$$l_j \text{ (Length of container in cell } j) = \sum_{i=1}^M L_i X_{ij}$$

$$h_j \text{ (Height of container in cell } j) = \sum_{i=1}^M H_i X_{ij}$$

$$w_j \text{ (Weight of container in cell } j) = \sum_{i=1}^M W_i X_{ij}$$

$$s_j \text{ (Racking strength of container in cell } j) = \sum_{i=1}^M S_i X_{ij}$$

The objectives and constraints of the CLP can now be formulated using the above notations as follows:

Objectives:

$$0-1: \text{ Maximize } \sum_{i=1}^M \sum_{j=1}^N X_{ij}$$

$$0-2: \text{ Minimize } \sum_{i=1}^{M-1} d(Y_{k_i}, Y_{k_{i+1}})$$

where  $d(.,.)$  is a distance function between yard locations of two consecutive containers loaded according to a sequence,  $k=(k_1, k_2, \dots, k_M)$ .

Constraints:

$$C-1: R_i = \sum_{j=1}^N X_{ij} \leq 1, i=1, \dots, M.$$

$$C_j = \sum_{i=1}^M X_{ij} \leq 1, j=1, \dots, N.$$

C-2: (Trim and stability limits)

a. Horizontal Stability

$$-D_y \leq CG_y \leq D_y$$

where  $CG_y$ : Center of gravity for loaded containers in athwartship direction.

$D_y$ : Magnitude of maximum allowable  $CG_y$  deviation.

Here, the athwartship center of gravity can be obtained by the following:

$$CG_y = \frac{\sum_{j=1}^N y_j \cdot w_j}{W_c}$$

where  $W_c$ : Total container weight.

$y_j$ : Y-axis location of center of gravity of cell  $j$ .

In the above relationship,  $y_j$ 's are known constants for each cell. Replacing  $w_i$  with its defining equation gives us the following:

$$CG_y = \frac{\sum_{j=1}^N y_j \left( \sum_{i=1}^M W_i \cdot X_{ij} \right)}{W_c}$$

Assuming  $W_c$  is a known constant, the horizontal stability constraint is a linear one.

#### b. Longitudinal (trim) stability

The longitudinal center of gravity ( $CG_x$ ) must be within certain limits from a desirable position. Thus, we have

$$P_1 - D_x \leq CG_x \leq P_1 + D_x$$

where  $P_1$ : Desirable position of  $CG_x$  from midships.

$D_x$ : Maximum allowed deviation of  $CG_x$  from  $P_1$ .

Here,  $CG_x$  can be obtained as follows:

$$CG_x = \frac{\sum_{j=1}^N x_j \cdot w_j}{W_c}$$

where  $x_j$ : Longitudinal distance to the center of gravity of cell  $j$  from the midships.

#### c. Vertical Stability

Upper and lower limits of a ship's metacentric height (GM) for a given ship displacement can be translated to the following constraints on its vertical center of gravity:

$$P_v - D_z \leq CG_z \leq P_v + D_z$$

where  $P_v$ : Desirable vertical location of  $CG_z$  from the keel,

$D_z$ : Maximum allowed deviation of  $CG_z$  from its desired position.

Here,  $CG_z$  can be computed as follows:

$$CG_z = \frac{\sum_{j=1}^N z_j w_j}{W_c}$$

C-3: (Container support constraints)

$$C_{j_1} - C_{j_2} \geq 0 \quad \text{for all vertically adjacent cells } j_1 \text{ and } j_2 \text{ within a bay.}$$

C-4: (Deck strength limits)

$$W_{j_1} + \dots + W_{j_k} \leq DS_k$$

where  $DS_k$ : Deck strength limit of bay  $k$  in which cells  $j_1, \dots, j_k$  are loaded in the same stack.

C-5: (Racking strength limits)

$$S_{j_1} \geq W_{j_2} + \dots + W_{j_k}$$

where  $j_1, \dots, j_k$  are cell indices of the same stack and  $j_1$  is below  $j_2, \dots, j_k$  with  $j_k$  the highest cell in the stack.

C-6: (Stack height limits)

$$h_{j_1} + \dots + h_{j_k} \leq SH_k$$

where  $SH_k$  is the stack height limit for bay  $k$  where cells  $j_1, \dots, j_k$  are located in a stack.

C-7: (Overstow limits)

$$d_{j_1} \geq d_{j_2}$$

where  $j_1$  and  $j_2$  are any vertically adjacent cells with  $j_2$  above  $j_1$  in a stack.

C-8: (Hazardous cargo placement restrictions)

This constraint is not considered here to simplify the problem. The fact that a very small fraction of containers are in this category, whereas an explicit inclusion of these constraints is an extremely complex task, justifies the omission (see Thoolen et al. (1978) for an attempt on this).

C-9: (Reefer placement restrictions)

This restriction is applied by making all  $X_{ij} = 0$  for all reefer container  $i$  and non-reefer cell  $j$ .

#### 4.1.2 A simplified problem

As an attempt to obtain familiarization of the integer assignment model in solving CLP's, a small CLP was constructed and an attempt was made to solve it using a linear program package.

In this simplified problem, there are forty containers to be sent to four ports. They are classified into three weight classes; heavy, medium, and light. The hypothetical ship has only five bays with nine cells in each. Container yard locations are known. They are assumed to be stacked in a single-tier in the same section, concentrated in one area. This simplified the

distance function. The problem, then, is to assign these forty containers to forty-five ship cells so that objectives 0-1 and 0-2 are achieved while satisfying constraints C-1, C-2, C-3, and C-7. The other constraints are left out to provide simplicity to the problem.

Container yard locations are as shown in Figure 4.1. Container data are as tabulated in Table 2. Ship structure with bay locations is shown in Figure 4.2. Centers of gravity of ship cells are as shown in Table 3.

The decision variables  $A_i$  are as defined in the following assignment matrix:

|                      |          | cells    |          |   |       |            |
|----------------------|----------|----------|----------|---|-------|------------|
|                      |          | 1        | 2        | 3 | . . . | 45         |
| Con-<br>tain-<br>ers | 1        | $A_1$    | $A_{41}$ |   | . . . | $A_{1761}$ |
|                      | 2        | $A_2$    | $A_{42}$ |   |       |            |
|                      | $\vdots$ | $\vdots$ | $\vdots$ |   |       |            |
|                      | $\vdots$ | $\vdots$ | $\vdots$ |   |       |            |
|                      | 40       | $A_{40}$ | $A_{80}$ |   |       | $A_{1800}$ |

$A_i = \begin{cases} 1, & \text{if assigned} \\ 0, & \text{otherwise} \end{cases}$   
 $(i=1, \dots, 1800)$

The constraints and objectives are constructed as follows:

C-1: (Physical Constraints)

a. Defining equations

$$A_1 + A_2 + \dots + A_{40} - S_1 = 0$$

$$A_{41} + A_{42} + \dots + A_{80} - S_2 = 0$$

$$\vdots$$

$$A_{1761} + A_{1762} + \dots + A_{1800} - S_{45} = 0$$

|        |   |    |    |    |    |    |    |    |    |     |
|--------|---|----|----|----|----|----|----|----|----|-----|
| Column | F | 1  |    |    |    |    |    |    |    |     |
|        | E |    | 17 | 38 | 19 | 11 | 30 | 5  | 16 |     |
|        | D | 10 | 6  | 23 | 2  | 21 | 15 | 24 | 37 |     |
|        | C | 22 | 18 | 7  | 8  | 4  | 9  | 25 | 3  |     |
|        | B | 28 | 27 | 14 | 20 | 26 | 13 | 29 | 12 |     |
|        | A | 36 | 39 | 31 | 34 | 33 | 40 | 35 | 32 |     |
| Row    |   | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | ... |

Figure 4.1 Yard storage pattern in simplified problem



TABLE 2. CONTAINER DATA FOR  
SIMPLIFIED PROBLEM

| Container<br>no. | Port<br>no. | Weight<br>(MT) | Row<br>no . | Container<br>no. | Port<br>no . | Weight<br>(MT) | Row<br>no. |
|------------------|-------------|----------------|-------------|------------------|--------------|----------------|------------|
| 1                | 1           | 5              | 1           | 21               | 3            | 10             | 5          |
| 2                | 1           | 5              | 4           | 22               | 3            | 10             | 1          |
| 3                | 1           | 5              | 8           | 23               | 3            | 10             | 3          |
| 4                | 1           | 5              | 5           | 24               | 3            | 10             | 7          |
| 5                | 1           | 5              | 7           | 25               | 3            | 10             | 7          |
| 6                | 1           | 5              | 2           | 26               | 3            | 15             | 5          |
| 7                | 1           | 5              | 3           | 27               | 3            | 15             | 2          |
| 8                | 1           | 5              | 4           | 28               | 3            | 15             | 1          |
| 9                | 1           | 5              | 6           | 29               | 3            | 15             | 7          |
| 10               | 1           | 5              | 1           | 30               | 3            | 15             | 6          |
| 11               | 2           | 10             | 5           | 31               | 4            | 15             | 3          |
| 12               | 2           | 10             | 8           | 32               | 4            | 15             | 8          |
| 13               | 2           | 10             | 6           | 33               | 4            | 15             | 5          |
| 14               | 2           | 10             | 3           | 34               | 4            | 15             | 4          |
| 15               | 2           | 10             | 6           | 35               | 4            | 15             | 7          |
| 16               | 2           | 15             | 8           | 36               | 4            | 15             | 7          |
| 17               | 2           | 15             | 2           | 37               | 4            | 15             | 1          |
| 18               | 2           | 15             | 2           | 38               | 4            | 15             | 8          |
| 19               | 2           | 15             | 4           | 39               | 4            | 15             | 3          |
| 20               | 2           | 15             | 4           | 40               | 4            | 15             | 6          |

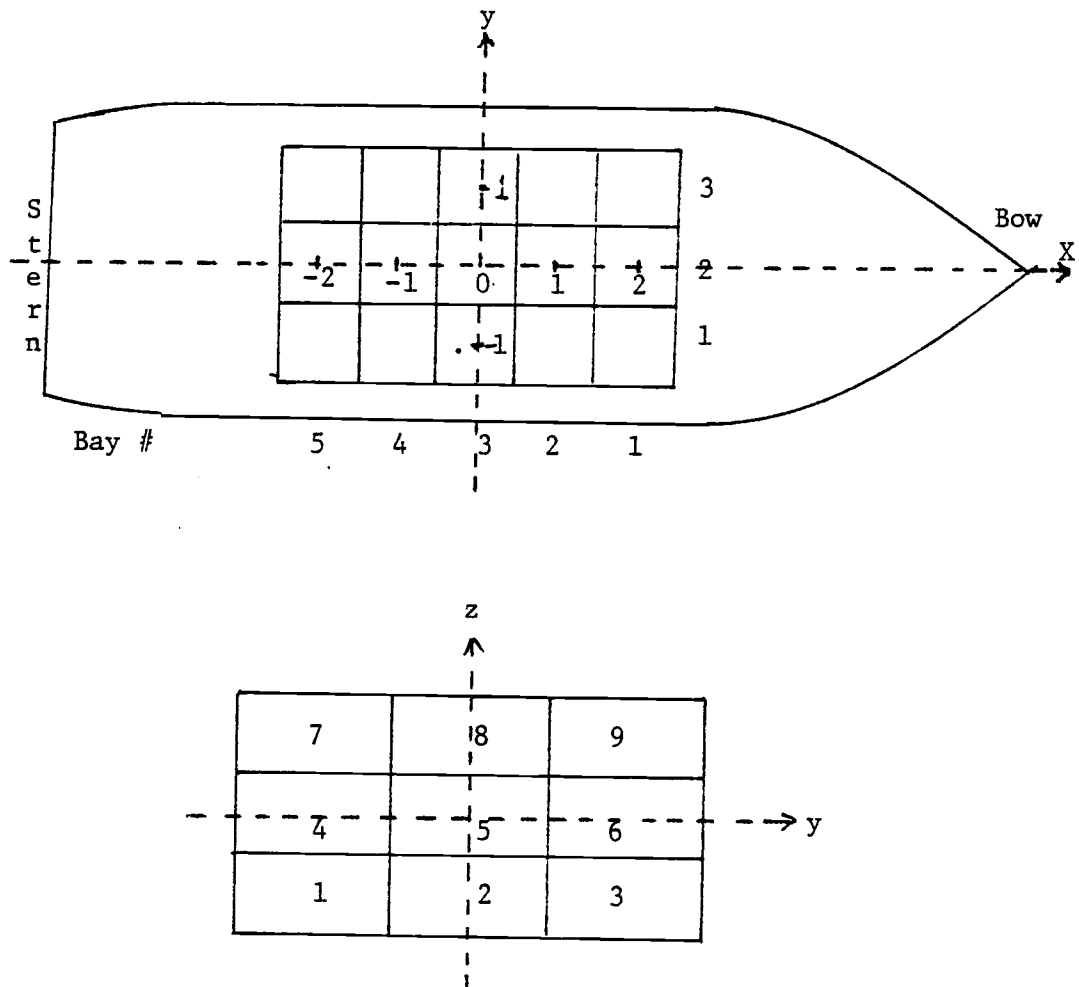


Figure 4.2 Hypothetical ship structure in example problem

TABLE 3. SHIP CELL DATA FOR  
SIMPLIFIED PROBLEM

| Cell | Bay | $\bar{X}$ | $\bar{Y}$ | $\bar{Z}$ | Cell | Bay | $\bar{X}$ | $\bar{Y}$ | $\bar{Z}$ |
|------|-----|-----------|-----------|-----------|------|-----|-----------|-----------|-----------|
| 1    | 1   | 2         | -1        | -1        | 26   | 3   | 0         | 0         | 1         |
| 2    | 1   | 2         | 0         | -1        | 27   | 3   | 0         | 1         | 1         |
| 3    | 1   | 2         | 1         | -1        | 28   | 3   | -1        | -1        | -1        |
| 4    | 1   | 2         | -1        | 0         | 29   | 4   | -1        | 0         | -1        |
| 5    | 1   | 2         | 0         | 0         | 30   | 4   | -1        | 1         | -1        |
| 6    | 1   | 2         | 1         | 0         | 31   | 4   | -1        | -1        | 0         |
| 7    | 1   | 2         | -1        | 1         | 32   | 4   | -1        | 0         | 0         |
| 8    | 1   | 2         | 0         | 1         | 33   | 4   | -1        | 1         | 0         |
| 9    | 1   | 2         | 1         | 1         | 34   | 4   | -1        | -1        | 1         |
| 10   | 2   | 1         | -1        | -1        | 35   | 4   | -1        | 0         | 1         |
| 11   | 2   | 1         | 0         | -1        | 36   | 4   | -1        | 1         | 1         |
| 12   | 2   | 1         | 1         | -1        | 37   | 5   | -2        | -1        | -1        |
| 13   | 2   | 1         | -1        | 0         | 38   | 5   | -2        | 0         | -1        |
| 14   | 2   | 1         | 0         | 0         | 39   | 5   | -2        | 1         | -1        |
| 15   | 2   | 1         | 1         | 0         | 40   | 5   | -2        | -1        | 0         |
| 16   | 2   | 1         | -1        | 1         | 41   | 5   | -2        | 0         | 0         |
| 17   | 2   | 1         | 0         | 1         | 42   | 5   | -2        | 1         | 0         |
| 18   | 2   | 1         | 1         | 1         | 43   | 5   | -2        | -1        | 1         |
| 19   | 3   | 0         | -1        | -1        | 44   | 5   | -2        | 0         | 1         |
| 20   | 3   | 0         | 0         | -1        | 45   | 5   | -2        | 1         | 1         |
| 21   | 3   | 0         | 1         | -1        |      |     |           |           |           |
| 22   | 3   | 0         | -1        | 0         |      |     |           |           |           |
| 23   | 3   | 0         | 0         | 0         |      |     |           |           |           |
| 24   | 3   | 0         | 1         | 0         |      |     |           |           |           |
| 25   | 3   | 0         | -1        | 1         |      |     |           |           |           |

$$\begin{aligned}
A_1 + A_{41} + A_{81} + \dots + A_{1761} - B_1 &= 0 \\
A_2 + A_{42} + A_{82} + \dots + A_{1762} - B_2 &= 0 \\
&\vdots \\
A_{40} + A_{80} + Z_{120} + \dots + A_{1800} - B_{40} &= 0
\end{aligned}$$

b. Constraints

$$\begin{aligned}
B_i &\leq 1, \quad i = 1, \dots, 40 \\
S_j &\leq 1, \quad j = 1, \dots, 45
\end{aligned}$$

C-2: (Trim and Stability Constraints)

a. Defining equations

$$\begin{aligned}
5A_1 + 5A_2 + \dots + 10A_{11} + \dots + 15A_{16} + \dots + 10A_{21} \\
+ \dots + 15A_{26} + \dots + \dots + 15A_{40} - W_1 &= 0
\end{aligned}$$

$$\begin{aligned}
5A_{41} + 5A_{42} + \dots + 10A_{51} + \dots + 15A_{56} + \dots + 10A_{61} \\
+ \dots + 15A_{66} + \dots + 15A_{80} - W_2 &= 0
\end{aligned}$$

$$\begin{aligned}
5A_{1761} + \dots + 10A_{1771} + \dots + 15A_{1776} + \dots \\
+ 10A_{1781} + \dots + 15A_{1786} + \dots + 15A_{1800} - W_{45} \\
&= 0
\end{aligned}$$

b. Longitudinal limits (X-direction)

$$\begin{aligned}
2W_1 + 2W_2 + \dots + 2W_9 + W_{10} + W_{11} + \dots + W_{18} - W_{28} \\
- W_{29} - \dots - 2W_{37} - \dots - 2W_{45} \geq -0.5
\end{aligned}$$

$$\begin{aligned}
2W_1 + 2W_2 + \dots + 2W_9 + W_{10} + \dots + \dots + W_{18} - W_{28} - \\
\dots - 2W_{37} - \dots - 2W_{45} \leq 0.5
\end{aligned}$$

c. Horizontal limits (Y-direction)

$$-W_1 + W_3 - W_4 + W_6 - W_7 + W_9 - \dots + W_{45} \geq -0.5$$

$$-W_1 + W_2 - W_4 + W_4 + W_7 + W_9 - \dots + W_{45} \leq 0.5$$

d. Vertical limits (Z-direction)

$$-W_1 - W_2 - W_3 + W_7 + W_8 + W_9 - \dots + W_{43} + W_{44} + W_{45} \geq -2.0$$

$$-W_1 - W_2 - W_4 + W_7 + W_8 + W_9 - \dots + W_{43} + W_{44} + W_{45} \leq 2.0$$

## C-3: (Container Support Constraints)

$$S_1 \geq S_4 \geq S_7$$

$$S_2 \geq S_5 \geq S_8$$

.  
.  
.

$$S_{38} \geq S_{41} \geq S_{44}$$

$$S_{39} \geq S_{42} \geq S_{45}$$

## C-4: (Overstow Restrictions)

a. Defining equations for port designators

$$A_1 + A_2 + \dots + 2A_{11} + \dots + 3A_{21} + \dots + 4A_{31} + \dots + 4A_{40} - D_1 = 0$$

$$A_{41} + A_{42} + \dots + 2A_{51} + \dots + 3A_{61} + \dots + 4A_{71} + \dots + 4A_{80} - D_2 = 0$$

.  
.

$$A_{1761} + \dots + 2A_{1771} + \dots + 3A_{1761} + \dots + 4A_{1791} + \dots \\ + 4A_{1800} - D_{45} = 0$$

b. Constraints

$$D_1 \geq D_4 \geq D_7$$

$$D_2 \geq D_5 \geq D_8$$

$$D_3 \geq D_6 \geq D_9$$

$$D_1 + D_2 + D_3 \geq D_4 + D_5 + D_6 \geq D_7 + D_8 + D_9$$

$$D_{10} \geq D_{13} \geq D_{16}$$

$$D_{11} \geq D_{14} \geq D_{17}$$

$$D_{12} \geq D_{15} \geq D_{18}$$

$$D_{10} + D_{11} + D_{12} \geq D_{13} + D_{14} + D_{15} \geq D_{16} + D_{17} + D_{18}$$

.

.

.

$$D_{37} \geq D_{40} \geq D_{43}$$

$$D_{38} \geq D_{41} \geq D_{44}$$

$$D_{39} \geq D_{42} \geq D_{45}$$

$$D_{37} + D_{38} + D_{39} \geq D_{40} + D_{41} + D_{42} \geq D_{43} + D_{44} + D_{45}$$

Objectives:

0-1: Maximize  $B_1 + B_2 + \dots + B_{40}$

0-2: Minimize transtainer move distance.

Since all containers are located within a single section, the total move distance is the sum of distances between each successive pair of yard locations. Thus,

a. Defining equations

$$A_1 + 4A_2 + 8A_3 + 5A_4 + \dots + 6A_{40} - P_1 = 0$$

$$A_1 + 4A_{42} + 8A_{43} + \dots + 6A_{80} - P_2 = 0$$

.

.

.

$$A_{1761} + 4A_{1762} + \dots + 6A_{1800} - P_{45} = 0$$

$$P_1 - P_2 + F_1 - G_1 = 0$$

$$P_2 - P_3 + F_2 - G_2 = 0$$

.

.

.

$$P_{44} - P_{45} + F_{44} - G_{44} = 0$$

b. Objective function

$$\text{Minimize } P_1 + F_1 + F_2 + \dots + F_{44} + G_1 + G_2 + \dots + G_{44}$$

The two objectives can be combined as follows:

$$\begin{aligned} \text{Maximize } B_1 + B_2 + \dots + B_{40} - (P_1 + F_1 + F_2 + \dots + F_{44} \\ + G_1 + G_2 + \dots + G_{44}) \end{aligned}$$

4.1.3 Experience with the simplified problem

The containership loading problem was formulated as a multi-constrained integer assignment problem. A CLP problem involving 1,000 containers is not an unusually big one. An integer problem of such a magnitude would involve at least one million 0-1 variables. It is immediately clear that the integer program

formulation is not a feasible approach to CLP solution.

This conclusion was supported by the experience of attempting to obtain an LP (linear programming) solution of the simplified problem. Formulated as an LP, the problem had 1,976 variables and 252 constraints. The non-zero density in its coefficient matrix was only 1.57 percent. Bounds were set as tight as possible on all variables. The PREVISED algorithm in the Multi-Purpose Optimization System (MPOS) developed by the Northwestern University was used to run the problem.

Initially, the problem was too big to fit into the core of CYBER 720 computer at O.S.U. Computer Center, which allowed a maximum of 320 KB (octal) of central memory to batch jobs. The problem fit into the core after eliminating the associated constraints and the part of objective function for minimizing transtainer move distance. With the execution time limit set at 250 seconds (octal), the MPOS gave no solution. The initial tableau output was not quite completed at the end of the time limit. Input data handling time alone took 69 seconds. The remainder of time was used up for data transfer and data conversion to form the initial tableau.

It was learned that an LP of this size would cost between \$300 and \$500 per run using the CDC APEX system. No attempt at solving the integer problem was made. It is clear, from this experience, that a day-to-day solution of CLP using the integer program structure is neither economical nor feasible.

#### 4.2 Other Formulations

As a second attempt at using a traditional optimization technique to solve CLP, a network formulation was tried. A general problem encompassing a subset of CLP with 0-2 objective excluded may be formulated as a "generalized assignment model,"



as defined in Ross and Zoltners (1979). Its structure is as follows:

$$\text{Minimize } \sum_{i \in I} \sum_{j \in J} c_{ij} \cdot X_{ij} \quad (4-1)$$

$$\text{Subject to } \sum_{i \in I} X_{ij} = 1 \text{ for all } j \in J \quad (4-2)$$

$$\sum_{j \in J} f_{ij} X_{ij} \leq b_i \text{ for all } i \in I \quad (4-3)$$

$$\text{and } X_{ij} = 0 \text{ or } 1 \text{ for all } i \in I, j \in J$$

The index set  $I$  and  $J$  would correspond to a set of containers and a set of ship cells, respectively. The objective coefficients  $c_{ij} = 0$  to accomplish (0-1). The (C-1) constraints fall into constraints of equation (4-1). The other constraints may be expressed by constraints of (4-3). Ross and Soland (1975) report construction of a computer code that solves generalized assignment problems.

Their program required  $(4v + 7m + 10n + 5300)$  words to solve a generalized assignment problem with  $m$  agents (index  $I$ ),  $n$  tasks (index  $J$ ) and  $v$  variables. This is translated to 425,300 words (1,477K octal) for a CLP of 1,000 containers and 1,000 ship cells. This excessive storage requirement makes the code unacceptable as a day-to-day solution technique for CLP. Furthermore, optimization of yard material handling (0-2) is not incorporated in this formulation. Computational speed report is limited to problems with  $m = 20$  and  $n = 50$ .

It is well known that a certain class of LP problems can be expressed as "pure networks." Pure networks, however, are unable to represent LP problems with fractional coefficients. Flows on arcs in pure networks are restricted to discrete integer values. Generalized network (GN) overcomes this restriction and allows

any fractional gain on arcs. Glover, Hultz, Klingman and Stutz (1976 and 1978) report establishment of GN as a computer-based planning tool.

The GN problem represents a large class of LP problems. Pure networks require at most two non-zero entries in each column of the coefficient matrix, where the entries must be 1 and -1. The GN problem, by allowing other non-zero doubletons as well as singletons in a column, is the broadest classification of linear network related problems. Applications of GN formulation arise in areas of resource allocation, production, distribution, scheduling, capital budgeting, and so on.

A generalized network, like a pure network, is represented as a directed graph.<sup>1</sup> The coefficient matrix is transformed so that if a column has two non-zero entries, at least one of these is -1. In this way, a directed arc is formed that leads from the node associated with the -1 to the node associated with the other non-zero entry. If both entries are -1, the arc may be directed either way. Columns with single non-zero entries cause arcs incident on only one node.

Consider the following illustrative LP problem as formulated by Glover et al. (1978).

$$\begin{array}{rcl}
 \text{Minimize} & X_{12} + 5X_{13} + 3X_{23} + X_{24} - 4X_{32} - 9X_{34} & \\
 \text{Subject to} & -X_{12} - X_{13} & = -5 \\
 & 2X_{12} - X_{23} - X_{24} + 1/3 X_{32} & = 0 \\
 & 1/2 X_{13} + X_{23} - X_{32} - X_{34} & = 0 \\
 & 1/5 X_{24} + 3X_{34} & = 10
 \end{array} \quad \left. \vphantom{\begin{array}{rcl}} \right\} (4-4)$$

---

<sup>1</sup>A graph consists of nodes and arcs connecting them. A directed graph has a sense of direction attributed to each arc, so that one node is the origin and another the destination of an arc.

$$0 \leq x_{12} \leq 3, 0 \leq x_{13} \leq 4, 0 \leq x_{23} \leq 6$$

$$0 \leq x_{24} \leq 5, 0 \leq x_{32} \leq 3, 0 \leq x_{34} \leq 7$$

The GN for this LP is shown in Figure 4.3. Each constraint corresponds to a node and each column in the coefficient matrix to an arc. The simplified problem of section 4.1, then, would involve 252 nodes and 1976 arcs.

The computer code which Glover et al. developed (NETG) reportedly occupied  $(8N + 4A + 8500)$  words of central memory, where  $N$  is the number of nodes and  $A$  number of arcs. For the simplified problem, we would need approximately 18,420 words (43K octal) of core space. For a CLP involving 1,000 containers and 1,000 ship cells, we would need at least 4.5 million words of central memory space. Moreover, the proprietary nature of NETG makes it unavailable to most researchers. Another major hindrance to this approach is the complexity of constructing a preprocessor program which takes in a primitive data set and converts it into data suitable for NETG. It is clear from these facts that the GN formulation would not be a practical model for CLP solution, although it offers a useful solution technique for many LP problems.

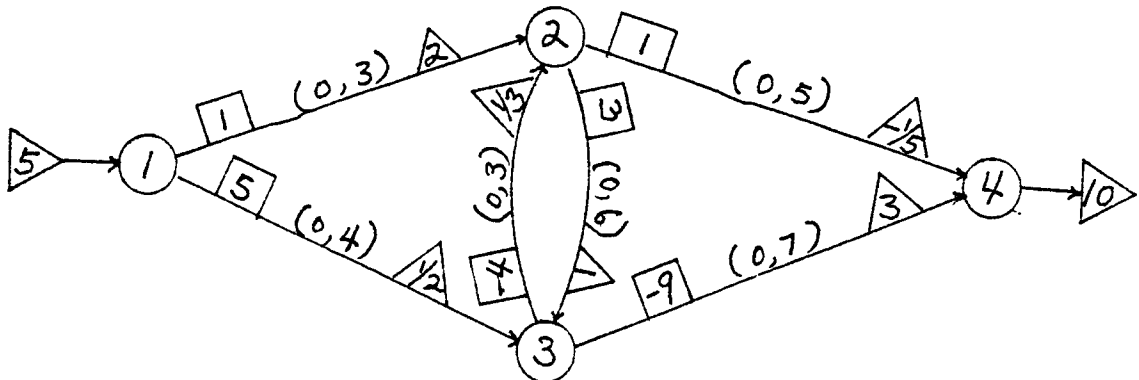


Figure 4.3 An example of generalized network

### 4.3 Dynamic Programming Formulation

#### 4.3.1 Transtainer Routing Problem (TRP)

Formulation attempts so far described did not materialize in an acceptable and satisfactory model for real-world application. They also revealed the difficulty of incorporating the objective of minimizing yard material handling with integer-programming related models. Since most ports are much concerned about minimizing their ship turnaround times in order to make themselves attractive to visiting ships, it seems imperative that a good CLP model should include this objective explicitly.

Unlike other real-world problems with many variables, the CLP problem requires that each container be treated as an identity and its exact shipboard assignment needs to be produced by the model. In handling containers, shipside cranes and yard transtainers move in discrete steps. For a pair consisting of a crane and a transtainer, the work sequence is the same for both pieces of equipment. The same set of containers is picked up by a transtainer from the yard slots and delivered to cranes by yard trucks in exactly the same order it is loaded into ship cells by a crane. Thus, the work sequence of a transtainer matches that of the crane in the same pair.

The movement of a transtainer, assuming one transtainer is used to pick up all containers in a loading set, is a multi-stage problem in the sense that after each container pick-up its next move consists of the yard location of the next container in a loading sequence.

A GLP, as discussed in section 3.1, is made with all constraints in mind except C-2 (stability), C-4 (deck strength), C-5 (racking strength), and C-6 (stack height). These constraints are not included in GLP because they cannot be verified

until complete container-cell assignments are produced. If we set aside these four constraints for the moment, then a CLP problem is one of finding an optimal transtainer movement sequence which satisfies the constraints constituting a GLP.

This multi-stage problem can be expressed as a network of Figure 4.4.

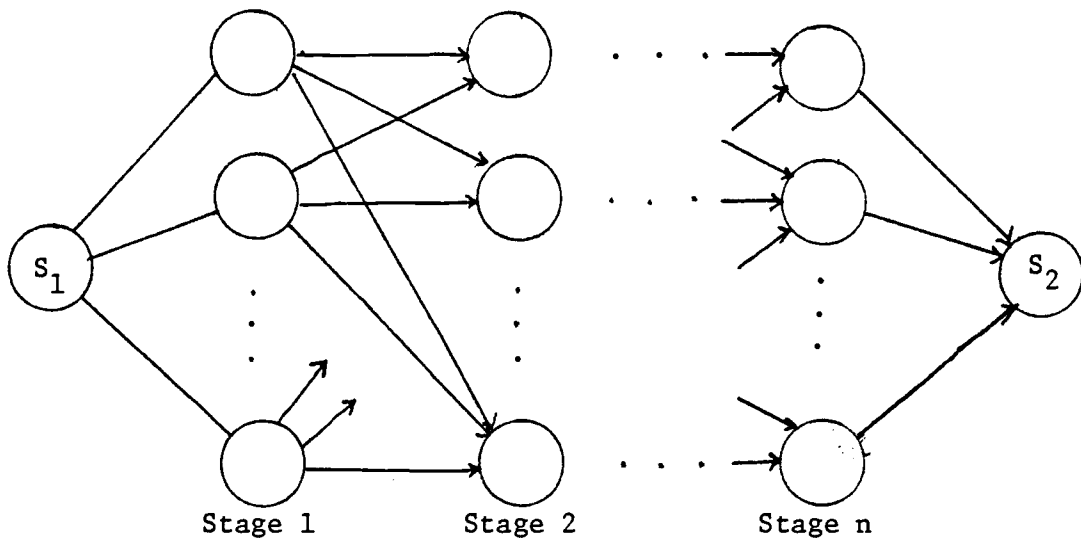


Figure 4.4 A multi-stage network

The terminal nodes  $S_1$  and  $S_2$  represent start and ending positions of transtainer, respectively. Each column of nodes represent possible container locations to which the transtainer can move from its previously occupied position. In its primitive form, this network involves as many nodes in each column as the total number of containers in a load set. The number of stages also equals the total number of containers.

Since a GLP specifies what container group to pick up in each stage, the above network is reduced to one with as many nodes as number of containers in the group for each stage. Thus,

for a 1,000 container CLP with 20 groups each consisting of 50 containers, the network would have approximately 50,000 nodes. It is further noted that as each container of a particular group is picked up, there remains one less container location for the transtainer to move to. Therefore, the number of nodes in each succeeding stage decreases by one in each group picked serially. Then, the network would contain

$$20 \sum_{i=1}^{50} n = 20 \left( \frac{50 \cdot 51}{2} \right) = 25,500 \text{ nodes.}$$

Arcs in the network represent possible transtainer movement plans. A movement plan consists of movement details that specify where to go, and which container to pick. Thus, arcs represent alternative ways for a transtainer to move and to pick the next container.

For the reduced network described above, we would have the following number of arcs:

$$20 \sum_{n=1}^{49} n(n+1) = 20 \left[ \sum_{n=1}^{49} n^2 + \sum_{n=1}^{49} n \right] = 833,000$$

The transtainer routing problem (TRP), then, can be formed as a deterministic dynamic programming problem with the following properties:

Stage: Selection of each container.

State: 1. Transtainer location.

2. Scatter pattern of remaining containers in the yard.

Objective function: Minimize TTMD.

Decision: What container to pick next.

The recursive relationship can be formulated as follows:

Recursive Relationship:

$$f_k^*(S) = \min_{X_k} \{d_{S, X_k} + f_{k-1}^*(S)\} \quad (4-5)$$

where  $X_k$ : The kth container to pick,  
 $S$ : Transtainer location and scatter pattern  
 at the beginning of stage k,  
 $d_{S, X_k}$ : Distance to be traveled by transtainer to  
 pick container  $X_k$  starting at position de-  
 scribed in state  $S$ ,  
 $f_{k-1}^*(S)$ : Minimum travel distance required to pick  
 k-1 containers and start stage k with  
 initial state  $S$ .

Each ending state of a stage becomes an initial state of the next stage. The state space,  $S$ , needs to be a complete array which not only indicates transtainer position but also location of each remaining container in the yard.

For a TRP involving 1,000 containers, it would require an array with a size of 833,833,000 to store state information of all arcs so the optimal path could be identified at the end of the forward pass. This is currently an impossible storage requirement. If a dynamic program formulation is to be used, a change must be made to reduce the storage requirement. It is observed that the problem complexity depends much on definition of state space and stage.

#### 4.3.2 Simplification for Manageability



The process of science and technology is primarily due to the human ability to identify similar objects and traits, and to apply categorization on them. Degree of complexity in any system

is determined by the level of detail that is adopted for observation and analysis. Here, an appropriate distance between the system of interest and the observer must be determined. The containership loading problem is certainly a very complex one under close examination. Now that a close-up examination is complete, power of simplification is called upon to render it a manageable one for practical solution.

Weinberg (1975) categorizes objects of study into three regions in terms of their complexity and randomness:

- I. Organized simplicity (machines)
- II. Unorganized complexity (aggregates)
- III. Organized complexity (systems)

This categorization is depicted in Figure 4.5.

 = Analytical treatment  
 = Statistical treatment

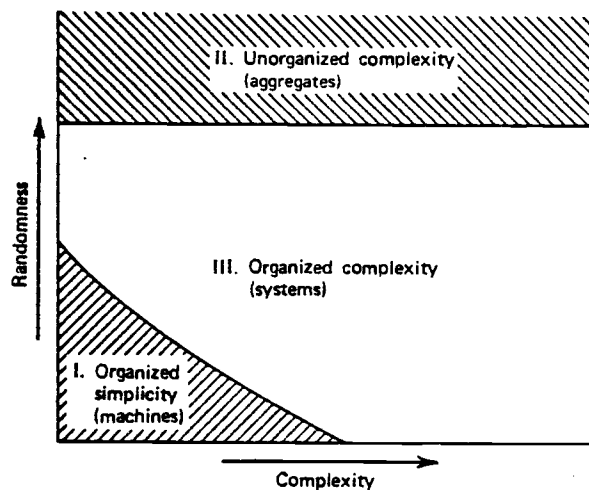


Figure 4.5 Types of systems with respect to complexity and organization  
 (Source: Weinberg, 1975)



Region I consists of small populations with a great deal of structure which could be expressed as a system of a reasonable number of equations. Populations in Region II have sufficient randomness to be analyzed using statistics. In Region III lie all systems that are too complex for analytic treatments and too structured for statistics. A containerport system falls in the last category in that its operations cannot be expressed as a set of equations and containers cannot be treated as random occurrences. Selective and judicious application of general systems approach techniques come into much usefulness here.

From the dynamic program formulation of TRP described in section 4.3.1, the number of stages can be reduced considerably by redefining the stage as consisting of picking all containers of a given group to be loaded into a specific bay. This approach is made possible because of the fact that a GLP essentially consists of groups of containers to be loaded into receiving bays. Furthermore, groups occupy segregated portions within a bay. Loading a bay is done group by group. So, each stage has a requirement of a given number of containers of a given group to pick in the yard. Now, a TRP of 1,000 containers would involve, say, 70 stages, as formulated with the revised definition of stage.

In each stage our attention is focused at a particular group to pick. The transtainer movement alternatives are then restricted to occupying rows of this group only. Now the decision at each stage is what containers among those remaining of this group to pick to fill the requirement of the stage. An appropriate heuristic rule must be structured so that a reasonable number of alternative transtainer movement plans can be investigated. As the algorithm progresses forward in the dynamic program solution process, the number of alternatives generated would decrease for a given initial condition and a group because the number of containers remaining in that group decreases.

The state space can also be redefined as follows: Since each stage is concerned with only one group, the state space,  $S$ , consists of the following conditions:

1. Initial transtainer row position.
2. Scatter pattern of remaining containers of the group picked in a stage.

It is anticipated, however, that this definition, although a much aggregated one from the original, would generate too many nodes to be manageable. Here, only the first condition, initial transtainer row position at the beginning of a stage, is used to define the state space. This is not a comprehensive definition, but it would reduce the TRP to a manageable size. Solutions from this formulation may not be truly optimal, but it is hoped that they would approach the optimal with sufficient proximity.

The recursive relationship for this formulation can be written as follows:

$$f_k^*(S) = \min_{j \in A_k} \{ f_{k-1}^*(\xi_{k-1}) + d_{S, M_j} \} \quad (4-6)$$

where  $f_k^*(S)$ : Minimum total transtainer move distance by the end of stage  $k$ , beginning the stage at row position  $S$ .

$A_k$ : Set of movement plan indices for stage  $k$  corresponding to initial state  $S$ .

$M_j$ : A movement plan,  $j \in A_k$ .

$d_{S, M_j}$ : Transtainer travel distance required to pick a given number of containers beginning at row position  $S$  and moving according to movement plan  $M_j$  in stage  $k$ .

$f_{k-1}^*(\xi_{k-1})$ : Minimum TTMD by the end of stage  $k-1$ , ending at row position  $\xi_{k-1} = S$ .

The dynamic program structure described above is embodied as a subroutine within the computerized CLP model which will be developed later. Whether this structure would maintain the corresponding computer program size at a manageable level needs to be ascertained. This work is completed through a simulation in which key problem parameters are varied and corresponding changes in program array requirements are noted. Explanations on this work will also appear later.

At this point, we know that the dynamic program structure will enable us to obtain an optimal transtainer routing plan for each load set. This routing plan would specify not only the sequence of rows to visit but also the number of containers of a particular group to pick among those located in a particular row. If the container list is sorted so that containers may be selected from top to bottom and from column A to column F in each visiting row, then an appropriate set of container-ship cell assignments could be obtained. Thus, an optimal transtainer routing plan would produce an optimal load plan. The next step, naturally, is to see whether a load plan thus obtained would satisfy all constraints. Finally, any violated constraint must be remedied so that an optimal and feasible load plan could be obtained. Detailed discussion on the model structure is presented in the next chapter.

## 5. COMPUTERIZED OPTIMIZATION MODEL FOR CONTAINERSHIP LOADING (COMCL)

### 5.1 Philosophy Behind the Model

Every model has a philosophy or thought behind it. This one is no exception. A number of facts were assembled to form our model structure.

First, although conventional ships were converted to carry containers aboard during the early stages of containerization, larger ships have been built from the keel up as fully dedicated containerships. These ships provide very stable loading platforms and their GM allowable ranges are rather broad. This makes ship stability constraint of less concern to ports which fill partial loads, although the ultimate nature of this constraint still holds true. The same can be said of trim limits. Also, the high cube limitation does not apply at all to most new ships. Thus, the structure of new containerships allows us to be more concerned about yard material handling than before when meeting ship stability conditions was of prime concern.

Second, ship turnaround time depends on how transtainers and cranes are routed. Crane and transtainer movements should be planned so that a smooth distribution of work load among work gangs is accomplished while the total loading time is minimized. Work sequencing of cranes and transtainers is a difficult task in that it demands consideration of a multitude of constraints and objectives. Computerization of sequence planning would constitute a good-sized research project and is left for further work. At present, our load planning model assumes that this sequencing activity is done properly by human planners. Then, the model is called upon to the task of minimizing the total transtainer move distance for a given bay-by-bay work sequence.

Here, we recognize the suitability of dynamic programming as a solution technique with each bay work as a stage in the problem structure.

Instead of sacrificing yard material handling effort to ensure loading restrictions, COMCL produces a load plan which minimizes total transtainer move distance. Then, any violation of stability or load restriction nature could be corrected through appropriate heuristic rules. These rules are constructed so that pairwise exchanges of container assignments are made towards attaining feasibility as fast as possible while the increase in total transtainer move distance (TTMD) due to exchanges is kept at a minimum.

With so many conditions and constraints to consider, this model would lack completeness if it did not include an interactive capability with the human operator. Designation of special cell locations for a few special containers should be made possible to the operator. Final adjustments on a computer-generated load plan should be easy to do. Thus, the model would take the nature of aiding the planner instead of replacing him.

These ideas were fully embraced by COMCL. Testing of the model using actual load data sets proved the validity of this approach, as discussed later.

## 5.2 General Structure of COMCL

COMCL is an optimization model in that it initially obtains a load plan which minimizes transtainer yard movement. Then, ship stability and loading restrictions are checked for feasibility. If any of these are violated, the model conducts one or more pairwise exchanges of container-cell assignments towards the direction of meeting them while minimizing the increase in transtainer travel distance caused by the interchanges. Finally, COMCL allows the planner to interact with it to make pairwise

exchanges that he feels necessary to implement any special considerations not incorporated explicitly in the model.

Figure 5.1 depicts the general structure of COMCL. Circular elements represent data files and rectangular blocks are programs/subroutines. COMCL consists of two separate programs: Program DATPREP and Program CLP. Program DATPREP takes in two files as input: CTRAW containing individual container information and RUNDAT with operational information such as date, ship's name, voyage number, port names, line names, etc. File CTRAW is in a form very close to primitive container information in actual use. Each record of CTRAW corresponds to one container. Program DATPREP generates "Operator Recapitulation Sheet" and "Tonnage by Line" tables, which are compilations of container statistics for a particular ship load. Then, DATPREP prepares data for the main program's use. In particular, it generates CTFIL which is read in by program CLP in the second state of COMCL execution. Both programs were written in FORTRAN IV language.

The following briefly describes the nature of each input file for Program CLP:

- YLOUT .. Yard structure file with yard section layout information.
- CTFIL .. Container file generated by DATPREP from CTRAW and RUNDAT.
- GPLAN .. Contains general loading plan and ship's condition prior to loading operation.
- FSHIP .. Fixed ship file containing ship structure and stability data.

File FSHIP is unique for each ship. Subsequent visits by the same ship do not necessitate changes or creation of a new FSHIP. File YLOUT is unique for a port. Once made, therefore,

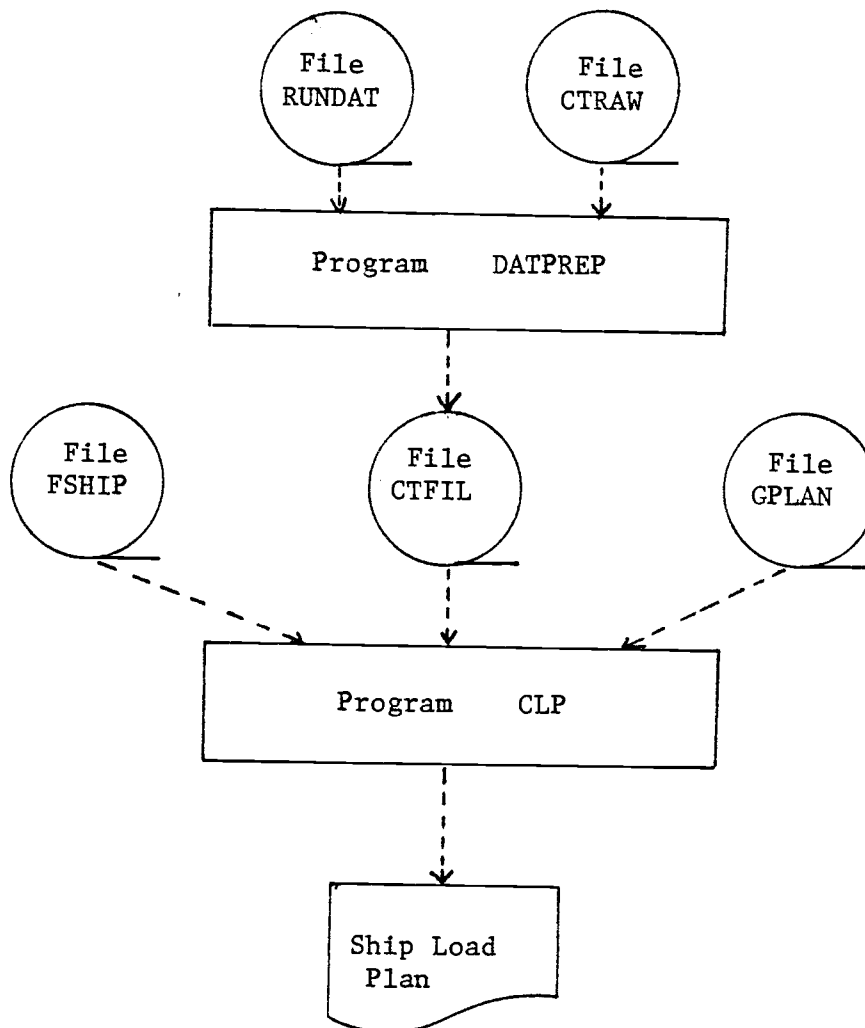


Figure 5.1 COMCL model structure

YLOUT need not change except to accommodate changes in port facilities. File RUNDAT needs some minor changes for each voyage. Files CTRAW and GPLAN provide the variable data unique for a given voyage load. Descriptions and explanations on data files follow.

### 5.3 Input Files

Data file specifications may be found in Appendix C. The following files, however, demand explanations as to why certain file structures are adopted.

#### 5.3.1 File CTFIL and container list

The major part of file CTFIL is the set of container records, each record corresponding to a single container. Program DATPREP conducts the following sorting in a number of stages in order to produce the container list in file CTFIL:

- a. Sort by port of destination according to order of call.
- b. Sort by size-type code in the following sequence:
  1. 20' Full
  2. 20' Reefers
  3. 20' Empties
  4. 20' Specials
  5. 40' Full
  6. 40' Reefers
  7. 40' Empties
  8. 40' Specials
- c. Sort by row position number in ascending order.
- d. Sort by yard column designator in ascending alphabetical order (from column A to column F).



3. Sort by yard tier designator in descending alphabetical order (Zulu, Top, Center, and Bottom).

This sorting is necessary so that container-to-ship cell assignments can be made in the order of proper picking among containers of a group stacked in a yard row position. Containers are usually picked from the nearest column (column A) to the seaward side of a section to the farthest column (column F), and from top to bottom within a column. This order of picking makes rehandling of containers unnecessary.

When program CLP reads in file CTFIL, each container record is stored in several different arrays with a common array index. The index value of a container is as determined within program DATPREP after completion of the sorting described above. Hence, a container is identified with a unique index number throughout program CLP execution.

### 5.3.2 File YLOUT and yard structure

In order to implement the objective of transtainer travel distance minimization, containerport yard structure needs to be specified in a relevant way. File YLOUT contains the number of yard sections, their identification numbers and relative dispositions of sections. Relative disposition of the  $i$ th and the  $j$ th sections with order significant is represented by array IAYS ( $i,j$ ), and its value is assigned according to the following rules:

| <u>IAYS (<math>i,j</math>)</u> | <u>Relative Disposition</u>  |
|--------------------------------|--|
| 1                              | Sections $i$ and $j$ are adjacent and section $i$ is to the right of section $j$ . |
| 2                              | Sections $i$ and $j$ are adjacent and section $j$ is to the right of section $i$ . |

- 3 Both sections are in the same column to the right of center traffic aisle.
- 4 Both sections are in the same column to the left of traffic aisle.
- 5 Sections i and j are in different columns with section i in the right column, and they are not adjacent.
- 6 Sections i and j are in different columns with section i in the left column, and they are not adjacent.
- 7 Sections i and j are one section ( $i=j$ ) and it is in the column left of the center traffic aisle.
- 8 Sections i and j are one section ( $i=j$ ) and it is in the column right of the center traffic aisle.

Figure 5.2 illustrates the relative disposition relationships of yard sections. The above rule was made to suite the structure of yards with rectangular yard sections and overhead-crane type material handling. Different yard sections may require a greater range of LAYS (.) values than the one defined above. It is assumed that yard sections are of the same size.

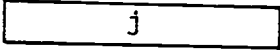
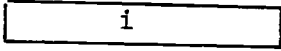
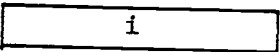
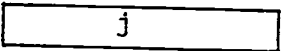
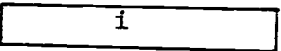
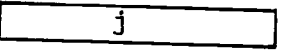
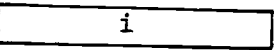
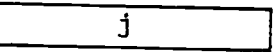
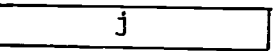

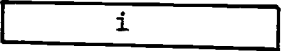
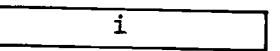

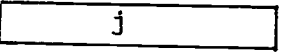
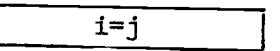
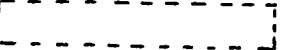

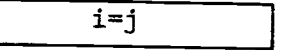
| <u>Disposition</u> | <u>Illustration</u>  |  |
|--------------------|--|--|
| 1                  |   |    |
| 2                  |   |    |
| 3                  |  | <br>     |
| 4                  | <br> |  |
| 5                  |   | <br>   |
| 6                  |   | <br> |
| 7                  |   |    |
| 8                  |   |    |

Figure 5.2 Relative dispositions of yard sections

### 5.3.3 File FSHIP and ship structure

Examination of a number of containership structures revealed the fact that a ship can be imaginably divided into a few sectors where all bays within a sector share a common set of vertical centers of gravity (VCG's). Tier levels are the same in the bays of a sector. It is reasonable, therefore, to specify data for the sectors and the sector in which each bay belongs.

Load restrictions applied to on-deck bays are different from those for under-deck bays. For this reason, an on-deck bay and an under-deck bay are regarded and dealt with as separate bays, although they may have a common bay number in many ships if they share the same location longitudinally.

An under-deck bay is assumed to take in only one-size containers, while some on-deck bays can have both 40-foot and 20-foot containers stacked in them. In a given column of a bay, however, only one-size containers are assumed to be allowed. For an on-deck bay with both sizes allowed, appropriate longitudinal center of gravity (LCG) and VCG values are entered in file FSHIP.

An important part of FSHIP consists of fixed stability data such as those for a light ship condition. GM and trim limits are also entered here. Trim limits may vary seasonally due to changes in sea conditions. Trim limits are the only part of FSHIP which may vary from one voyage to another. Once created and thoroughly checked for accuracy, FSHIP for a ship should not need any change thereafter.

### 5.3.4 File GPLAN structure

Data for file GPLAN come from the following source documents:

| <u>Document</u>                 | <u>Data</u>  |
|---------------------------------|--|
| a. Color Schematic<br>(GLP)     | <ul style="list-style-type: none"> <li>•• Number, type, size, and destination of containers for each bay to load.</li> <li>•• Total number of bays to load.</li> <li>•• Total number of containers to load.</li> <li>•• Loading patterns for each bay.</li> </ul>  |
| b. Ship Bay Plans               | <ul style="list-style-type: none"> <li>•• Weight sum, vertical moment (VM) and longitudinal moment (LM) of containers on-board prior to loading in each bay.</li> <li>•• Number of high cubes already loaded in each under-deck column.</li> <li>•• Weight of already loaded containers in each on-deck column.</li> </ul> |
| c. Ship Work Sequence<br>Sheets | <ul style="list-style-type: none"> <li>•• Order in which bays are to be loaded.</li> <li>•• Number and size of containers to load in each bay by group.</li> <li>•• Cumulative count of containers handled by the end of each work stage.</li> </ul>   |

The DATPREP output lists containers categorized into a number of groups. A group is a homogeneous collection of containers with the same destination, size and type of content. A GLP, then, specifies how many of each group to load into what bays in what load patterns. Work Sequence sheets specify the order of loading for bay-group combinations. A load pattern (LPAT) specifies how many to load starting in which cell in what tier of a bay.

## 5.4 Program CLP Organization

Program CLP is the main part of COMCL. It consists of a number of subroutines and function subprograms. General program flow is as shown in Figure 5.3. The following provide a description of each routine in terms of its function and output. Detailed flowcharts and program listings are found in the Appendices.

### 5.4.1 Program CLP

This is the main routine which reads in input data files and calls different subroutines. The final stage of this program consists of an interactive feature. Thus, program CLP provides input/output service, subroutine routing logic, and the interactive capability.

After input data files are read, CLP calls subroutine DYNAMIC to generate an optimum transtainer routing plan which minimizes the TTMD. Then, subroutine ASSGN is called to make individual container-cell assignments according to the generated transtainer routing plan. Stability and loading restrictions such as number of high cubes and weight on-deck are checked on this load plan in subroutine FCHECK. Any violated constraints and locations of such violations are identified in it also. Then, FCHECK calls subroutine STABLE to compute the most favorable values of GM and trim with the given set of containers regardless of yard locations if either GM or trim limits is violated.

By this stage, "Stability Calculation" sheet and "Stability Summary Table" would have been printed out along with a list of any violations for the operator to see. Then, the operator is asked by the program whether he desires to improve the stability conditions through subroutine REMEDY. This query is posed only

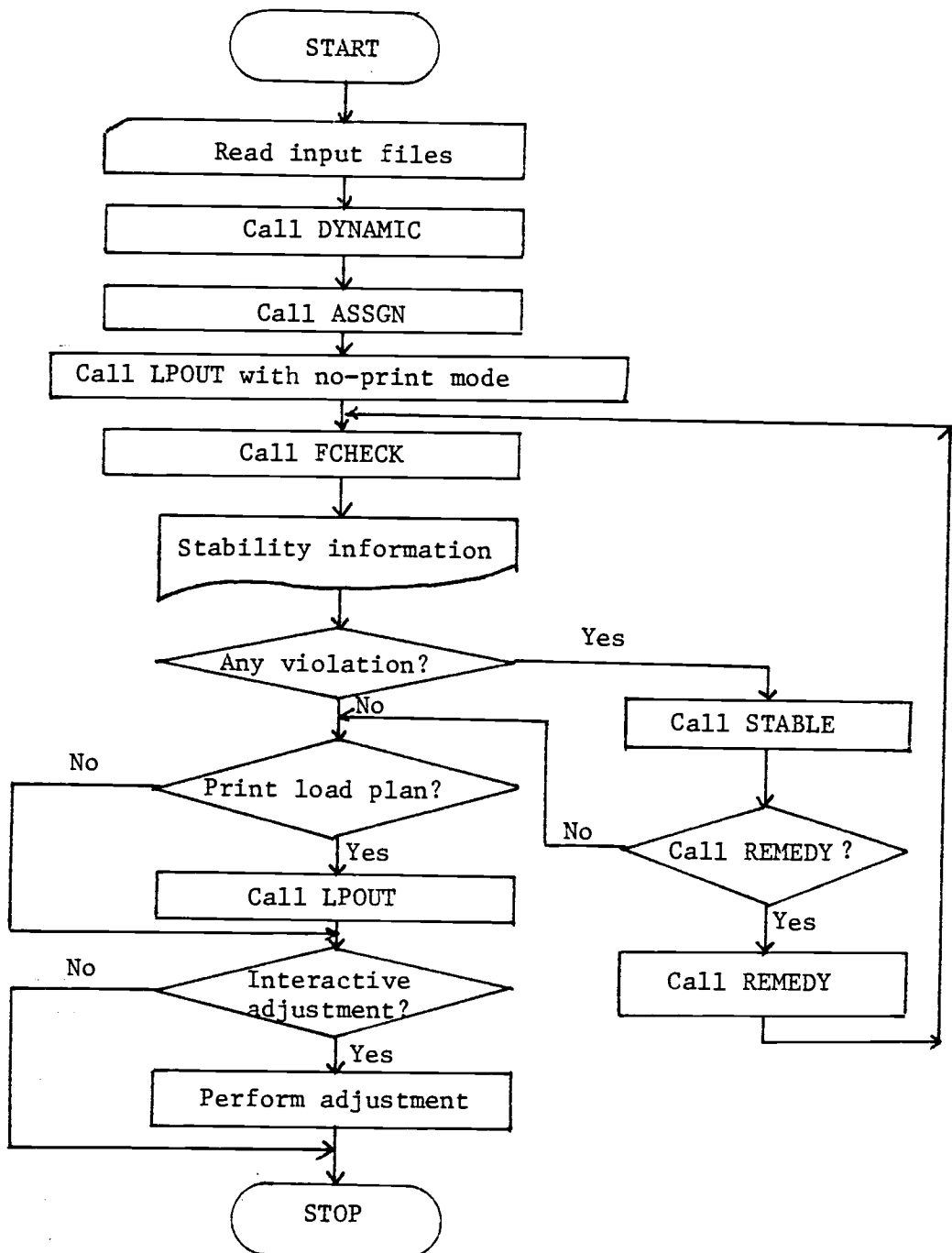


Figure 5.3 Program CLP Flowchart

if any violations have been detected. Otherwise, the operator is asked whether he desires the load plan with entire container-cell assignments be printed out.

Subroutine REMEDY makes iterations towards feasibility. Each iteration consists of a pairwise exchange of assignments. At conclusion of subroutine REMEDY execution, program CLP calls subroutine FCHECK again to recompute stability figures. Then, the desirability of load plan output is asked again.

Program LPOUT lists the load plan (container-to-cell assignments) thus obtained. Each line of output corresponds to one container-cell assignment. This load plan provides the loading instructions to both the yard crew and ship gangs because the order of picking containers in yard is identical to the order of loading them on-board.

Finally, program CLP allows the operator to make use of its interactive feature to make any last-minute adjustments to the load plan generated by COMCL. Each exchange is made by entering container sequence numbers of a pair of containers whose locations are desired to be switched. Feasibility of each requested exchanged is checked internally before any exchange is made.

#### 5.4.2 Subroutine DYNAMIC

The job of loading a particular group of containers into a certain bay constitutes a stage in the dynamic programming formulation of the transtainer routing problem (TRP). The solution to a TRP is given by a set of move details for each stage. An initial state of a stage is the location of a transtainer(s) before picking activity for that stage begins. An initial state of a stage is an ending state of the immediate preceding stage. The objective of a TRP is to minimize TTMD.

In order to measure TTMD effectively, we employ the concept



of a row equivalent (RE). A row equivalent is the length of time required to travel a distance of a 20-foot container side dimension. A row movement within a yard section and a section change move need to be made commensurable. Since the common factor shared by these two different types of transtainer moves is time, RE is defined as the above. The RE's for section change moves are established as follows:

Given:

1. Time to travel along an entire yard section: 1.5 minutes
2. Time to move to a non-adjacent section: 10 minutes
3. Length of a yard section: 74 row positions.
4. Distance between two adjacent sections: 10 row positions.

Conversion:

1. Transtainer move to an adjacent section (NER1):  
10 RE's.
2. Transtainer move to a non-adjacent section (NER2):  
 $(10 \text{ min}/1.5 \text{ min}) \times 74 \text{ RE's} = 493 \text{ RE's}.$

Initially, subroutine DYNAMIC was formulated to handle the one-transtainer TRP. It was later expanded to handle two-transtainer TRP as well. Since most modern containerhips require two gangs or more, each consisting of a crane and a transtainer team, it is only natural that COMCL possess such a capability.

Stages are constructed from the work sequence data. They provide cumulative counts of containers handled. These include unloaded containers and rehandles, and provide a measure of each gang's work progress. Assuming each container, regardless of its size, type, and bay to load, takes an equal length of

handling time, it constitutes a measure of work progress on a common scale. Here, three alternative methods of executing subroutine DYNAMIC are open to a planner:

- a. Parallel Method: Both transtainers operate simultaneously.
- b. Serial Method (T1-T2): Transtainer #1 operation is planned first and is followed by transtainer #2 operation.
- c. Serial Method (T2-T1): Transtainer #2 operation is planned first and is followed by transtainer #1 operation.

The work sequence for a parallel operation is constructed by using the aforementioned cumulative container counts. Stages are formed in the increasing order of their cumulative counts. In the second method, a higher priority is given to transtainer #1 than transtainer #2. That is, all stages pertaining to transtainer #1 is conducted before those of transtainer #2. A concatenated TRP of two TRP's is formed in this manner. The order of concatenation is the opposite in the third method of planning.

Of these three methods, the parallel method is the most realistic in that it handles one transtainer's stage at a time sequentially according to the way two transtainers are expected to operate in the yard. It must be pointed out, however, that the operational restriction of two transtainers not allowed in one section at the same time is not checked by subroutine DYNAMIC. Some type of a simulation would be necessary to implement this restriction rigorously.

The state space consists of initial row positions of both transtainers in the parallel method. It is obvious that the

number of arcs generated and nodes kept would be considerably greater in this method than in the other two. Also, each trans-  
tainer's total move distance (TMD) is kept for each combination  
of ending states. Movement plans correspond to only one trans-  
tainer within a given stage, however.

A solution within a stage corresponds to a set of move de-  
tails which define transtainer moves necessary to pick up a  
given number of containers of the group specified for the stage.  
Thus, each solution (movement plan) corresponds to an arc in the  
network representation of TRP. A node represents a transtainer  
state. Each arc is associated with a beginning node (initial row  
position) and an ending node (ending row position). This rela-  
tionship is depicted in Figure 5.4.

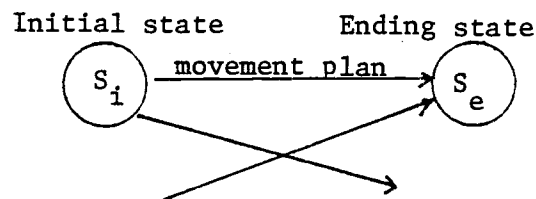


Figure 5.4 Arc-node relationship in  
dynamic program

Let the beginning node of an arc be called its root node.  
By keeping track of each arc's root node, the optimal solution  
path can be traced back starting at the ending node with the  
least TTMD in the final stage. Also, this feature allows compu-  
tation of TMD at each stage a simple task.

A move detail is specified by a move type and a target row  
number. A target row is one towards which a move is to be made.  
Move types are defined as follows:

- 1: Within-section picking move, increasing row direction.
- 2: Within-section picking move, decreasing row direction.
- 3: Empty move to a new row in the same section.
- 4: Empty move to a new row in another section.

Alternative ways of routing a transtainer within a stage are generated according to the following transtainer routing logic. A possessing row is defined as one that has at least one container of the group to pick. A possessing section contains at least one possessing row. The center refers to the traffic lane between two columns of yard sections.

#### TRANSTAINER ROUTING LOGIC

Step 1. Check if the current section is possessing. If so, go to Step 4.

Step 2. If current section is the only possessing section, go to 2A. Otherwise go to Step 3.

2A. Generate an upward pick move first. If enough has been picked, go to the pruning logic. Otherwise, go to 2B.

2B. Generate a downward pick move. If enough has been picked, go to the pruning logic. Otherwise, go to 2A.

Step 3. If current section is in the right column, go to 3A. Otherwise, go to 3B.

3A. Generate a "move down x rows, then pick upward" solution with x incremented by one at each iteration. Go to Step 4.

3B. Generate a "move up x rows, then pick downward" solution with x incremented by one at each iteration. Go to Step 4.

Step 4. Pick a possessing section to go to and see if enough can be picked there. If so, go to 4A. Otherwise, select the second possessing section and go to 4A.

4A. Form a section change move to the first possessing section.

4B. Generate a center-to-outward pick move. If enough has been picked, go to the pruning logic, then to 4C. If not, go to 4A with the second possessing section.

4C. Generate an out-to-center pick move. If enough has been picked, go to the pruning logic. If not, go to 4A with the second possessing section.

The advantage of dynamic programming in solving a problem of great complexity is in that only a fraction of all possible solution paths needs to be examined to obtain the optimal solution. This property is due to the pruning done in each stage. Different solutions which led to the same state by the end of a stage are compared on the basis of their TMD's up to that stage and only the best one is retained. The rest are discarded. This pruning function is embodied mathematically in the recursive relationship. Subroutine DYNAMIC utilizes the fact that we need to concern ourselves with only the possessing rows of one group in each stage. Other rows are temporarily left out of attention. This fact gives simplicity to the model structure.

The following notations are needed to explain the pruning logic:

- m : Old solution number with the same ending state in the same stage.
- n : New solution number.
- n' : Root solution of solution n.

- $f_m$  : TMD for solution m.  
 $d_n$  : Transtainer move distance corresponding to solution n within this stage.

The pruning is done in Subroutine DISCOMP as follows:

### PRUNING LOGIC

For each new solution n generated in a given stage,

- Step 1. Compute  $f_n = f_m + d_n$   
Step 2. Search the solution list of the same stage to find one with the same ending state as solution n. If none exists, add solution n to the solution list. If one is found, go to Step 3.  
Step 3. If  $f_n < f_m$ , replace solution m with solution n. Otherwise, discard solution n.

We know that the ending state of a solution in a dynamic program stage affects the outcome of solutions generated in the coming stages. A state should consist of the following two conditions, as discussed previously:

- a. Ending row position of transtainer.
- b. Location pattern of remaining containers of the group as a result of the current solution.

Only the first item was included in the definition of state in our formulation. An explicit consideration of the second condition would make the problem too complex to handle due to the great number of arcs which must be generated and investigated. As an indirect way of incorporating it, a penalty concept was investigated. If all remaining containers are located in one

section, the transtainer need not travel much to pick them up in remaining picking stages of the group. Conversely, if, regardless of the number, they are scattered over several sections, the required TMD would be considerable. Thus, remaining containers' scatter pattern affects TTMD in coming stages and, hence, the determination of the optimal solution.

Let

$P_n$  : Penalty distance for remaining yard pattern of this group corresponding to solution  $n$ .

$$f'_n = f_n + p_n$$

Here,  $p_n$  is computed as follows:

$$p_n = \sum_{i \in I_n} d(Y_i, Y_{i+1})$$

where  $I_n$  : possessing row number indices corresponding to solution  $n$ .

Thus,  $p_n$  is the distance required to pick all remaining containers of the affected group from the lowest possessing row number to the last. The pruning logic Step 3 was modified as follows:

Step 3: Compare  $f'_n$  with  $f'_m$ .

If  $f'_n < f'_m$  replace solution  $m$  with solution  $n$ . Otherwise, discard solution  $n$ .

This feature was sometimes successful when applied to the last three pick stages of groups with at least three pick stages associated. But the improvements were marginal and the lack of consistency in its results forced its rejection from the model. The original pruning logic was used to obtain all test results.

### 5.4.3 Subroutine DISCOMP

Computation of TMD using RE's as well as the pruning work for subroutine DYNAMIC are performed within subroutine DISCOMP. Distance computation is facilitated by using function NDIST. The pruning logic had been discussed. Distance computation is done as follows:

Let

- $Y_1$  : Yard position of the initial transtainer position,  
where  $r_1$  is the row number of  $Y_1$
- $Y_2$  : Yard position of the ending transtainer position at  
the end of a move, where  $r_2$  is the row number of  $Y_2$ .
- IAYS ( $Y_1, Y_2$ ) : Disposition code of two sections contain-  
ing rows  $Y_1$  and  $Y_2$ .
- NER1 : Number of RE's required for an adjacent section  
transfer move.
- NER2 : Number of RE's required for a non-adjacent sec-  
tion transfer move.
- $d(Y_1, Y_2)$  : TMD required to move from row  $Y_1$  to row  $Y_2$ .

Then,  $d(Y_1, Y_2)$  is computed as follows according to section dis-  
position code:

| <u>IAYS (<math>Y_1, Y_2</math>)</u> | <u><math>d(Y_1, Y_2)</math></u>     |
|-------------------------------------|-------------------------------------|
| 1                                   | $74 - r_1 + \text{NER1} + r_2$      |
| 2                                   | $r_1 + \text{NER1} + 74 - r_2$      |
| 3                                   | $74 - r_1 + \text{NER2} + 74 - r_2$ |
| 4                                   | $r_1 + \text{NER2} + r_2$           |
| 5                                   | $74 - r_1 + \text{NER2} + r_2$      |
| 6                                   | $r_1 + \text{NER2} + 74 - r_2$      |



$$\begin{array}{rcl} 7 & & | r_1 - r_2 | \\ 8 & & | r_1 - r_2 | \end{array}$$

A few examples of NDIST computation are as follows:

$$d(4237,5225) = 37 + 10 + 74 - 25 = 96 \text{ RE's (IAYS=2)}$$

$$d(4237,5725) = 37 + 493 + 74 - 25 = 579 \text{ RE's (IAYS=6)}$$

$$d(4237,4270) = 70 - 37 = 33 \text{ RE's (IAYS=8)}$$

#### 5.4.4 Subroutine ASSGN

Container-to-cell assignments are made by subroutine ASSGN according to the optimal transtainer routing plan obtained in subroutine DYNAMIC. The following rules are applied in making these assignments:

- a. Special assign containers are assigned before others.
- b. Loading is done by tier for a given container group to be loaded into a bay (from bottom and up).
- c. Within a tier, loading is done from port to starboard side.
- d. An entire group's loading must be finished within a bay before loading other groups into it.
- e. Within a yard row position, containers are picked from column A to column F, from top to bottom.

The last rule is implemented through a sorting procedure for file CTFIL, as described in 5.3.1.

#### 5.4.5 Subroutine FCHECK

The following ship stability and loading restrictions are

checked with the load plan obtained by subroutines DYNAMIC and ASSGN:

- a. GM limits,
- b. Trim limits,
- c. Deck strength limits,
- d. High cube limits.

Vertical and longitudinal moment contributions by each container to load is computed. Sums of weights and both moments just prior to loading the current set of containers have been read in as input data. The resulting GM and trim values are checked against their upper and lower limits. Total weight stacked in each on-deck column is checked against the limit applied to the bay. Any violation is printed out to inform the operator. Also, bay-by-bay "Stability Calculation" sheet as well as "Ship Stability Summary" is printed out in the format that is currently used at most ports.

Let

W : Total ship displacement including containers (MT)  
 D : Draft at midship at a given displacement (M)  
 MTC : Moment to change trim by 1 Cm at a given displacement (MT-M)  
 KM : Keel to metacenter distance (M)  
 LCB : Longitudinal center of buoyancy (M)  
 LCF : Longitudinal center of flotation (M)  
 VM : Total ship vertical moment (MT-M)  
 LM : Total ship longitudinal moment (MT-M)  
 FS : Free-surface effect moment by liquid stores (MT-M)  
 LBP : Length of ship between two perpendiculars (M)

Then, the following relationships are used to compute a ship's GM and trim values:

$$GM = KM - (VM + FS)/W$$

$$TR = (LM - LCB \times W)/(MTC \times 100)$$

$$\text{Forward draft } (D_f) = D - 0.5 TR - TR \times LCF/LBP$$

$$\text{Aft draft } (D_a) = D + 0.5 TR - TR \times LCF/LBP$$

If either GM or trim limit is violated, subroutine FCHECK prints out the upper and lower limits of applicable moments due to the current set of containers to load. These are easily computed from known GM or trim limits and moment contributions from light ship condition, constant load, liquid stores, and containers already loaded. Then subroutine STABLE is called to see if it is possible to attain feasibility with the given set of containers. Subroutine STABLE assigns containers in the most favorable way with regard to the violated limit, and checks to see if this ideally obtained moment condition intersects the feasible region. Yard locations of containers are completely disregarded in this hypothetical assignment. It is to be remembered that what subroutine STABLE does has no effect on the original set of assignments at all. Its function is simply to determine if feasibility can be obtained with the current set or not. If not, the GLP should be reviewed for possible modifications and an informative message of that effect is printed out for the operator's benefit.

If there is no violation, the operator can choose to have the entire load plan printed out and terminate the planning. With any violation noted, he can consult the ship's personnel as to whether the violation can be tolerated or not. If it is deemed to be of no significant consequence, the run terminates by printing out the load plan as it is. If a violation needs to be corrected, however, subroutine REMEDY can be called for it. These options are entered manually from the terminal keyboard by the operator. Human judgments are thus effectively incorporated into the model. Subroutine FCHECK is called again after

REMEDY execution to print out the revised stability figures.

#### 5.4.6 Subroutine LPOUT

The primary function of subroutine LPOUT is to print out a generated load plan in the sequence of loading. In case of two crane operation, the list for the aft crane follows that of the forward crane. This sequence is also the sequence of trans-tainer picking in the yard.

When LPOUT is called with a "no-print" option, it simply computes TTMD for each transtainer according to the current load plan. This feature enables the operator to see how much increase in TTMD was caused by subroutine REMEDY execution or by interactive adjustments. Load assignments for special assign containers are printed separately from the main set.

#### 5.4.7 Subroutine REMEDY

The objective of subroutine REMEDY is to attain feasibility with as little sacrifice in TTMD as possible. For a given load set, the number of possible assignment combinations is so huge that it is not worthwhile to enumerate them all. An efficient heuristic is needed to achieve the objective. The following explain the logic employed in the REMEDY heuristic:

FIRST. The more serious violation is dealt with before others. GM and trim limit violations are considered more serious than stack weight or high-cube violations. Seriousness of GM and trim violations are compared using the quantity called "degree of violation (DOV)" defined as follows:

$$\text{DOV} = \frac{\text{Amount of moment deviation from the limit}}{\text{Moment improvement hypothetically possible}}$$

SECOND. Correction of one type of violation will affect consequences of other violated constraints. Here, as we deal with violations in decreasing order of their DOV values, we need to ensure that candidate pairs for exchanges are chosen so that they will not adversely affect other violated constraints. For example, suppose GM is too high and trim is excessive by the bow. Also, assume that GM violation has a higher DOV value. Then, a candidate pair for improving GM could consist of a heavy container assigned to a high tier and a light container assigned to a low tier within the same group regardless of bays. However, if the heavy one is in a bay forward of the light one, this exchange will worsen the trim condition. Such candidate pairs are to be excluded from consideration.

THIRD. A large LM change may be created if, say, the lightest container assigned to the most forward receiving bay is exchanged with the heaviest one in the most aft receiving bay. Likewise, a great VM change is possible if a heavy container assigned to a high on-deck tier and a light one assigned to a low under-deck tier are exchanged. The resulting increase in TTMD, however, must be kept at a minimum. To implement these dual objectives, a "distance-to-improvement ratio (D/I ratio)" is used. The D/I ratio is defined as follows:

$$D/I = \frac{\text{Distance between yard locations of a pair}}{\text{Moment improvement possible by a pair}}$$

By selecting a candidate with the smallest ratio for a given first container of a pair, TTMD sacrifice can be minimized while moment improvement is maximized. Some extreme conditions, however, may result to hamper our intent. Consider, for example, D/I ratios of two candidate pairs are both equal to 1.0 and obtained as follows:

$$D/I_1 = 100/100 = 1.0 \text{ and } D/I_2 = 500/500 = 1.0$$

Preference decision between these two involves a tradeoff between the denominator and the numerator of the ratio. How much increase in LM is equivalent to the 100 RE increase in TTMD? It is a difficult question to answer. We can get around this problem using "screens."

Two types of screens are used. The first type is one that limits the minimum amount of moment improvement possible by each iteration. The second type of screen limits the relative locations of candidate pairs. The latter screens are applied in three stages. In the first stage, only containers in the same yard section are considered for exchanges. Then, in the second stage, those that are in two adjacent sections are examined. Finally, containers in any locations are considered. This gradual relaxation of Type II screen ensures that we do not sacrifice TTMD overly for a gain in moment. At the same time, Type I screen allows only those with at least a certain amount of moment improvement to be considered as candidates. The size of Type I screen was arbitrarily determined as 50 MT-M for GM and 100 MT-M for trim violation, respectively. Appropriate fine-tuning of these screen sizes can be made easily for particular situations. The above screen concept is summarized below:

Type I Screen: Specifies minimum moment improvement.

For GM violation : 50 MT-M

For trim violation : 100 MT-M

Type II Screen: Restricts yard location of a candidate.

Stage I : Within the same section.

Stage II : Within the adjacent section.

Stage III : In any section.

FOURTH. The above concepts are used to construct a list of candidate pairs from which exchange pairs are sequentially

selected according to their D/I ratios until either feasibility is obtained or no more pair is available. Arbitrarily, the maximum number of candidate pairs is limited to 200. This limit is necessary to set array dimensions in the subroutine. Thus, of all possible candidate pairs, most desirable pairs are included first in the list followed by others with relaxed screens. Then, the final exchanges are made according to a single measure of merit, the D/I ratio. These exchanges are performed by calling subroutine XCHNGE.

Remedial routines for deck-strength violation and high-cube violation ensure that an exchange does not cause the same trouble in another location. Deck-strength violation is a major concern to a planner for large modern containerships which often take on containers up to four tiers high on-deck. The routine searches the group list for the container which will make the greatest reduction of stack weight among those located in the same section as the one to be replaced. If no such candidate is found, then the search is expanded to the adjacent section. This process proceeds from the top container in the violated stack and moves downward. This is necessary because ship personnel would like to have lighter containers in top tiers than in lower tiers.

There is a tradeoff between the amount of stack weight reduction and the increase in TTMD. Naturally, we desire to minimize the latter while expediting the former process. In view of the absoluteness of deck strength constraints felt by most ship personnel, the formulation was done so that more emphasis is placed on achieving feasibility as soon as possible.

In resolving a high-cube limit violation, subroutine REMEDY looks for the candidate of the same group with the smallest distance difference from the one to be replaced. Since a container is either a high-cube or of regular height, this search is done by computing distance differences of those regular-height

containers in the same group. Thus, the problem in a stack is resolved while increase in TTMD is kept at a minimum.

Subroutine FCHECK is called again after REMEDY execution is terminated. Then, the operator is given a choice between having the load plan printed out and not. Finally, the interactive adjustment option is open, as discussed previously. At termination of the interactive phase, subroutines FCHECK and LPOUT are called again to give the final printouts of stability tables and the load plan.

### 5.5 Simulation of Subroutine DYNAMIC

The greatest concern associated with most dynamic programming formulation is perhaps the phenomenon called "curse of dimensionality." As we saw in section 4.3, information regarding all nodes which have not been pruned away needs to be retained so that the final solution may be traced back in the last stage of a dynamic program algorithm. Thus, the number of nodes imposes certain computer core storage requirements in a particular formulation. The definition of state space adopted in COMCL was inevitable for this reason. Now, it was necessary to see if this formulation could handle problems of expected sizes. A simulation program that includes subroutine DYNAMIC as a major part was constructed to answer this question.

An actual load set from the Port of Portland was used to generate the following relationships:

1. Number of containers in a group (x)  
; Beta distributed with  $\alpha = 0.70$  and  $\gamma = 3.45$
2. Number of possessing rows (Y) as a function of number of containers in a group (x)  
;  $Y = 0.15 X + 0.29$



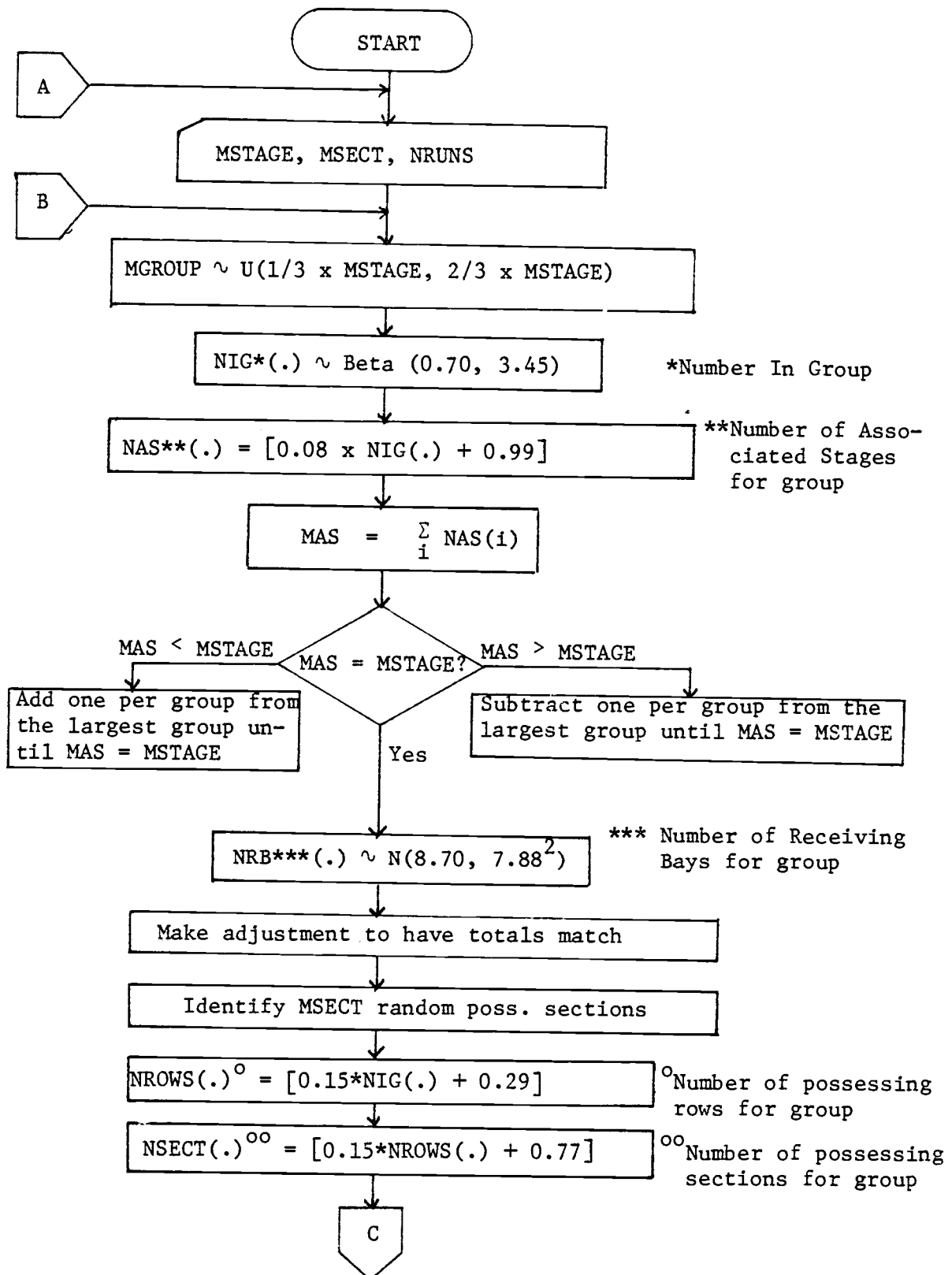
3. Number of possessing sections (Z) as a function of number of rows (Y)  
;  $Z = 0.15Y + 0.77$
4. Number of receiving bays (B) as a function of number of containers in a group (X)  
;  $B = 0.08X + 0.99$
5. Number of containers to load in a stage (L)  
; Normally distributed with mean of 8.70 and standard deviation of 7.88.

The Beta parameter estimation followed the procedure explained in Greer (1970). Beta random number generation was done according to the method contained in Pritsker (1974). The input parameters were as follows:

MSTAGE : Number of states,  
MSECT : Number of total possessing yard sections,  
NRUNS : Number of runs to make with given MSTAGE and MSECT values.

Only the single transtainer problem was attempted in all runs. Figure 5.5 describes the steps of simulation. Basically, a ship loading situation is created using random numbers. Then, subroutine DYNAMIC is executed using this data set. Total number of solutions selected, total number of solutions generated including those pruned away, and total computation time were noted for each run. A total of 36 runs were made while varying MSTAGE and MSECT. Three runs each were made for each parameter combination. MSTAGE was varied from 10 to 65 and MSECT from 3 to 5. Simulation results are listed in Table 4.

The results were analyzed using the Statistical Interactive Package System (SIPS) at O.S.U. Computer Center. The following



(Continued)

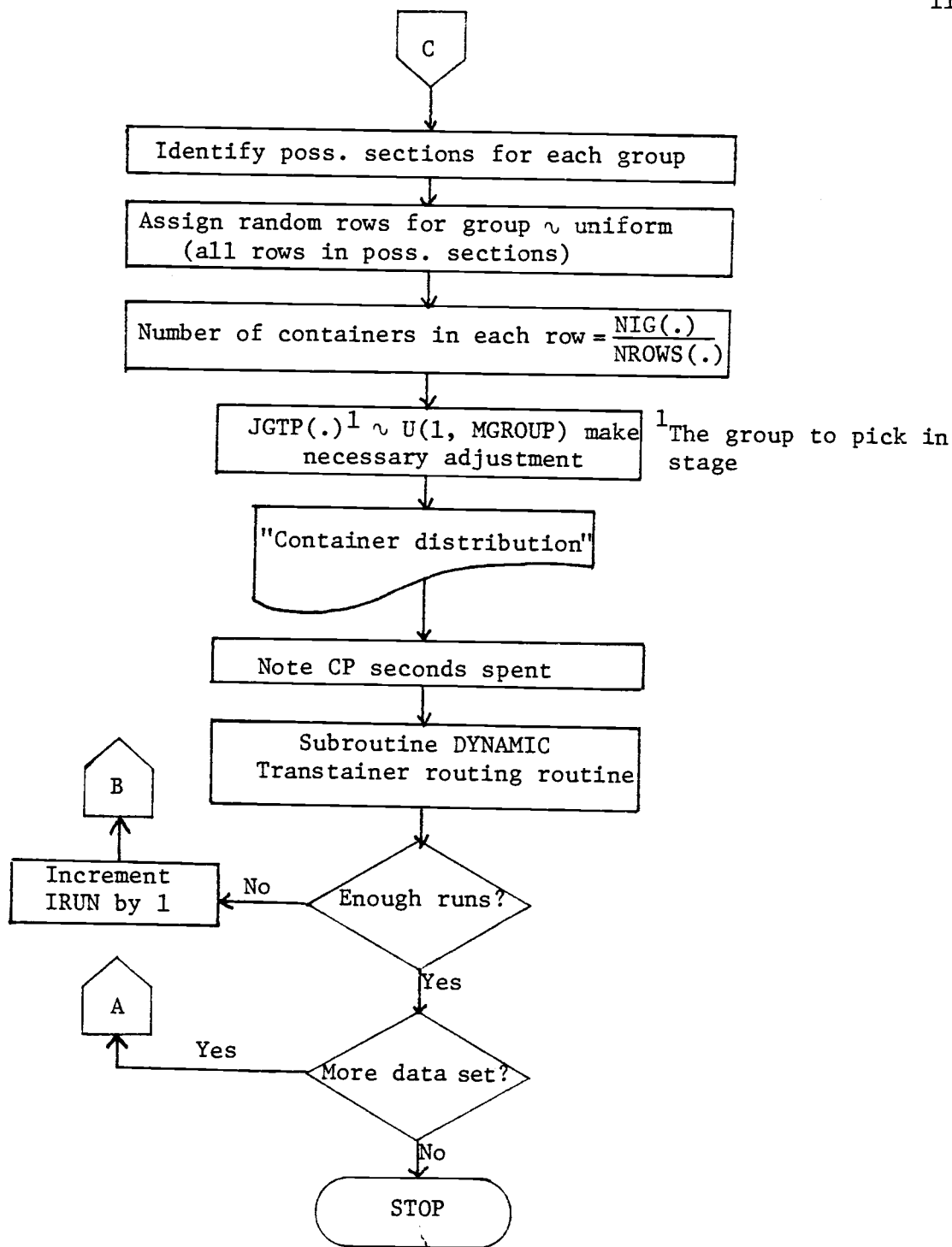


Figure 5.5 Flowchart for subroutine DYNAMIC simulation

TABLE 4. RESULTS OF SUBROUTINE  
DYNAMIC SIMULATION

| RUN<br>NO. | 1<br>MSTAGE | 2<br>MSECT | 3<br>MGROUP | 4<br>NCTLD | 5<br>NTSOL | 6<br>NTGEN | 7<br>TRUN |
|------------|-------------|------------|-------------|------------|------------|------------|-----------|
| 1          | 15          | 4          | 7           | 189        | 43         | 110        | 0.51      |
| 2          | 15          | 4          | 6           | 90         | 21         | 33         | 0.22      |
| 3          | 15          | 4          | 6           | 96         | 45         | 110        | 0.52      |
| 4          | 25          | 3          | 15          | 321        | 57         | 131        | 0.68      |
| 5          | 25          | 3          | 14          | 342        | 64         | 174        | 0.82      |
| 6          | 25          | 3          | 15          | 263        | 46         | 85         | 0.49      |
| 7          | 35          | 4          | 13          | 354        | 92         | 257        | 1.12      |
| 8          | 35          | 4          | 13          | 261        | 81         | 205        | 1.06      |
| 9          | 35          | 4          | 11          | 164        | 57         | 101        | 0.52      |
| 10         | 45          | 5          | 26          | 667        | 128        | 434        | 1.79      |
| 11         | 45          | 5          | 27          | 563        | 136        | 429        | 1.95      |
| 12         | 45          | 5          | 21          | 397        | 82         | 187        | 0.95      |
| 13         | 55          | 5          | 25          | 390        | 98         | 201        | 0.99      |
| 14         | 55          | 5          | 25          | 417        | 98         | 196        | 1.01      |
| 15         | 55          | 5          | 30          | 506        | 110        | 285        | 1.34      |
| 16         | 65          | 5          | 23          | 519        | 141        | 322        | 1.64      |
| 17         | 65          | 5          | 30          | 610        | 155        | 485        | 2.01      |
| 18         | 65          | 5          | 24          | 479        | 147        | 363        | 1.67      |
| 19         | 10          | 3          | 5           | 148        | 27         | 58         | 0.32      |
| 20         | 10          | 3          | 3           | 70         | 23         | 41         | 0.25      |
| 21         | 10          | 3          | 4           | 51         | 14         | 17         | 0.13      |
| 22         | 20          | 4          | 10          | 218        | 50         | 132        | 0.61      |
| 23         | 20          | 4          | 9           | 156        | 39         | 69         | 0.39      |
| 24         | 20          | 4          | 10          | 237        | 73         | 286        | 1.01      |
| 25         | 30          | 4          | 16          | 283        | 57         | 116        | 0.57      |
| 26         | 30          | 4          | 17          | 371        | 81         | 271        | 1.14      |
| 27         | 30          | 4          | 10          | 186        | 73         | 180        | 0.83      |
| 28         | 40          | 4          | 18          | 457        | 103        | 261        | 1.29      |
| 29         | 40          | 4          | 20          | 327        | 57         | 93         | 0.53      |
| 30         | 40          | 4          | 18          | 308        | 73         | 162        | 0.78      |
| 31         | 50          | 4          | 29          | 587        | 161        | 565        | 2.11      |
| 32         | 50          | 4          | 30          | 718        | 149        | 465        | 2.06      |
| 33         | 50          | 4          | 22          | 444        | 101        | 256        | 1.18      |
| 34         | 60          | 4          | 29          | 620        | 131        | 385        | 1.78      |
| 35         | 60          | 4          | 26          | 370        | 103        | 225        | 1.14      |
| 36         | 60          | 4          | 30          | 606        | 132        | 407        | 1.64      |

relationships were obtained from the work:

Let

- $Y_1$  : NTSOL (total number of nodes adopted),
- $Y_2$  : TRUN (CPU execution time for Subroutine DYNAMIC),
- $X_1$  : MSTAGE,
- $X_2$  : MSECT,
- $X_3$  : MGROUP (number of container groups),
- $X_4$  : NCTLD (number of containers to load),
- $X_5$  : NTGEN (total number of solutions generated).

$$Y_1 = 1.08 + 0.86X_1 + 0.14X_4$$

$$Y_2 = -0.034 + 0.0019X_1 + 0.0028X_4$$

$$Y_2 = 0.0016 + 0.0063Y_1 + 0.0022X_5$$

The results show that MGROUP and MSECT do not affect either NTSOL or TRUN significantly. Instead, MSTAGE and NCTLD are the key factors of both dependent variables. All three relationships obtained showed very good fits to the linear models adopted. For a problem involving 1,000 containers and 100 stages, estimated values of NTSOL and TRUN from the above relationships are

$$\text{NTSOL} = 227 \quad ; \quad \text{TRUN} = 2.95 \text{ seconds.}$$

This shows that the present dynamic programming formulation would not require excessive central memory or computational time requirements. Computationally, it seems quite efficient although we lack a basis for comparison.

This analysis provided the assurance that the current formulation would be a manageable one in the least, and that the model would find its usefulness at most containerports only if

its capability meets our intent and expectations in terms of load plan generation.

### 5.6 Analogy with the Dual Simplex Method

Lemke's dual simplex method solves an LP starting with a dual feasible (primal superoptimal) solution and conducts pivots towards primal feasibility while maintaining dual feasibility.

In order to understand the workings of the dual simplex method, consider the following linear programming problem.

$$\begin{aligned} \text{Minimize} \quad & \underline{c} \underline{x} \\ \text{Subject to} \quad & \underline{A} \underline{x} = \underline{b} \\ & \underline{x} \geq \underline{0} \end{aligned}$$

The following represents a basic solution at some iteration.

|          | $z$      | $x_1$       | ... | $x_j$       | ... | $x_k$       | ... | $x_n$       | RHS                 |
|----------|----------|-------------|-----|-------------|-----|-------------|-----|-------------|---------------------|
| $z$      | 1        | $z_1 - c_1$ | ... | $z_j - c_j$ | ... | $z_k - c_k$ | ... | $z_n - c_n$ | $\frac{c_B b}{c_B}$ |
| $x_{B1}$ | 0        | $y_{11}$    | ... | $y_{1j}$    | ... | $y_{1k}$    | ... | $y_{1n}$    | $b'_1$              |
| $x_{B2}$ | 0        | $y_{21}$    | ... | $y_{2j}$    | ... | $y_{2k}$    | ... | $y_{2n}$    | $b'_2$              |
| $\vdots$ | $\vdots$ | $\vdots$    |     | $\vdots$    |     |             |     |             | $\vdots$            |
| $x_{Br}$ | 0        | $y_{r1}$    | ... | $y_{rj}$    | ... | $y_{rk}$    | ... | $y_{rn}$    | $b'_r$              |
| $\vdots$ | $\vdots$ | $\vdots$    |     | $\vdots$    |     |             |     |             | $\vdots$            |
| $x_{Bm}$ | 0        | $y_{m1}$    | ... | $y_{mj}$    | ... | $y_{mk}$    | ... | $y_{mn}$    | $b'_m$              |

Suppose that the tableau is dual feasible (that is,  $z_j - c_j \leq 0$  for a minimization problem). If all right-hand-side values

$(b'_i)$  are positive, this is the optimal solution. Otherwise, we select the pivot row  $r$  with

$$b'_r = \text{Minimum } \{b'_i\} .$$

In row  $r$ , if all  $y_{rj} \geq 0$ , the dual is unbounded and the primal is infeasible. Otherwise, select the pivot column  $k$  by the following ratio test:

$$\frac{z_k - c_k}{y_{rk}} = \text{Minimum}_j \left\{ \frac{z_j - c_j}{y_{rj}} ; y_{rj} < 0 \right\}$$

Then, a pivot is made at  $y_{rk}$  and the algorithm is thus repeated.

This process is depicted in Figure 5.6. Here, only a two-dimensional situation is used to illustrate the point.

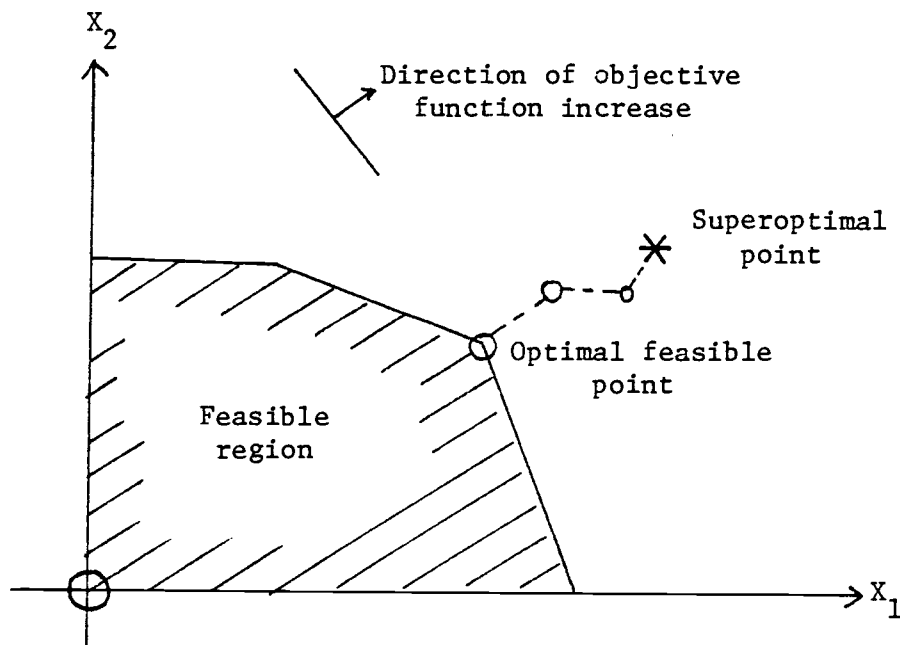


Figure 5.6 Working of dual simplex algorithm

Subroutine DYNAMIC in COMCL obtains a superoptimal solution which may or may not be primal feasible. If it is found to be primal feasible (that is, all constraints are satisfied), it is the optimal feasible solution. Otherwise, subroutine REMEDY performs pairwise exchange iterations which are quite analogous to dual simplex pivot operations. The philosophical parallel can be made regardless of the fact that the dual simplex solves LP problems while a CLP problem is essentially a 0-1 integer program.

The workings of subroutine REMEDY can be described by the following 0-1 integer program:

$$\text{Minimize } \sum_i \sum_j d_{ij} \cdot X_{ij} \quad (1)$$

$$\text{Subject to } \sum_i \sum_j g_{ij} X_{ij} \geq \delta \quad (2)$$

$$\sum_j X_{ij} \leq 1 \quad (3)$$

$$\sum_i X_{ij} \leq 1 \quad (4)$$

and

$$X_{ij} = 0 \text{ or } 1 \quad (5)$$

Here,  $X_{ij} = \begin{cases} 1, & \text{if container } i \text{ and container } j \text{ are to be} \\ & \text{exchanged,} \\ 0, & \text{otherwise.} \end{cases}$

$$d_{ij} = d(Y_i, Y_j)$$

$$g_{ij} = CG_i \cdot W_j - CG_j \cdot W_i$$

$CG_i$  : Center of gravity of cell currently assigned  
to container  $i$

$W_i$  : Weight of container  $i$ .

The objective (1) is to minimize the TTMD increase due to



exchanges. Constraint (2) states that the moment improvement due to all exchanges must be at least  $\delta$  in order to ensure a stable load condition. Only one load constraint is expressed in this problem structure. This is appropriate because subroutine REMEDY handles one violation at a time. Note that stability constraints are chiefly in consideration here.  $CG_i$  may either refer to LCG or VCG of the cell assigned to container  $i$  by subroutines DYNAMIC and ASSGN, depending on which violation is being handled.

Note that a solution  $X_{ij} = 0$  for all  $i$  and  $j$  is a super-optimal solution to the above problem. It is exactly the solution obtained by subroutine DYNAMIC. It is not primal feasible if constraint (2) is not satisfied. In fact, the left-hand-side value of constraint (2) is zero for this initial solution. We assume  $\delta$  is a positive quantity. Constraints (3) and (4) are added because "static" exchanges are desired. In other words, once a container has been switched to a different cell, it must not be moved again to another location. This condition is implemented by calling function LXAMIN, which examines if a container has been included as a candidate for exchange previously.

Of all feasible candidate pairs, subroutine REMEDY sequentially selects exchange pairs according to the magnitude of the following D/I ratio:

$$D/I = \frac{d_{rk}}{g_{rk}} = \text{Minimum}_{i,j} \left\{ \frac{d_{ij}}{g_{ij}} : g_{ij} \geq G \right\}$$

Pairs with  $g_{ij} \leq 0$ , that is, those that would result in negative contribution to stability improvement are rejected without further consideration. In order to prevent the phenomenon of many tiny improvements occurring, an arbitrary screen was used to eliminate those pairs with moment improvement less than amount  $G$ . Note that such a screen could have been implemented in the

dual simplex method if it was addressed to a particular type of problems. In our case, such a screen is possible and proves to be an efficient method of rapidly obtaining primal feasibility.

A similar analogy can be said of the high-cube correction routine, but somewhat different logic was adopted for the deck-strength correction, as discussed previously.

## 6. TESTING OF COMCL

### 6.1 Test Environments and Procedure

Testing the effectiveness of COMCL was conducted using actual load sets of two ships which visited the Fulton Terminal (T-6) of the Port of Portland, Oregon. These ships were Japan Apollo Maru and Alaska Maru. They both belong to the Japanese Consortium which consists of the following shipping companies:

- Japan Line
- K Line
- MOSK Line
- NYK Line
- Showa Line
- YS Line

Ships in this consortium commonly carry on board containers of other member companies as well as their own. They usually visit Vancouver, British Columbia; Seattle, Washington; and Portland, Oregon before making cross-Pacific voyages. On the other side of the Pacific, they usually call at Kobe, Nagoya, Shimizu and Tokyo or Yokohama in Japan.

Both Japan Apollo Maru and Alaska Maru are typical modern cellular container ships which can carry both 20-foot and 40-foot containers. The tested voyages are as follows:

- Test #1: Japan Apollo Voyage #5
- Test #2: Japan Apollo Voyage #10
- Test #3: Alaska Maru Voyage #92
- Test #4: Alaska Maru Voyage #93
- Test #5: Alaska Maru Voyage #91

All of the above voyages occurred between August 1980 and March 1981.

Actual loading operations according to COMCL-generated load plans were not conducted due to absence of an agreement to enable it. Technical difficulties associated with computer terminal connection and data input effort forced post-voyage testing in all cases.

All three modes of planning described in 5.4.2 were used and feasible load plans were obtained in each test. The interactive adjustment feature was not used, although its usefulness and feasibility were tested separately. The CDC Cyber 72 computer at O.S.U. Milne Computer Center was used for the testing. Terminal mode of operation was used in all tests.

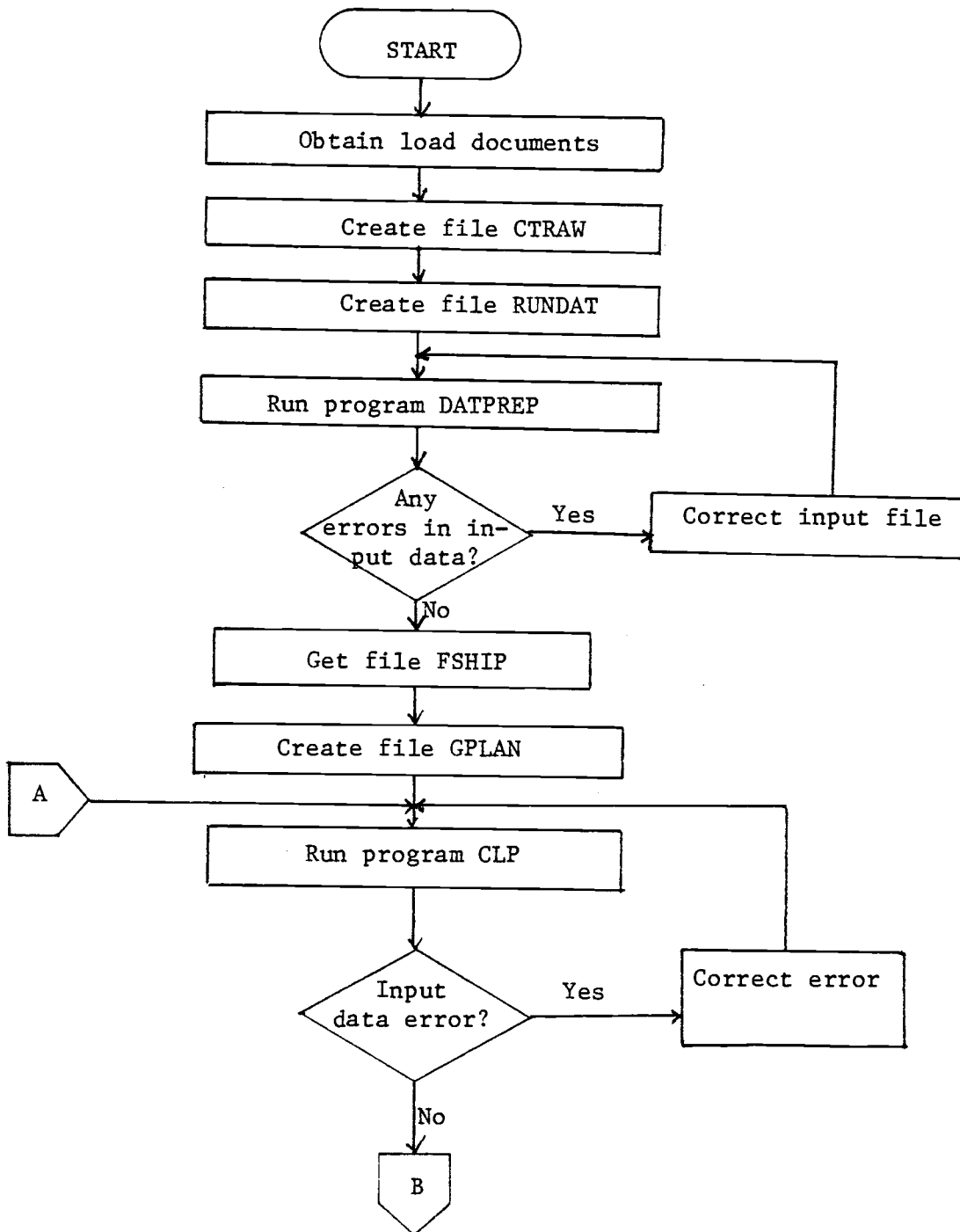
The test procedure used is as shown in Figure 6.1. All documents showing input data as well as manually generated plans were provided by the operations personnel at the Fulton Terminal.

The initial positions of transtainers were fixed as follows for all tests. The TTMD's of both manually generated plans and COMCL-generated plans were computed based on these initial transtainer positions:

T1 (Forward Transtainer): 4201  
T2 (Aft Trainstainer): 5201

These positions were arbitrarily chosen because of the following two reasons: First, sections #42 and #52 are the most frequently used sections for storing outbound containers at the Fulton Terminal. Second, fixing of their initial positions provides a common basis of TTMD comparisons among several different load plans generated for a load set.

The same work sequence used in manual planning was used for COMCL runs in each test except that a slight modification was applied on that of test #5 in order to simplify the operation.



(Continued)

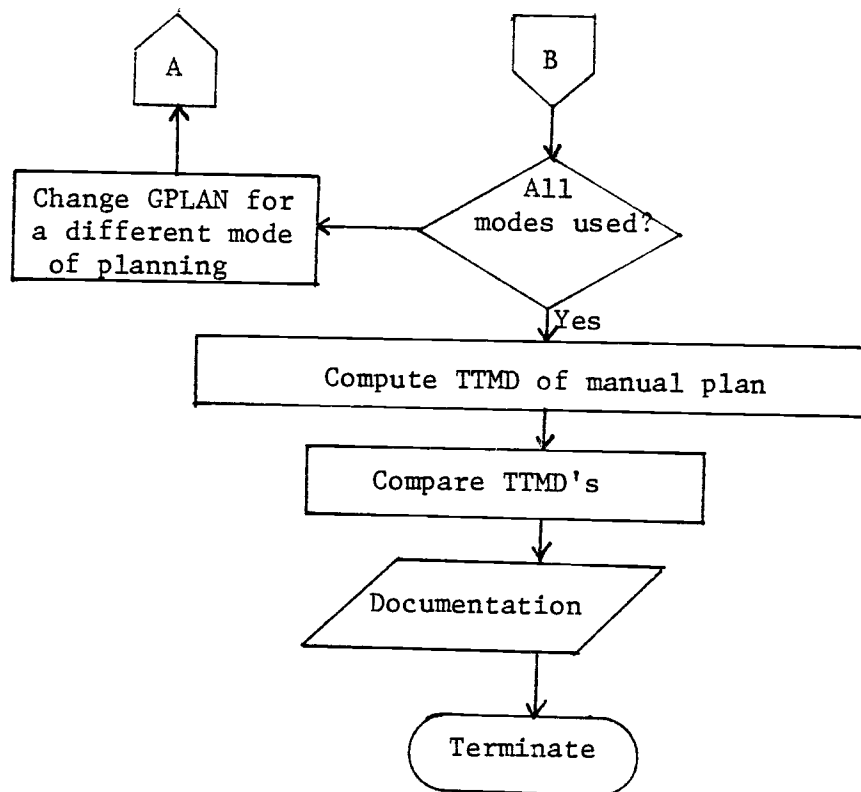


Figure 6.1 COMCL test procedure

## 6.2 Test Results

The test results are summarized in Tables 5 through 9. All three modes of planning (parallel, T1-T2, and T2-T1) produced feasible load plans whose stability indicators were acceptable to ship's personnel. The allowable limits in GM and trim for both ships were from 0.0M to 2.0M in both categories with 0.1M excess in trim tolerated in certain situations. The tables show comparison of TTMD values in each test. The units of TTMD are row equivalents (RE's), as previously defined.

Summary comparison of TTMD values of the best result by COMCL and the manual plan for each test may be tabulated as in Table 10. The overall performance of COMCL through these five tests is a 3.2 percent reduction in yard material handling from what the manual plans require.

Some statistics relating number of containers to pick per stage and number of stages are shown in Table 11. In this table, we may note that the average number to pick is the smallest in test #4 in which COMCL showed a poor performance. The ratio of MSTAGE to NCTLD was the highest in this test, also. These indicate that this particular test involved loading of small numbers of containers with many crane movements among different bays. In a situation such as this, subroutine DYNAMIC may not produce the true optimal routing plan due to the limitations in the state space definition. In other words, some good solution paths are pruned out at choke points, where a container group with only one or two possessing rows is involved. This observation, however, can be verified through further testing effort.

The number of special-assign containers constituted only 0.9 percent of the total. In test #2, a large number of special content containers (onions) required a designated storage area

TABLE 5. SUMMARY OF TEST #1 RESULTS

| Ship: Japan Apollo         |                    | Voyage No.: 5     |                   |                  |
|----------------------------|--------------------|-------------------|-------------------|------------------|
|                            | Parallel<br>Method | Serial<br>(T1-T2) | Serial<br>(T2-T1) | Manual<br>Method |
| T1 TMD (RE's)              | 3354               | 3156              | 3354              | 4626             |
| T2 TMD (RE's)              | 2561               | 4023              | 2561              | 2745             |
| TTMD (RE's)                | 5915               | 7179              | 5915              | 7371             |
| Violation                  | None               | None              | None              | --               |
| GM (M)                     | 1.76               | 1.74              | 1.76              | 1.75             |
| Trim (M)                   | 1.09               | 1.18              | 1.09              | 0.92             |
| % TTMD of<br>Manual Method | 80.2*              | 97.4              | 80.2*             | 100.0            |

Number of stages: 20  
 Number of containers to load: 261  
 Special assign containers: None  
 Number of container groups: 9  
 Number of bays to load into: 18  
 Possessing yard sections: 4

\* Best results.



TABLE 6. SUMMARY OF TEST #2 RESULTS

| Ship: Japan Apollo            |      |                    |                   |                   | Voyage No.: 10   |  |  |  |  |
|-------------------------------|------|--------------------|-------------------|-------------------|------------------|--|--|--|--|
|                               |      | Parallel<br>Method | Serial<br>(T1-T2) | Serial<br>(T2-T1) | Manual<br>Method |  |  |  |  |
| T1 TMD                        | BR*  | 3284               | 2010              | 3502              | 2372             |  |  |  |  |
|                               | AR** | --                 | 2362              | --                | --               |  |  |  |  |
| T2 TMD                        | BR   | 3879               | 4397              | 3919              | 3955             |  |  |  |  |
|                               | AR   | --                 | 4613              | --                | --               |  |  |  |  |
| TTMD                          | BR   | 7163               | 6407              | 7421              | 6327             |  |  |  |  |
|                               | AR   | --                 | 6975              | --                | --               |  |  |  |  |
| Violation                     |      | None               | Trim              | None              | --               |  |  |  |  |
| GM                            | BR   | 1.97               | 1.93              | 1.94              | 1.98             |  |  |  |  |
|                               | AR   | --                 | 1.94              | --                | --               |  |  |  |  |
| Trim                          | BR   | 2.04               | 2.17              | 2.01              | 2.06             |  |  |  |  |
|                               | AR   | --                 | 2.08              | --                | --               |  |  |  |  |
| Number of iterations          |      | 0                  | 9                 | 0                 | --               |  |  |  |  |
| % TTMD of Manual Method       |      | 113.2              | 110.2***          | 117.3             | 100.0            |  |  |  |  |
| Number of stages:             |      |                    |                   |                   | 37               |  |  |  |  |
| Number of containers to load: |      |                    |                   |                   | 329              |  |  |  |  |
| Special assign containers:    |      |                    |                   |                   | None             |  |  |  |  |
| Number of container groups:   |      |                    |                   |                   | 15               |  |  |  |  |
| Number of bays to load into:  |      |                    |                   |                   | 21               |  |  |  |  |
| Possessing yard sections:     |      |                    |                   |                   | 4                |  |  |  |  |

\* Before REMEDY figures.

\*\* After REMEDY figures.

\*\*\* Best result.

TABLE 7. SUMMARY OF TEST #3 RESULTS

| Ship: Alaska Maru             |    | Voyage No.: 92     |                   |                   |                  |
|-------------------------------|----|--------------------|-------------------|-------------------|------------------|
|                               |    | Parallel<br>Method | Serial<br>(T1-T2) | Serial<br>(T2-T1) | Manual<br>Method |
| T1 TMD                        | BR | 3283               | 4201              | 3283              | 3998             |
|                               | AR | 3181               | 3389              | 3579              | --               |
| T2 TMD                        | BR | 5095               | 5201              | 4883              | 5159             |
|                               | AR | 5323               | 5497              | 5011              | --               |
| TTMD                          | BR | 8378               | 8402              | 8116              | 9157             |
|                               | AR | 8504               | 8886              | 8590              | --               |
| Violation                     |    | DS(4)*             | DS(8)             | DS(8)             |                  |
| GM                            | BR | 1.48               | 1.49              | 1.46              | 1.48             |
|                               | AR | 1.50               | 1.50              | 1.49              | --               |
| Trim                          | BR | 0.07               | 0.22              | 0.24              | 0.04             |
|                               | AR | 0.15               | 0.08              | 0.13              | --               |
| Number of<br>iterations       |    | 4                  | 12                | 12                | --               |
| % TTMD of<br>Manual<br>Method |    | 92.9**             | 97.0              | 93.8              | 100.0            |

Number of stages: 47

Number of containers to load: 338

Special assign containers: 5

Number of groups: 18

Number of receiving bays: 23

Number of possessing sections: 4

Special condition: 4-high stacks on-deck

\* Deck strength violation in four columns.

\*\* Best result.

TABLE 8. SUMMARY OF TEST #4 RESULTS

| Ship: Alaska Maru              |        |                    | Voyage No.: 93    |                   |                  |
|--------------------------------|--------|--------------------|-------------------|-------------------|------------------|
|                                |        | Parallel<br>Method | Serial<br>(T1-T2) | Serial<br>(T2-T1) | Manual<br>Method |
| T1 TMD                         | BR     | 1644               | 1618              | 3707              | 1667             |
|                                | AR     | 1674               | 1642              | 3707              | --               |
| T2 TMD                         | BR     | 4604               | 4816              | 3926              | 4101             |
|                                | AR     | 4604               | 4816              | 3956              | --               |
| TTMD                           | BR     | 6248               | 6434              | 7633              | 5768             |
|                                | AR     | 6278               | 6458              | 7663              | --               |
| Violations                     |        | DS(2)              | DS(1)             | DS(2)             | --               |
| GM                             | BR     | 1.31               | 1.30              | 1.30              | 1.32             |
|                                | AR     | 1.31               | 1.30              | 1.30              | --               |
| Trim                           | BR     | 0.97               | 1.07              | 1.03              | 1.04             |
|                                | AR     | 0.99               | 1.08              | 1.05              | --               |
| Number of iterations in REMEDY | 3      |                    | 1                 | 3                 | --               |
| % TTMD of Manual Method        | 108.8* |                    | 112.0             | 132.9             | 100.0            |

Number of stages: 40  
 Number of containers to load: 218  
 Special assign containers: 7  
 Number of container groups: 16  
 Number of bays to fill: 19  
 Possessing yard sections: 4

\* Best result.

TABLE 9. SUMMARY OF TEST #5 RESULTS

| Ship: Alaska Maru              |    | Voyage No.: 91     |                   |                   |                  |
|--------------------------------|----|--------------------|-------------------|-------------------|------------------|
|                                |    | Parallel<br>Method | Serial<br>(T1-T2) | Serial<br>(T2-T1) | Manual<br>Method |
| T1 TMD                         | BR | 1763               | 1597              | 1929              | 1917             |
|                                | AR | --                 | 1621              | --                | --               |
| T2 TMD                         | BR | 4235               | 4817              | 4149              | 4249             |
|                                | AR | --                 | 4841              | --                | --               |
| TTMD                           | BR | 5998               | 6414              | 6078              | 6166             |
|                                | AR | --                 | 6472              | --                | --               |
| Violations                     |    | None               | DS(1)             | None              | --               |
| GM                             | BR | 1.48               | 1.47              | 1.46              | 1.52             |
|                                | AR | --                 | 1.47              | --                | --               |
| Trim                           | BR | 1.12               | 1.04              | 1.19              | 1.24             |
|                                | AR | --                 | 1.04              | --                | --               |
| Number of iterations in REMEDY |    | 0                  | 1                 | 0                 | --               |
| % TTMD of Manual Method        |    | 97.3*              | 1.05              | 98.6              | 100.0            |
| Number of stages:              |    |                    |                   | 39                |                  |
| Number of containers to load:  |    |                    |                   | 264               |                  |
| Special assign containers:     |    |                    |                   | 0                 |                  |
| Number of container groups:    |    |                    |                   | 12                |                  |
| Number of bays to fill:        |    |                    |                   | 20                |                  |
| Possessing yard sections:      |    |                    |                   | 4                 |                  |
| * Best result.                 |    |                    |                   |                   |                  |

TABLE 10. COMPARISON OF COMCL PLANS  
VERSUS MANUAL PLANS

| Test  | Best Method<br>COMCL | (1)<br>TTMD<br>(COMCL) | (2)<br>TTMD<br>(Manual) | % ((1)÷(2)) |
|-------|----------------------|------------------------|-------------------------|-------------|
| #1    | Parallel             | 5915                   | 7371                    | 80.2        |
| #2    | T1-T2                | 6975                   | 6327                    | 110.2       |
| #3    | Parallel             | 8504                   | 9157                    | 92.9        |
| #4    | Parallel             | 6278                   | 5768                    | 108.8       |
| #5    | Parallel             | 5998                   | 6166                    | 97.3        |
| Total | --                   | 33,670                 | 34,789                  | 96.8        |

TABLE 11. MSTAGE AND NUMBER OF  
CONTAINERS TO PICK

| Test  | (1)<br>MSTAGE | (2)<br>Av. NPICK* | (3)<br>NCTLD | (1)÷(3) | No. of spe-<br>cial-assign<br>containers |
|-------|---------------|-------------------|--------------|---------|--|
| #1    | 30            | 8.7               | 261          | .115    | 0  |
| #2    | 37            | 9.1               | 329          | .112    | 0  |
| #3    | 47            | 7.1               | 338          | .139    | 5  |
| #4    | 41            | 3.4               | 218          | .183    | 7  |
| #5    | 39            | 6.8               | 264          | .148    | 0  |
| Total | 194           | 6.9               | 1,410        | .138    | 12                                       |

\* NPICK: Number of containers to pick in a stage.

for proper ventilation. This requirement was satisfied by classifying them into an exclusive 40-foot specials (40-foot S) group and forming appropriate LPAT's for the designated area. Load locations for the few special-assign containers were entered as designated and these were excluded from the normal assignment process.

The REMEDY routine for trim violations was effectively used to reduce the trim value in test #2. Tests #3 and #4 amply demonstrated the efficiency of the REMEDY routine for deck strength violations. An average of 1.4 iterations was required to correct each deck strength violation. Test #3 particularly involved four-high stacking on-deck in eight columns. The routines for GM and high-cube violations had to be tested by creating artificial violations because no such violations occurred. They were found to function properly as designed.

Computer CPU run times required for each run varied from eight seconds to 23 seconds. Whenever subroutine REMEDY had to be called, the run time tended to be large. Program CLP required 135 KB (octal) of core memory to run. The maximum core storage limit of the CYBER computer used is currently 310KB (octal).

### 6.3 Discussion of Results

The following observations are made based on the test results and experiences gained through the tests:

- a. COMCL generated feasible load plans in all tests within reasonable computer execution times.
- b. COMCL can be accommodated by any container port with a medium to large computer facility. Program CLP which is the larger part in the model, requires approximately 140 KB (octal) of core memory space.

- c. In all tests except one, the parallel mode of planning gave the best result. It is deemed that this mode creates a combined work sequence which is closest to the real one. In this sense, COMCL performance is very much dependent on work sequence. A sequence plan involving excessive number of stages hinder the ability of COMCL, as seen in the test #4 result.
- d. COMCL-produced load plans required overall four percent less transtainer movement than manually generated plans. Although the true potential of COMCL has not been revealed through this limited number of tests, this reduction will bring about significant economic consequences in terms of shorter ship turnaround times and better utilization of port facilities.
- e. A much greater potential benefit that COMCL can render is in computerization of the load planning itself. Currently, three operations personnel are required to work at least one full day to complete load planning for an average-size load set at the Fulton Terminal. This does not take into account the work by data processing personnel for making stability checks using a mini-computer system. With a computer system organized as shown in Figure 2.16, the planning staff only need to input file GPLAN for a COMCL run. Two major parts of GPLAN are current load conditions and GLP information. If current load conditions of incoming vessels can be transferred in a common format from the previous port of call through data line, what is left would be construction of a work sequence and encoding of GLP information into LPAT's. These jobs should not take any more than an hour to an experienced planner. With time

allowed for error correction, a COMCL run can be completed within a two-hour span involving one planning person and one data entry person.

These estimates mean over 90 percent reduction in manpower cost of planning. Thus freed manpower may be used for other administrative tasks as well as for absorbing increase in planning work due to a forecasted increase in container business.

- f. With more than one data entry display terminals connected to a main computer where COMCL would reside, load planning for more than one vessel can be conducted simultaneously. For a large port such as Seattle, Washington, this would mean better utilization of planning staff as well as shorter planning time.
- g. Another important advantage that computerization offers is accuracy of results. Small discrepancies in stability computations were noted between COMCL-generated plans and manually generated plans during testing. Implementation of COMCL can eliminate the possible hazard of an unstable voyage across the Pacific due to a computational error.

These observations point to the usefulness and possibly tremendous impact on port productivity improvement that a proper implementation of COMCL can bring to the container port industry. It is noted here that any good model is not good at all unless its true value is appreciated and a proper management attention is expended for an effective design of a system which incorporates it.



## 7. CONCLUSIONS AND RECOMMENDATIONS

### 7.1 Usefulness and Contributions of the Model

The model developed through this study is a limited one in many respects as discussed in the next section. However, a few conclusions may be drawn on its merits as follows:

FIRST. COMCL is a unique containership load planning model in that no other of like nature has been developed. Existing models merely try to produce feasible load plans while COMCL optimizes in material handling as well as ensuring feasibility. The tests seemed to prove this point.

SECOND. The philosophy of the model was drawn from established optimization techniques and therefore, must be a sound one. The tests show that starting with an optimal plan which may be either feasible or infeasible is an efficient method of reaching an optimal feasible solution.

THIRD. Cost savings from COMCL utilization would come not only from resulting reduction of loading time but also from reduction of planning effort required to produce loading plans. These savings would be gained with a proper computer facility available at a containerport. To a particular port, the corresponding reduction in ship turnaround times would mean not only a more efficient utilization of port facilities, but also an eventual increase in container business as more ships begin to call it. Also, use of this model would obviate the long-term training of planning personnel usually required for efficient load plan generation.

FOURTH. COMCL provides an important software block in containerport automation. With a computer network linking all major

ports of a geographic region, as described in Section 2.3.4, a ship's current load condition can be relayed to another port rapidly and in a format suitable for load planning at another port. General loading plans can be entered graphically from a CRT terminal with a magnetic sensor pencil by the shipping line representative. This would obviate translating of color-coded schematics, thus shortening ship turn-around times considerably. The technology to do the above is currently available and it is a matter of port management innovation and cooperation among port authorities. When such an automation system is installed at major containerports, port efficiency increase could be phenomenal.

## 7.2 Restrictions in the Model

The following restrictions apply to COMCL:

- a. The model is addressed only to containerports with overhead crane and yard truck handling system in the yard. These ports are characterized by rectangular yard sections. Other yard configurations would require some reformulation work if the same ideas are to be utilized.
- b. The efficiency and cost savings from using the model would be obtained with a proper computer system which can accommodate the model and maintain/update data files constantly. A desirable configuration of a computer system is described in Section 2.3.4. Reduction in planning effort and stevedoring labor are somewhat offset by increased data processing needs in hardware, software development/maintenance, and personnel costs. Benefit cost analysis is necessary for each individual port. Most ports, however, already have medium-to-large

size computer systems. Operating and installation of an interactive terminal at a quay operations office would not require any substantial investment. Current data processing personnel can be better utilized to handle the software maintenance. One or two data entry personnel, however, would be required at the quay terminal point.

- c. COMCL takes, as a part of its input data, human generated equipment work sequence. Potentially great improvements in yard material handling would be possible through use of an effective work sequencing program.

As of now, the quality of a work sequence effectively sets an upper bound as to how well COMCL can perform. This point is only a restriction to the model, and not a demerit. Manual planning is done likewise currently.

- d. Equipment interference between two transtainers are not rigorously checked in subroutine DYNAMIC. The time element necessary to do this comes from cumulative container counts and these may not constitute an accurate measure of gangs' work progress. When such an interference occur, operational adjustments must be made as necessary. This restriction is equally shared by the current method of manual planning.
- e. Exact measurements of activity times were not possible due to strong labor resistance. Given a more amenable atmosphere, however, time study results can be used to modify function NDIST with relative ease.

### 7.3 Possible Areas for Future Research

This study is only a small step into the future of port automation. One of the major hindrances to port automation is certainly the lack of appreciation of the usefulness of existing optimization techniques to efficient port operations. Future research should be conducted with a clear picture of how an automated containerport should be like. Under present set-up of container handling at most ports, the greatest part of ship turnaround times, besides actual unload/load time, is spent by "the preparation work." This involves transfer of current ship load data and GLP to port personnel by ship and shipping line personnel, generation of tags for manual planning, computation of data for stability check, data accuracy check and record keeping of pertinent data. An automated system should reduce such preparation time to a minimum. It can be done if the following problems are successfully solved:

- a. A computer network connecting all major ports of a geographic region such as the West Coast of the U.S. and Canada need to be established. This network shall share the common software and enable ship data transfer from one port to another by simply opening access to it to all node points of the network. Optimum design of such a network with related software and hardware requirements and specifications is an important area of future research for port automation.
- b. Current technology in computer graphics can be fully utilized for containership load planning work. Each shipping line representative currently travels to next port of call of a ship where he color-codes general loading strategy on ship schematics. This work can be

done at a CRT terminal using a magnetic sensor pen interactively. Computer first asks the planner what group (port, size and type of cargo) to plan for. Then the bay numbers to load this group are queried. For each receiving bay, a full display of the bay plan may appear on the screen with all cells. The planner can designate the cells for a given group using a magnetic sensor pen. This input can be converted to input data for COMCL within the computer. After each step, the computer may provide an informative display showing number of remaining containers of each group to plan. Also, the computer may aid the planner by computing expected GM and trim values with the strategy entered. This can be done by using average weights of container groups. Thus, the planner can have prior knowledge of what stability situation will result from a loading.

- c. Equipment work sequencing is done manually today. This problem was not attempted in this study because the problem without it was of sufficient complexity. A better utilization of crane and transtainer time, however, can result by an efficient heuristic program which considers both ship load requirements and yard distribution of groups. With two gangs operating, it is a problem of four-machine sequencing. Since a transtainer and a crane operate as a pair, however, it is, in effect, a two-machine sequencing problem. A common measure of effectiveness, such as time or dollar, needs to be established for it. Activity times such as for opening and closing hatches, passing over a superstructure, transit to a different bay, transtainer

transit times, etc., need to be established through time and motion studies. A plausible structure of such a heuristic is one that generates a fixed number, say 100, of alternative sequences and checks their feasibility in terms of machine work interference restrictions. Then, the best sequence may be used for the COMCL use. If COMCL results in an undesirable plan in terms of stability, an interactive feature may allow selection of the next best work sequence for another COMCL run. Such a situation can be avoided, however, if the sequencing heuristic is properly structured.

It is hoped that there would be a surge of interest within the container industry in searching for effective ways to improve port efficiency as a result of this study.

## BIBLIOGRAPHY

Anonymous, Methodology for Estimating Capacity of Marine Terminals, Vol. I: Standardized Methodology, MA/RD-970-t 76070 VI, February, 1976.

Apple, James M., Plant Layout and Material Handling, Third Edition, John Wiley and Sons, New York, 1977.

ARINC Research Corp., Automated Management Control System for Public Marine Terminals, Report to Maritime Administration, AD-A061-632, Annapolis, Maryland, November, 1978.

Bazaraa, M. S. and J. J. Jarvis, Linear Programming and Network Flows, John Wiley and Sons, New York, 1977.

Bazaraa, Mokhtar S. and C. M. Shetty, Nonlinear Programming: Theory and Algorithms, John Wiley and Sons, New York, 1979.

Beliech, D. E., Proposed Method for Efficient Preload Planning for Containerized Cargo Ships, Master's thesis, USNGPS, Monterey, California, June, 1974.

Bolton, R. L., J. H. Gner and M. S. Miller, Manual Procedures for Estimating Marine Terminal Throughput, AD A 055 535, March, 1978.

Bolton, R. L. et al., Marine Terminal Operations, AD 055 537, March, 1978.

Borovits, J., "Determination of Optimum Delivery Size of Containers," Operations Research Quarterly, Vol. 27, 1976.

Boyer, W. C., "Containerization--A System Still Evolving," Journal of the Waterways and Coastal Engineering Division of ASCE, November, 1972.

Bradley, Gordon H., "Survey of Deterministic Networks," AIIE Transactions, Vol. 7, No. 3, September, 1975.

Bradley, Gordon, G. Brown, and G. Graves, "Design and Implementation of Large-Scale Primal Transshipment Algorithms," Management Science, Vol. 24, No. 1, September, 1977.

Bureau of Naval Personnel, Principles of Naval Engineering, U.S. Government Printing Office, Washington, D.C., 1970.

Canada, John R. and John A. White, Jr., Capital Investment Decision Analysis for Management and Engineering, Prentice-Hall, Inc., New Jersey, 1980.

Charnes, A. and W. W. Cooper, Management Models and Industrial Applications of Linear Programming, Vol. II, John Wiley, New York, 1961.

Cojeen, H. P. and P. Van Dyke, "The Automated Planning and Sequencing of Containers for Containership Loading and Unloading," Proceedings of the Second Ship Operation Automation Symposium, Washington, D.C., September, 1976.

Cohen, Claude and Jack Stein, Multi-purpose Optimization System: User's Guide, Version 4, Vogelback Computing Center, Northwestern University, Illinois, 1978.

Collins, D., Marine Terminal Control System, NTIS, Springfield, Va., July, 1976.

Control Data Corporation, FORTRAN Extended Version 4 Reference Manual, CDC, 1979.

Dannenbring, D. G. and B. M. Khumawala, "An Investigation of Branch and Bound Methods for Solving the Segregated Storage Problem," AIIE Transactions, September, 1973.

Edmond, E. D., "Operating Capacity of Container Berths for Scheduled Services by Queue Theory," Dock and Harbor Authority, November, 1975.

\_\_\_\_\_, "How Useful are Queue Models in Port Investment Decisions for Container Berths," Journal of Operational Research Quarterly, Vol. 29, No. 8, August, 1978.

Erichsen, Stain, Optimum Capacity of Ships and Port Terminals, Department of Naval ARchitecture and Marine Engineering, University of Michigan, April, 1971.

Ermol'ev, Yu M. et al., "Planning of Shipping Empty Seaborne Containers," Cybernetics, Vol. 12, No. 4, July-August, 1976.

Fishman, G. S., Concepts and Methods in Discrete Event Digital Simulation, John Wiley and Sons, Inc., New York, 1973.

Flores, Ivan, Data Structure and Management, Prentice-Hall, Inc., 1970.



Francis, R. and J. A. White, Facility Layout and Location: An Analytical Approach, Prentice-Hall, Inc., New Jersey, 1974.

Frankel, E. (ed.), Port Design and Analysis Methodology, Report No. MITSG 74-31, Massachusetts Institute of Technology, Mass., December, 1974.

Gass, Saul I., Linear Programming: Methods and Applications, Fourth Edition, McGraw-Hill, New York, 1975.

Geisler, R. and W. Troutnite, "Automation of a Container Terminal Using Process Computers," Siemen's Review, Vol. 44, No. 8, West Germany, August, 1977.

Glover, F., J. Hultz, D. Klingman, and J. Stutz, "Generalized Networks: A Fundamental Computer-Based Planning Tool," Management Science, Vol. 24, No. 12, August, 1978.

Glover, F., J. Hultz, D. Klingman, and J. Stutz, A New Computer-Based Planning Tool, Research Report CCS 289, University of Texas, Austin, Texas, December, 1976.

Glover, F., D. Karney, D. Klingman, and A. Napier, "A Computation Study on Start Procedures, Basis Change Criteria, and Solution Algorithms for Transportation Problems," Management Science, Vol. 20, No. 5, January, 1974.

Gordon, Geoffrey, System Simulation, Prentice-Hall, New Jersey, 1969.

Graves, Robert J. and Leon F. McGinnis, "Toward Multi-Criteria Location and Layout: Ship Facilities Layout Analysis," Proceedings--1980 AIIE Spring Annual Conference, 1980.

Greer, Willis R., Jr., "Capital Budgeting Analysis with the Timing of Events Uncertain," The Accounting Review, January, 1970.

Hanson, J. B., "Optimizing Ports through Computer Simulation Sensitivity Analysis of Pertinent Parameters," Operational Research Quarterly, Vol. 23, No. 4, 1972.

Hara, T. and Y. Takahashi, "Container Terminal System," Toshiba Review, No. 89, Japan, January-February, 1974.

Hillier, Frederick S. and Gerald Lieberman, Operations Research, Second Ed., Holden-Day, Inc., California, 1967.

Howard, H. H., "Container Handling in Deep Sea Terminals," World Ports, February-March, 1969.

Hwang, C. C., Analysis of Container Port through System Simulation, Ph.D. dissertation, North Carolina State University, Raleigh, N.C., 1978.

Jimenez, L. A., "Simulation Model to Assist in Port Capacity Planning Decisions," Proceedings of Summer Computer Simulation, AFIPS Press, Montvale, N.J., 1976.

Jones, L., Deep Sea Container Ports: Systems Appraisal and Simulation Modelling, Open University Press, U.K., 1972.

Katzan, Harry, Jr., Computer Data Management and Data Base Technology, Van Nostrand Reinhold Co., New York, 1975.

Klingman, D., A. Napier, and J. Stutz, "NETGEN: A Program for Generating Large-Scale Capacitated Assignment, Transportation, and Minimum Cost Flow Network Problems," Management Science, Vol. 20, No. 5, January, 1974.

Klingman, D., P. Randolph, and S. Fuller, "A Cotton Ginning Problem," Operations Research, Vol. 24, No. 4, July-August, 1976.

McCarthy, T. J. and J. J. Carter, Data Analysis Techniques for a Containerized Export Cargo Transportation System, NTIS, Springfield, Va., June, 1974.

Miller, J. R. et al., Digital Computer Simulation of the Rivergate Container Terminal, a report to the Port of Portland, Oregon.

Morse, Philip M., Queues, Inventories and Maintenance, John Wiley and Sons, New York, 1958.

Mulvey, John M., "Testing of a Large-Scale Network Optimization Program," Mathematical Programming, 1978.

Mytton, M. and N. Walker, "Computer Models in Planning Container Terminals," Dock and Harbor Authority, April, 1979.

Nehrling, B. C., Container Ship Loading and Unloading Simulation, Department of Naval Architecture and Marine Engineering, University of Michigan, April, 1970.

Neter, John and William Wasserman, Applied Linear Statistical Models, Richard D. Irwin, Inc., Illinois, 1974.

Ninomiya, Y. and Y. Nabishima, "Container Handling System with Computer Controlled Crane at Freight Terminals," Hitachi Review, Vol. 25, No. 5, Japan, May, 1976.

Novaes, A. G. N., An Appraisal of Queuing Model for the Solution of Port Capacity Problem, Commodity Transportation and Economic Development Laboratory, M.I.T., Cambridge, Mass., 1972.

Parsons, Ron and Lawrence Hill, Analysis and Simulation of a Seaport, Interim Report, The Sea Grant Office, M.I.T., Cambridge, Mass., May, 1971.

Peleg, K. and E. Peleg, "Container Dimensions for Optimum Utilization of Storage and Transportation Space," Computer Aided Design, Vol. 8, No. 3, U.K., July, 1976.

Pierre, Creton, An Approach to the Optimal Design of a Container Terminal, master's thesis, M.I.T., 1973.

Price, Willard and Peri Muretta, Seaport Management: A Bibliography, Institute for Marine and Coastal Studies, University of Southern California, Los Angeles, December, 1979.

Pritsker, A. Allan B., The GASP IV Simulation Language, John Wiley and Sons, New York, 1974.

Rath, Eric, Container Systems, John Wiley and Sons, New York, 1973.

Rawson, K. J. and E. C. Tupper, Basic Ship Theory, Volume 1, Longman, Inc., London, 1976.

Reeves, S. J., "Evaluation of Port Functioning Characteristics," Dock and Harbor Authority, March, 1970.

Reitman, Julian, Computer Simulation Applications, John Wiley and Sons, New York, 1971.

Riggs, James L., Production Systems: Planning, Analysis and Control, John Wiley and Sons, New York, 1976.

Ross, Terry and R. M. Soland, "A Branch and Bound Algorithm for the Generalized Assignment Problem," Mathematical Programming, 1975.

Ross, Terry and R. M. Soland, "Modelling Facility Location Problems as Generalized Assignment Problems," Management Science, November, 1977.

Ross, Terry and A. Zoltners, "Weighted Assignment Models and Their Applications," Management Science, Vol. 25, No. 7, July, 1979.

Rowe, Kenneth et al., Statistical Interactive Programming System, Command Reference Manual, Department of Statistics, Oregon State University, Corvallis, Oregon, September, 1978.

Salkin, Harvey M., Integer Programming, Addison-Wesley Publishing Co., Reading, Massachusetts, 1975.

Schmidt, J. W. and R. E. Taylor, Simulation and Analysis of Industrial Systems, Richard D. Irwin, Inc., Illinois, 1970.

— Scott, David K. and Der-San Chen, A Loading Model for a Container Ship, a paper presented at ORSA/TIMS Joint Meeting, Los Angeles, California, November, 1978.

Simmons, Donald M., Linear Programming for Operations Research, Holden-Day, Inc., San Francisco, 1972.

Thoolen, S. et al., Operational Evaluation of a Computer Loading Program for Container Vessels, MA/RD-930-79008, May, 1978.

Tozzoli, A. T., "Containerization and its Impact on Ports," Journal of the Waterways, Harbors and Coastal Engineering Division of ASCE, August, 1972.

Weinberg, Gerald M., An Introduction to General Systems Thinking, John Wiley and Sons, New York, 1975.

West, Tom M., Edward McDowell and Doug W. Cho, "Simulation of Operations in a Marine Container Port," Paper presented in Joint ORSA/TIMS Meeting at Colorado Springs, Colorado, November, 1980.

Whittaker, J. R., Containerization, Second Ed., John Wiley and Sons, New York, 1975.

Whitehouse, Garry E., Systems Analysis and Design Using Network Techniques, Prentice-Hall, Inc., New Jersey, 1973.

## APPENDICES

## APPENDIX A

Summary of Guidelines for Loading  
Containers Shipboard

(Source: Rath)

1. To minimize susceptibility to severe rolling, a containership should have a small bilge radius, carried well forward and aft, in conjunction with bilge keels.
2. To minimize susceptibility to severe pitching, a containership should have a low displacement/length ratio.
3. Containerships should be fitted with bilge keels to minimize susceptibility to rolling.
4. Bilge keels should be located right at the turn of the bilge for maximum effectiveness and should be greater than 18 inches in depth, to insure that they extend beyond the boundary layer.
5. Model tests should be made to determine optimum trace of bilge keels for minimum resistance, and to determine resistance and effectiveness trade-off data for establishing optimum depth.
6. If a passive-tank roll-damping system is used, it should be thoroughly model tested before installation to assure effectiveness over the full range of the sea conditions expected in service.
7. Activated stabilization systems on containerships should include safeguards in the system to prevent induced or aggravated rolling in the event of derangement of the normal control system.
8. For the development of intact trim and stability calculations, the following criteria should be applied:
  - a. Container weights should be based on data from the operator's experience on the particular trade involved. Where there are no data, according to Maritime Administration Guidelines, weight should be assumed as 60 percent of the rated maximum.
  - b. Permeabilities should be assumed 0.60 for volume occupied by containers and 0.95 for remainder of hold.

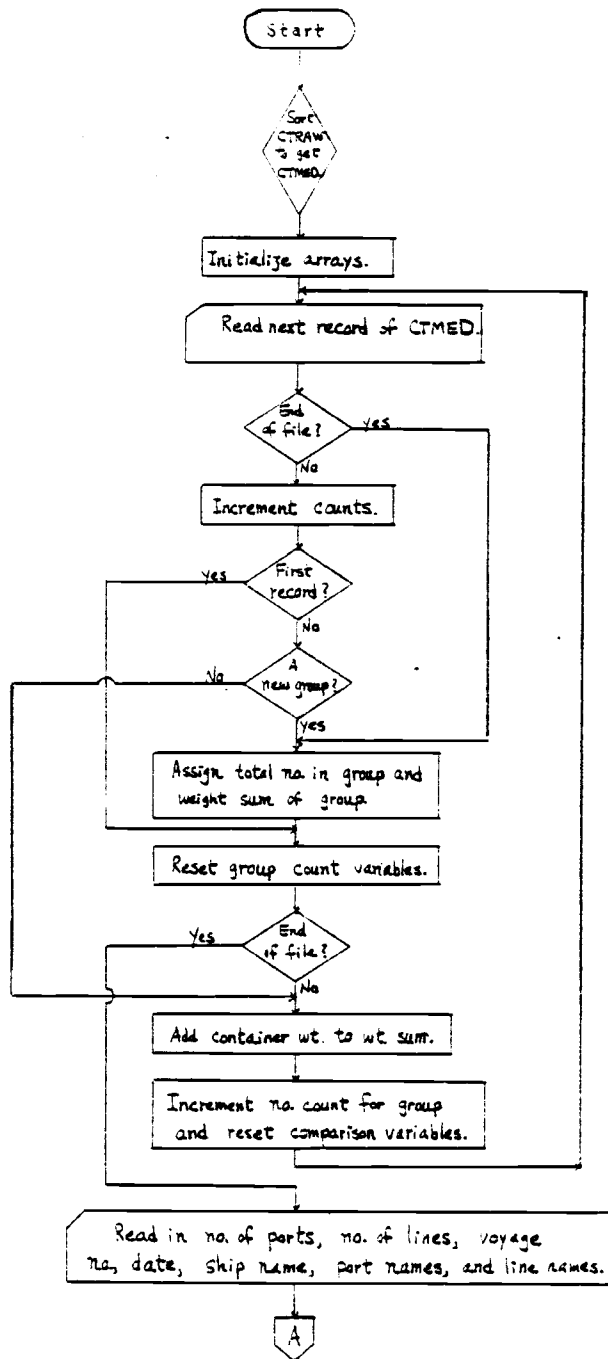
9. Permanent ballast should be installed in the minimum amounts necessary to satisfy the U.S. Coast Guard stability standards.
10. Permanent ballast required because of possible deck cranes or other installations to be made in the future should not be fitted until the future installation is actually made.
11. Freeboard of a containership midbody should be at least 60 percent greater than the minimum required by the regulations.
12. Freeboard at the bow of a containership should be at least 6 percent of the overall length of the ship.
13. Investigations for adequacy of freeboard in severe ocean conditions on the routine contemplated should be made in the preliminary design of a containership.
14. Flare on the order of at least 25 to 35 degrees should be incorporated into the forward sections of a containership to reduce the possibilities of boarding seas at the bow.
15. Model tests should be made of any new containership design to determine optimum flare.
16. A containership should have a substantial breakwater as secondary protection for the forwardmost deck-stowed containers against green water shipped over the bow.
17. Bulwarks for protection of containers stowed on deck from boarding seas should be provided on ships operating with freeboards less than about 25 percent of the beam.
18. Any ship which routinely carries containers on deck should have a container restraint system with permanently installed and portable fittings specifically designed for the size and weight of containers being carried and for the particular ship involved.
19. Selection of the type of securing system for deck-stowed containers on a containership should be based on trade-off studies of structural, lashed, and locked systems, considering life-cycle costs and degree of security afforded.
20. On-deck containers should be positioned on restraint fittings.
21. Positioning and restraint fittings for on-deck containers should be located within specified tolerances so as to fit the containers to be carried.

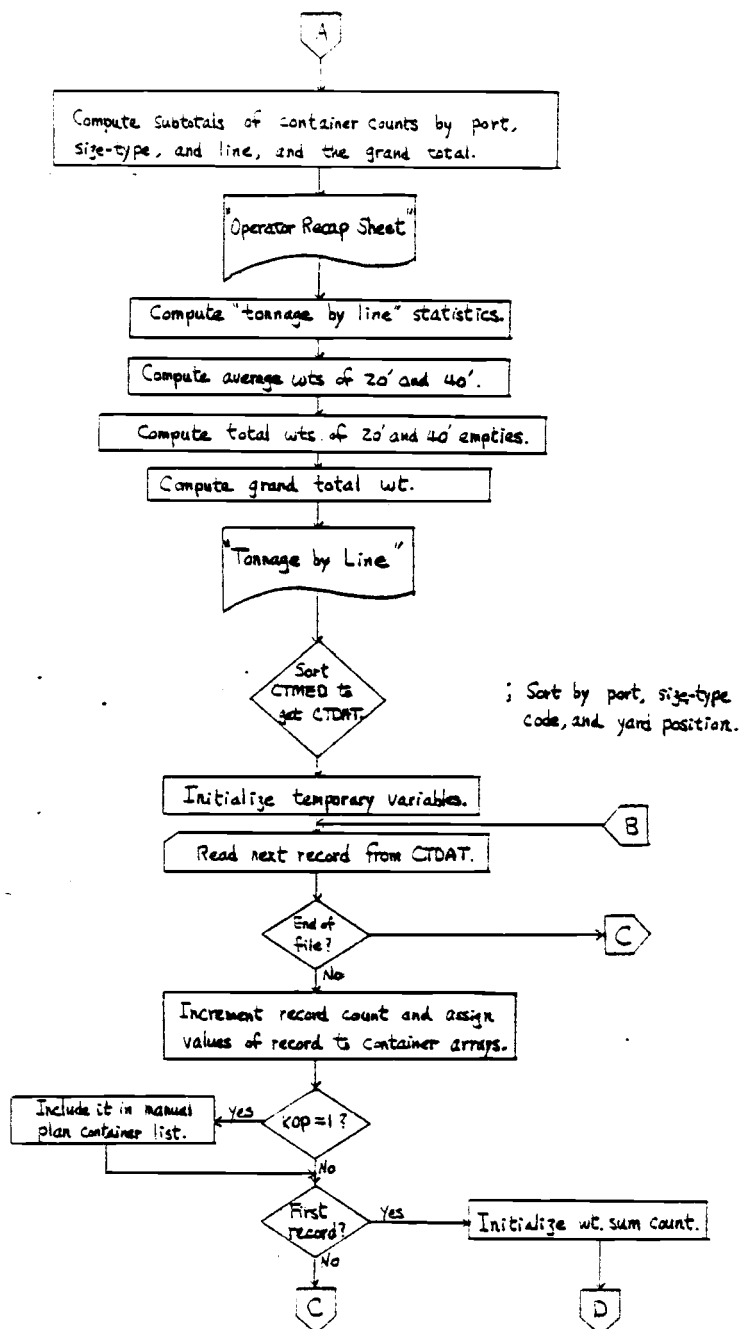
22. Ship structure under decks supporting containers should have sufficient local strength to support the container weight concentrated at the container corner fittings.
23. Where containers are carried over hatches, the strength of the hatch covers should be sufficient to carry the load of the containers, or supports of sufficient strength should be fitted across the hatch openings.
24. The supports for containers on deck should be adequate to carry the longitudinal and transverse loads from ship motions into the ship's primary hull structure.
25. The design of structural supports for on-deck containers should be based on the combined gravity and maximum dynamic loads from ship motions, and on a factor of safety of three on the ultimate strength of the material.
26. Lashing assemblies for securing containers on deck should include provisions for readily tensioning the lashing when installed to remove all slack and avoid excessive pretension in the assembly.
27. Stacking fittings should be used between tiers of containers lashed on deck, at each corner fitting, to prevent horizontal movement.
28. Bridge fittings should be used to provide horizontal support between rows of containers within a stack, and should be of a type to also resist tipping.
29. The number and arrangement of lashings in a lashed securing system for containers should be determined from calculations based on predicted sea conditions and the weight and stiffness characteristics of the containers to be carried.
30. The required strength of container lashings should be based on maximum loads determined from lashing calculations, a factor of safety of two on the breaking strength for lashing assemblies and attachment fittings, and a factor of safety of three on the ultimate strength of supporting structure.
31. Lashing assemblies should be proof tested and samples should be tested to failure to verify designed performance.
32. The arrangement of a buttress securing system for on-deck containers should provide for minimum interference of the structural posts with container handling and stowage, and for ready handling of the stowing frames by container-lifting equipment.

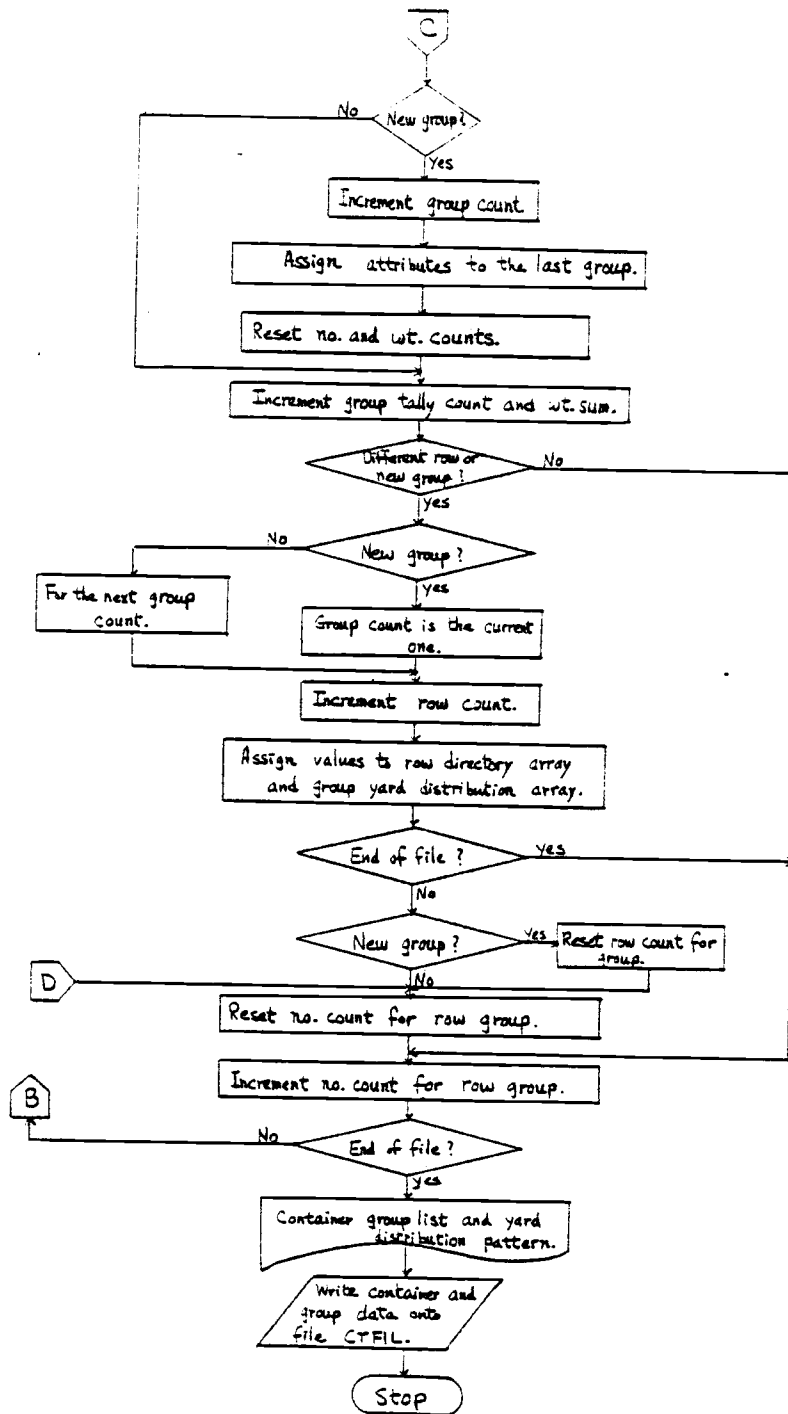


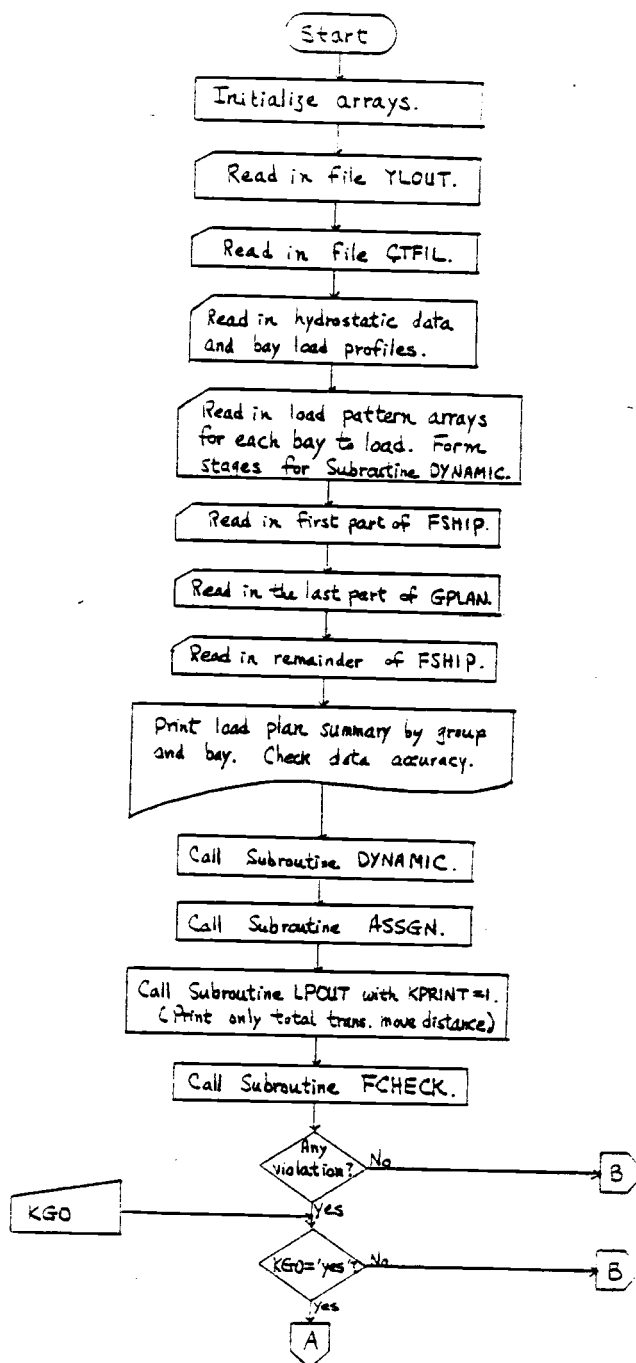
33. The strength of a buttress system should be based on maximum combined gravity and dynamic loads for expected extreme sea conditions and a factor of safety of three on the ultimate strength of the material of the structure.
34. In selection of arrangements for on-deck stowage of containers, athwartships arrangements should be included in the alternatives for consideration and evaluation.
35. Arrangements for container stowage on deck should include fore and aft access for personnel between container stowages and the deck edge with clear passageways a minimum of about three feet in width.
36. Where container stowages on deck require access to make up and inspect the restraint system, the arrangement should provide for a minimum of 30 inches of spacing between the ends of containers in adjacent rows.
37. Where reefer containers are carried on deck, the arrangement should provide a minimum of 30 inches' clearance for the container ends in which the reefer units are mounted.
38. Where ventilated containers with covers required to be opened and closed at sea are carried on deck, the arrangement should provide a minimum of 30 inches' clearance between the ends of containers for access to the covers and to permit the covers to swing open.
39. Fire stations, sounding tubes, and other features on deck requiring access at sea should be located clear of container stowages.
40. Where containers are carried directly on deck outboard of a hatch, stowage fittings should be provided under the container corners to hold the containers securely in position and to elevate them clear of a wet deck.

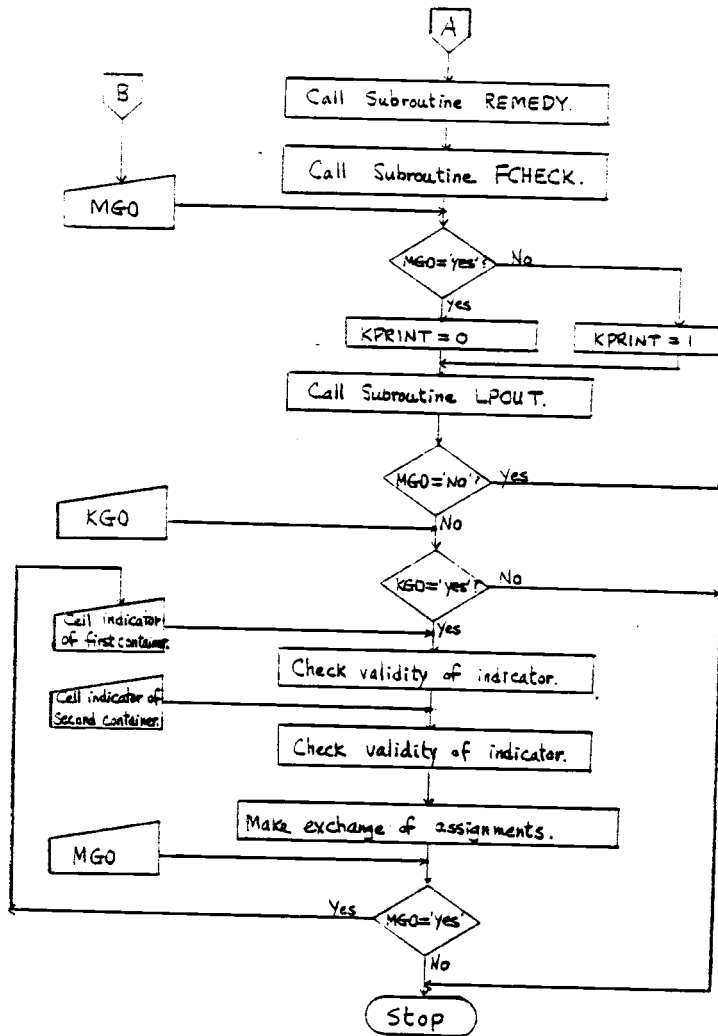
## APPENDIX B

COMCL Flowcharts1. Program DATPREP

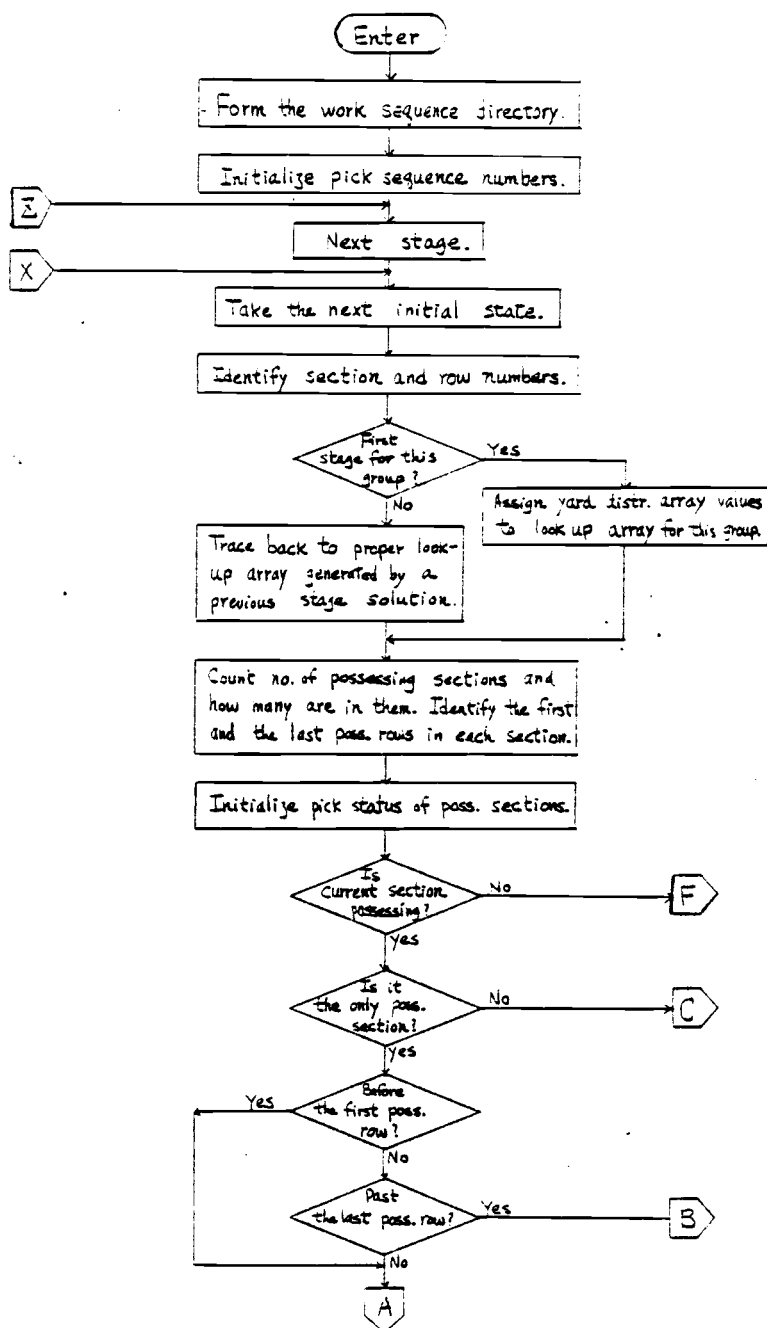


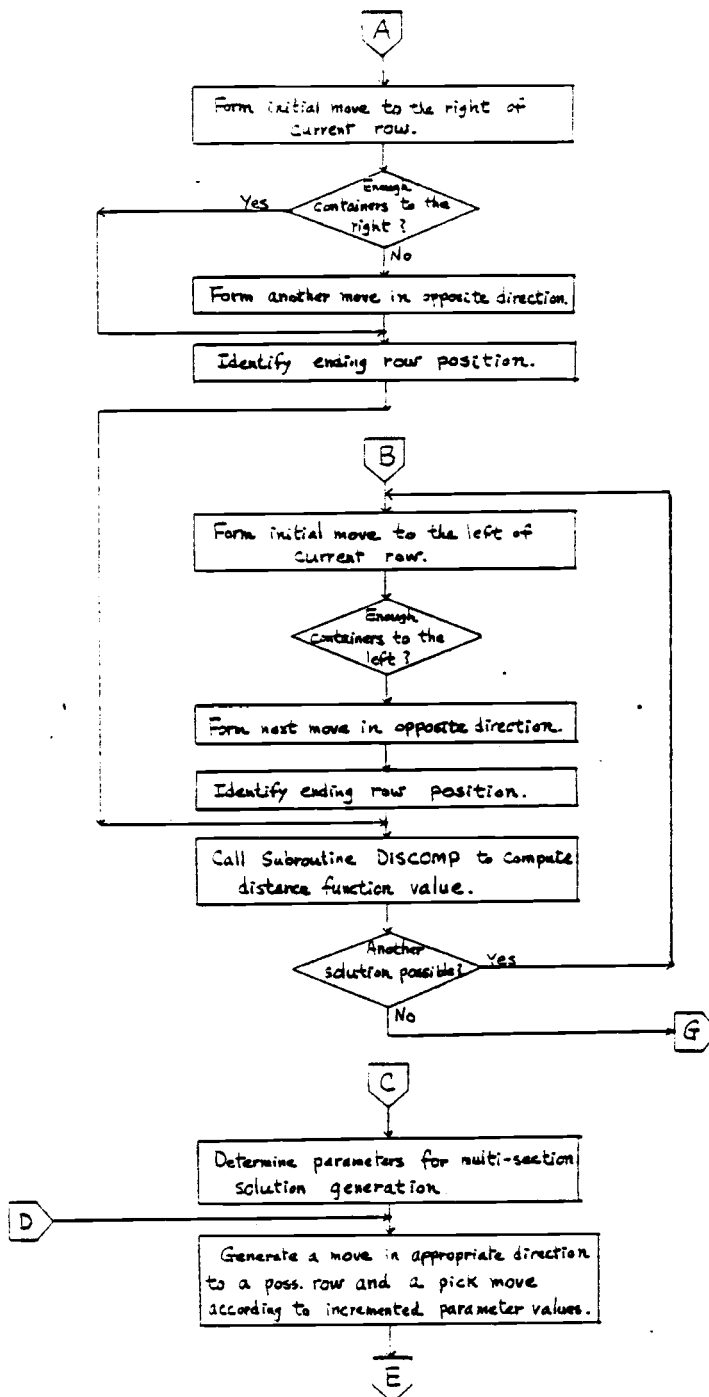


2. Program CLP

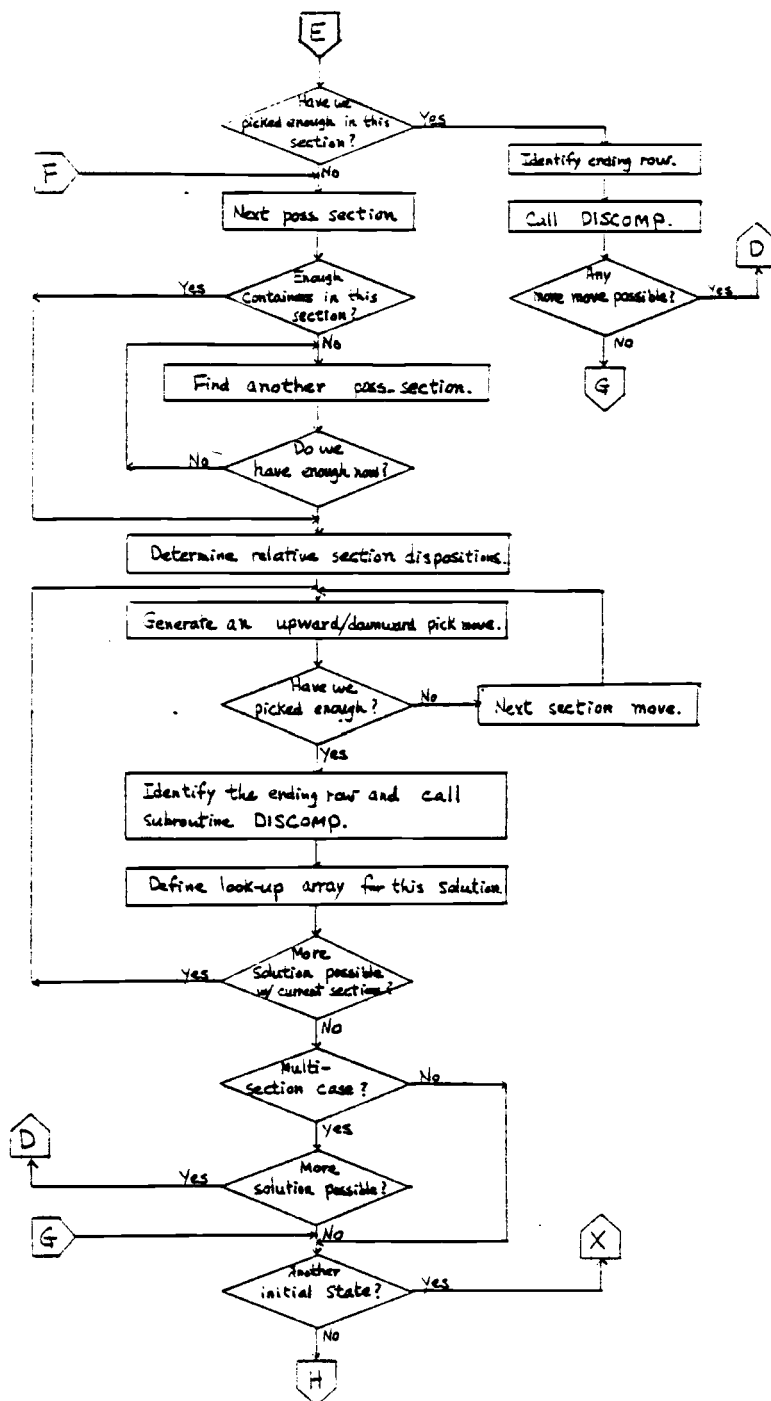


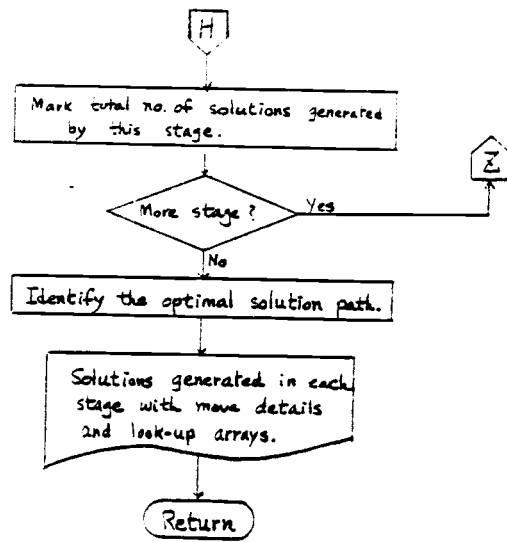
# Subroutine DYNAMIC

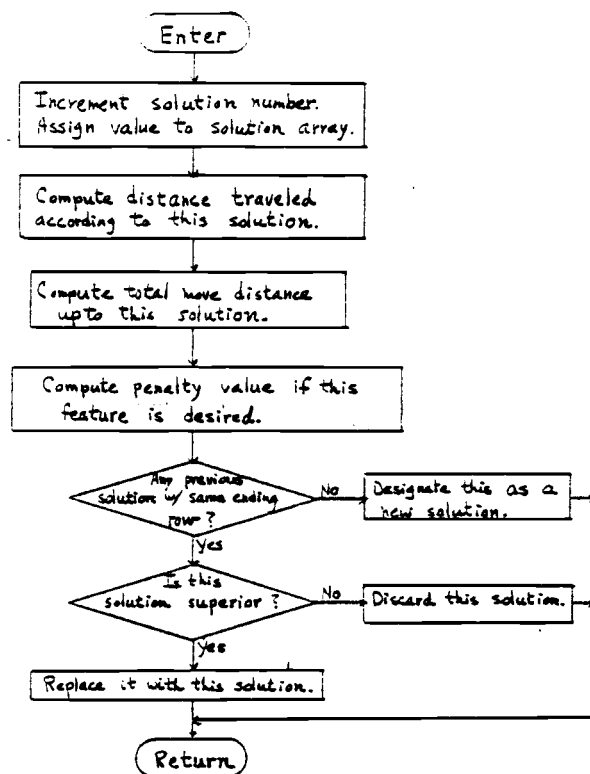
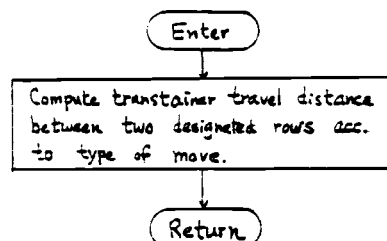


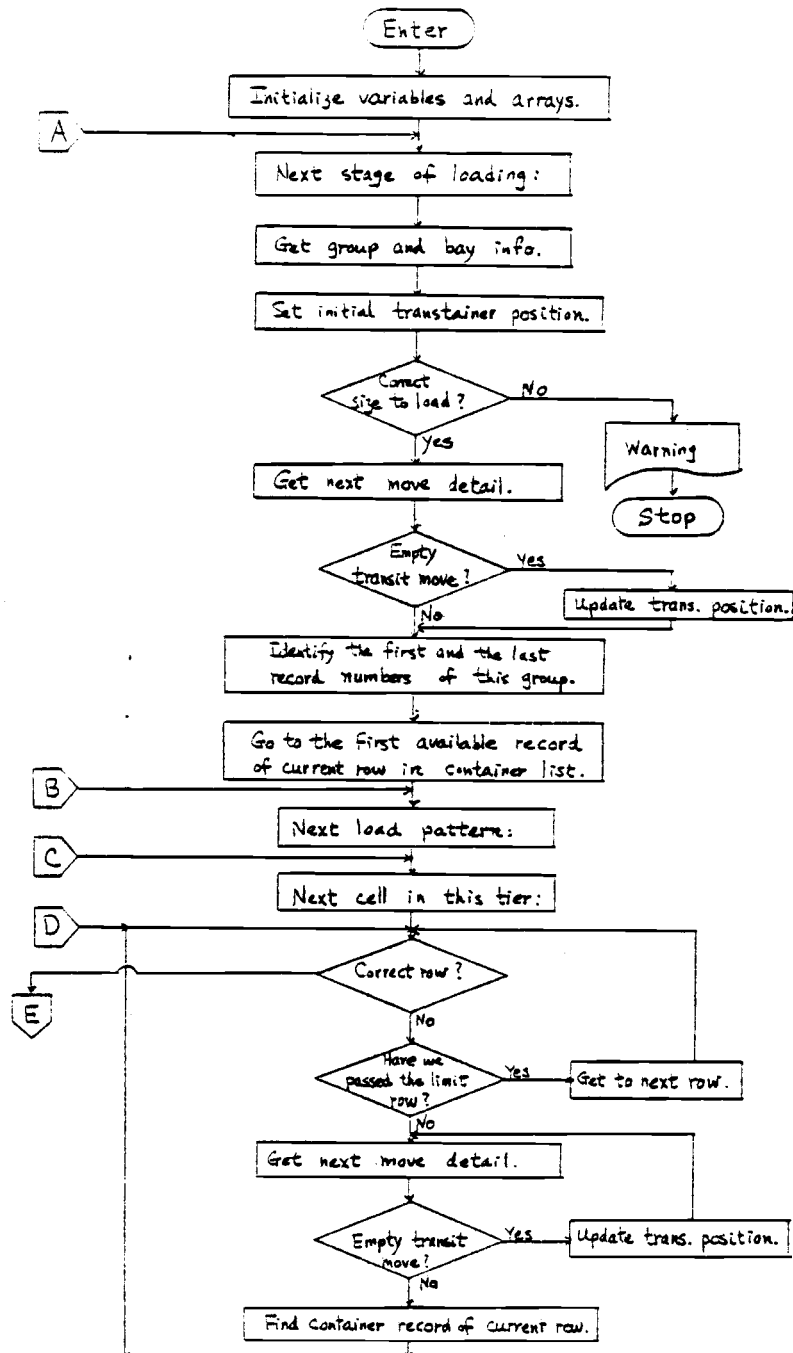


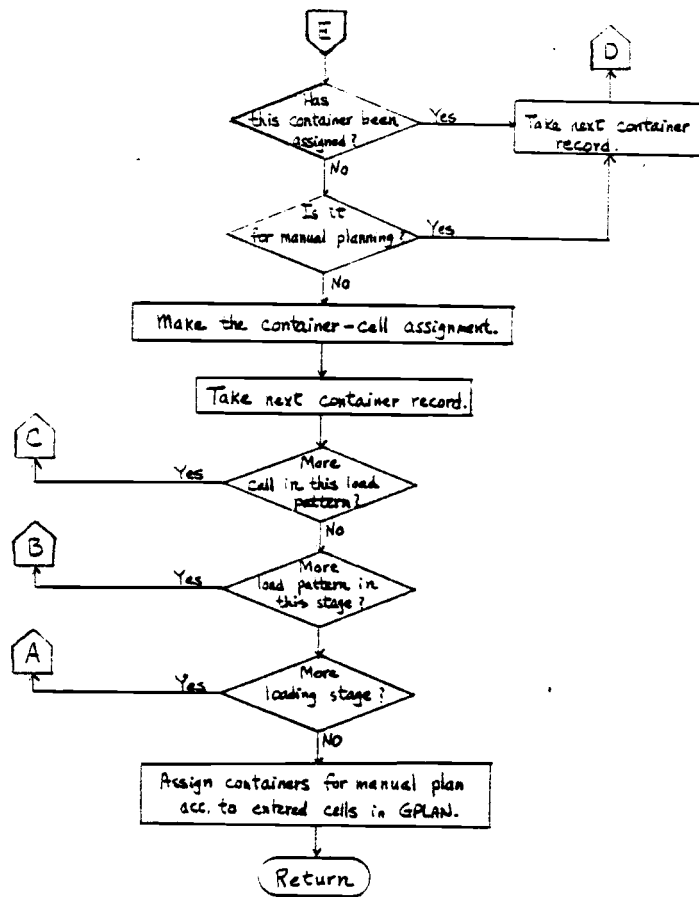


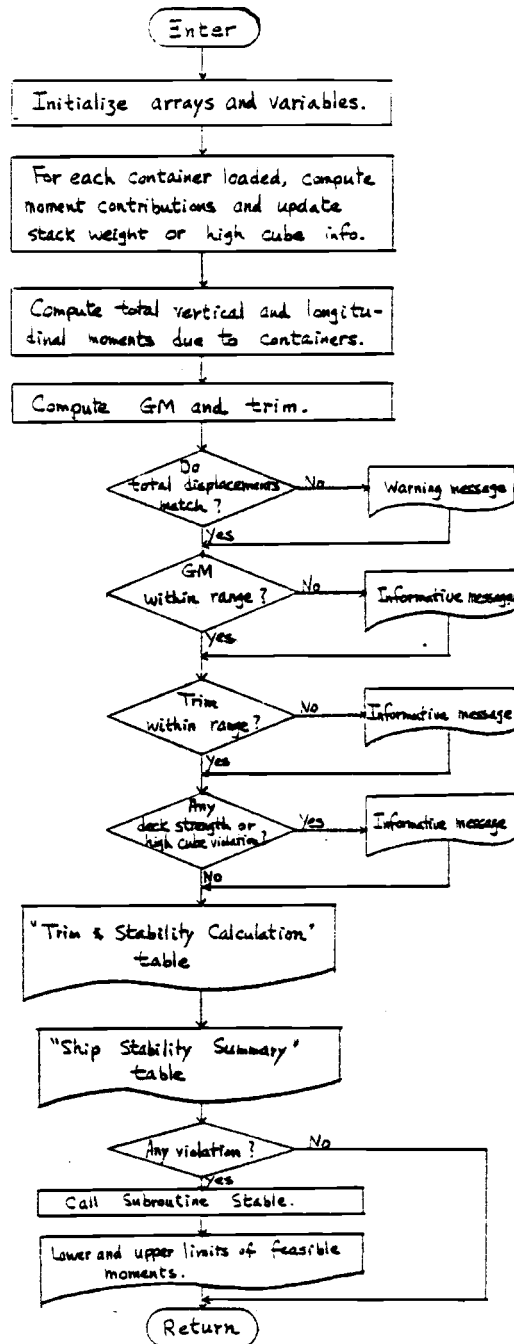


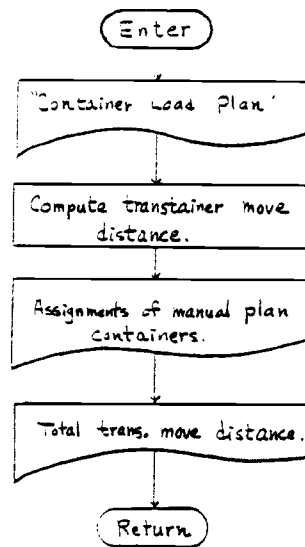


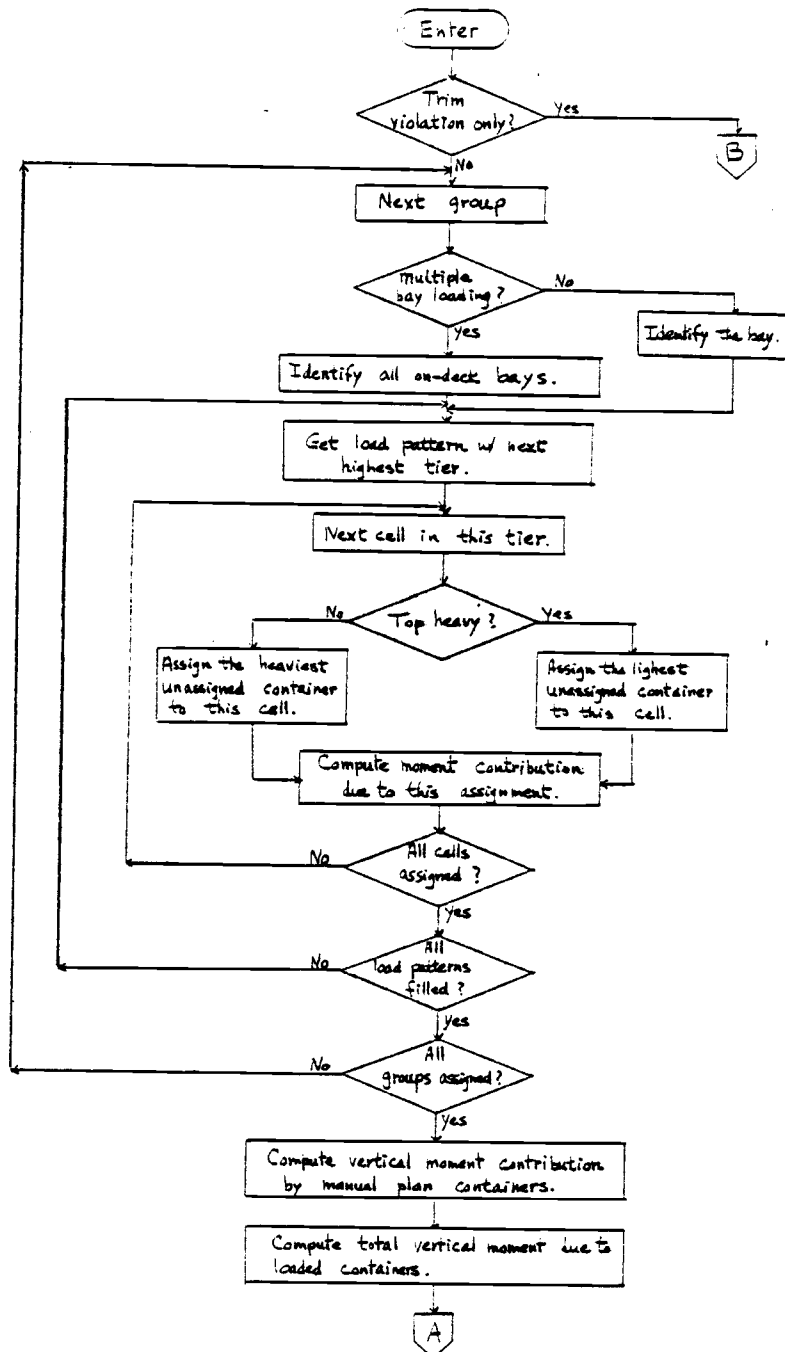
Subroutine DISCOMPFunction NDIST Flowchart

Subroutine ASSGN

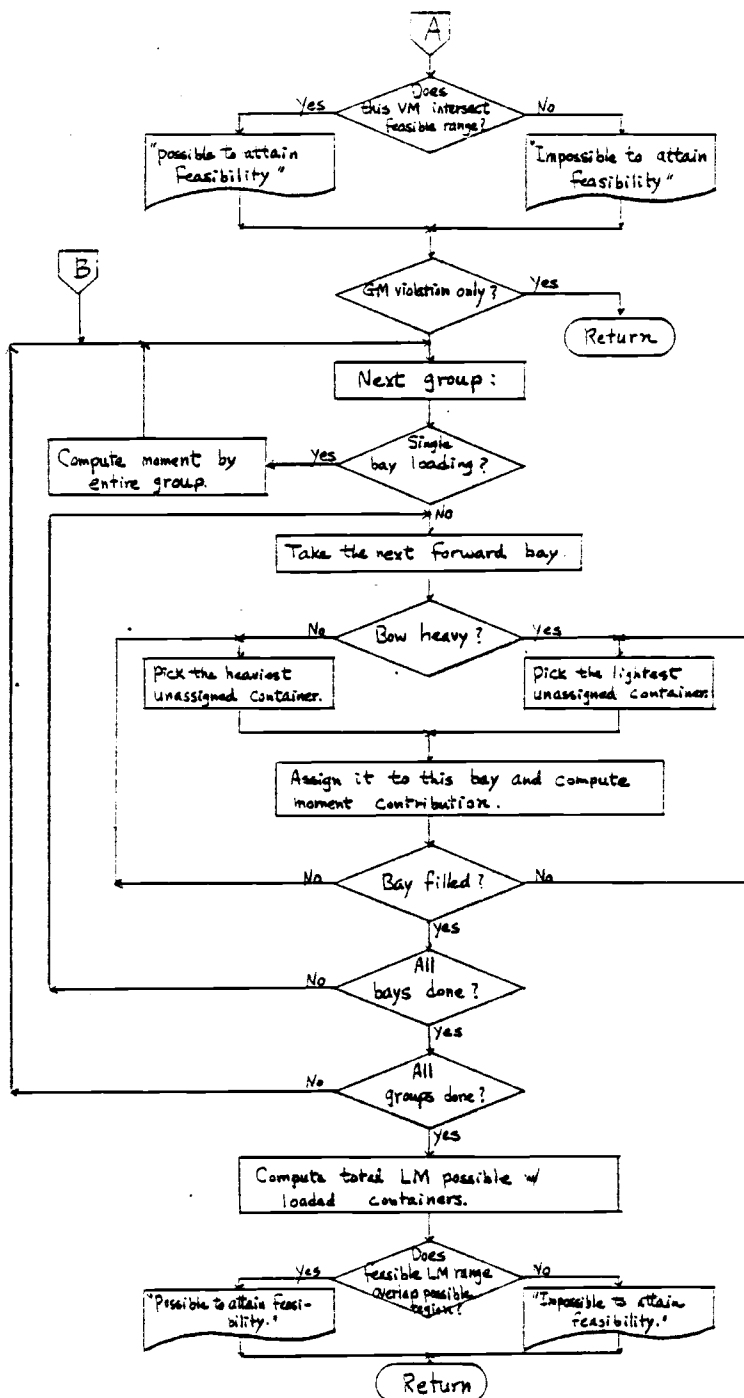


Subroutine FCHECK

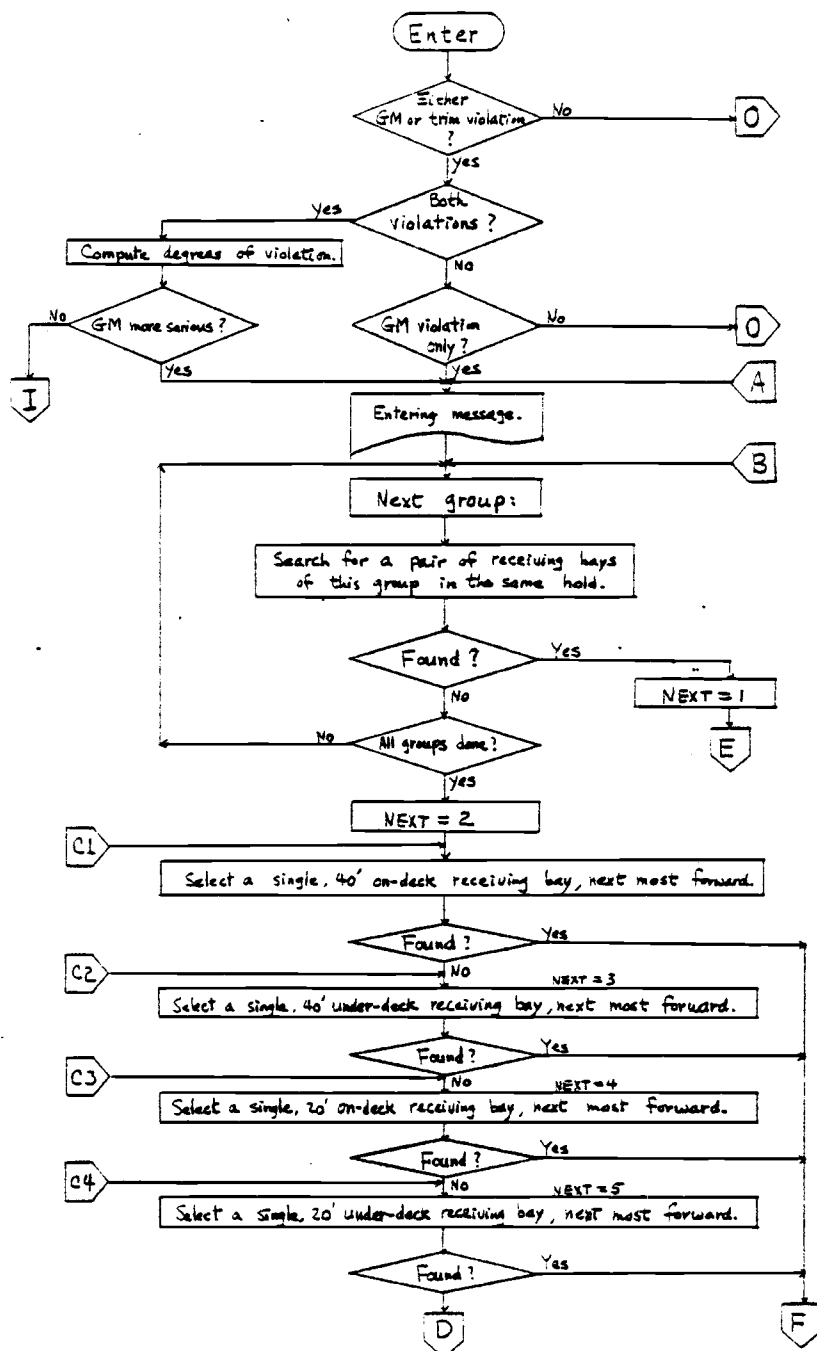
Subroutine LPOUT

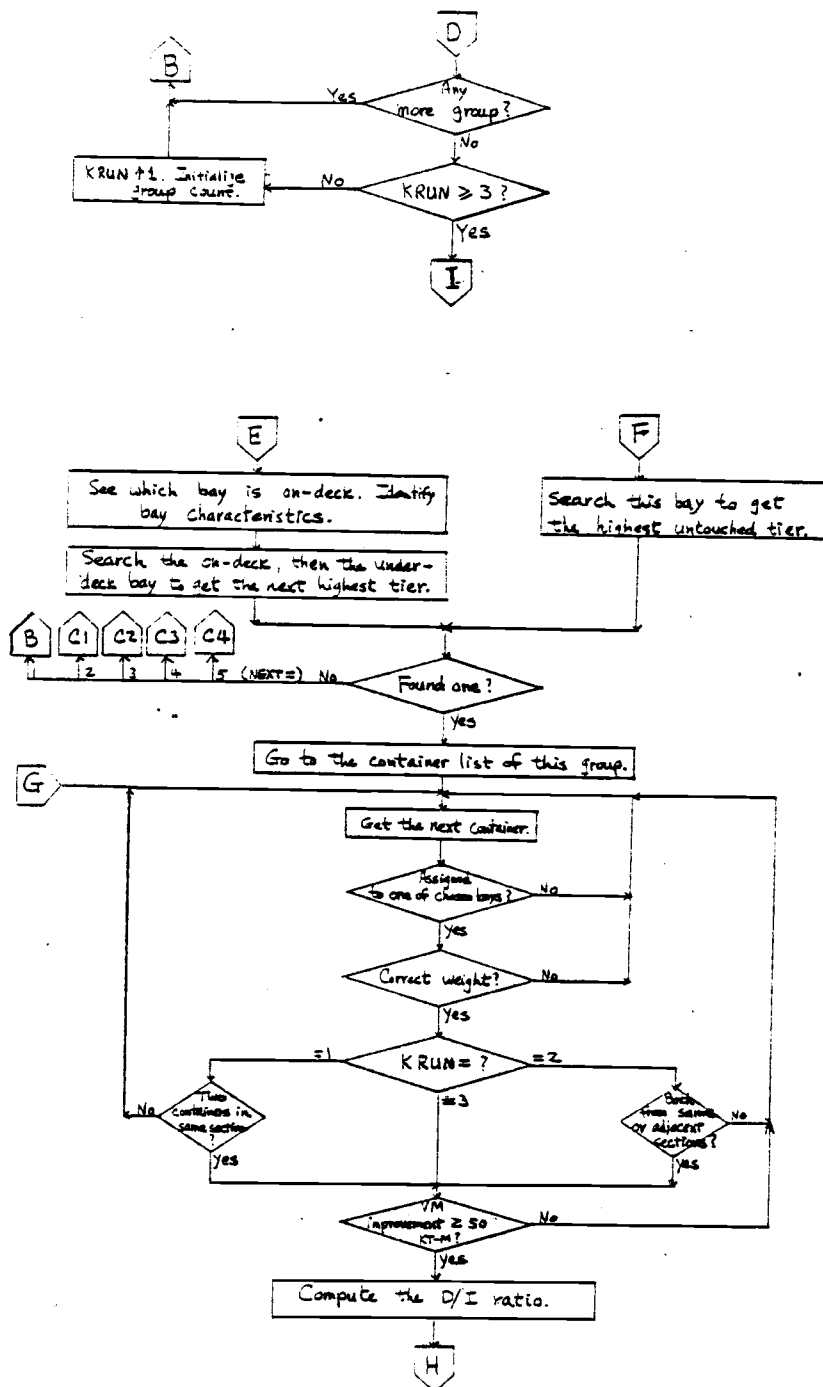
Subroutine STABLE

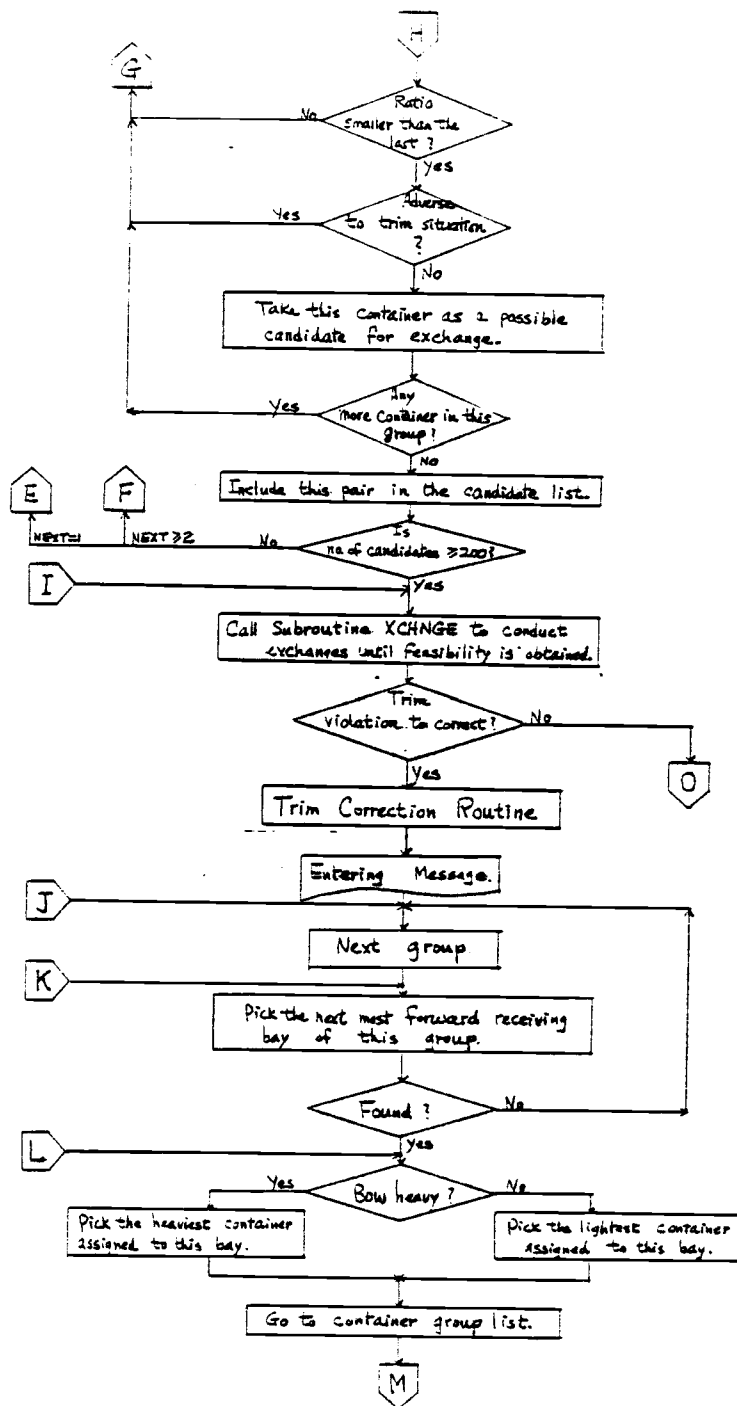


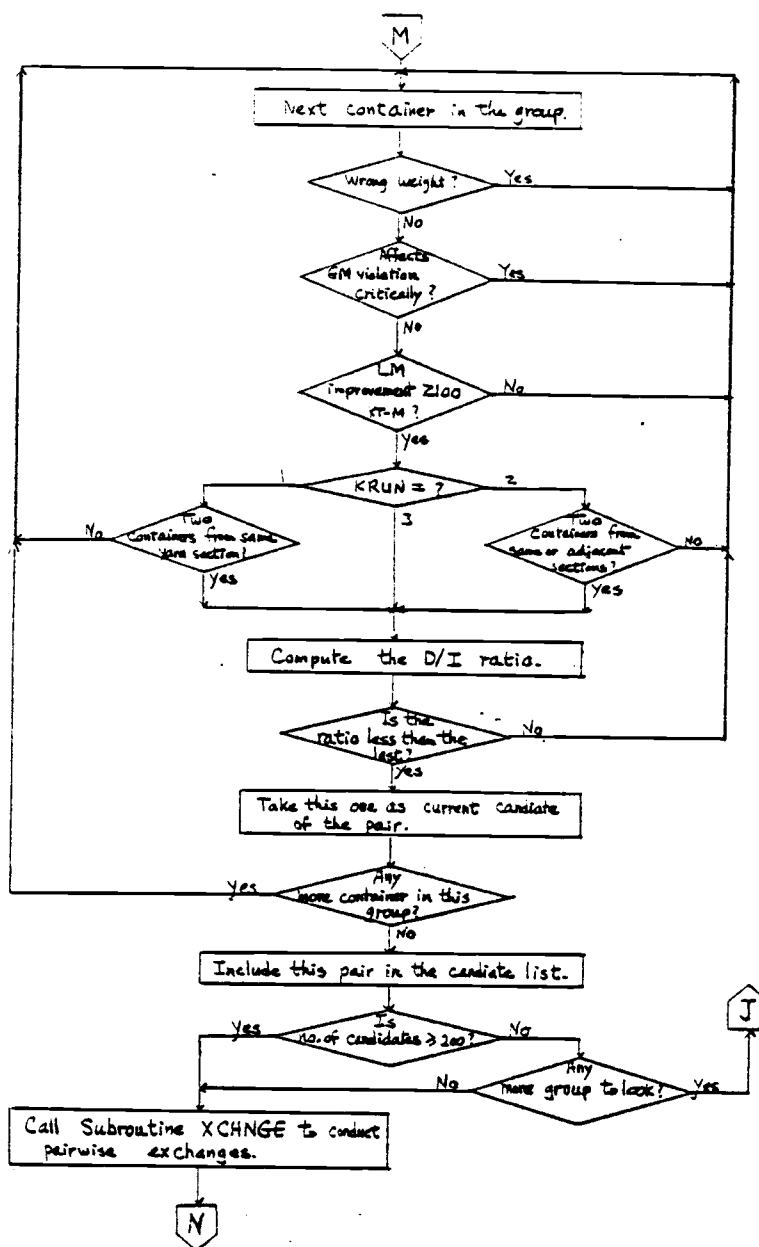


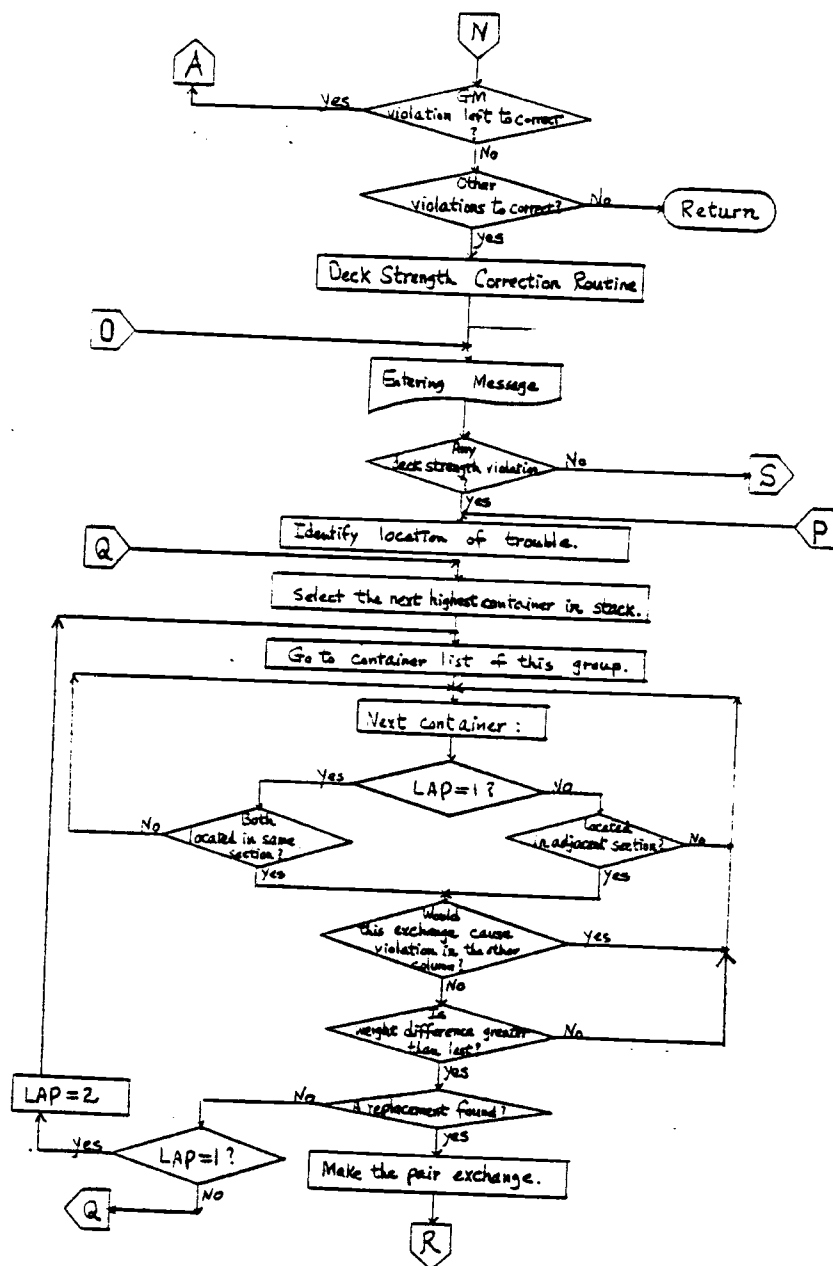
# Subroutine REMEDY

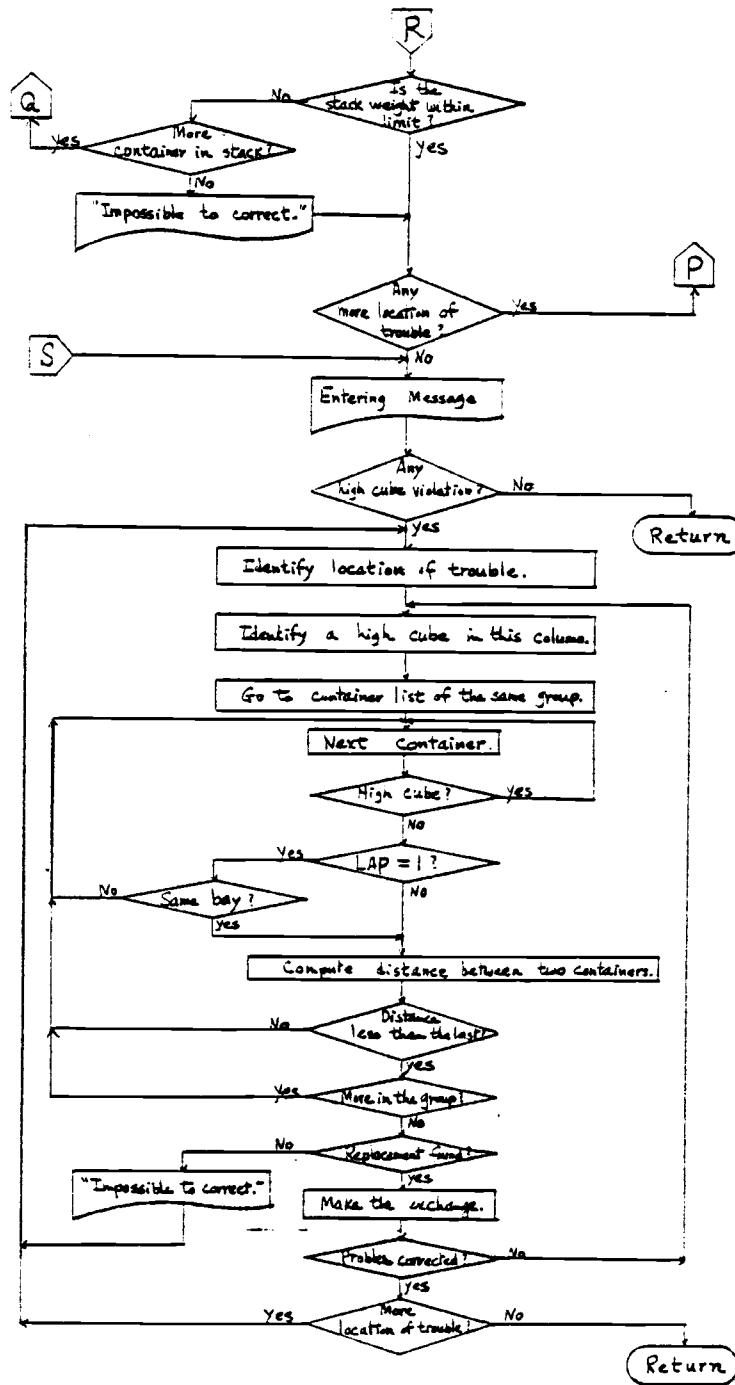


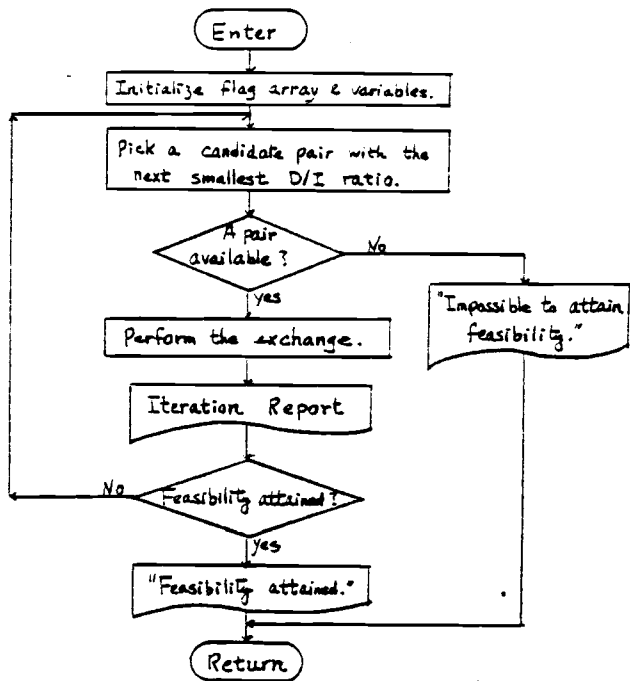
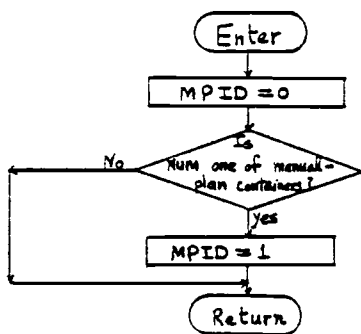




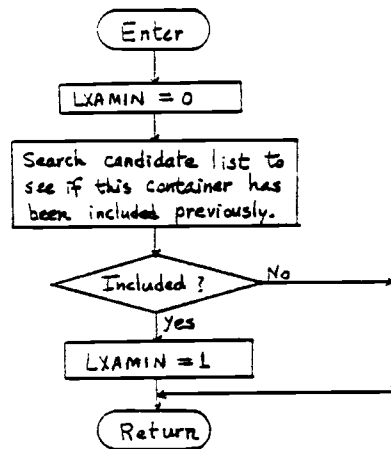






Subroutine XCHNGEFunction MPID Flowchart



Function LXAMIN

## APPENDIX C

COMCL Input File Specifications

1. File CTRAW Specification
2. File RUNDAT Specification
3. File YLOUT Specification
4. File FSHIP Specification
5. File GPLAN Specification

1. File CTRAW Specification

Each record corresponds to each container to load.

| <u>Column</u> | <u>Format</u> | <u>Data</u>   |
|---------------|---------------|---|
| 1             | I1            | PORT<br><br>(1:Kobe 2:Nagoya 3:Shimizu 4:Yokohama)  |
| 3             | I1            | SIZE-TYPE<br><br>1:20' Full    2:20' Reefer    3:20' Special<br>4:20' Empty    5:40' Full    6:40' Reefer<br>7:40' Special    8:40' Empty |
| 5             | A6            | YLOC: yard location<br><br>(e.g., 4225BT)   |
| 12            | I1            | LINE<br><br>1:Japan Line    2:K Line<br>3:MOSK    4:NYK<br>5:SHOWA Line    6:YS Line  |
| 14            | I2            | COMM: type of commodity<br><br>1: Dry    2: Reefer    3: Explosives<br>4: O/T    5: ORM    6: Corrosives                                  |

|    |      |   |           |           |
|----|------|---|-----------|-----------|
|    |      | 7: Flamm.   | 8: Oxides | 9: F/R    |
|    |      | 10: Bulk  | 11: Hide  | 12: Other |
| 17 | F4.1 | WT: weight in MT<br>(e.g., 23.5)  |           |           |
| 22 | I1   | HC: High-cube or not<br>0: if not high cube<br>1: if high cube  |           |           |
| 24 | A11  | ID: container ID number   |           |           |
| 35 | I1   | Op Code<br><br>0: if regular operation<br>1: if special operation (cell location to<br>be designated by operator) |           |           |

## 2. File RUNDAT Specification

| <u>Record</u> | <u>Column</u> | <u>Format</u> | <u>Data</u>             | <u>Description</u>    |
|---------------|---------------|---------------|-------------------------|-----------------------|
| 1.            | 1             | I2            | IP                      | No. of ports          |
|               | 3             | I2            | LN                      | No. of lines          |
|               | 6             | L6            | NDATE                   | Date (month-day-year) |
|               | 12            | I2            | NVOY                    | Voyage no.            |
|               | 15            | A10           | SHIPID                  | Ship name             |
| 2.            | 1             | 8A10          | PNAME (I),<br>I = 1, IP | Port names            |
| 3.            | 1             | 8A10          | LNAME (I),<br>I = 1, LN | Line names            |

## 3. File YLOUT Specifications

General format: list-directed input (no format)

| <u>Record</u> | <u>Data</u> | <u>Description</u>                               |
|---------------|-------------|--|
| 1.            | NYSEC       | No. of yard sections                             |
|               | NER1        | Equivalent RE's for adjacent<br>section transfer |

|    |   |   |
|----|---|---|
|    | NER2  | Equivalent RE's for non-adjacent section transfer     |
| 2. | IYSEC(I), I = 1,NYSEC                         | Individual section numbers                            |
| 3  | (For I = 1, NYSEC)<br>IAYS(I, J), J = 1,NYSEC | Relative disposition code for section i and section j |

#### 4. File FSHIP Specifications

##### A. Definition of Variables and Arrays

| <u>Var/Array</u> | <u>Description/Definition</u>  |
|------------------|--|
| NBAYS            | Total number of bays in the ship   |
| IBNO(I)          | Bay number of the ith bay, from bow toward stern (on-deck and under-deck bays are separate)  |
| LDSYS            | Bay number designation system<br>=1 for Japan Apollo, Lions Gate Bridge, and Alaska Maru<br>=2 for Hikawa Maru, Beisha Maru, and Hotaka Maru |
| LSUP             | Location of the superstructure. The first bay number after of the superstructure is used   |
| NBSEC            | Number of bay sectors  |
| MCOLD(K)         | Maximum number of columns, on-deck in sector K   |
| MCOLH(K)         | Maximum number of columns, under-deck in sector K  |
| MTIRD(K)         | Maximum number of tiers, on-deck in sector K   |
| MTIRH(K)         | Maximum number of tiers, under-deck in sector K  |
| KBKEY(I)         | Characteristic code for the ith bay  |

a b cc dd ee ff

a: Location (1:on-deck, 2:under-deck)

b: Type (1:20'only, 2:40'only, 3:both)

cc: Sector number

dd: adjacent 20' bay no. which gets  
occupied by loading a 40' container  
in this bay (00 if not applicable)

ee: Number of columns in this bay

ff: Number of tiers

JHOLD(I) Sector number of this bay

DEKL2(I) Deck strength limit in MT for 20' stacks  
in the ith bay (for on-deck bays only)

DEKL4(I) Deck strength limit in MT for 40' stacks  
in the ith bay (for on-deck bays only)

CGL2(I) Longitudinal center of gravity for 20'  
container cells in the ith bay

CGL4(I) Longitudinal center of gravity for 40'  
container cells in the ith bay

NCEL(I,K) Number of cells in the Kth column, ith bay

MHCH(I,K) Maximum number of high cubes allowed in  
the Kth column, ith bay (for under-deck  
bays only)

WTLS Displacement at light ship condition

VMLS Vertical moment at ship condition

TMLS Longitudinal moment at ship condition

GMMAX Upper limit of GM

GMMIN Lower limit of GM

TRMAX Upper limit of trim

TRMIN Lower limit of trim

BPL Length of ship from bow to propeller (M)

VCG2D(M,L) VCG for 20' cells in the lth tier, on-  
deck, sector M

VCG4D(M,L) VCG for 40' cells in sector M

VCG2H(M,L) VCG for 20' cells in the lth tier, under-  
deck, sector M

VCG4H(M,L) VCG for 40' cells in sector M

## B. Record Structure (List-directed Input)

| <u>Record</u> | <u>Structure</u>                                     | <u>How Many</u> |
|---------------|--|-----------------|
| 1             | NBAYS, LSUP, LDSYS, NBSEC                            | 1               |
| 2             | MCOLD(K), MCOLH(K), MTIRD(K),<br>MTIRH(K)            | NBSEC           |
| 3             | WTLS, VMLS, TMLS, GMMAY, GMMIN,<br>TRMAX, TRMIN, BPL | 1               |
| 4A            | IBNO(I), KBKEY(I), JHOLD(I)                          | NBAYS           |
| 4B            | DEKL2(I), DEKL4(I) (on-deck only)                    |                 |
| 4C            | CGL2(I), CGL4(I)                                     | NBAYS           |
| 4D            | NCEL(I,K)  | No. of columns  |
| 4E            | MHCH(I,K) (under-deck only)                          | No. of columns  |
| 5A            | VCG2D(M,L)   | MTIRD(M)        |
| 5B            | VCG4D(M,L)   | MTIRD(M)        |
| 5C            | VCG2H(M,L)   | MTIRH(M)        |
| 5D            | VCG4H(M,L)   | MTIRH(M)        |

## 5. File GPLAN Specifications

### A. Variables/Arrays Definition

| <u>Var/Array</u> | <u>Definition</u>                                |
|------------------|--|
| DISP             | Total ship displacement after loading (MT)       |
| DRAFT            | Mean draft at the displacement (M)               |
| HKM              | Metacentric height at the displacement (M)       |
| CMT              | Moment required to change trim by 1 Cm<br>(MT-M) |
| CBL              | Longitudinal center of buoyancy (M)              |
| CFL              | Longitudinal center of flotation (M)             |
| TKWT             | Total weight of tank stores (MT)                 |
| TKVM             | Vertical moment due to tank stores (MT-M)        |
| TKLM             | Longitudinal moment tank stores (MT-M)           |

|           |  |
|-----------|--|
| FSMT      | Free surface moment due to tank sotres<br>(Mt-M)   |
| COWT      | Constant loads, weight (MT)  |
| COVM      | Constant loads, vertical moment (MT-M)   |
| COLM      | Constant loads, longitudinal moment<br>(MT-M)  |
| NBAYL     | Number of bays to load   |
| IBAYL(I)  | Bay number of the ith bay in load sequence   |
| LOCA      | Location of bay<br>(=1 if on-deck, =2 if under-deck)   |
| NCBL(I)   | Number of columns to receive at least one<br>container in the ith bay to load  |
| NTGB(K)   | Bay-group load descriptor for the Kth<br>stage of loading  |
| NTGB(.)   | NTGB(.) = aa bb cc<br><br>aa : Bay number<br>bb : Group number<br>cc : Number of load pattern arrays (LPAT's)  |
| LPAT(K,N) | Loan pattern descriptor in the Kth stage<br>of loading. The second argument specifies<br>the detailed order of loading in this stage<br><br>LPAT(.,.) = aa bb cc<br><br>aa : Tier number to load. Tiers are num-<br>bered sequentially from bottom and up<br><br>bb : Column number to start loading in<br>tier aa. Columns are numbered se-<br>quentially from port to starboard<br><br>cc : Number of cells to fill consecutively<br>from column bb in tier aa |
| KOLL(I,N) | Column number of the nth receiving column<br>in this bay   |
| NHC(I,N)  | Number of high cubes currently loaded in<br>the nth receiving column in this under-<br>deck bay  |
| WTOD(I,N) | Total stack weight in the nth receiving<br>column in this on-deck bay  |

|          |  |
|----------|--|
| MSTAGE   | Number of loading stages                                 |
| PLWT2(I) | Total weight of 20' containers currently loaded in bay i |
| PLLM2(I) | LM due to 20' containers currently loaded in bay i       |
| PLVM2(I) | VM due to 20' containers currently loaded in bay i       |
| PLWT4(I) | Total weight of 40' containers currently loaded in bay i |
| PLLM4(I) | LM due to 40' containers currently loaded in bay i       |
| PLVM4(I) | VM due to 40' containers currently loaded in bay i       |
| MPLAN    | Number of special plan containers                        |
| MPCL(K)  | Assigned cell number of the Kth special plan container   |
|          | MPCL(.) = aa bb cc                                       |
|          | aa : Bay number  |
|          | bb : Column number                                       |
|          | cc : Tier number   |
| JBROW    | Initial transtainer location                             |
|          | JBROW = aa bb  |
|          | aa : Section number                                      |
|          | bb : Row number  |

B. Record Structure (List-directed input)

| <u>Record</u> | <u>Data</u>                     | <u>How Many</u> |
|---------------|---------------------------------|-----------------|
| 1             | DISP, DRAFT, HKM, CMT, CBL, CFL | 1               |
| 2             | TKWT, TKUM, TKLM, FSMT          | 1               |
| 3             | COWT, COVM, COLM                | 1               |
| 4             | NBAYL, JBROW                    | 1               |
| 5A            | IBAYL(I), LOCA, NCBL(I)         | NBAYL           |
| 5B            | KOLL(I,N), N=1, NCBL(I)         |                 |



|    |   |        |
|----|---|--------|
| 5C | NHC(I,N), N=1, NCBL(I) (under-deck only)  |        |
|    | or WTOD(I,N), N=1, NCKL(I) (on-deck only) |        |
| 6  | MSTAGE                                    |        |
| 7A | NTGB(K)                                   | MSTAGE |
| 7B | LPAT(K,N), N=1, NPAT                      | MSTAGE |
| 8A | PLWT2(I), PLLM2(I), PLVM2(I)              | NBAYS  |
|    | and/or                                    |        |
| 8B | PLWT4(I), PLLM4(I), PLVM4(I)              |        |
| 9  | MPCL(K)                                   | MPLAN  |

## APPENDIX D

COMCL Program Listings1. Program DATPREP

```

PROGRAM DATPREP      73/172  OPT=0  TPACE      ETN 4.8+52A

1      PPROGRAM DATPREP(RUNDAT,CTRAW,OUTPUT,CTMED,CTOAT,CTFIL,
2      TAPE1=RUNDAT,TAPE2=CTRAW,TAPE3=CTMED,TAPE4=CTOAT,
3      TAPE6=OUTPUT,TAPE7=CTFIL)
C
5      C---THIS PROGRAM TAKES CTRAW FILE TO SORT IT AND COMPILE TALLYS
C---FROM THE SORTED FILE CTMED. OPERATOR RECAPITULATION SHEET AND
C---TONNAGE SUMMARY BY LINE ARE GENERATED FROM THIS. THEN CTRAW
C---IS SORTED AGAIN TO YIELD CTOAT, WHICH IS READ IN TO GENERATE
C---DATA FOR CTFIL. GENERATED DATA ARE PRINTED OUT TO ENABLE
10     C---ACCURACY CHECK.
C-----
      DIMENSION NTALLY(5,6,A),SUMWT(5,6,8),ISROW(32,20),PNAME(5),
1      LNAME(6),NSCA(5,A),NSR(6),NSCR(5),WTPL2(5,6),WTPL4(5,6),
2      WTP2(5),WTP4(5),EMT2(6),EMT4(6),WTGT(6),IPST(500),
15     YPOS(500),LCHC(500),IDC(500),WGT(500),IGCH(32),
4      NIG(32),NIGP(32,20),WTL2(6),WTL4(6),NSTG(32),NTPL2(5,6),
5      NTPL4(5,6),AVWT(32),MPIR(32),TNAME(A),MPC(30),KOMM(12)
      LOGICAL ENDFLG,NEWGP
C
20     C---SORT CTRAW TO YIELD CTMED.
C
      CALL SMSORT(40)
      CALL SMFILE("SORT","FORMATTED",5LCCTRAW,"REWIND")
25     CALL SMKEY(1,1,1,0,"DISPLAY","C90L6","A")
      CALL SMKEY(12,1,1,0,"DISPLAY","C90L6","A")
      CALL SMKEY(13,1,1,0,"DISPLAY","C90L6","A")
      CALL SMFILE("OUTPUT","FORMATTED",5LCCTMED,"REWIND")
      CALL SMEND
30     C
C---INITIALIZE APRAYS.
C-----
      DO 500 I=1,5
      DO 500 J=1,6
      DO 500 K=1,4
35     NTALLY(I,J,K)=0
500     SUMWT(I,J,K)=0.0
      DO 510 M=1,32
      DO 510 L=1,20
40     NIGR(M,L)=ISROW(M,L)=0
C
C---PEAD CTMED AND GET NUMBER AND WEIGHT TALLYS.
C-----
      NUM=NSEC=IX=0      *SWT=0.0
45     ENDFLG=.FALSE.
10     CONTINUE
C
      READ(3,3000)IPORT,ISTC,ISEC,LINE,WT
      FCFMAT(I1,I2,I3,IX,I1,4X,F4.1)
50     IF (EOF(3).NE.0.) ENDFLG=.TRUE.
      IF (ENDFLG)GOTO 30
      NUM=NUM+1
      IF (NUM.EQ.1) GO TO 35
      IF ((IPORT.NE.IP).OR.(ISTC.NE.IS).OR.(LINE.NE.LN)) 30,40
55     30     NTALLY(IP,LN,IS)=NUMG
      SUMWT(IP,LN,IS)=SWT
35     SWT=0.0      *NUMG=0
      IF (ENDFLG) GO TO 50
40     SWT=SWT+WT
60     45     NUMG=NUMG+1
      IP=IPORT      *IS=ISTC      *LN=LINE
      GO TO 10
C
65     50     CONTINUE
C
C---READ IN NUMBER OF PORTS AND NUMBER OF LINES.
C-----
      READ(1,3100)IP,LN,NDATE,NVCY,SHIPIN
70     FCFMAT(I2,I7,I3,IX,410)
C
C---COMPUTE MONTH, DATE, AND YEAR COMPONENTS OF NDATE.
C-----
75     MONTH=NDATE/10000
      NUM1=MOD(NDATE,10000)
      NDT=NUM1/100
      NYR=MOD(NUM1,100)

```

PROGRAM DATPREP 73/172 OPT=0 TRACE

FTN 4.8+528

```

C
C---READ IN PORT NAMES, LINE NAMES, TYPE NAMES, AND COMMODITY TYPES.
C-----
80  READ(1,3110) (PNAME(I),I=1,IP)
    READ(1,3110) (LNAME(I),I=1,LN)
    3110 FORMAT(8A10)
85  TNAME(1)=3H20F  TNAME(2)=3H20P  TNAME(3)=3H20S
    TNAME(4)=3H20M  TNAME(5)=3H40F  TNAME(6)=3H40P
    TNAME(7)=3H40S  TNAME(8)=3H40M
    KOMM(1)=3H0RY  KOMM(2)=3H0FR  KOMM(3)=4H0PL  KOMM(4)=2HOT
    KOMM(5)=3H0PM  KOMM(6)=4H0PR  KOMM(7)=4H0LH  KOMM(8)=3H0XI
90  KOMM(9)=2HFR  KOMM(10)=4H0LK  KOMM(11)=4H0HDE
    KOMM(12)=5H0THER
C
C---COMPUTE SUBTOTALS AND GRAND TOTAL OF CONTAINER COUNTS.
C-----
95  DO 70 J=1,LN
    NSUM=0
    DO 60 I=1,IP
    DO 50 K=1,A
    NSUM=0
100  DO 55 JJ=1,LN
    NSUM=NSUM+NTALLY(I,JJ,K)
    NSCA(I,K)=NSUM
60  NSUM=NSUM+NTALLY(I,J,K)
70  NSR(J)=NSUM
C
105  NSUM=0
    DO 80 I=1,IP
    NSUM=0
    DO 75 K=1,A
    NSUM=NSUM+NSCA(I,K)
110  NSCB(I)=NSUM
    NSUM=NSUM+NSCB(I)
80  NCTLO=NSUM
C
C---PRINT OUT OPERATOR RECAPITULATION SHEET.
C-----
115  PRINT 3200
    FORMAT(1H1,///,35X,"***OPERATOR RECAPITULATION***",//)
    PRINT 3210, (PNAME(I),I=1,IP)
120  FORMAT(5X,4(14X,A10),9X,"TOTALS",/)
    PRINT 3220
    FORMAT(16X,4(" 2F", " 2R", " 2S", " 2M", " 4F", " 4R", " 4S", " 4M",2X),/)
    PRINT 3215
    FORMAT("LINE")
    DO 90 J=1,LN
125  PRINT 3230, J, ((NTALLY(I,J,K),K=1,A),I=1,IP),NSR(J)
    FORMAT(2X,I2,2X,4(13,2X),2X,I2,2X,I6)
    PRINT 3240, ((NSCA(I,K),K=1,A),I=1,IP)
    FORMAT(/,"SUBT",2X,4(13,2X),/)
130  PRINT 3250, (NSCB(I),I=1,IP),NCTLO
    FORMAT(/,"TOTALS",4(11X,6X),7X,I5,///)
C
C---COMPUTE TONNAGE TALLYS.
C-----
135  DO 120 I=1,IP
    SUM1=SUM2=0.0
    DO 110 J=1,LN
    SUM=0.0
    HUN=0
    DO 100 K=1,A
140  HUN=HUN+NTALLY(I,J,K)
    SUM=SUM+SUMNT(I,J,K)
    100  NTPL2(I,J)=SUM
    NTPL2(I,J)=HUN
    SUM=0.0
    HUN=0
145  C
    DO 105 K=5,7
    HUN=HUN+NTALLY(I,J,K)
150  SUM=SUM+SUMNT(I,J,K)
    105  NTPL4(I,J)=HUN
    NTPL4(I,J)=SUM
    SUM1=SUM1+NTPL2(I,J)
    SUM2=SUM2+NTPL4(I,J)
110  CONTINUE

```

PROGRAM DATPREP 73/172 OPT=0 TRACE

FTN 4.8+528

```

155      WTP2(I)=SUM1
      WTP4(I)=SUM2
120      CONTINUE
      SUN1=SUN2=0.0
      NUN1=NUN2=0
160      DO 130 J=1, LN
      SUM1=SUM2=0.0
      NUM1=NUM2=0
      DO 125 I=1, IP
165      SUM1=SUM1+WTP2(I,J)
      NUM1=NUM1+NTPL2(I,J)
      NUM2=NUM2+NTPL4(I,J)
125      SUM2=SUM2+WTP4(I,J)
      WTL2(J)=SUM1
      WTL4(J)=SUM2
170      NUN1=NUN1+NUM1
      NUN2=NUN2+NUM2
      SUN1=SUN1+WTL2(J)
      SUN2=SUN2+WTL4(J)
175      130      CONTINUE
      NT2=NUN1
      NT4=NUN2
      WT2=SUN1
      WT4=SUN2

C
180      C---COMPUTE AVERAGE HEIGHTS OF 20 FT AND 40 FT LOADED CONTAINERS.
      C-----
      AV2=WT2/FLOAT(NT2)
      AV4=WT4/FLOAT(NT4)

C
185      C---COMPUTE EMPTY TOTAL HEIGHTS.
      C-----
      SUN1=SUN2=0.0
      DO 140 J=1, LN
      SUM1=SUM2=0.0
190      DO 135 I=1, IP
      SUM1=SUM1+SUMHT(I,J,4)
      SUM2=SUM2+SUMHT(I,J,4)
135      EMT2(J)=SUM1
      EMT4(J)=SUM2
195      SUN1=SUN1+EMT2(J)
      SUN2=SUN2+EMT4(J)
140      CONTINUE
      EMTY2=SUN1
      EMTY4=SUN2

200      C
      C---COMPUTE GRAND TOTAL WEIGHT.
      C-----
      SUM=0.0
      DO 150 J=1, LN
205      WGT(J)=WTL2(J)+WTL4(J)+EMT2(J)+EMT4(J)
      SUM=SUM+WGT(J)
150      WTLO=SUM

C
210      C---PRINT OUT TONNAGE SUMMARY BY LINE.
      C-----
      PRINT 3300
      FORMAT(11H1,///,35X,"*****TONNAGE BY LINE*****",/)
      PRINT 3301,SHIPID,NVOY,MONTH,NOT,NYP
215      3301      FORMAT(//,2X,"SHIPID:",A10,20X,"VOYAGE NO:",I3,20X,"DATE:",
      1 I2,"/",I2,"/",I2)
      PRINT 3302,AV2,AV4
      FORMAT(//,2X,"20FT AVERAGE WEIGHT:",F5.1,20X,
      1 "40FT AVERAGE WEIGHT:",F5.1)
220      PRINT 3310
      FORMAT(//,10X,"***20 FT CONTAINERS***")
      PRINT 3320,(LNAME(J),J=1,LN)
      3320      FORMAT(2X,"PORT",11X,6(2X,A10),4X,"TOTALS",/)
      DO 200 I=1, IP
      PRINT 3330,(NAME(I),(WTP2(I,J),J=1,LN),WTP2(I))
225      200      CONTINUE
      3330      FORMAT(2X,10X,7F13.1,5X,F7.1)
      PRINT 3340,(WTL2(J),J=1,LN),WT2
      3340      FORMAT(2X,"20" LOAD",2X,7F13.1,5X,F7.1)
      PRINT 3350
      3350      FORMAT(//,10X,"***40 FT CONTAINERS***")
      DO 210 I=1, IP

```

PROGRAM DATPREP 73/172 OPT=0 TPACE

FTN 4.8+528

```

210 PRINT 3330.PNAME(I),(WTL4(I,J),J=1,LN),WTP4(I)
CONTINUE
235 3360 (WTL4(J),J=1,LN),WTL4
FORMAT(2X,"40° LOAD",2X,F13.1,5X,F7.1)
PRINT 3370,(WTL2(J),J=1,LN),WTL2
3370 FORMAT(2X,"20° EMPTY",1X,F13.1,5X,F7.1)
PRINT 3380,(WTL4(J),J=1,LN),WTL4
240 3390 (WTL2(J),J=1,LN),WTL2
PRINT 3390,(WTL2(J),J=1,LN),WTL2
3390 FORMAT(2X,"40° EMPTY",1X,F13.1,5X,F7.1)
C
C---SORT CTMD BY POST, BY SIZE-TYPE CODE, AND BY YARD POSITION.
C---THE SORTED FILE IS CTDAT.
245 C-----
REWIND 3
CALL SMSCOT(40)
CALL SMFILE("SORT","FORMATTED",5LCCTMD)
250 CALL SMKEY(1,1,1,0,"DISPLAY","COBOL6","A")
CALL SMKEY(3,1,1,0,"DISPLAY","COBOL6","A")
CALL SMKEY(5,1,4,0,"DISPLAY","COBOL6","A")
CALL SMKEY(7,1,1,0,"DISPLAY","COBOL6","A")
CALL SMKEY(10,1,1,0,"DISPLAY","COBOL6","D")
255 CALL SMFILE("OUTPUT","FORMATTED",5LCCTDAT,"REWIND")
CALL SMEND
C
C---READ IN CTDAT AND COMPUTE NUMBER TALLYS.
C-----
260 NUM=MG=0 3NSTG(1)=NUMG1=1
MPLAN=0
SUMG1=0.0
KROW=0
ENDFLG=.FALSE.
NEWGP=.FALSE.
265 220 CONTINUE
C
4000 READ(4,4000) IPST,ISTC,YPOST,LINE,ICOMM,WT,IHC,ID,KOP
FORMAT(I1,I2,1X,A6,I2,1X,I2,F5.1,1X,I1,1X,A10,2X,I1)
270 IF(EOF(4).NE.0.) ENDFLG=.TRUE.
IF(ENDFLG) GOTO 242
NUM=NUM+1
IPST(NUM)=IPST+10+ISTC
LCHG(NUM)=LINE+1000+ICOMM+10+IHC
275 YPOS(NUM)=YPOST
IDC(NUM)=ID
WGT(NUM)=WT
IF(KOP.NE.1) GOTO 225
MPLAN=MPLAN+1
MPC(MPLAN)=NUM
280 225 DECODE(6,4010,YPOST) JSPCW,XCT
4010 FORMAT(I4,F2)
IF(NUM.GT.1) GOTO 230
SUMG1=WT
285 GOTO 265
C
230 IF(IPST(NUM).EQ.JPST) GOTO 260
242 MG=MG+1
NEWGP=.TRUE.
NIG(MG)=NUMG1
290 NRIG(MG)=KROW+1
AVWT(MG)=SUMG1/FLOAT(NUMG1)
IGCH(MG)=JPST
NSTG(MG+1)=NSTG(MG)+NUMG1
NUMG1=J
SUMG1=0.0
295 C
250 NUMG1=NUMG1+1
SUMG1=SUMG1+WT
IF((JSPCW.NE.JJROW).OR.NEWGP) 262,270
300 262 IF(NEWGP) 263,264
263 MGC=MG
GOTO 268
264 MGC=MG+1
305 268 KROW=KROW+1
JSPCW(MGC,KROW)=JSPCW
NIG(MGC,KROW)=NUMG1
IF(ENDFLG) GOTO 240
IF(NEWGP) NEWGP=.FALSE.

```

PROGRAM DATPREP 73/172 OPT=0 TRACE

FTN 4.8+528

```

310      265      IF (IPST(NUM).NE.JPST) KPOW=0
          270      NUMG2=0
          NUMG2=NUMG2+1
          IF (ENDFLG) GOTO 280
          JJROW=JSROW
          JSTC=JSTC $JPST=IPST(NUM)
315      C
          C---CHECK DATA INPUT ACCUPACY.
          C-----
          IF (WT.LT.50.0) GOTO 275
          PRINT 4510,ID,WT
320      4510      FORMAT(//,"WEIGHT OF CONTAINER",2X,A10,2X,"EXCEEDS 50 K-TONS.",
          1 //,"ENTERED VALUE=",F12.2)
          STOP
          IF (JSPOW.GE.4101.AND.JSPOW.LE.5773) GOTO 220
          PRINT 4520,ID,YPOST
325      4520      FORMAT("WPONG YARD POSITION ENTERED FOR CONTAINER",2X,A10,2X,/,
          1 "ENTERED VALUE=",2X,A10)
          STOP

C
330      280      CONTINUE
          C
          C---PRINT OUT CONTAINER AND GROUP DATA
          C-----
335      PRINT 4021
          4021      FORMAT(1H1,///,30X,"**CONTAINER GROUP DATA**",/)
          PRINT 4030,NCTLD,WTLD
          4030      FORMAT(10X,"TOTAL NO. TO LOAD",I4,20X,"TOTAL WEIGHT(KT)",F6.1)
340      4040      PRINT 4040
          4040      FORMAT(//,5X,"SEQ NO.",3X,"IPST",3X,"YPOS",5X,"LCHC",5X,
          1 "IDC",10X,"WGT")
          PRINT 4041
          4041      FORMAT(5X,"*****",3X,"*****",3X,"*****",5X,"*****",5X,"*****",
          1 10X,"*****")
345      DO 300 M=1,MG
          JP=IGCH(M)/10
          JG=MOD(IGCH(M),10)
          PRINT 4050,M,PNAME(JP),TNAME(JG),AVHT(M)
          K1=NSTG(M) $K2=K1+NRIG(M)-1
350      DO 290 K=K1,K2
          PRINT 4050,K,IPST(K),YPOS(K),LCHC(K),IDC(K),WGT(K)
          4050      FORMAT(5X,I5,3X,I5,3X,A6,5X,I4,3X,A10,3X,F5.1)
          4060      FORMAT(//,2X,"GP NO",I3,2X,"PORT",A10,"TYPE",A10,
          1 2X,"AV. WT",F5.1,/)
355      300      CONTINUE
          C
          C---PRINT OUT MANUAL PLAN CONTAINER LIST.
          C-----
360      IF (MPLAN.EQ.0) GOTO 309
          PRINT 4111
          4111      FORMAT(///,30X,"**LIST OF MANUAL PLAN CONTAINERS**",/)
          PRINT 4090,(MPC(K),K=1,MPLAN)
          C
365      309      PRINT 4200
          4200      FORMAT(1H1,///,30X,"**DISTRIBUTION OF CONTAINERS IN YARD**",
          1 //)
          PRINT 4220,(N,N=1,20)
          4220      FORMAT("REP NO",3X,20I5)
          DO 310 I=1,MG
          NN=NRIG(I)
370      PRINT 4230,I,(JSPOW(I,N),N=1,NN)
          PRINT 4240,(NIG(I,N),N=1,NN)
          4240      FORMAT(11X,20I5)
          4230      FORMAT(I5,6X,20I5)
375      C
          C---WRITE CTREL.
          C-----
          WRITE (7,4065) NCTLD,MG,MPLAN,WTLD
          4065      FORMAT(15,F10.1)
          DO 400 I=1,NCTLD
          400      WRITE (7,4070) IPST(I),YPOS(I),LCHC(I),IDC(I),WGT(I)
          4070      FORMAT(I7,A6,I4,A10,F4.1)
          DO 410 M=1,MG
          WRITE (7,4080) IGCH(M),NTG(M),NSTG(M),NRIG(M),AVHT(M)
385      4080      FORMAT(4F4,F5.1)

```

PROGRAM DATPREP 73/172 OPT=0 TRACE

FIN 4.8+52A

```

      KR=NRIG(M)
      WRITE(7,4090) (ISPOW(M,N),N=1,KR)
      WRITE(7,4090) (NIGP(M,N),N=1,KP)
390   4090  FORMAT(20I4)
      410  CONTINUE
      C
      WRITE(7,3100) IP,LM,VDATE,NVOY,SHIPID
      WRITE(7,3110) (PNAME(I),I=1,IP)
      WRITE(7,3110) (LNAME(I),I=1,LM)
395   3110  (KOMM(I),I=1,12)
      WRITE(7,5155) (MPC(K),K=1,MPLAN)
      IF(MPLAN.EQ.0) STOP
5155   5155  FORMAT(12A5)
      400  STOP
      END

```

2. Program CLP

```

PROGRAM CLP      73/172  OPT=0 TRACE  P40MP      FTV 4.8+528

1      C      CLP
C      *BASE=100.
C      *BLANK
C      *COLL
5      C      *SIFT
C      *CC4M
C      *INDENT=3.
C      *IDIN=2.
C      *ROUT
10     C      *RIGHT
C      *STAT=10.

      PROGRAM CLP (YLOUT=65,CTFIL,GPLAN,FSHIP=512,INPUT=65,OUTPUT,TAPE1=
      1 YLOUT,TAPE2=CTFIL,TAPE3=GPLAN,TAPE4=FSHIP,TAPE5=OUTPUT,DEBUG=
      2 COUTPUT)
15     COMMON /DCOM/ TYSEC(20),IAYS(20,20),JBROW(2),NER1,NER2,NIGP(32,20)
      1,NYSEC
      COMMON /DACOM/ ISROW(32,20),JGTP(70),MDETA(70,6),NPICK(70),NOPT(70)
      1,NP1G(32),K9L(70),KOPE
20     COMMON /FCOM/ EPL,CNT,C9L,CFL,COWT,COVM,COLM,CISP,DRAFT,FSMT,GMMAX
      1,G4MIN,G4,HKM,IRNO(40),MTIR(15),MCOLH(15),MCOLD(15),NBAYS,NVOY,
      2,NDATE,TWT,TVM,IL4,PLWT2(40),PLWT4(40),PLM2(40),PLL4(40),PLVM2(40)
      3,PLVM4(40),TFIN,IOSHIP,TKWT,TKVM,TKLM,TMLS,TMAX,TRPIN,VPLS,WPLS,
      4,WTL0
25     COMMON /FSCOM/ CGL2(40),CGL4(40),KOLL(25,12),KOIN(25,12),NCBL(25),
      1,MTIRH(15),NHC(25,10),VCG20(15,4),VCG40(15,4),VCG2H(15,10),VCG4H(15
      2,10),WTOO(25,12),PMCH(40,12),DEKL2(40),DEKL4(40),LOVE,NVIOL,JVIOL
      3(25),LRAD(25),ACVM1,ACVM2,ACLM1,ACLM2,ACV4,ACLM,ACUSE(25,10),WCECK
      4(25,10)
30     COMMON /AFCOM/ LLOC(500),NCTLD,NSTG(32),NPCL(25),MPC(25),MPLAN,
      1,KEY(40),NRDAY(32),NLCA0(32,30)
      COMMON /ALCOM/ LPAT(70,15),NTG9(70),IGCH(32),JYPOS(500)
      COMMON /FLCOM/ IPST(500),WGT(500)
      COMMON /LCOM/ JSC(500),LNAME(10),IPORT(5),KPRIAT,KOMM(12)
35     COMMON /GCOM/ TRAYL(25),NRAYL,NSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
      1,JHOLD(40),MGPCUP,KTRAN(70)
      DIMENSION NTLCA0(32),KUM4(70),KTEM(15)

      ---THIS IS THE MAIN PROGRAM FOR COMPUTRIZED LCAO PLANNING MODEL F
      ---CONTAINER SHIPS.
40     -----
      ---INITIALIZE ARRAYS.
      -----
45     DO 110 I=1,40
      PLVM4(I) = 0.0
      PLVM2(I) = PLVM4(I)
      PLLM4(I) = PLVM2(I)
      PLLM2(I) = PLLM4(I)
      PLWT4(I) = PLLM2(I)
50     PLWT2(I) = PLWT4(I)
      110 CONTINUE
      DO 120 K=1,15
      KTEM(K) = 0
55     120 CONTINUE

      ---READ IN FILE YLOUT.
      -----
60     READ (1,*) NYSEC,NER1,NER2
      READ (1,*) (IYSEC(I),I=1,NYSEC)
      DO 130 I=1,NYSEC
      READ (1,*) (IAYS(I,J),J=1,NYSEC)
65     130 CONTINUE

      ---READ IN FILE CTFIL, GENERATED BY PROGRAM DATPREP FROM CTAW.
      -----
70     READ (2,*) NCTLD,NGROUP,MPLAN,WTL0
      DO 140 I=1,NCTLD
      READ (2,*) IPS,JYPOS(I),LCH,IOC(I),WGT(I)
      IPST(I) = IPS*10000+LCH
75     140 CONTINUE
      DO 150 I=1,NGROUP
      READ (2,*) IGCH(I),NIG(I),NSTG(I),NRIG(I),GMEAN(I)
      NR = NRIG(I)
      READ (2,*) (ISPOW(I,N),N=1,NR)
      READ (2,*) (NIGR(I,N),N=1,NR)
80     150 CONTINUE
      READ (2,*) IF,LN,NDATE,NVOY,IOSHIP

```



PROGRAM CLP

73/172 CPT=0 TRACE PHOMP

FTN 4.8+52A

```

80      READ (2,880) (JPORT(I),I=1,IP)
      READ (2,890) (LNAME(I),I=1,LN)
      READ (2,890) (NOMM(I),I=1,12)
      IF (NPLAN.EQ.0) GO TO 160
      READ (2,*) (MPC(K),K=1,NPLAN)

C
C
C      ---READ IN FILE GPLAN.
C      ---READ HYDROSTATIC DATA AND RAY LOAD PROFILES.
160     READ (3,*) DISP,DRAFT,HKM,CMT,CBL,CFL
      READ (3,*) TKWT,TKVM,TKLM,FSMT
      READ (3,*) COWT,COWM,COLM
      READ (3,*) NRAYL,JRROW(1),JRROW(2)
      DO 170 I=1,NRGROUP
        NRRAY(II) = 0
        NTLOAD(II) = 0
170     CONTINUE
      DO 260 I=1,NBAYL
        DO 180 II=1,MGROUP
          NLOAD(III,I) = 0
180     CONTINUE
      READ (3,*) JRAYL(II),LOCA,NCRL(II)
      IF (LOCA.EQ.1.OR.LCCA.EQ.2) GO TO 190
      GO TO 200
190     IF (NCRL(II).GT.0.AND.NCRL(II).LE.12) GO TO 210
200     PRINT 630, JRAYL(II),I
      STOP
210     KK = NCRL(II)
      READ (3,*) (KOLL(I,K),K=1,KK)
      DO 220 K=1,KK
        IF (KOLL(I,K).LE.15.AND.KOLL(I,K).GT.0) GO TO 220
        PRINT 640, K,JRAYL(II)
      STOP
220     CONTINUE
      IF (LOCA.EQ.1) GO TO 240
      READ (3,*) (NCUBE(I,K),K=1,KK)
      DO 230 K=1,KK
        IF (NCUBE(I,K).GT.0.AND.NCUBE(I,K).LE.10) GO TO 230
        PRINT 650, JRAYL(II),KOLL(I,K),NCUBE(I,K)
      STOP
230     CONTINUE
      GO TO 260
240     READ (3,*) (WDECK(I,K),K=1,KK)
      DO 250 K=1,KK
        IF (WDECK(I,K).GT.0.AND.WDECK(I,K).LT.100.) GO TO 250
        PRINT 660, JRAYL(II),KOLL(I,K),WDECK(I,K)
      STOP
250     CONTINUE
260     CONTINUE

C
C
C      ---READ IN LPAT'S FOR EACH STAGE OF LOADING.
130     READ (3,*) NSTAGE1,NSTAGE2,KOPE
      NSTAGE = NSTAGE1+NSTAGE2
      DO 380 KS=1,NSTAGE
        IF (KS.LE.NSTAGE1) KTR = 1
        IF (KS.GT.NSTAGE1) KTR = 2
135     READ (3,*) KDESIG,KUME
      IBR = KDESIG/10000
      NTT = MOD(KDESIG,10000)
      JGG = NTT/100
      NPAT = 400*(NTT,100)
      KBL(KS) = IF3
      JGTP(KS) = JGG
      IF (JGG.GT.1.AND.JGG.LE.MGROUP) GO TO 280
270     PRINT 670, KS,JGG
      STOP
145     280     READ (3,*) (KTEM(K),K=1,NPAT)
      NUM = 0
      DO 300 K=1,NPAT
        IF (KTEM(K).GE.10101.AND.KTEM(K).LT.110000) GO TO 290
150     PRINT 680, KS,K
      STOP
      NUM = NUM+MOD(KTEM(K),100)
290     CONTINUE
      NPICK(KS) = NUM
      DO 310 IB=1,NRAYL

```

```

PROGRAM CLP          73/172  CPT=0 TRACE  PHOMP          FTN 4.8+528

155          IF (IIR.EQ.IBAYL(I)) GO TO 320
310          CONTINUE
320          NLOAD(JGG,IE) = NUM
160          NRAY(JGG) = NRAY(JGG)+1
          NTLOAD(JGG) = NTLOAD(JGG)+NUM
          CCG
          ---SORT AND PREPARE DATA FOR SUBROUTINE DYNAMIC RUN.
          NTGB(KS) = KDESIG
165          KUMM(KS) = KUME
          KTRAN(KS) = KTR
          DO 330 K=1,15
            LPAT(KS,K) = KTEM(K)
170          330 CONTINUE
          IF (KOPE.EQ.1) GO TO 380
          IF (KS.EQ.1) GO TO 340
          KSS = KS-1
          DO 370 LS=1,KSS
175          IF (KUME.GE.KUMM(LS)) GO TO 370
          DO 350 MS=LS,KSS
            KSS = KSS+LS-MS
            NTGB(MSS+1) = NTGB(MSS)
            KBL(MSS+1) = KBL(MSS)
            JGTP(MSS+1) = JGTP(MSS)
            KUMM(MSS+1) = KUMM(MSS)
            KTRAN(MSS+1) = KTRAN(MSS)
            NPICK(MSS+1) = NPICK(MSS)
            DO 340 LP=1,15
180          LPAT(MSS+1,LP) = LPAT(MSS,LP)
185          340 CONTINUE
          350 CONTINUE
          NTGB(LS) = KDESIG
          KUMM(LS) = KUME
          KBL(LS) = IB8
190          JGTP(LS) = JGG
          KTRAN(LS) = KTR
          NPICK(LS) = NUM
          DO 360 LP=1,15
            LPAT(LS,LP) = KTEM(LP)
195          360 CONTINUE
          GO TO 340
          370 CONTINUE
          380 CONTINUE
          CCG
          ---READ IN FILE FSHIP, THE FIXED SHIP STRUCTURE FILE.
          READ (4,*) NBAYS,LSUP,LOSYS,NRSEC
          DO 390 K=1,NRSEC
200          READ (4,*) MCOLD(K),MCOLH(K),MTIRO(K),MTIRH(K)
205          390 CONTINUE
          READ (4,*) WTLS,VNLS,TNLS,GMMAX,GMMIN,TRMAX,TRPIN,BPL
          DO 450 I=1,NBAYS
            READ (4,*) IBNO(I),K9KEY(I),JHOLD(I)
            KEY = K9KEY(I)
            K1 = MOD(KEY,10000)
            KOL = K1/100
            NTR = MOD(K1,100)
            K2 = KEY/100000000
            LOK = K2/10
            IT = MOD(K2,10)
            IF (LOK.EQ.2) GO TO 400
            READ (4,*) CEKL2(I),DEKL4(I)
            READ (4,*) CGL2(I),CGL4(I)
            IF (LOK.EQ.2) GO TO 420
            DO 410 K=1,KOL
210          NCEL(I,K) = NTR
            410 CONTINUE
            IF (LOK.NE.2) GO TO 430
            READ (4,*) INCEL(I,K),K=1,KOL
            READ (4,*) IMHCH(I,K),K=1,KOL
215          420
220          430
225          CCG
          ---READ IN THE LAST PART OF FILE GPLAN.
          IF (IT.EQ.2) GO TO 440
          READ (3,*) PLWT2(I),PLLM2(I),PLV42(I)
230          IF (IT.EQ.1) GO TO 450

```

PROGRAM CLP

73/172 CPT=0 TRACE PHOMP

FTN 4.8+529

```

235 440 READ (3,*) FLMT4(I),PLLM4(I),PLVM4(I)
450 CONTINUE
IF (MPLAN.EQ.0) GO TO 460
READ (3,*) (MPCL(K),K=1,MPLAN)
C
C ---FINISH READING IN FSHIP.
C
240 460 DO 480 M=1,NBSEC
MTD = MTIRD(M)
MTH = MTIRF(M)
IF (MTD.EQ.0) GO TO 470
READ (4,*) (VCG2D(M,L),L=1,MTD)
245 470 READ (4,*) (VCG4D(M,L),L=1,MTD)
IF (MTH.EQ.0) GO TO 480
READ (4,*) (VCG2H(M,L),L=1,MTH)
READ (4,*) (VCG4H(M,L),L=1,MTH)
480 CONTINUE
C
250 C ---PRINT OUT LOAD PLAN OF GROUPS BY BAY AND CHECK DATA ACCURACY.
C
PRINT 690
PRINT 700
PRINT 710
255 PRINT 720, (IBAYL(I),I=1,NBAYL)
NUM = 0
DO 490 JG=1,MGROUP
NUM = NUM+NTLOAD(JG)
PRINT 730, JG,NTLOAD(JG),NIG(JG),INLOAD(JG,K),K=1,NPAYL)
260 490 CONTINUE
NCTLP = NUM+MPLAN
PRINT 740, NCTLP,NCTLD
DO 500 JG=1,MGROUP
IF (NTLOAD(JG).LE.NIG(JG)) GO TO 500
265 PRINT 750, JG,NIG(JG),NTLOAD(JG)
STOP
500 CONTINUE
C
C ---CALL SUBROUTINE DYNAMIC
C
C
C ---CALL SUBROUTINE ASSCN.
C
275 CALL ASSCN
KPRINT = 1
CALL LPOUT
C
C ---CONDUCT FEASIBILITY CHECKS.
C
CALL FCHECK
IF (INVOL.EQ.0) GO TO 530
C
C ---CALL SUBROUTINE REMEDY IF THE OPERATOR DESIRES.
C
285 510 PRINT 760
READ 770, KGO
IF (KGO.EQ.1HA) GO TO 530
IF (KGC.NE.1HY) GO TO 510
CALL REMEDY
290 520 CALL FCHECK
C
C ---PRINT OUT THE LOAD PLAN IF DESIRED.
C
295 530 PRINT 790
READ 770, MGO
IF (MGO.EQ.1HA) GO TO 540
IF (MGO.NE.1HY) GO TO 530
KPRINT = 0
GO TO 550
300 540 KPRINT = 1
550 CALL LPOUT
IF (MGO.EQ.1HA) STOP
C
C ---INTERACTIVE ADJUSTMENT ROUTINE.
C
305 560 PRINT 790
READ 770, KGO

```

PROGRAM CLP

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

310      IF (MGO.EQ.1HN) GO TO 620
        IF (MGO.NE.1HY) GO TO 560
570     PRINT 590
        READ *,ID1,ID2
        DO 580 IG=1,MGROUP
315      IF (ID1.LT.NSTG(IG)) GO TO 590
580     CONTINUE
        GO TO 630
590     IF (ID2.GE.NSTG(IG-1).AND.ID2.LT.NSTG(IG)) GO TO 610
600     PRINT 510
        GO TO 570
320     610    LLL = LLOC(ID1)
        LLOC(ID1) = LLOC(ID2)
        LLOC(ID2) = LLL
        PRINT 820
        READ 770, MGO
325     IF (MGO.EQ.1HY) GO TO 570
        IF (MGO.EQ.1HN) GO TO 520
        GO TO 560
620     CONTINUE
        STOP
330     CCCCCC
        ---FORMAT STATEMENTS.
        -----
335     630    FORMAT (///,44+ERROR IN FIRST RECORD OF GPLAN DATA FOR BAY,I5,3X,
        19HOF ORDER=,I3)
        640    FORMAT (///32HWRONG DATA IN LOAD COLUMN NUMBER,I5,2X,3H#AY,I5)
        650    FORMAT (///,46HWRONG DATA ENTERED FOR NO OF HIGH CUBES IN BAY,I5,3X
340     1,7HCOLUMN=,I5,/,14HENTERED VALUE=,I7)
        660    FORMAT (///,46HWRONG DATA ENTERED FOR WEIGHT ON DECK FOR BAY=,I5,3X
        1,7HCOLUMN=,I5,/,14HENTERED VALUE=,F10.2)
        670    FORMAT (///,5X,31HERROR IN GPLAN--NTGB FOR STAGE=,I5,3X,6HGROUP=,I3
        1)
        680    FORMAT (///,5X,20HLPAT ERROR IN STAGE=,I4,3X,3HNO=,I4)
345     690    FORMAT (///,36,15HGROUP LOAD PLAN)
        700    FORMAT (30X,15+*****+)
        710    FORMAT (/,5HGROUP,2X,7HTO LOAD,2X,4HIN GROUP,25X,14HRECEIVING BAYS
        1)
350     720    FORMAT (25X,24I4,/)
        730    FORMAT (I5,2X,I7,2X,I9,4X,24I4)
        740    FORMAT (/,5HTOTAL,2I10)
        750    FORMAT (///,46HTOO MANY CONTAINERS PLANNED TO LOAD FROM GROUP,I7,/,
        120HTOTAL IN THIS GROUP=,I5,5X,22HTOTAL PLANNED TO LOAD=,I5)
355     760    FORMAT (/,75HDO YOU WANT TO IMPROVE THE STABILITY BY EXECUTING
        1 SURROUTINE REPEYD+,/,48HANSWER BY ENTERING EITHER *YES* OR
        2 *NO*.)
        770    FORMAT (A1)
        780    FORMAT (/,44HDO YOU WANT CONTAINER LOAD PLAN PRINTED OUT+,/,
        144HANSWER BY ENTERING EITHER*YES* OR *NO* HERE.)
360     790    FORMAT (///,65HDO YOU WANT TO MAKE ANY ADJUSTMENT TO THE GENERATED
        1 LOAD PLAN+,/,24HANSWER EITHER *YES* OR *NO*.)
        800    FORMAT (/,113HADJUSTMENTS CAN BE MADE BY ENTERING CELL INDICATORS
        1OF A PAIR OF CONTAINERS WHICH YOU WISH TO HAVE INTERCHANGED.,/,
        24HENTER BELOW ID NUMBERS OF CONTAINERS TO EXCHANGE.,/)
365     810    FORMAT (3X,41HTHESE TWO CANNOT BE EXCHANGED. TRY AGAIN.)
        820    FORMAT (/,7HINTERCHANGE PERFORMED. DO YOU WANT TO CONTINUE+)
        830    FORMAT (3I9,F10.1)
        840    FORMAT (I2,A6,I4,A10,F4.1)
        850    FORMAT (4I4,F10.1)
370     860    FORMAT (20I4)
        870    FORMAT (2I2,I7,I3,1X,A10)
        880    FORMAT (9A10)
        890    FORMAT (12A5)
        ENQ

```

SUBROUTINE DYNAMIC 73/172 CPT=0 TRACE PMOMP

FTN 4.8+529

```

1      C      DP
      C      SUBROUTINE DYNAMIC
5      C      ---THIS SUBROUTINE SOLVES A DYNAMIC PROGRAMMING PROBLEM TO OBTAIN
      C      ---AN OPTIMAL TRANSTAINER ROUTING PLAN IN PICKING A GIVEN NO. OF
      C      ---CONTAINERS OF EACH GROUP IN MSTAGE STAGES FOR A LOADING OPERATI
      C      ---THE OBJECTIVE IS TO MINIMIZE TOTAL TRANSTAINER MOVE DISTANCE.
10     C      -----
      C      COMMON /CCOM/ TYSEC(20),IAYS(20,20),JBROW(2),NER1,NER2,NIGR(32,20)
      C      1,NYSEC
      C      COMMON /DACOM/ ISROW(32,20),JGTP(70),MDETA(70,6),NPICK(70),NOPT(70
15     C      1),NRIG(32),KRL(70),KOPE
      C      COMMON /GCOM/ IRAYL(25),NRAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
      C      1,JHOLD(40),MGRUP,KTRAN(70)
      C      COMMON /CCOM/ KICKS(5),KICKR(20),LTEM(20),MSOLN(410),LES(410),LFV
20     C      1(410),NSEC,MROW,JGG,MOVE(410,6),NTSOL(70),NSOL,IPG(32),NAS(32),KTT
      C      2,JJRR,LAST,ISTATE,LGR(410,17)
      C      DIMENSION KAS(32,15),LUA(20),LOVE(4),NGIS(10),LRA(20),ISCG(4)
      C      1,MGIS(4),KFR(4),KLR(4),NREST(70)
      C      ---PRINT OUT WORK SEQUENCE TABLE.
      C      -----
25     C      PRINT 1120
      C      PRINT 1130
      C      PRINT 1140
      C      DO 10 KS=1,MSTAGE
      C      PRINT 1150,KS,KTRAN(KS),JGTP(KS),NPICK(KS),KRL(KS)
30     C      10 CONTINUE
      C      ---FORM THE WORK SEQUENCE DIRECTORY.
      C      -----
35     C      DO 20 J=1,MGRUP
      C      NAS(J) = 0
      C      20 CONTINUE
      C      DO 30 K=1,MSTAGE
      C      JGP = JGTP(K)
      C      NAS(JGP) = NAS(JGP)+1
      C      L = NAS(JGP)
      C      KAS(JGP,L) = K
40     C      30 CONTINUE
      C      ---INITIALIZE ARRAYS.
      C      -----
45     C      DO 40 J=1,MGRUP
      C      IPG(J) = 0
      C      40 CONTINUE
      C      DO 50 NS=1,350
      C      DO 50 NV=1,6
      C      MOVE(NS,NV) = 0
50     C      50 CONTINUE
      C      ---FOR THE NEXT STAGE:
      C      -----
55     C      NSOL = 0
      C      ISOL = NSOL
      C      DO 1030 K=1,MSTAGE
      C      NPP = NPICK(K)
      C      JGG = JGTP(K)
      C      KTT = KTRAN(K)
      C      MROW = NPP(JGG)
      C      IPG(JGG) = IPG(JGG)+1
      C      IF (K.GT.1) GO TO 60
      C      NFS = 1
      C      NIS = NFS
      C      GO TO 70
65     C      ---FOR THE NEXT INITIAL STATE:
      C      -----
70     C      IF (K.EQ.2) NIS = 1
      C      IF (K.GE.3) NIS = NTSOL(K-2)+1
      C      NFS = NTSOL(K-1)
      C      DO 1020 J=NIS,NFS
      C      ---IDENTIFY SECTION NUMBER AND ROW NUMBER.
      C      -----
75     C      IF (KOPE.GT.1) GO TO 40
      C      IF (K.EQ.1) JR = JBROW(1)

```

SUBROUTINE DYNAMIC 73/172 CPT=0 TRACE PNDMP

FTN 4.8+528

```

80      IF (K.GT.1) JR = LES(J)
      GO TO 90
      IF (K.EQ.1) ISTATE = JBROW(1)*10000+JBROW(2)
      IF (K.GT.1) ISTATE = LES(J)
      IF (KTT.EQ.1) JR = ISTATE/10000
      IF (KTT.EQ.2) JR = MOD(ISTATE,10000)
85      IF (KTT.EQ.1) JJRR = MOD(ISTATE,10000)
      IF (KTT.EQ.2) JJRR = ISTATE/10000
      JROW = MCD(JR,100)
      JSEC = JR/100
      TFLAG = 0
90      DO 100 NS=1,NYSEC
      IF (IYSEC(NS).EQ.JSEC) GO TO 110
      CONTINUE
C 100      ---DESIGNATE PROPER LOCK UP ARRAY, LUA, FOR FIRST TIME LOOK UP.
C      IF (IPG(JGG).GT.1) GO TO 130
      DO 120 M=1,MROW
      LTEM(M) = NIGR(JGG,M)
      LUA(M) = NIGR(JGG,M)
100      CONTINUE
      GO TO 170
C 110      ---TRACE BACK TO PROPER LOOK UP ARRAY GENERATED BY A PREVIOUS STAG
C      *IZZZ = IPG(JGG)-1
      KASS = KAS(JGG,IZZZ)
      KRT = J
      KF = K-KASS-1
      IF (KF.LT.1) GO TO 150
      DO 140 NN=1,KF
      JRT = MSOLN(KRT)/10
      KRT = JRT
110      CONTINUE
      DO 160 M=1,MROW
      LTEM(M) = LGR(KRT,M)
      LUA(M) = LGR(KRT,M)
115      CONTINUE
C 120      ---COUNT HOW MANY SECTIONS CONTAIN ONE OR MORE OF THIS GROUP AND A
C      ---HOW MANY ARE IN EACH CONTAINING SECTION. IDENTIFY THE FIRST AND
C      ---CONTAINING ROWS IN EACH SECTION.
      KOUNT = 0
      IRR = KOUNT
      NUM = IFF
      NS = NUM
      ISS = NS
      JRR = JR
125      DO 200 I=1,MROW
      IF (LUA(I).EQ.0) GO TO 200
      KOUNT = KOUNT+1
      ISR = ISRCW(JGG,I)
      IS = ISR/100
      IF ((I.EQ.1).OR.(KOUNT.EQ.1)) GO TO 190
      IF (IS.EQ.ISS) GO TO 190
      NS = NS+1
      ISCG(NS) = ISS
      NGIS(NS) = NUM
      KFR(NS) = JFR
      KLR(NS) = JLR
      NUM = 0
      JFR = I
      NUM = NUM*LUA(I)
      ISS = IS
      JLR = I
135      CONTINUE
      NSEC = NS+1
      ISCG(NSEC) = ISS
      KFR(NSEC) = JFR
      KLR(NSEC) = JLR
      IF (LUA(MROW).EQ.0) KLR(NSEC) = JLR
      NGIS(NSEC) = NUM
145      CONTINUE
C 150      ---INITIALIZE PICK STATUS OF POSSESSING SECTIONS.
C      DO 210 IS=1,NSEC
      KICKS(IS) = 0
      CONTINUE
210

```

SUBROUTINE DYNAMIC 73/172 CPT=0 TRACE PMOMP

FTN 4.4+525

```

155      DO 220 IF=1,MROW
          KICKR(IR) = 0
          220      CONTINUE
C
C      ---SEE IF THE CURRENT SECTION IS POSSESSING.
160      DO 230 N=1,NSEC
          IF (ISCG(N).EQ.JSEC) GO TO 240
          230      CONTINUE
          GO TO 710
C
C      ---SEE IF CURRENT SECTION IS THE ONLY POSSESSING ONE.
240      IF (NSEC.EQ.1) GO TO 250
          GO TO 430
C
C      ---CURRENT SECTION IS THE SINGLE POSSESSING. FIRST,SEE IF WE ARE
C      ---AN EXTREME ROW.
250      KODE = N
          KOIS = IAYS(MS,MS)
          KFF = KFR(KOCE)
          KLL = KLF(KODE)
          IF (JR.LE.ISROW(JGG,KFF)) GO TO 260
          IF (JR.GE.ISROW(JGG,KLL)) GO TO 270
          GO TO 280
175      260      KX = 1
          IFLAG = 1
          GO TO 290
          270      KX = 0
          IFLAG = 1
          GO TO 290
180      270      IFLAG = 1
          GO TO 290
C
C      ---CURRENT ROW POSITION IS BETWEEN TWO EXTREME CONTAINING ROWS. F
C      ---LEFT MOVE SOLUTION FIRST.
190      280      KX = 1
          IFLAG = 0
          ---INCREMENT SOLUTION NO.
          290      ISOL = ISOL+1
          NMV = 0
          IF (KX) 300,300,360
          C
          C      ---SEE IF THERE IS ENOUGH TO THE RIGHT OF CURRENT ROW.
          300      NSUM = 0
          TAG = 0.0
          NMV = NMV+1
          MOVE(ISOL,NMV) = 2000+KFF*10+KOIS
          DO 310 M=KFF,KLL
              MP = KLL-M+KFF
              IF (ISROW(JGG,MP).GT.JR) GO TO 310
              TAG = TAG+1.0
              IF (TAG.EQ.1.0) MM = MR
              NSUM = NSUM+LUA(MP)
              IF (NSUM.GE.NPP) GO TO 350
              LTEM(M) = 0
          310      CONTINUE
          C
          C      ---NOT ENOUGH TO THE RIGHT OF CURRENT ROW. WE NEED TO TURN TO CPP0
          C      ---DIRECTION.
          NMV = NMV+1
          MOVE(ISOL,NMV) = 3000+(MM+1)*10+KOIS
          NMV = NMV+1
          MOVE(ISOL,NMV) = 1000+KLL*10+KOIS
          DO 320 M=KFF,KLL
              IF (ISROW(JGG,M).LE.JP) GO TO 320
              NSUM = NSUM+LUA(M)
              IF (NSUM.GE.NPP) GO TO 340
              LTEM(M) = 0
          320      CONTINUE
          330      PRINT 1160, JGG,K
          STOP
          C
          C      ---COMPLETE PICKING LEFT OF INITIAL ROW WITH INITIAL MOVE RIGHT.
          340      LAST = ISROW(JGG,M)
          LTEM(M) = NSUM-NPP
          GO TO 410

```

SUBROUTINE DYNAMIC 73/172 CPT=0 TRACE PMDMP

FTN 4.8+528

```

C      ---ENOUGH CONTAINERS ON THE RIGHT SIDE. COMPLETE PICKING THERE.
235 350 LTEM(MR) = NSUM-NPP
      LAST = ISROW(JGG,MR)
      GO TO 410
C      ---INITIAL LEFT MOVE. SEE IF THERE IS ENOUGH TO THE LEFT OF
C      ---CURRENT ROW.
240 360 NSUM = 0
      TAG = 0.0
      NMV = NMV+1
      MOVE(ISOL,NMV) = 1000+KLL*10+KOIS
245 DO 370 MR=KFF,KLL
      IF (ISROW(JGG,MR).LT.JR) GO TO 370
      TAG = TAG+1
      IF (TAG.EQ.1.) MM = MR
      NSUM = NSUM+LUA(MR)
      IF (NSUM.GE.NPP) GO TO 400
250 LTEM(MR) = 0
370 CONTINUE
C      ---NOT ENOUGH TO THE LEFT. TURN AND PICK TO THE RIGHT OF INITIAL
C
255 NMV = NMV+1
      MOVE(ISOL,NMV) = 3000+(MM-1)*10+KOIS
      NMV = NMV+1
      MOVE(ISOL,NMV) = 2000+KFF*10+KOIS
260 DO 380 MR=KFF,KLL
      MRR = KLL-M+KFF
      IF (ISROW(JGG,MRR).GE.JR) GO TO 380
      NSUM = NSUM+LUA(MRR)
      IF (NSUM.GE.NPP) GO TO 390
265 LTEM(MRR) = 0
380 CONTINUE
      GO TO 330
C      ---ENOUGH PICKED TO THE RIGHT OF INITIAL ROW.
C
270 390 LAST = ISROW(JGG,MRR)
      LTEM(MRR) = NSUM-NPP
      GO TO 410
C      ---PICKED ENOUGH TO THE LEFT OF BEGINNING ROW.
C
275 400 LAST = ISROW(JGG,MR)
      LTEM(MR) = NSUM-NPP
C
C 410 CALL DISCOMP (NRT,K,J,ISOL,JSOL,NMV,JR)
C      ---IF POSSIBLE, FORM THE NEXT SOLUTION OF LEFT MOVE FOR THIS ROW P
280 IF ((KX+IFLAG).GT.0) GO TO 1020
      DO 420 M=1,MROW
      LTEM(M) = LUA(M)
285 420 CONTINUE
      KX = 1
      IFLAG = 1
      GO TO 290
C
C      ---WE ARE AT ONE OF THE MULTIPLE POSSESSING SECTIONS. SEE IF WE A
C      ---RIGHT COLUMN SECTION OR LEFT-COLUMN.
290 430 KODE = N
      KFF = KFF(KODE)
      KLL = KLL(KODE)
      KASE = 3
      NMV1 = 0
      NSUM1 = NMV1
      GO TO 720
295 440 IRM = 0
      KFF = KFF(KODE)
      KLL = KLL(KODE)
      IF (JSEC.LT.50) GO TO 510
300 C
C      ---RIGHT COLUMN SECTION.
C
305 IF (JR.GE.ISROW(JGG,KFF)) GO TO 450
      NRM = 1
      KSP = KFF
      GO TO 500
450 IF (JR.LE.ISROW(JGG,KLL)) GO TO 460

```



SUBROUTINE DYNAMIC

73/172 CPT=0 TRACE PMOMP

FTN 4.8+528

```

310      IRM = -1
      KSR = KLL
      GO TO 490
460      DO 470 KKR=KFF,KLL
      IF (ISROW(JGG,KNR).GE.JR) GO TO 480
315      CONTINUE
480      KSP = KKF
490      NRM = KSR-KFF+1
500      KOIR = -1
      GO TO 580

C
C
C      ---LEFT COLUMN SECTION.
C
510      IF (JR.LE.ISROW(JGG,KLL)) GO TO 520
      NRM = 1
      KSR = KLL
325      GO TO 570
520      IF (JR.GE.ISROW(JGG,KFF)) GO TO 530
      IRM = -1
      KSR = KFF
330      GO TO 560
530      DO 540 KKR=KFF,KLL
      IF (JR.GE.ISROW(JGG,KNR)) GO TO 550
540      CONTINUE
550      KSR = KKF
560      NRM = KLL-KSR+1
335      KOIR = 1

C
C
C      ---SOLUTION GENERATION LOOP FOR MULTI-SECTION CASE.
C
580      IF (IRM.GE.0) GO TO 600
      IRM = IRM+1
      KASE = 1
      NSUM1 = 0
      NMV1 = NSUM1
      NSUM = NMV1
345      NMV = NSUM
      DO 590 KKF=1,MROW
      LRA(KKF) = LUA(KR)
590      CONTINUE
      GO TO 720
350      IRM = IRM+1
      KFF = KFF(KODE)
      KLL = KLF(KODE)
      KIR = KSF+KOIR*(IRM-1)
      NRS = KIF-KFF+1
355      IF (KOIR.LT.0) NRS = KLL-KIR+1
      DO 610 IP=1,4
      LOVE(IP) = 0
610      CONTINUE
      NSUM = 0
      NMV = NSUM
      NSUM1 = NMV
      NMV1 = NSUM1
      ISOL = ISOL+1
      NMV = 1
365      LOVE(NMV) = 3000*KIR*10+IAYS(MS,MS)
      NMV = NMV+1
      LOVE(NMV) = 1000*KLL*10+IAYS(MS,MS)
      IF (KOIR.GT.0) LOVE(NMV) = 2000*KFF*10+IAYS(MS,MS)
      DO 620 KKF=1,MROW
      LTEM(KR) = LUA(KR)
370      CONTINUE
620      CONTINUE

C
C
C      ---SEE IF WE HAVE PICKED ENOUGH IN CURRENT SECTION.
C
375      DO 630 NR=1,NRS
      NPP = KIR-KOIR*(NR-1)
      NSUM = NSUM+LUA(NPP)
      IF (NSUM.GE.NPP) GO TO 640
      LTEM(NRR) = 0
      KICKR(NRR) = 1
380      CONTINUE
      GO TO 680
640      LAST = ISROW(JGG,NRR)
      LTEM(NRR) = NSUM-NPP
385      DO 650 NV=1,NMV

```

SUBROUTINE DYNAMIC

73/172 CPT=0 TRACE PHOMP

FTN 4.8+529

```

      MOVE (ISOL,NV) = LOVE(NV)
      CONTINUE
      CALL DISCOMP (NRT,K,J,ISOL,JSOL,NMV,JR)
      DO 650 KR=1,MROW
390      KICKR(KR) = 0
      CONTINUE
      IF (IRM.GE.NRM) GO TO 1020
      DO 670 KR=1,MROW
395      LTEM(KR) = LUA(KR)
      CONTINUE
      GO TO 600
C
C      ---NOT ENOUGH. GO TO THE SECTION MOVE LOOP.
C
400      NSUM1 = ASUM
      NMV1 = NMV
      KASE = 1
      DO 690 KR=1,MROW
      LRA(KR) = LTEM(KR)
405      CONTINUE
      DO 700 IP=1,NMV1
      MOVE (ISOL,IP) = LOVE(IP)
      CONTINUE
410      ISOL = ISOL-1
      GO TO 720
C
C      ---CURRENT SECTION IS NON-POSSESSING. START A SOLUTION BY MOVING
C      ---TO A POSSESSING SECTION.
C
415      NMV1 = 0
      NSUM1 = NMV1
      KASE = 2
C
C      ---SECTION MOVE LOOP.
C
420      DO 1000 JJK=1,NSEC
      JFLAG = 0
      IS = ISCG(JJK)
      IF (IS.EQ.JSEC) GO TO 1000
425      KF1 = KFR(JJK)
      KL1 = KLR(JJK)
      DO 730 KS=1,NSEC
      KICKS(KS) = 0
      CONTINUE
430      ---SEE IF THIS SECTION HAS ENOUGH TO PICK.
      NSUN = 0
      LTYPE = 2
      ISOL = ISCL+1
435      DO 740 KR=KF1,KL1
      NSUN = NSUN+LUA(KR)
      CONTINUE
      IF (NSUN.GE.(NPP-NSUM1)) LTYPE = 1
      IF (LTYPE.EQ.1) GO TO 790
440      ---FIND THE SECOND POSSESSING SECTION.
      JK = 0
      DO 760 JL=1,NSEC
      IF (KICKS(JL).EQ.1) GO TO 760
      IF (JL.EQ.JJK.OR.JL.EQ.JK) GO TO 760
      IF (ISCG(JL))
      IF (ISS.NE.JSEC) GO TO 770
450      CONTINUE
      ISOL = ISOL-1
      GO TO 1000
      770      KF2 = KFR(JL)
      KL2 = KLR(JL)
      KICKS(JL) = 1
      JK = JL
455      ---SEE IF THESE TWO SECTIONS CONTAIN ENOUGH TO PICK.
      NSUN2 = 0
      DO 780 KR=KF2,KL2
      NSUN2 = NSUN2+LUA(KR)
460

```

SUBROUTINE DYNAMIC

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

465      780      CONTINUE
          NSOM = NSUM+NSUN2+NSUM1
          IF (NSOM.GE.NPP) GO TO 790
          GO TO 750
C
C      ---DETERMINE RELATIVE SECTION DISPOSITIONS.
470      790      DO 800 NS=1,NYSEC
          IF (IYSEC(NS).EQ.IS) GO TO 810
800      CONTINUE
810      KDIS1 = IAYS(NS,NS)
475      IF (LTYPE.EQ.1) GO TO 840
          DO 820 LS=1,NYSEC
          IF (IYSEC(LS).EQ.ISS) GO TO 830
820      CONTINUE
830      KDIS2 = IAYS(NS,LS)
C
C      ---INITIAL SECTION CHANGE MOVE.
840      KFF = KF1
          KLL = KL1
485      KDIS = KDIS1
          NSUM = NSUM1
          NMV = NMV1
C
C      ---GENERATE AN UPWARD PICK MOVE.
490      850      NMV = NMV+1
          MOVE(ISOL,NMV) = 4000*KFF*10+KDIS
          NMV = NMV+1
          MOVE(ISOL,NMV) = 1000+KLL*10+IAYS(NS,NS)
495      DO 860 M=KFF,KLL
          NSUM = NSUM+LUA(M)
          IF (NSUM.GE.NPP) GO TO 870
          LTEM(M) = 0
860      CONTINUE
          GO TO 910
          LTEM(M) = NSUM-NPP
          LAST = ISROW(JGG,M)
          IF (K.NE.6) GO TO 910
          GO TO 910
500
505      C
          C      ---GENERATE A DOWNWARD PICK MOVE.
          C
510      880      NMV = NMV+1
          MOVE(ISOL,NMV) = 4000+KLL*10+KDIS
          NMV = NMV+1
          MOVE(ISOL,NMV) = 2000+KFF*10+IAYS(NS,NS)
          DO 890 M=KFF,KLL
          MR = KLL-M+KFF
515      NSUM = NSUM+LUA(MR)
          IF (NSUM.GE.NPP) GO TO 900
          LTEM(MR) = 0
890      CONTINUE
          GO TO 910
          LTEM(MR) = NSUM-NPP
520      LAST = ISROW(JGG,MR)
          C
          C      ---ROUTING-BACK ROUTINE.
          C
525      910      JFLAG = JFLAG+1
          IF (LTYPE.GT.1) GO TO 920
          GO TO 940
          GO TO (930,940,930,940,930,940,930,940), JFLAG
920      KFF = KF2
          KLL = KL2
          KDIS = KDIS2
          GO TO (950,940,930,940,950,940,930,940), JFLAG
930      CALL TISCCMP (NRT,K,J,ISOL,JSOL,NMV,JF)
          DO 960 KR=1,NROW
          KICKR(KR) = 0
530      CONTINUE
          GO TO 940
          KR=1,M=0
          LTEM(KR) = LUA(KR)
          IF (KASE.EQ.1) LTEM(KR) = LRA(KF)
          CONTINUE
          960

```

SUBROUTINE DYNAMIC

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

540      IF (NMV1.LT.1) GO TO 980
          DO 970 I=1,NMV1
              MOVE (ISOL+1,IM) = LOVE(IM)
          CONTINUE
970      IF (LTYPE.EQ.1.AND.JFLAG.GT.1) GO TO 1000
545      NSUM = NSUM1
          NMV = NMV1
          ISOL = ISOL+1
          IF (LTYPE.EQ.1) GO TO 880
          KFF = KF1
          KLL = KL1
          KOIS = KOIS1
          JF = JFLAG/2
          GO TO (850,880,980,990), JF
555      JFLAG = 0
          GO TO 750
C
C
1000     CONTINUE
          IF (KASE.EQ.3) GO TO 440
          IF (KASE.EQ.2) GO TO 1020
          IF (IRM.EE.NRM) GO TO 1020
          DO 1010 KR=1,NROW
              LITEM(KR) = LUA(KR)
1010     CONTINUE
          GO TO 600
1020     CONTINUE
C-----MARK TOTAL NUMBER OF SOLUTIONS GENERATED IN THIS STAGE.
C-----
          NTSOL(K) = ISOL
1030     CONTINUE
C-----IDENTIFY THE OPTIMAL SOLUTION PATH.
C-----
575     ISS = NTSOL(MSTAGE-1)+1
          IST = NTSOL(MSTAGE)
          LEAST = 9999999999
          IGOOD = ISS
          DO 1040 I=ISS,IST
              LFF = LFF(I)
              LFF = LFF/100000
              LEST = LFF/1E10+MOD(LFF,100000)
              IF (LEST.GE.LEAST) GO TO 1040
          IGOOD = I
          LEAST = LEST
1040     CONTINUE
C
          DO 1080 I=1,MSTAGE
              II = MSTAGE-I+1
              NREST(II) = IGOOD
              KSOL = MSOLN(IGOOD)
              KROOT = KSOL/10
              IF (KOE.GT.1) GO TO 1050
              IF (II.EQ.1) NOPT(1) = JRPOW(1)*10000+KSOL
              GO TO 1060
              IF (II.EQ.1) NOPT(1) = JROW(1)*1E8+JROW(2)*1E4+KSOL
              IF (II.GT.1) NOPT(II) = LEST(KROOT)*10000+KSOL
              NMV = 100*(KSOL,10)
              DO 1070 J=1,NMV
                  MOETA(II,J) = LOVE(IGOOD,J)
1070     CONTINUE
              K1 = KSOL/10
              IGOOD = K1
1080     CONTINUE
C
C
C-----PRINT OUT GENERATED SOLUTIONS AND THE OPTIMAL SOLUTION IN EACH
C-----STAGE.
C-----
          PRINT 1170
          PRINT 1180
          PRINT 1190
          ISS = 1
          DO 1110 M=1,MSTAGE
              JGG = JGTP(M)
              NR = NRIG(JGG)
              IF (M.EQ.1) GO TO 1090

```

SUBROUTINE DYNAMIC 73/172 CPT=0 TRACE PHOMP

FTN 4. 2+528

```

1090     ISS = NTSOL(M-1)+1
        IST = NTSOL(M)
620     DO 1100 I=ISS,IST
        PRINT 1200, P,KTRAN(M),JGG,NPICK(M),I,MSOLN(I),LES(I),LFV(I)
        1      (LGR(I),N=1,NP)
        M1 = MODMSOLN(I,10)
        PRINT 1210, (MOVE(I,J),J=1,M1)
625     1100 CONTINUE
        PRINT 1220, MREST(M)
        1110 CONTINUE
        PRINT 1230, MSCL
        PRINT 1240, LEAST
630     RETURN
C
1120 FORMAT (//,20X,19HWORK SEQUENCE TABLE)
1130 FORMAT (20X,19H***** )
1140 FORMAT (//,5X,5HSTAGE,5X,5HTRANS,5X,5HGROUP,5X,5HHOW MANY,5X,5HBAY
635     1,/)
1150 FORMAT (110,5X,I5,6X,I5,7X,I5,5X,I5)
1160 FORMAT (3X,19HNOT ENOUGH OF GP CUP,I4,2X,8HIN STAGE,I4)
1170 FORMAT (1H1,///,5X,42H***** DP SOLUTIONS GENERATED IN STAGES***** ,/
640     1,/)
1180 FORMAT (2X,5HSTAGE,2X,5HTRANS,2X,5HGROUP,2X,5HNPICK,2X,6HMSOL NO,2X
1,4HSOLUTION,2X,7HEND ROW,2X,6HFN VAL,2X,16HROW STATUS ARRAY)
1190 FORMAT (2X,5H***** ,2X,5H***** ,2X,5H***** ,2X,5H***** ,2X,6H***** ,2X
1,8H***** ,2X,7H***** ,2X,6H***** ,2X,7H***** ,7X,16H***** )
645     2***** ,/)
1200 FORMAT (2X,I3,4X,I3,2X,I3,3X,I4,3X,I4,2X,I5,2X,I5,3X,I15,3X,20I3)
1210 FORMAT (30X,13HMOVE DETAILS,10I7)
1220 FORMAT (12X,16H*OPT SOLUTION NO,I4,/)
1230 FORMAT (33HTOTAL NO. OF GENERATED SOLUTIONS=,I5)
1240 FORMAT (5X,5HTTMO=,I7)
END

```

SUBROUTINE DISCOMP 73/172 CPT=0 TRACE PHOMP

FTN 4.9+529

```

1      C      DISC
          SUBROUTINE DISCOMP (NRT,K,J,ISOL,JSOL,NMV,JR)
5      COMMON /DACOM/ ISROW(32,20),JGTP(70),MOETA(70,6),NPICK(70),NOPT(70)
          1),NRIG(32),KBL(70),KOPE
          COMMON /CCOM/ KICKS(5),KICKR(20),LTEM(20),MSOLN(410),LES(410),LFV
          1(410),NSEC,MROB,JGG,MOVE(410,6),NTSOL(70),NSOL,IPG(32),NAS(32),KTT
          2,JJRR,LAST,ISTATE,LGR(410,17)
10     C      ---SOLUTION BACK-KEEPING FUNCTION.
          C
          NSOL = NSOL+1
          DO 10 NS=1,NSEC
          KICKS(NS) = 0
15     10 CONTINUE
          IF(KOPE.GT.1) GO TO 20
          LES(ISOL)=LAST
          GO TO 30
20     IF (KTT.EQ.1) LES(ISOL) = LAST*10000+JJRR
          IF (KTT.EQ.2) LES(ISOL) = JJRR*10000+LAST
25     30 MSOLN(ISOL) = J*10+NMV
          DO 40 IR=1,MROW
          LGR(ISOL,IR) = LTEM(IR)
          40 CONTINUE
          NRT = 0
          IROW = JR
          DO 90 NV=1,NMV
          MOV = MOVE(ISOL,NV)
          MK = MOV/1000
          M1 = MOD(MOV,1000)
          MR = M1/10
          KD = MOD(M1,10)
          LROW = ISROW(JGG,MR)
          GO TO (50,50,50,70), MK
          IF (NV.EQ.NMV) GO TO 60
          NRT = NRT+IARS(LROW-IROW)
          GO TO 80
          60 NRT = NRT+IARS(LAST-IROW)
          GO TO 90
          70 NRT = NRT+NTIST(IROW,LROW)
          80 IROW = LROW
          90 CONTINUE
          C      ---COMPUTE PENALTY VALUE FOR THIS SOLUTION.
          C
          MR = 0
          LPEN = MR
          GO TO 120
          IF (NRIG(JGG).LT.5) GO TO 120
          DO 110 IR=1,MROW
          IF (LTEM(IR).EQ.0) GO TO 110
          NR = ISROW(JGG,IR)
          IF (MR.GT.0) GO TO 100
          MR = NR
          GO TO 110
          100 LPEN = LPEN+NDIST(MR,NR)
          110 CONTINUE
          IF (KOPE.GT.1) GO TO 130
          IF (K.EQ.1) LFV(ISOL) = NRT*100000+LPEN
          IF (K.GT.1) LFV(ISOL) = (LFV(J)/100000)*100000+NRT*100000+LPEN
          GO TO 160
          130 IF (K.GT.1) GO TO 140
          IF (KTT.EQ.1) LFV(ISOL) = NRT*1E10+LPEN
          IF (KTT.EQ.2) LFV(ISOL) = NRT*100000+LPEN
          GO TO 160
          140 IF (KTT.EQ.2) GO TO 150
          LV1 = (LFV(J)/100000)*100000
          LFV(ISOL) = LV1+NRT*1E10+LPEN
          GO TO 160
          150 LV2 = (LFV(J)/100000)*100000
          LFV(ISOL) = LV2+NRT*1E5+LPEN
          160 CONTINUE
          C      ---PRUNE SOLUTIONS AS GENERATED BY COMPARING FUNCTION VALUES.
          C
          JSOL = 0
          IF (K.EQ.1) NSE = 1

```

SUBROUTINE DISCOMP

73/172 CPT=0 TRACE PMOMP

FTN 4.8+524

```

      IF (K.GT.1) NSS = NTSOL(K-1)+1
      IF (ISOL.EQ.NSS) GO TO 200
80      NTS = ISOL-1
      DO 190 N=NSS,NTS
        IF (LES(N).NE.LES(ISOL)) GO TO 190
        LF1 = LFV(N)
        LF2 = LF1/1E5
85        LFOLD = LF1/1E10+MOD(LF1,100000)+MOD(LF2,100000)
        LF3 = LFV(ISOL)
        LF4 = LF3/100000
        LFNEW = LF3/1E10+MOD(LF3,100000)+MOD(LF4,100000)
90        IF (LFNEW.GE.LFOLD) GO TO 210
        LFV(N) = LFV(ISOL)
        LES(N) = LES(ISOL)
        MSOLN(N) = MSOLN(ISOL)
        DO 170 NV=1,NMV
95          170  MOVE(N,NV) = MOVE(ISOL,NV)
          CONTINUE
          JSOL = N
          DO 180 IR=1,MROW
100           180  LGR(N,IR) = LGR(ISOL,IR)
          CONTINUE
          GO TO 210
190        CONTINUE
200      JSOL = ISOL
      RETURN
105      210  ISOL = ISOL-1
      C
      RETURN
      END

```

FUNCTION NDIST

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

1      C      NDIST
          FUNCTION NDIST (JSROW,KSROW)
5      COMMON /CCOM/ IYSEC(20),IAYS(20,20),JBROW(2),NER1,NER2,NIGR(32,20)
          1,NYSEC
          JSEC = JSROW/100
          KSEC = KSROW/100
          JROW = MOD(JSEC*100,100)
          KROW = MOD(KSEC*100,100)
10     DO 10 MS=1,NYSEC
          IF (IYSEC(MS).EQ.JSEC) GO TO 20
          10 CONTINUE
          20 DO 30 NS=1,NYSEC
          IF (IYSEC(NS).EQ.KSEC) GO TO 40
15     30 CONTINUE
          40 KOIS = IAYS(MS,NS)
          GO TO (50,60,70,80,90,100,110,120), KOIS
          50 NDIST = 74-JROW+NER1+KROW
          GO TO 120
          60 NDIST = JROW+NER1+74-KROW
          GO TO 120
          70 NDIST = 74-JROW+NER2+74-KROW
          GO TO 120
          80 NDIST = JROW+NER2+KROW
          GO TO 120
          90 NDIST = 74-JROW+NER2+KROW
          GO TO 120
          100 NDIST = JROW+NER2+74-KROW
          GO TO 120
          110 NDIST = IARS(JFOW-KROW)
          120 RETURN
          END

```



SUBROUTINE ASSGN

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

1      C      ASS
          SUBROUTINE ASSGN
          COMMON /DACOM/ ISROW(32,20),JGTP(70),MDETA(70,6),NPICK(70),NOPT(70
5      1),NRIG(32),KRL(70),KOPE
          COMMON /AFCOM/ LLOC(500),NCTLC,NSTG(32),MPCL(25),MPC(25),MPLAN,
          1,KRKEY(40),NRPAY(32),NLCAO(32,30)
          COMMON /ALCOM/ LPAT(70,15),NTGB(70),IGCH(32),JYPOS(500)
10     1,COMMON /GCOM/ IRAYL(25),NRAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
          1,JHOLD(40),MGRFCUP,KTRAN(70)
          C      ---THIS SUBROUTINE PERFORMS ASSIGNMENT OF INDIVIDUAL CONTAINERS TO
          C      ---CELLS IN THE SHIP ACCORDING TO THE RESULTS OF SUBROUTINE DYNAMI
          C      -----
15     C      ---INITIALIZE VARIABLES AND THE LOAD INDICATOR ARRAY.
          C      -----
          DO 10 K=1,NCTLC
          LLOC(K) = 0
20     10 CONTINUE
          C      ---MAKE ASSIGNMENTS FOR MANUAL PLAN CONTAINERS.
          C      -----
          IF (MPLAN.EQ.0) GO TO 30
          DO 20 K=1,MPLAN
          I = MPC(K)
          LLOC(I) = MPCL(K)
25     20 CONTINUE
          C      ---FOR THE NEXT STAGE OF LOADING:
          C      -----
          30 DO 270 NS=1,MSTAGE
          KTR = KTRAN(NS)
          C      -----
35     C      ---IDENTIFY BAY NO. AND BAY CHARACTERISTICS.
          C      -----
          NC1 = NTGB(NS)
          I3 = NC1/10000
          KEY = KRKEY(I3)
          KY = KEY/10000000
          KTP = MOD(KY,10)
          KOUNT = 0
          NC2 = NC1/100
          JGG = MOD(NC2,100)
40     C      -----
          C      ---SET INITIAL TRANSTAINER POSITION FOR THIS STAGE.
          C      -----
          NPAT = MOD(NC1,100)
          JJ = 1
          KSOLN = NOPT(NS)
          K1 = KSOLN/10000
          IF (KOPE.GT.1) GO TO 40
          K2 = K1
          GO TO 50
          55     40 IF (KTR.EQ.1) K2 = K1/10000
          IF (KTR.EQ.2) K2 = MOD(K1,10000)
          NR = NRIG(JGG)
          DO 60 NF=1,NR
          IF (K2.EQ.ISROW(JGG,NF)) GO TO 70
          60     60 CONTINUE
          KSP = 0
          GO TO 80
          70     KSR = NF
          80     CONTINUE
          KAR = IGCH(JGG)
          KS7 = MOD(K2,10)
          C      -----
          C      ---CHECK TO SEE IF WE ARE ASSIGNING A CORRECT SIZE BAY TO THE GRO
          C      -----
70     C      GO TO (90,90,90,90,110,110,110,110), KS7
          IF ((KTP.EQ.1).OR.(KTP.EQ.3)) GO TO 120
          90     PRINT 230, JGG,I3
          100     STOP
          110     IF ((KTP.EQ.2).OR.(KTP.EQ.3)) GO TO 120
          GO TO 100
          C      -----
          C      ---CHECK THE NEXT MOVE DETAIL. IF EMPTY TRANSIT, SET CURRENT ROW P

```

SUPROUTINE ASSGN

73/172 CPT=0 TRACE PMDMP

FTN 4.8+528

```

C      ---AS THE TARGET ROW OF MOVE.  IF A PICKING MOVE, START THE PICKIN
C
90      120      NPP = NPICK(NS)
          MVV = MDETA(NS,JJ)
          MGO = MVV/1000
          MT = MVV/10
95      130      MTROW = MOD(MT,100)
          GO TO (140,140,130,130), MGO
          KSR = MTROW
          JJ = JJ+1
          GO TO 120

C      ---SET TRANSTAINER POSITION TO CORRECT INITIAL POSSESSING POW PEFO
C      ---IDENTIFY THE BEGINNING AND ENDING RECORD NUMBERS FOR THIS GROUP
140      IF (JJ.GE.2) GO TO 140
          IF (KSR.NE.0) GO TO 180
          KDIR = 1
          NEW = K2
150      IF (MGO.EQ.2) KDIR = -1
          NEW = NEW+KDIR
          DO 160 NF=1,N2
100      IF (NEW.EQ.ISPOW(JGG,NF)) GO TO 170
          CONTINUE
          GO TO 150
170      KSR = NF

C      ---GO TO THE CURRENT ROW IN CONTAINER FILE.
180      JSN = KFINO(JGG,KSR)

C      ---FOR THE NEXT LOAD PATTERN FOR THIS GROUP IN THIS BAY:
          DO 260 NC=1,NPAT

C      ---IDENTIFY LOAD PATTERN FOR THIS COLUMN--BEGINNING CELL AND
C      ---HOW MANY TO LOAD INTO.
115      LPN = LPAT(NS,NC)
          NCL = MCC(LP,100)
          LP1 = LPN/100
          JT = LP1/100
120      KS = MOD(LP1,100)
          KF = KS+NCL-1

C      ---FOR EACH CELL IN THE TIER:
          DO 260 KW=KS,KF

C      ---CHECK IF ROW IS CORRECT. IF SO, CHECK IF THE CONTAINER HAS BEEN
C      ---IF NOT, CHECK IF WE HAVE MOVED PAST THE LIMIT ROW.
130      190      JSN = KFINO(JGG,KSR)
          IF (JSN.GT.3) GO TO 220
          IF (MGO.EQ.2) GO TO 200
          KSR = KSR+1
          IF (KSR.GT.MTROW) GO TO 210
          GO TO 190

C      ---WE HAVE BEEN PICKING DOWN THE ROWS. IF PAST LIMIT, CHECK THE N
C      ---MOVE DETAIL. OTHERWISE, MOVE DOWN THE NEXT ROW.
140      200      KSR = KSR-1
          IF (KSR.LT.MTROW) GO TO 210
          GO TO 190

C      ---IF THE NEXT MOVE DETAIL IS AN EMPTY TRANSIT. UPDATE ROW POSITIO
C      ---AND CHECK ANOTHER MOVE DETAIL. IF A PICKING MOVE, GO TO THE FIR
C      ---CONTAINING ROW FOR THIS PICK.
145      210      JJ = JJ+1
          MVV = MDETA(NS,JJ)
          MGO = MVV/1000
          MT = MVV/10
150      MTROW = MOD(MT,100)
          IF (MGO.LE.2) GO TO 190
          KSR = MTROW

```

SUBROUTINE ASSGN

73/172 CPT=0 TRACE PHOMP

FTN 4.9+529

```

155          GO TO 210
C
C
C      ---WE ARE AT THE CORRECT ROW. IF THIS CONTAINER HAS BEEN LOADED, L
C      ---THE NEXT RECORD. IF NOT, MAKE AN ASSIGNMENT TO CURRENT CELL.
C      -----
160      220      IF (LLOC(JSN).EQ.0) GO TO 230
          GO TO 190
          230      IF (MPLAN.EQ.C) GO TO 250
          OO 240 MP=1,MPLAN
          IF (MPC(MP).EQ.JSN) GO TO 190
165      240      CONTINUE
          250      LLOC(JSN) = IP*10000+KK*100+JT
          KOUNT = KOUNT+1
          IF (KOUNT.GE.NPP) GO TO 270
          JSN = JSN+1
170      260      CONTINUE
          270      CONTINUE
          RETURN
C
175      280      FORMAT (///,61:YOU ARE TRYING TO LOAD WRONG SIZE CONTAINERS O
          1      GROUP,I4,J4H INTO 9AY NO.,I4.48H. MAKE CORRECTIONS IN FILE
          2      GPLAN,PLEASE.)
          ENO

```

```

1      C      KFIND
3      FUNCTION KFINC (JGG,KSR)
5      COMMON /DACOM/ ISRCW(32,20),JGTP(70),MOETA(70,6),NPICK(70),NOPT(70),
7      NRIG(32),KBL(70),KOPE
9      COMMON /AFCOM/ LLOC(500),NCTLD,NSTG(32),MPCL(25),MPC(25),MPLAN,
11     KKEY(40),NRBAY(32),NLCAD(32,30)
13     COMMON /ALCOM/ LPAT(70,15),NTGB(70),IGCH(32),JYPOS(500)
15     COMMON /GCOM/ JBAYL(25),NBAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
17     1,JHOL(40),MGRUP,KTRAN(70)
19     C      ---THIS FUNCTION FINDS THE FIRST UNASSIGNED CONTAINER IN THE CORRE
21     C      ---YARD ROW AND TRANSMITS ITS SEQUENCE NUMBER. IF KFINO=0, IT IMPL
23     C      ---NO SUCH CONTAINER EXISTS.
25     C      -----
27     KFINO = 0
29     J1 = NSTG(JGG)
31     J2 = J1+MIG(JGG)-1
33     DO 10 JC=J1,J2
35         IYP = JYPOS(JC)
37         DECODE (5,20,IYP) JSROW,JUNK
39         IF (JSROW.NE.ISROW(JGG,KSR)) GO TO 10
41         IF (LLOC(JC).GT.0) GO TO 10
43         KFINO = JC
45     RETURN
47     10 CONTINUE
49     RETURN
51     C      20 FORMAT (I4,R2)
53     END

```

SUBROUTINE FCHECK

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

1      C      FCM
2      SUBROUTINE FCHECK
3      COMMON /FCOM/ FPL,CMT,C9L,CFL,COWT,COVM,COLM,CISP,DRAFT,FSMT,GPMAX
5      1,GMMIN,GY,HKM,IBNO(40),MTIRO(15),MCOLH(15),MCOLD(15),NBAYS,NVOY,
6      2,NDATE,TWT,TVM,TLN,PLWT2(40),PLWT4(40),PLL2(40),PLL4(40),PLVM2(40
7      3),PLVM4(40),TRIM,ICSHIP,TKMT,TKVM,TKLM,THLS,TRMAX,TRMIN,VMLS,WTLS,
8      4,WTLO
9      COMMON /FSCOM/ CGL2(40),CGL4(40),KOLL(25,12),KOLN(25,12),NCBL(25),
10     1,MTIRH(15),NHC(25,10),VCG2D(15,4),VCG4D(15,4),VCG2H(15,10),VCG4H(15
11     2,10),WTOD(25,12),MHCH(40,12),DEKL2(40),DEKL4(40),LOVE,NVIOL,JVIOL
12     3(25),LPAO(25),ACVM1,ACVM2,ACLM1,ACLM2,ACVM,ACLP,NCUBE(25,10),WCECK
13     4(25,10)
14     COMMON /AFCOM/ LLCC(500),NCTLO,NSTG(32),MPCL(25),MPC(25),MPLAN,
15     1,KKEY(40),NRBAY(32),NLCA(32,30)
16     COMMON /FLCOM/ IPST(500),WGT(500)
17     COMMON /GCOM/ JBAYL(25),NBAYL,NSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
18     1,JHOLD(40),MGRUP,KTRAN(70)
19     DIMENSION AWT2(40),AWT4(40),ALM2(40),ALM4(40),AVM2(40),AVM4
20     1(40),CWT2(40),CWT4(40),CLM2(40),CLM4(40),CVM2(40),CVM4(40)
21
22     CCCCCC
23     ---IN THIS SUBROUTINE, FEASIBILITY CHECKS ARE MADE ON GENERATED LO
24     ---PLAN WITH REGARDS TO SHIP STABILITY RESTRICTIONS AND OTHER LOAD
25     ---RESTRICTIONS INHERENT TO SHIP STRUCTURE.
26     -----
27     ---INITIALIZE ARRAYS AND VARIABLES.
28     -----
29     KF = 0
30     DO 10 I=1,NBAYS
31         AVM4(I) = 0.0
32         AVM2(I) = AVM4(I)
33         ALM4(I) = AVM4(I)
34         ALM2(I) = ALM4(I)
35         AWT4(I) = AWT2(I)
36         AWT2(I) = AWT4(I)
37     10 CONTINUE
38     DO 40 I=1,NBAYL
39         IB = IBAYL(I)
40         KEY = KKEY(IB)
41         K1 = KEY/10000.0000
42         LOC = K1/10
43         KOL = NCBL(I)
44         DO 30 K=1,KOL
45             IF (LOC.EQ.2) GO TO 20
46             WTOD(I,K) = WCECK(I,K)
47             GO TO 30
48             NHC(I,K) = NCUBE(I,K)
49     20 CONTINUE
50     30 CONTINUE
51     40 CONTINUE
52
53     CCCCCC
54     ---FOR EACH CONTAINER LOADED, COMPUTE MOMENT CONTRIBUTIONS
55     ---AND UPDATE STACK WEIGHT AND HIGH CUBE STATUS.
56     -----
57     DO 180 I=1,NCTLO
58         IF (LLCC(I).EQ.0) GO TO 180
59         LOC = LLCC(I)
60         I9 = LOC/10000
61         L1 = MOD(LOC,10000)
62         IC = L1/100
63         IT = MOD(L1,100)
64         IF (MPLAN.EC.0) GO TO 60
65         DO 50 J=1,MPLAN
66             IF (I.NE.MPC(J)) GO TO 50
67             KF = 2
68             GO TO 60
69     50 CONTINUE
70     C      DO 70 J9=1,NBAYL
71         IF (IB.NE.IBAYL(J9)) GO TO 70
72         IF (KF.EC.2) KF = 1
73         GO TO 90
74     70 CONTINUE
75     C      NC = NCBL(JF)
76     DO 90 JK=1,NC
77         IF (IC.NE.KOLL(J9,JK)) GO TO 90

```

SUBROUTINE FCHECK

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

      IF (KF.EQ.1) KF = 0
      GO TO 100
90      CONTINUE
C
100     KEB = IPST(I)/10000
        KSZ = MOD(KEB,10)
        KHC = MOD(IPST(I),10)
85      KEY = KKEY(I)
        K1 = KEY/100000
        KSEC = MOD(K1,100)
        K2 = K1/100
        NTIR = MOD(KEY,100)
90      LOK = K2/10
        KTP = MOD(K2,10)
C
C-----FOR 20 FT CONTAINERS.-----
95      IF (KSZ.GT.4) GO TO 140
        AWT2(I) = AWT2(I) + WGT(I)
        ALM2(I) = ALM2(I) + WGT(I)*CGL2(I)
        IF (LOK.EQ.2) GO TO 120
        IF (KF.NE.0) GO TO 110
        WTD(JB,JK) = WTD(JB,JK) + WGT(I)
        KOIN(JB,JK) = 1
110     AVM2(I) = AVM2(I) + WGT(I)*VCG2D(KSEC,IT)
        GO TO 180
120     IF (KF.NE.0) GO TO 130
        IF (KHC.EQ.1) NHC(JB,JK) = NHC(JB,JK) + 1
130     ITT = NTIR*(KSEC) - NTIR + IT
        AVM2(I) = AVM2(I) + WGT(I)*VCG2H(KSEC,ITT)
        GO TO 180
C
C-----FOR 40 FT CONTAINERS.-----
140     ALM4(I) = ALM4(I) + WGT(I)*CGL4(I)
        AWT4(I) = AWT4(I) + WGT(I)
        IF (LOK.EQ.2) GO TO 150
        IF (KF.NE.0) GO TO 150
        WTD(JB,JK) = WTD(JB,JK) + WGT(I)
        KOIN(JB,JK) = 2
150     AVM4(I) = AVM4(I) + WGT(I)*VCG4D(KSEC,IT)
        GO TO 180
160     IF (KF.NE.0) GO TO 170
        IF (KHC.EQ.1) NHC(JB,JK) = NHC(JB,JK) + 1
170     ITT = 4*NTIR*(KSEC) - NTIR + IT
        AVM4(I) = AVM4(I) + WGT(I)*VCG4H(KSEC,ITT)
180     CONTINUE
C
C-----COMPUTE TOTAL MOMENTS DUE TO CONTAINERS.-----
        SUM = 0.0
130     SUN = SUM
        SUN2 = SUM
        SUN1 = SUM
        SUM2 = SUM
        SUM4 = SUM
        DO 190 I=1,NRAYS
135     CWT2(I) = PLWT2(I) + AWT2(I)
        CWT4(I) = PLWT4(I) + AWT4(I)
        CLM2(I) = PLLM2(I) + ALM2(I)
        CLM4(I) = PLLM4(I) + ALM4(I)
        CVM2(I) = PLVM2(I) + AVM2(I)
140     CVM4(I) = PLVM4(I) + AVM4(I)
        SUM = SUM + PLWT2(I) + PLWT4(I)
        SUN = SUN + AWT2(I) + AWT4(I)
        SUM1 = SUM1 + PLVM2(I) + PLVM4(I)
        SUM2 = SUM2 + PLLM2(I) + PLLM4(I)
145     SUM1 = SUM1 + AVM2(I) + AVM4(I)
        SUM2 = SUM2 + ALM2(I) + ALM4(I)
190     CONTINUE
        ECWT = CWT
        ECVM = CVM
        ECLM = CLM
        ACWT = SUM
        ACVM = SUM1
        ACLM = SUM2
        TCWT = ECWT + ACWT

```

SUBROUTINE FCHECK

73/172 CPT=0 TRACE PMOMP

FTN 4.4+529

```

155      TCVM = ECVN+ACVM
      TCLM = ECLM+ACLM
      C
      C ---COMPUTE GM AND TRIM.
      C
160      TWT = WTLS+TKWT+CONT+TCWT
      TVM = VMLS+TKVW+COVM+TCVM
      TLM = TMLS+TKLW+COLM+TCLM
      GM = HKY-(TVM+FSMT)/TWT
      TRIM = (TLM-CBL*TWT)/(CNT*100)
165      OFOR = DRAFT-TRIM/2.0-TRIM*CFL/BPL
      ODAFT = DRAFT+TRIM/2.0-TRIM*CFL/BPL
      C
      C ---CHECK TOTAL DISPLACEMENT FIGURES.
      C
170      DIFF = ABS(TWT-DISP)
      IF (DIFF.LE.5.C) GO TO 200
      PPINT 320, TWT,DISP
      C
      C ---CHECK GM RANGE.
      C
175      NVIOL = 0
      200 IF (GM.GE.GMMIN.AND.GM.LE.GMMAX) GO TO 210
      NVIOL = NVIOL+1
      JVIOL(NVIOL) = 1
      L3AD(NVIOL) = 1
180      IF (GM.GT.GMMAX) L3AD(NVIOL) = 2
      PPINT 330
      C
      C ---CHECK TRIM RANGE.
      C
185      210 IF (TRIM.GE.TRMIN.AND.TRIM.LE.TRMAX) GO TO 220
      NVIOL = NVIOL+1
      JVIOL(NVIOL) = 2
      L3AD(NVIOL) = 1
190      IF (TRIM.GT.TRMAX) L3AD(NVIOL) = 2
      PRINT 340
      C
      220 CONTINUE
      C
195      C ---DECK STRENGTH LIMIT CHECK AND HIGH CUBE NUMBER CHECK.
      C
      DO 270 I=1,N9AYL
      IB = IBAYL(I)
      KEY = K9KEY(IB)
200      K1 = KEY/10
      LOC = K1/10
      ITP = MOD(K1,10)
      KOL = NCBL(I)
      250 K=1,KOL
      KOLA = KCLL(I,K)
      IF (LOC.EQ.2) GO TO 250
      WT = WCOL(I,K)
      IF (KOLN(I,K).EQ.2) GO TO 240
      IF (WT.LE.DEKL2(IB)) GO TO 260
210      230 NVIOL = NVIOL+1
      JVIOL(NVIOL) = 3
      L3AD(NVIOL) = I*100+K
      PRINT 350, IB,KOLA
      GO TO 260
215      240 IF (WT.LE.DEKL4(IB)) GO TO 260
      GO TO 230
      C
      C ---HIGH CUBE CHECK FOR UNDER-DECK RAYS.
      C
220      250 IF (NHC(I,K).LE.MHCH(IB,KOLA)) GO TO 260
      NVIOL = NVIOL+1
      JVIOL(NVIOL) = 4
      L3AD(NVIOL) = I*100+K
      PRINT 360, IB,KOLA
225      260 CONTINUE
      270 CONTINUE
      C
      C ---PRINT OUT TABLE OF STABILITY CALCULATION.
      C
230      MONTH = NDATE/10000
      NYR = MOD(NDATE,100)

```

SURROUTINE FCHECK

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

235      ND1 = NDATE/100
      NDT = MOO(ND1,100)
      PRINT 370
      PRINT 380
      PRINT 390, IDSHIP, NVOY, MONTH, NDT, NYR
      PRINT 400
      PRINT 410
240      DO 310 I=1,NBAYS
      KEY = K9KEY(I)
      K1 = KEY/IEF
      LOC = K1/10
      ITP = MOD(K1,10)
245      IF (ITP.EQ.2) GO TO 290
      IF (LOC.EQ.2) GO TO 280
      PRINT 420, I, I9NO(I), CWT2(I), CVM2(I), CGL2(I), CLM2(I)
      GO TO 310
280      PRINT 430, I, I9NO(I), CWT2(I), CVM2(I), CGL2(I), CLM2(I)
      GO TO 310
250      I99 = I9NO(I)-1
290      IF (LOC.EQ.2) GO TO 300
      PRINT 420, I, I99, CWT4(I), CVM4(I), CGL4(I), CLM4(I)
      IF (ITP.EQ.2) GO TO 310
      PRINT 420, I, I9NO(I), CWT2(I), CVM2(I), CGL2(I), CLM2(I)
255      GO TO 310
300      PRINT 430, I, I99, CWT4(I), CVM4(I), CGL4(I), CLM4(I)
      IF (ITP.EQ.2) GO TO 310
      PRINT 430, I, I9NO(I), CWT2(I), CVM2(I), CGL2(I), CLM2(I)
260      310 CONTINUE
      PRINT 440, TCWT,TCVM,TCLM
      C
      C
      C      ---PRINT OUT SHIP STABILITY SUMMARY
265      PRINT 470
      PRINT 480
      PRINT 490
      PRINT 500, WTLS,VMLS,TMLS
      PRINT 510, COVT,COVM,COLM
270      PRINT 520, TKWT,TKVM,TKLM,FSMT
      PRINT 530, TCWT,TCVM,TCLM
      PRINT 540, TWT,TVM,TLM,FSMT
      PRINT 550, HKM,CMT,CBL,DRAFT
275      PRINT 560, HKM,TVM,FSMT,TWT,GM
      PRINT 570, TLM,CBL,TWT,CMT,TPIM
      PRINT 580, DFOR,DAFT
      C
      C
      C      ---COMPUTE ALLOWABLE RANGES OF VERTICAL AND LONGITUDINAL MOMENTS T
280      ---IN THE FEASIBLE STABILITY REGION, IF NECESSARY.
      IF (NVVOL.EQ.0) RETURN
      LOVE = 0
      IF ((JVIOL(1).LE.2) LOVE = JVIOL(1)
      IF ((JVIOL(1)+JVIOL(2)).EQ.3) LOVE = 3
285      IF (LOVE.EQ.0) RETURN
      ACVM1 = (HKM-GMAX)*TWT-FSMT-VMLS-TKVM-COVM-ECVM
      ACVM2 = (HKM-GMIN)*TWT-FSMT-VMLS-TKVM-COVM-ECVM
      ACLM1 = TRMIN*CMT*100+CBL*TWT-TMLS-TKLM+COLM-ECLM
290      ACLM2 = TRMAX*CMT*100+CBL*TWT-TMLS-TKLM+COLM-ECLM
      C
      C
      C      ---PRINT OUT ABOVE INFORMATION AND CALL SUBROUTINE STABLE.
      PRINT 450, ACVM1,ACVM2
295      PRINT 460, ACLM1,ACLM2
      C
      C
      C      CALL STABLE
      RETURN
300      C
      320 FORMAT (///,39H TOTAL DISPLACEMENT FIGURES DO NOT MATCH,/,
      139H CHECK DISPLACEMENT ENTRY IN FILE CPLAN.,3X,14H CORRECT VALUE=,F7.1)
      2.1,5X,14H ENTERED VALUE=,F7.1)
330 FORMAT (///,19H IS OUT OF RANGE.)
340 FORMAT (//,13H TWT CUT OF RANGE.)
305      350 FORMAT (//,36H CHECK STRENGTH LIMIT EXCEEDED IN BAY1,I4,3X,4H COL.,I4)
      360 FORMAT (//,27H TOO MANY HIGH CURVES IN BAY1,I4,3X,4H COL S=01,I4)
      370 FORMAT (141,///,20X,30H TRIM AND STABILITY CALCULATION)
      380 FORMAT (20X,29H *****)

```



SUBROUTINE FCHECK

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

310      390 FORMAT (2X,5MSHIP:A10,20X,10HVOYAGE NO:,I3,20X,5HDATE:,I2,1H/,I2,
      400 FORMAT (5X,5HSEIZ E,2X,9HCONTAINER,2X,2HVERTICAL,3X,3HLCG,7X,
      410 FORMAT (12X,6HHEIGHT,5X,6HMOMENT,14X,7HFORE(-),3X,6HAFT(+),/)
315      420 FORMAT (I3,3X,I2,1H/,3F10.1,3X,F10.1)
      430 FORMAT (I3,3X,I2,1H/,3F10.1,3X,F10.1)
      440 FORMAT (/6HTOTALS,3X,2F10.1,12X,F10.1)
      450 FORMAT (///,40HLOWER LIMIT OF FEASIBLE VERTICAL MOMENT=F10.1,
      15H KT-M,/,40HUPPER LIMIT OF FEASIBLE VERTICAL MOMENT=F10.1,
      25H KT-M)
320      460 FORMAT (/37HLCWER LIMIT OF FEASIBLE LONG. MOMENT=F10.1,5H KT-M,/,
      1,37HUPPER LIMIT OF FEASIBLE LONG. MOMENT=F10.1,5H KT-M,/)
      470 FORMAT (1H1,///,30X,22MSHIP STABILITY SUMMARY)
      480 FORMAT (30X,22H*****STABILITY SUMMARY*****
325      490 FORMAT (///,20X,10HHEIGHT(KT),10X,12HVERT. MOMENT,10X,12HLONG. MOM
      1NT,10X,10HFS. MOMENT,/)
      500 FORMAT (10HLIGHT SHIP,10X,F10.1,10X,F11.1,10X,F12.1)
      510 FORMAT (15HCONST. A OTHERS,5X,F10.1,10X,F11.1,10X,F12.1)
      520 FORMAT (11HTANK TOTALS,9X,F10.1,10X,F11.1,10X,F12.1,10X,F10.1)
330      530 FORMAT (10HCONTAINERS,10X,F10.1,10X,F11.1,10X,F12.1)
      540 FORMAT (/5HTOTAL,15X,F10.1,10X,F11.1,10X,F12.1,10X,F10.1)
      550 FORMAT (/5X,4HTKM=F6.2,5X,4HNTC=F7.2,5X,4HLCG=F5.2,5X,6HDRAFT=
      1,F6.2)
      560 FORMAT (/10X,9HG M =,F6.2,4H - (,F10.1,2H +,F6.1,3H )/,F7.1,
      12H =,F5.2,2H M)
335      570 FORMAT (/10X,11HT R I M = (,F10.1,2H -,F5.2,2H X,F6.1,4H )/(,F7.2
      1,11H X 100) =,F5.2,2H M)
      580 FORMAT (/10X,10HFWO DRAFT=F5.1,5X,10HAFT DRAFT=F5.1,/)
      END

```

SUBROUTINE LPCUT

73/172 CPT=0 TRACE PMOMP

FTN 4.2+523

```

1      C      OUT
          SUBROUTINE LPCUT
5      COMMON /ALCOM/ LPAT(70,15),NIGR(70),IGCH(32),JYPOS(500)
          COMMON /FLCOM/ IPST(500),WGT(500)
          COMMON /LCOM/ IOC(500),LNAME(10),IPOST(5),KPRINT,KOMM(12)
          COMMON /GCOM/ ITRAYL(25),NRAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
10     1,JHOLD(40),MGRUP,KTRAN(70)
          COMMON /AFCOM/ LLOC(500),NCTLD,NSTG(32),MPCL(25),MPC(25),MPLAN,
          1,KHKEY(40),NRBAY(32),NLCAO(32,30)
          COMMON /OCOM/ TYSEC(20),TAYS(20,20),JBROW(2),NER1,NER2,NIGR(32,20)
          1,NYSEC
          DIMENSION NOIS(2)
15     C      ---THIS SUBROUTINE IS FOR PRINTING OUT THE GENERATED LOAD PLAN.
          C      -----
          IF (KPRINT.EQ.1) GO TO 10
          PRINT 130
          PRINT 140
20     PRINT 150
          DO 100 KTR=1,2
          NOIS(KTR) = 0
          ISR = JBROW(KTR)
          PRINT 160, KTR,JBROW(KTR)
25     DO 90 IS=1,MSTAGE
          IF (KTR.NE.KTRAN(IS)) GO TO 90
          NC1 = NIGR(IS)
          IB = NC1/10000
          NC2 = NC1/100
          JGG = MGC(NC2,100)
          DO 20 NR=1,NRAYL
30     IF (IP.EQ.ITRAYL(NR)) GO TO 30
          20 CONTINUE
          NPAT = MCD(NC1,100)
          DO 80 NC=1,NPAT
35     LPN = LPAT(IS,NC)
          NCL = MGC(LP,100)
          LPL = LPN/100
          JT = LP/100
          KS = MOD(LPL,100)
          KF = KS+NCL-1
          DO 70 KK=KS,KF
          LOKN = IB*10000+KK*100+JT
          DO 40 JC=1,NCTLD
45     IF (LLCC(JC).EQ.LOKN) GO TO 50
          40 CONTINUE
          GO TO 70
          ISEQ = JC
          KEY = IPST(ISEQ)/10000
          IF = KEY/10
          IT = MOD(KEY,10)
          KEQ = MOD(IPST(ISEQ),10000)
          IL = KEQ/1000
          KE = MOD(KEQ,1000)
          IC = KEY/10
          IF (KPRINT.EQ.1) GO TO 60
          PRINT 170, IB,IB,KS,JT,IPOST(IP),IOC(ISEQ),LNAME(IL),
          KOMM(ICOM),WGT(ISEQ),JYPOS(ISEQ)
          JP = JYPOS(ISEQ)
          DECODE(6,10,JP) JSR,JUNK
          NCIS(KTR) = NOIS(KTR)+NOIS(ISR,JSR)
          ISF = JSR
          70 CONTINUE
          80 CONTINUE
          90 CONTINUE
65     100 CONTINUE
          IF (MPLAN.EQ.0) GO TO 120
          IF (KPRINT.EQ.1) GO TO 120
          PRINT 130
70     DO 110 I=1,MPLAN
          IO = MPC(I)
          LOK = MPCL(I)
          IA = LOK/10000
          LI = MOD(LOK,10000)
          IC = LI/100
          IT = MOD(LI,100)
          KEY = IPST(IO)/10000
75

```

SUBROUTINE LPCUT

73/172 CPT=0 TRACE PMDMP

FTN 4.8+52A

```

      IP = KEY/10
      KEQ = 400(IFST(ID),10000)
80      IL = KEQ/1000
      KEX = 400(KEQ,1000)
      ICOM = KEX/10
      PRINT 170, IB, IE, IC, IT, IPORT(IP), IDC(ID), LNAME(IL), KOMM(ICOM),
85      WGT(ID), JYPCS(IC)
      1
110 CONTINUE
120 PRINT 200, NOIS(1), NOIS(2)
      RETURN
C
130 FORMAT (1H1, //, 30X, 19HCONTAINER LOAD PLAN)
140 FORMAT (30X, 19H*****//)
90 150 FORMAT (2X, 3HBEY, 3X, 9HLOCATION, 3X, 4HPORT, 10X, 2HID, 10X, 4HLINE, 7X,
      14HTYPE, 7X, 6HHEIGHT, 3X, 13HYARD LOCATION, //)
160 FORMAT (//, 2X, 7HFOR NC, I2, 2X, 6HTRANS, 10X, 17HINITIAL POSITION, I6
      1)
95 170 FORMAT (I5, 3X, I2, 1H-, 2I2, 3X, A7, 3X, A12, 3X, A10, 3X, A10, F5.1, 5X, A10)
180 FORMAT (I6, R2)
190 FORMAT (///, 47H***MANUALLY PLANNED CONTAINER LOAD POSITIONS***)
200 FORMAT (//, 33HTOTAL TRANSTAINER MOVE DISTANCES=, 2I10)
      END

```

SUBROUTINE STABLE

73/172 CPT=0 TRACE PHOMP

FTN 4.4+528

```

1      C      STAR
          SUBROUTINE STABLE
5      COMMON /DACOM/ ISROW(32,20),JGTP(70),MDETA(70,6),NPICK(70),NOPT(70)
          1) ,NPIG(32),KRL(70),KOPF
          COMMON /AFCOM/ LLOC(500),NCTLO,NSTG(32),MPCL(25),MPC(25),MPLAN,
          1) ,KKEY(40),NRRAY(32),NLCA(32,30)
          COMMON /FSCOM/ CGL2(40),CGL4(40),KOLL(25,12),KOIN(25,12),NCAL(25),
10      1) ,NTHP(15),NHC(25,10),VCG2D(15,4),VCG4D(15,4),VCG2H(15,10),VCG4H(15
          2,10),WTD(25,12),MPCH(40,12),DEKL2(40),DEKL4(40),LOVE,NVIOL,JVIOL
          3(25),LRAD(25),ACVM1,ACVM2,ACLM1,ACLM2,ACVM,ACLM,NCU9E(25,10),WCECK
          4(25,10)
          COMMON /ALCOM/ LPAT(70,15),NTGB(70),IGCH(32),JXPOS(500)
15      COMMON /FLCOM/ IPST(500),WGT(500)
          COMMON /GCOM/ JRAYL(25),NRAYL,MSTAGE,GMEAN(32),NTG(32),NCEL(40,12)
          1) ,JHOLD(40),NGROUP,KTRAN(70)
          DIMENSION JOONE(50), LCOONE(50), KOONE(100)
          -----
          ---FOR GM VIOLATION:
          -----
20      CCCCCC
          ---INITIALIZE VARIABLES AND APPAYS.
          -----
          DO 10 I=1,50
          JOONE(I) = 0
25      10 CONTINUE
          NEXT = 0
          HLM = 0.0
          HVM = HLM
          GO TO (20,330,20), LOVE
          CCCCCC
          ---FOR NEXT GROUP:
          -----
30      20 LOR = LRAD(1)
          DO 230 JGG=1,NGROUP
          IF (JOONE(JGG).EQ.1) GO TO 230
          NSS = NSTG(JGG)
          NLL = NSS*NIG(JGG)-1
          NCIG = NIG(JGG)
          DO 30 N=1,NCIG
          KOONE(N) = 0
          30 CONTINUE
          CCCCCC
          ---FOR SINGLE RAY LOADING GROUP, GET THE NEXT HIGHEST TIER.
          -----
          DO 40 L=1,50
          LCOONE(L) = 0
          40 CONTINUE
          IF (NRBAY(JGG).GT.1) GO TO 100
          DO 50 KS=1,PMSTAGE
          IF (JGG.EQ.JGTP(KS)) GO TO 60
          50 CONTINUE
          IB = KRL(KS)
          NTH = NTHR(KS)
          NPAT = NTH(ATT,100)
          DO 70 J=1,NRAYL
          IF (IRAYL(J).EQ.KRR) GO TO 80
          70 CONTINUE
          C
          60 NUM = 0
          LHIGH = NUM
          C
          ---PICK THE HIGHEST UNFILLED TIER.
          -----
          DO 90 N=1,NPAT
          NN = NPAT-N+1
          NUM = NUM+1
          IF (LCOONE(NUM).EQ.1) GO TO 90
          LP = LPAT(KS,NN)
          ITIR = LP/10000
          IF (ITIR.LE.LHIGH) GO TO 90
          70 KSS=KS
          NN = NN
          LHIGH = ITIR
          NUM = NUM
          90 CONTINUE
          IF (LHIGH.GT.0) GO TO 150
          GO TO 230
75

```

SUBROUTINE STABLE

73/172 CPT=0 TRACE PHOMP

FTN 4.8+529

```

C
C
C
90      ---FOR MULTIPLE BAY LOADING, PICK THE HIGHEST TIER AMONG ALL
C      ---TIERS OF RECEIVING BAYS THROUGHOUT.
C      -----
100     NUM = 0
        LHIGH = NUM
85     DO 140 KS=1,MSTAGE
        IF (JGTP(KS),NE,JGG) GO TO 140
        NXB = KBL(KS)
        LOC = KBKEY(NXB)/1000000000
        IF (NEXT.EQ.1) GO TO 110
        IF (LOC.EQ.2) GO TO 140
90     GO TO 120
        IF (LOC.EQ.1) GO TO 140
110     NTT = NTER(KS)
120     NPAT = MCO(NTT,100)
95     DO 130 N=1,NPAT
        NN = NPAT-N+1
        NUM = NUM+1
        IF (LCONE(NUM).EQ.1) GO TO 130
        LP = LPAT(KS,NN)
        ITIR = LP/10000
        IF (ITIR.LE.LHIGH) GO TO 130
100    KSS=KS
        NO = NN
        LHIGH = ITIR
        NUMB = NUM
        IB = NXB
105    CONTINUE
130    CONTINUE
140    IF (LHIGH.GT.0) GO TO 150
        IF (NEXT.EQ.1) GO TO 230
        NEXT = 1
        GO TO 100
C
C
C
115    ---FOR NEXT CELL IN THAT TIER:
C      -----
150    LP = LPAT(KSS,NO)
        LL = LP/100
        ITIP = LL/100
        KOL = MOD(LL,100)
        NCOL = MOD(LP,100)
120    KEY = KBKEY(IB)
        K1 = KEY/1000000
        LOK = K1/1000
        KSEC = MOD(K1,100)
125    IKK = IGCH(JGG)
        ICT = MOD(IKK,10)
        LOB = LHAD(1)
        NTIP = MOD(KEY,100)
        ITT = MTIRH(KSEC)-NTIP+ITIP
C
C
C
130    ---ASSIGN EITHER THE NEXT HEAVIEST, OR THE NEXT LIGHTEST TO THIS T
C      ---THEN COMPUTE VERTICAL MOMENT CONTRIBUTION FROM THIS EXCHANGE.
C      -----
        DO 220 NC=1,NCCL
135    STWT = 0.0
        IF (LOB.EQ.1) STWT = 100.
        DO 180 M=NSS,4LL
        NOM = 4-NSS+1
        IF (LCONE(NOM).EQ.1) GO TO 180
        IF (MFTD(M).EQ.1) GO TO 180
140    IF (LCB.EQ.2) GO TO 180
        IF (WGT(M).GE.STWT) GO TO 180
        GO TO 170
        IF (WGT(M).LE.STWT) GO TO 180
160    NO4B = NOM
170    MASS = M
        STWT = WGT(MASS)
180    CONTINUE
        KDCNE(NC+1) = 1
        IF (ICT.GT.4) GO TO 230
        IF (LOK.EQ.2) GO TO 190
150    HVM = HVB+WGT(MASS)*VCG2C(KSEC,ITIR)
        GO TO 220
        HVM = HVB+WGT(MASS)*VCG2H(KSEC,ITT)
190    GO TO 220

```

SUBROUTINE STABLE

73/172 CPT=0 TRACE PMDMP

FTN 4.A+523

```

155      200      IF (LOK.EQ.2) GO TO 210
           HVM = HVP+WGT(MASS)*VCG4C(KSEC,ITIR)
           GO TO 220
160      210      HVM = HVP+WGT(MASS)*VCG4H(KSEC,ITT)
           220      CONTINUE
           LDONE(NUMB) = 1
           IF (NRDAY(JGG).EQ.1) GO TO 80
           GO TO 100
165      230      CONTINUE
C
C      ---COMPUTE MOMENT CONTRIBUTION BY MANUAL PLAN CONTAINERS.
C      -----
           IF (MPLAN.EQ.0) GO TO 240
           DO 270 M=1,MPLAN
           I = MPC(M)
           LOC = MPCL(I)
           I9 = LOC/10000
           L1 = MOD(LOC,10000)
           IL = MOD(L1,100)
           KEY = K9KEY(ITR)
           KEB = IPST(I)/10000
           KSZ = MOD(KEB,10)
           LOC = KEY/1000000000
           K1 = KEY/1000000
           KSEC = MOD(K1,100)
           NTIR = MOD(KEY,100)
           IF (LOC.EQ.2) GO TO 250
           IF (KSZ.GT.4) GO TO 240
           HVM = HVM+WGT(I)*VCG2D(KSEC,IL)
           GO TO 270
170      240      HVM = HVM+WGT(I)*VCG4C(KSEC,IL)
           GO TO 270
           250      ITT = NTIRH(KSEC)-NTIR+IL
           IF (KSZ.GT.4) GO TO 260
           HVM = HVM+WGT(I)*VCG2H(KSEC,ITT)
           GO TO 270
175      260      HVM = HVM+WGT(I)*VCG4H(KSEC,ITT)
           270      CONTINUE
C
C      ---CHECK TO SEE IF THE FEASIBLE RANGE OVERLAPS WITH THE HYPOTHETIC
C      ---POSSIBLE RANGE JUST OBTAINED.
C      -----
185      280      PRINT 540, ACVP,HVM
           IF (LOF.EQ.2) GO TO 310
           IF (HVM.LT.ACVP*2) GO TO 300
           290      PRINT 560
           GO TO 320
           300      PRINT 550
           GO TO 320
           310      IF (HVM.GT.ACVP*1) GO TO 300
           GO TO 290
190      320      IF (LOVE.NE.3) RETURN
           ---FOR TRIM VIOLATION:
           ---INITIALIZE ARRAYS AND VARIABLES.
           -----
           330      LOR = L9AD(2)
           IF (LOVE.EQ.2) LOR = L9AD(1)
           DO 340 I=1,50
           JOONE(I) = 0
195      340      CONTINUE
C
C      ---FOR THE NEXT GROUP:
C      -----
200      460      JGG=1,MGROUP
           IF (JOONE(JGG).EQ.1) GO TO 460
           NSS = NSTG(JGG)
           NLL = NSS+NIG(JGG)-1
           NCIG = NIG(JGG)
           DO 350 N=1,NCIG
           KDONE(N) = 0
205      350      CONTINUE
C
C      ---FOR SINGLE RAY LOADING: LOAD ALL CONTAINERS IN GROUP AND
C      ---COMPUTE LONG. MOMENT CONTRIBUTION.
C      -----
210
215
220
225
230

```

SUBROUTINE STAPLE

73/172 CPT=0 TRACE PMOMP

FTN 4.8+528

```

      IKK = IGCH(JGG)
      IS7 = 400(IKK,10)
      IF (NRBAY(JGG).GT.1) GO TO 390
235      DO 360 KS=1,MSTAGE
            IF (JGG.EQ.JGTP(KS)) GO TO 370
      360      CONTINUE
            GO TO 460
240      370      KRB = KRL(KS)
            TGWT = GHEAN(JGG)*FLOAT(NGIG)
            IF (ISZ.GT.4) GO TO 380
            HLM = HLM+TGWT*CGL2(KRB)
            GO TO 460
245      380      HLM = HLM+T(WT*CGL4(KRB))
            GO TO 460
C
C
C      ---FOR MULTIPLE BAY LOADING, GO TO THE MOST FORWARD UNFILLED RECEI
C      ---BAY.  FILL THIS BAY WITH EITHER THE LIGHTEST OR THE HEAVIEST SUB
C      ---THEN, COMPUTE LONG. MOMENT CONTRIBUTION.
C      -----
250      390      DO 450 K=1,NBAYL
            IF (NLOAC(JGG,K).EQ.0) GO TO 450
            IR = IRAYL(K)
            NQ = NLOAC(JGG,K)
255            RWT = 0.0
            NIM = 0
            400            PWT = 100.
            IF (LOR.EQ.2) PWT = 0.
            DO 430 N=NSS,NLL
                NUM = N-NSS+1
                IF (MSTD(N).EQ.1) GO TO 430
                IF (KTONE(NUM).EQ.1) GO TO 430
                IF (LC7.EQ.2) GO TO 410
                IF (WGT(N).GE.PWT) GO TO 430
265                GO TO 420
            410                IF (WGT(N).LE.PWT) GO TO 430
            420                PWT = WGT(N)
                NOM = NUM
270            430            CONTINUE
                KDOON(NOM) = 1
                NIM = NIM+1
                BWT = RWT+PWT
                IF (NIM.LT.NQ) GO TO 400
                IF (ISZ.GT.4) GO TO 440
275                HLM = HLM+BWT*CGL2(IR)
                GO TO 450
            440                HLM = HLM+BWT*CGL4(IR)
            450            CONTINUE
            460            CONTINUE
C
C
C      ---ASSIGN THE MANUAL LOAD CONTAINERS AND COMPUTE LM CONTRIBUTIONS.
C      -----
285      IF (MPLAN.EQ.0) GO TO 490
      DO 480 M=1,MPLAN
            I = MDC(M)
            LOC = MPCL(M)
            IR = LOC/10000
            KEB = IPST(I)/10000
            KSZ = 400(KEB,10)
290            IF (KSZ.GT.4) GO TO 470
            HLM = HLM+WGT(I)*CGL2(IP)
            GO TO 480
            470            HLM = HLM+WGT(I)*CGL4(IR)
            480            CONTINUE
295      C
      C
      C      ---SEE IF THE LM RANGES OVERLAP.
      C      -----
300      490      POINT 570, ACL,HLM
            IF (LOR.EQ.2) GO TO 520
            IF (HLM.GT.ACL*1) GO TO 510
            500      POINT 530
            GO TO 540
            510      POINT 540
            GO TO 530
305      520      IF (HLM.LT.ACL*2) GO TO 510
            GO TO 503
            530      RETURN
C

```

SUBROUTINE STAPLE 73/172 CPT=0 TRACE PMOMP

FTN 4.8+524

```

310      540 FORMAT (/,45HCURRENT VERT. MOMENT DUE TO ADDED CONTAINERS=,F10.1,/
      1,37HHYPOTHETICALLY POSSIBLE VERT. MOMENT=,F10.1)
      550 FORMAT (/,65HIT IS POSSIBLE TO ATTAIN GM FEASIBILITY THROUGH
      1  REMEDY.)
      560 FORMAT (/,75HIT IS NOT POSSIBLE TO ATTAIN GM FEASIBILITY UNDER THE
      1  CURRENT CONDITIONS.,/,73H PLEASE MAKE CHANGES IN GENERAL LOAD
315      2  PLAN OR CONTAINER COMPOSITION.)
      570 FORMAT (/,45HCURRENT LONG. MOMENT DUE TO ADDED CONTAINERS=,F10.1,/
      1,27HHYPOTHETICALLY POSSIBLE LM=,F10.1)
      580 FORMAT (/,42HIT IS POSSIBLE TO ATTAIN TRIM FEASIBILITY.)
      590 FORMAT (/,77HIT IS NOT POSSIBLE TO ATTAIN TRIM FEASIBILITY UNDER
320      1  EXISTING CONDITIONS.,/,75HPLEASE MAKE CHANGES IN THE GENERAL
      2  LOAD PLAN OR CONTAINER COMPOSITION.)
      END

```



SUBROUTINE REMEDY

73/172 CPT=0 TRACE PMDMP

FTN 4, A+52A

```

1      C      REM
2      SUBROUTINE REMEDY
3      COMMON /OACOM/ ISROW(32,20),JGTP(70),MOETA(70,6),NPICK(70),NOPT(70
5      1),NRIG(32),KRL(70),KOPE
6      COMMON /ALCOM/ LPAT(70,15),NTGR(70),IGCH(32),JYPOS(500)
7      COMMON /FSCOM/ CGL2(40),CGL4(40),KOLL(25,12),KOIN(25,12),NC9L(25),
8      1,MTIRH(15),NHC(25,10),VCG20(15,4),VCG40(15,4),VCG2H(15,10),VCG4H(15
9      2,10),MTOD(25,12),MHCH(40,12),DEKL2(40),DEKL4(40),LCVE,NVIOL,JVIOL
10     3(25),LRAD(25),ACVM1,ACVM2,ACLM1,ACLM2,ACVN,ACLN,NCURE(25,10),WCECK
11     4(25,10)
12     COMMON /AFCOM/ LLOC(500),NCTLD,NSTG(32),MPCL(25),MPC(25),MPLAN,
13     1,KRKEY(40),NRBAY(32),NLQAD(32,30)
14     COMMON /FLCOM/ IPST(500),WGT(500)
15     COMMON /GCOM/ IRAYL(25),NRAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
16     1,JHOLD(40),MGRUP,KTRAK(70)
17     DIMENSION KSHR(10),KJUN(32),MARK(10,10)
18     COMMON /XCOM/ LDONE(200),IDEX(200),ROI(200),ENT(200),HPT(200),ITER
19
20     C      ---INITIALIZE VARIABLES.
21     C      -----
22     C      10 IF (LOVE.EQ.0) GO TO 750
23     C
24     C      ---COMPUTE DEGREE OF VIOLATION FOR EACH TYPE. CHOOSE TO RESOLVE TH
25     C      ---VIOLATION WITH A HIGHER DEGREE.
26     C      -----
27     GO TO (30,500,20), LOVE
28     20 DEN = ABS(ACVM1-ACVM1)
29     ABOV = ABS(ACVM1-ACVM2)
30     IF (LRAD(1).EQ.2) ABOV = ABS(ACVM1-ACVM4)
31     FACT1 = ABOV/DEN
32     DENOM = ABS(ACLM2-ACLM1)
33     ABOVE = ABS(ACLM1-ACLM1)
34     IF (LRAD(2).EQ.2) ABOVE = ABS(ACLM1-ACLM2)
35     FACT2 = ABOVE/DENOM
36     LOR1 = LRAD(1)
37     LOR2 = LRAD(2)
38     IF (FACT2.GT.FACT1) GO TO 500
39
40     C      ---GM CORRECTION.
41     C      -----
42     C      ---PICK A PAIR OF RECEIVING BAYS IN THE SAME HOLO.
43     C      -----
44     30 PRINT 1110
45     LOR1 = LRAD(1)
46     ITER = 1
47     KRUN = 1
48     40 KTGP = 1
49     NEXT = KTGP
50     NSH9 = 0
51     IF (NEXT.GT.1) GO TO 170
52     IF (KTGP.GT.NGROUP) GO TO 100
53     DO 90 IG=KTGP,*GROUP
54     NRB = NRRAY(IG)
55     IF (NRB.LE.1) GO TO 90
56     KTT = 0
57     DO 80 IR=1,NRAYL
58     IF (INLOC(IG,IR).EQ.0) GO TO 80
59     KTT = KTT+1
60     KJ = IRAYL(IR)
61     KHH = JHOLD(KJ)
62     IF (KTT.EQ.1) GO TO 70
63     IF (JHH.NE.KHH) GO TO 70
64     IF (NSHB.EQ.0) GO TO 210
65     DO 60 K=1,NSHB
66     KSH = KSHR(K)
67     KGP = KSH/100
68     KHD = MOD(KSH,100)
69     IF (KGP.NE.IG) GO TO 60
70     IF (KHD.EQ.KHH) GO TO 70
71
72     60 CONTINUE
73     GO TO 210
74     J3 = K9
75     JHH = KHH
76     80 CONTINUE
77     90 CONTINUE

```

SUBROUTINE REMEDY

73/172 (PT=0 TRACE PMOMP

FTN 4.9+528

```

      NEXT = 2
      KST = 1
100  JST = 0
C
C      ---PICK A SINGLE RAY ACCORDING TO THE LOADING SEQUENCE.
C
110  DO 160 IST=KST,4STAGE
      JGG = JGTP(IST)
      JBB = KRL(IST)
      DO 110 K=1,NSH9
          IF ((JGG*100+JHOLD(JBB)).EQ.KSHR(K)) GO TO 160
      CONTINUE
      KEY = KBKEY(JBB)
      IKK = IGCH(JGG)
      KY = KEY/10000000
      ISZ = MOD(IKK,10)
      LOC = KY/10
      ITP = MOD(KY,10)
      GO TO (50,120,130,140,150,490), NEXT
      IF (ISZ.LE.4.OR.LOC.EQ.2) GO TO 160
      JST = IST
      GO TO 190
      IF (ISZ.LE.4.OR.LOC.EQ.1) GO TO 160
      JST = IST
      GO TO 190
      IF (ISZ.GT.4.OR.LOC.EQ.2) GO TO 160
      JST = IST
      GO TO 190
      IF (ISZ.GT.4.OR.LOC.EQ.1) GO TO 160
      JST = IST
      GO TO 190
160  CONTINUE
170  NEXT = NEXT+1
      IF (NEXT.GT.5) GO TO 190
      KST = 1
      GO TO 130
180  IF (KRUN.GE.3) GO TO 490
      KRUN = KRUN+1
      GO TO 40
190  DO 200 I=1,NRAYL
      IF (IRAYL(I).EQ.JBB) JRAY = I
200  CONTINUE
      NSS = NSTG(JGG)
      NLL = NSS+NIG(JGG)-1
      NBB = JBB
      NRAY = JRAY
      KST = IST+1
      GO TO 250
C
C      ---FOR THE SAME HOLD RAYS CHOSEN, GET THE HIGHEST TIER CELL THAT H
C      ---EEN AFFECTED.
C
130  210  KYJ = KBKEY(JB)
      NSS = NSTG(IG)
      NLL = NSS+NIG(IG)-1
      KJ1 = KYJ/1000000
      ISEC = MOD(KJ1,100)
      LOC = KJ1/100
      IKK = IGCH(IG)
      ISZ = MOD(IKK,10)
      DO 220 I=1,NRAYL
          IF (IRAYL(I).EQ.JB) JRI = I
          IF (IRAYL(I).EQ.KB) KRI = I
220  CONTINUE
      KJGP = IG
      NSH9 = NSH9+1
      KSH9(NSH9) = IC*100+JH4
      IF (LOC.EQ.2) GO TO 230
      KBB = JB
      JBB = JB
      KRAY = JRI
      JRAY = KRI
      GO TO 240
145  230  KBB = KB
      JBB = JB
      KRAY = KRI
      JRAY = JRI

```

SUBROUTINE REMEDY 73/172 CPT=0 TRACE PHOMP

FTN 4.8+52A

```

155      240 M9B = K9B
          LAP = 1
          MBAY = K9AY
160      250 DO 260 I=1,10
          DO 260 J=1,10
              MARK(I,J) = 0
165      260 CONTINUE
          KEY = K9KEY(M9B)
          LOC = KEY/1000000000
          KY = MOD(KEY,10000)
170      265 NGOL = KY/100
          NTIR = MOD(KY,100)
          270 DO 300 I=1,NTIR
              IT = NTIR-I+1
              DO 300 IC=1,NGOL
                  IF (MARK(IT,IC).EQ.1) GO TO 300
                  LOKN = M9B*10000+IC*100+IT
                  DO 280 JC=1,NCTLC
                      IF (LOC(JC).EQ.LOKN) GO TO 290
175      280 CONTINUE
                  GO TO 300
180      290 IDO = JC
                  IF (IDO.LT.NSS.OR.IDO.GT.NLL) GO TO 300
                  IF (MPIID(IDO).EQ.1) GO TO 300
                  WTT = WGT(IDO)
185      300 ICC = IC
                  ITT = IT
                  MARK(ITT,ICC) = 1
                  IF (ITER.LE.1) GO TO 320
                  IF (LXAMIN(IDO).GE.1) GO TO 300
                  GO TO 320
190      310 CONTINUE
                  IF (NEXT.GT.1.AND.KST.LE.MSTAGE) GO TO 100
                  IF (NEXT.EQ.1) GO TO 310
                  GO TO 170
195      320 IF (LAP.EQ.2) GO TO 50
          LAP = 2
          M9B = J9B
          MBAY = JBAY
          GO TO 250

C
C
C      ---SEARCH THE CONTAINER GROUP LIST FOR THE ENTERING CONTAINER WITH
C      ---THE MINIMUM RATIO.
C      -----
200      320 JO = JYPOS(IDO)
          DECODE (6,1120,JO) JSROW,JUNK
          STAP = 10000.0
          K1 = KEY/1000000
          ISC = MOD(K1,100)
          DO 430 N=NSS,NLL
205      330 LOK = LLOC(N)
              IF (LOK.EQ.0) GO TO 480
              IB = LOK/10000
              IF (IB.EQ.J9B) GO TO 330
              IF (NEXT.GE.2) GO TO 480
              IF (IB.NE.K9B) GO TO 480
210      330 IF (LOB1.EQ.2) GO TO 480
              IF (WGT(N).GE.WTT) GO TO 480
              GO TO 350
215      340 IF (WGT(N).LE.WTT) GO TO 480
          350 KP = JYPOS(N)
          DECODE (6,1120,KP) KSP0W,JUNK
          JSEK = JSROW/100
          KSEK = KSP0W/100
220      360 IF (KRUN.EQ.1) GO TO 360
              IF (IAIS(JSEK-KSEK).EQ.10) GO TO 370
              IF (JSEK.EQ.KSEK) GO TO 370
225      370 IF (KRUN.LT.3) GO TO 480
              ND = NLIST(JSROW,KSP0W)
              DIST = FLOAT(ND)
              JTIP = MOD(LOK,100)
              WWT = WGT(N)
              LOK1 = LOK/100
              JCOL = MOD(LOK1,100)
              KEM = K9KEY(ITR)
230      LL = KEM/1000000
              MTIR = MOD(KEM,100)

```

SUBROUTINE REMEDY

73/172 CPT=0 TRACE PHOMP

FTN 4.8+528

```

      JSEC = MOD(LL,100)
      JLOC = LL/1000
      IF (LOC.EQ.2) GO TO 330
      IF (ISZ.GT.4) GO TO 380
235      VCG1 = VCG2(ISEC,ITT)
      GL1 = CGL2(I*99)
      GO TO 410
240      380      VCG1 = VCG4(ISEC,ITT)
      GL1 = CGL4(I*99)
      GO TO 410
      390      ITT = MTIRH(ISEC)-MTIR+ITT
      IF (ISZ.GT.4) GO TO 400
      VCG1 = VCG2(ISEC,ITT)
245      GL1 = CGL2(I*99)
      GO TO 410
      400      VCG1 = VCG4(ISEC,ITT)
      GL1 = CGL4(I*99)
250      410      IF (JLOC.EQ.2) GO TO 430
      IF (ISZ.GT.4) GO TO 420
      VCG2 = VCG2(ISEC,JTIR)
      GL2 = CGL2(I*99)
      GO TO 450
255      420      VCG2 = VCG4(ISEC,JTIR)
      GL2 = CGL4(I*99)
      GO TO 450
      430      JJT = MTIRH(JSEC)-MTIR+JTIR
      IF (ISZ.GT.4) GO TO 440
260      VCG2 = VCG2(JSEC,JJT)
      GL2 = CGL2(I*99)
      GO TO 450
      440      VCG2 = VCG4(JSEC,ITT)
      GL2 = CGL4(I*99)
265      C 450      VME = WTT*VCG1+WMT*VCG2
      TME = WTT*GL1+WMT*GL2
      VMN = WMT*VCG1+WTT*VCG2
      TMN = WMT*GL1+WTT*GL2
270      IF (LOB1.EQ.1.AND.VMN.GE.VME) GO TO 480
      IF (LOB1.EQ.2.AND.VMN.LE.VME) GO TO 480
      IF (ITER.EQ.1) GO TO 460
      IF (LXAMN(A).GE.1) GO TO 480
275      460      DELVM = ABS(VMN-VME)
      IF (DELVM.LE.50.) GO TO 480
      IF (LXPIDN(A).EQ.1) GO TO 480
      RATIO = DIS/DELVM
      IF (RATIO.GE.STAR) GO TO 480
      IF (LOVE.EQ.1) GO TO 470
280      IF (LOB2.EQ.1.AND.TMN.LE.TME) GO TO 480
      IF (LOB2.EQ.2.AND.TMN.GE.TME) GO TO 480
      470      STAR = RATIO
      NN = N
      ITER = IR
285      480      CONTINUE
      IF (STAR.GE.10000.0) GO TO 270
      IDEX(ITER) = ICD*10000+NN
      FDI(ITER) = STAR
      WMT(ITER) = VME
      WTT(ITER) = VMN
290      IF (ITER.GE.20) GO TO 490
      ITER = ITER+1
      GO TO 270
295      C ---MAKE EXCHANGES BY CALLING SUBROUTINE XCHANGE.
      C
      490      CALL XCHNGE (1,LOB1)
      IF (LOVE.EQ.1) GO TO 750
      IF (LOVE.EQ.3.AND.FACT2.GT.FACT1) GO TO 750
300      C ---FOR TRIM CORRECTION!
      C ---
      C ---FOR THE NEXT GROUP!
      C ---
305      C
      500      PRINT 1130
      LOA2 = LRA0(1)

```

SUBROUTINE REMEDY 73/172 CPT=0 TRACE PMOMP

FTN 4.8+528

```

310      ITER = 1
      IF (LOVE.EQ.3) LOR2 = L9AD(2)
      KRUN = 1
510 DO 730 JGG=1,MGROUP
      IF (NRBAY(JGG).LE.1) GO TO 730
      IKK = IGCH(JGG)
315      ISZ = MOD(IKK,10)
      NSS = NSTG(JGG)
      NLL = NSS+NTG(JGG)-1
      KK = 0

C
C      ---TAKE THE NEXT MOST FORWARD RECEIVING BAY OF THIS GROUP.
C
      DO 720 K=1,NBAYL
      IF (NLOAC(JGG,K).EQ.0) GO TO 720
      KK = KK+1
      IF (KK.GE.NRBAY(JGG)) GO TO 720
      NUMC = NLOAC(JGG,K)
      KB = IBAYL(K)
      CGE = CGL2(KB)
      KEY = KKEY(KB)
330      KY = KEY/1000000
      KSEC = MCD(KY,100)
      MTIR = MCD(KEY,100)
      KLOC = KY/1000
335      IF (ISZ.GT.4) CGE = CGL4(KB)
      DO 710 NC=1,NUMC
      COMP = 0.0
      IF (LC92.EQ.2) COMP = 100.0

C
C      ---PICK THE HEAVIEST OR THE LIGHTEST CONTAINER IN THIS BAY.
C
340      DO 540 L=NSS,NLL
      IF (LLOC(L).EQ.0) GO TO 540
      LOK = LLOC(L)
      LB = LOK/10000
      IF (LB.NE.MB) GO TO 540
      IF (MPIO(L).EQ.1) GO TO 540
      IF (LOR2.EQ.2) GO TO 520
      IF (WGT(L).LE.COMP) GO TO 540
350      GO TO 510
      IF (WGT(L).GE.COMP) GO TO 540
      COMP = WGT(L)
      LL = L
      LOKK = LOK
540      CONTINUE
      IF (CCMP.EQ.0.OR.COMP.EQ.100.0) GO TO 720
      IF (ITER.EQ.1) GO TO 550
      IF (LXAM(NLL).GE.1) GO TO 710

C
C      ---SEARCH THE GROUP LIST AND PICK THE ONE WITH THE SMALLEST RATIO
C      ---AS THE ENTERING CONTAINER.
C
360      DO 550
      LK1 = LOKK/100
      LCOL = MOD(LK1,100)
      LTIP = MOD(LOKK,100)
365      JYP = JYPOS(LL)
      DECODE (6,1120,JYP) JSROW,JUNK
      STAR = 100.0
      IF (KLOC.EQ.2) GO TO 570
      IF (ISZ.GT.4) GO TO 560
      VC1 = VCG20(KSEC,LTIP)
370      GO TO 590
      VC1 = VCG40(KSEC,LTIP)
      GO TO 590
      JJT = 4TIRH(KSEC)-MTIR+LTIP
375      IF (ISZ.GT.4) GO TO 580
      VC1 = VCG2H(KSEC,JJT)
      GO TO 590
      VC1 = VCG4H(KSEC,JJT)
      GO TO 590
380      DO 700 M=NSS,NLL
      IF (LLOC(M).EQ.0) GO TO 700
      IF (ITER.GE.1) GO TO 600
      IF (LXAMIN(M).GE.1) GO TO 700
      MOK = LLOC(M)
      MB = MCK/10000
385      MLEV = 40C(MOK,100)

```

```

SUBROUTINE REMEDY      73/172  CPT=0 TRACE  PHOMP      FTN 4.8+528

      IF (MB.EQ.KR) GO TO 700
      IF (LOB2.EQ.2) GO TO 610
      IF (WGT(M).GE.COMP) GO TO 700
      GO TO 620
390      IF (WGT(M).LE.COMP) GO TO 700
      CGA = CGL2(MB)
      IF (ISZ.GT.4) CGN = CGL4(MB)
      MEY = KJKEY(M)
335      MY = MEY/1000000
      MSEC = MOD(MY,100)
      JTIR = MOD(MEY,100)
      MFC = MY/1000
      IF (MLOC.EQ.2) GO TO 640
      IF (ISZ.GT.4) GO TO 630
400      VC2 = VCG20(MSEC,MLEV)
      GO TO 660
      VC2 = VCG40(MSEC,MLEV)
      GO TO 650
405      LEV = MTIR*(MSEC)-JTIR*MLEV
      IF (ISZ.GT.4) GO TO 650
      VC2 = VCG20(MSEC,LEV)
      GO TO 660
      VC2 = VCG40(MSEC,LEV)
410      TME = COMP*CGE+WGT(M)*CGN
      TMA = COMP*CGN+WGT(M)*CGE
      VME = COMP*VC1+WGT(M)*VC2
      VMA = COMP*VC2+WGT(M)*VC1
      DELTM = ABS(TME-TMA)
415      IF (DELT.ME.100.) GO TO 700
      IF (LOBE.EQ.2) GO TO 670
      IF (LOB1.EQ.1.AND.VMN.GE.VME) GO TO 700
      IF (LOB1.EQ.2.AND.VMN.LE.VME) GO TO 700
      IF (LOB2.EQ.1.AND.TMN.LE.TME) GO TO 700
      IF (LOB2.EQ.2.AND.TMN.GE.TME) GO TO 700
420      MYF = JYPOS(M)
      DECODE (6,1120,MYP) MSROW,MUNK
      MSEC = MSROW/100
      JSEC = JSROW/100
425      IF (KRUN.EQ.1) GO TO 680
      MSEC = IABS(MSEC-JSEC)
      IF (MSD.EQ.10) GO TO 690
      IF (MSEC.EQ.JSEC) GO TO 690
      IF (KRLN.LT.3) GO TO 700
430      NO = NCIST(JSROW,MSROW)
      DIST = FLCAT(NO)
      RATIO = DIST/DELT.M
      IF (RATIO.GE.STAP) GO TO 700
      IF (MPI2(M).EQ.1) GO TO 700
435      STAR = RATIO
      MM = M
      MCKK = MOK
      MEY = M9
      VMEE = VME
      VMAN = VMN
      TME = TME
      TMN = TMN
440      CONTINUE
      IF (STAR.EQ.100.0) GO TO 710
445      C      INDEX(ITER) = LL*10000+MM
      RDI(ITER) = STAR
      GMT(ITER) = TME
      HMT(ITER) = TMN
      ITER = ITER+1
      IF (ITER.GT.200) GO TO 740
      CONTINUE
710      CONTINUE
720      CONTINUE
730      CONTINUE
      IF (KRUN.GE.3) GO TO 740
455      KRUN = KRUN+1
      GO TO 510
      C
      C      ---MAKE EXCHANGES BY CALLING SUBROUTINE XCHNGE.
      C
460      740 CALL XCHNGE (2,LOB2)
      IF (LOBE.EQ.2) GO TO 730
      IF (LOBE.EQ.3.AND.FACT2.GT.FACT1) GO TO 70

```

```

0000      ---TYPE 3 CORRECTION (CHECK STRENGTH)---
0010 PRINT 1140
0020 DO 760 I=1,NVICL
0030     KOUN(I) = 0
0040 CONTINUE
0050 DO 780 I=1,NVICL
0060     IF (JVIOL(I).NE.3) GO TO 780
0070     IF (KOUN(I).EQ.1) GO TO 780
0080     KOUN(I) = 1
0090     ISAD = LSAD(I)
0100     GO TO 790
0110 CONTINUE
0120 GO TO 940

0130      ---IDENTIFY LOCATION OF TROUBLE.---
0140 JBL = ISAD/100
0150 KOL = MOD(ISAD,100)
0160 IQ = IRAYL(JBL)
0170 ICOL = KOLL(JBL,KOL)
0180 ISZ = KOIN(JBL,KOL)
0190 CLIM = DEKL2(IE)
0200 IF (ISZ.EQ.2) CLIM = DEKL4(IB)
0210 DIFF = WTOP(JBL,KOL)-CLIM
0220 KEY = KRKEY(IE)
0230 NTIR = MOD(KEY,100)
0240 DO 800 NT=1,NTIR
0250     LDONE(NT) = 0
0260 CONTINUE

0270      ---GET THE HEAVIEST CONTAINER IN THIS COLUMN.---
0280 DO 930 NT=1,NTIR
0290     IT = NTIR-NT+1
0300     IF (LDONE(IT).GE.1) GO TO 930
0310     LOKN = I3*IC000+ICCL*100+IT
0320     DO 810 IC=1,NCTLO
0330         IF ((LOC(IC).EQ.LOKN) GO TO 820
0340     CONTINUE
0350     GO TO 930
0360     IF (HPID(IC).EQ.1) GO TO 930
0370     IDD = IC
0380     COMP = WGT(IDD)
0390     ITT = IT
0400     LDONE(ITT) = 1
0410     DO 830 IG=1,MGROUP
0420         IF (NSTG(IG).LE.IDD) GO TO 830
0430         IGG = IG-1
0440         GO TO 840
0450     CONTINUE
0460     IGG = MGROUP

0470      ---FIND THE CONTAINER THAT WILL FIT AND YET NOT VIOLATE THE DECK
0480      ---STRENGTH LIMIT IN ITS OWN COLUMN BY THE EXCHANGE.---
0490 NSS = NSTG(IGG)
0500 NLL = NSSAIG(IGG)-1
0510 IYP = JYPOS(ICC)
0520 DECODE (6,1120,IYP) MSROW,JZZ
0530 MSEK = MSROW/100
0540 LAP = 1
0550 WDEF = 0.0
0560 HWOD = WDEF

0570 DO 910 JC=NSS,NLL
0580     IF ((LOC(JC).EQ.0) GO TO 910
0590     IF (JC.EQ.ICC) GO TO 910
0600     JLCC = LOC(JC)
0610     JBAY = JLCC/10000
0620     JL1 = JLCC/100
0630     JCGL = MCD(JL1,100)
0640     JTIR = MCD(JLCC,100)
0650     IF (JCGL.NE.ICOL) GO TO 860
0660     IF (JBAY.EQ.I3) GO TO 910

```

```

SUPROUTINE REMEDY      73/172  CPT=0 TRACE  P40NP      FTN 4.0+528

540      860      JAY = KKEY(JRAY)
              JLOC = JAY/1000000000
              IF (JLOC.EQ.2) GO TO 900
              DO 880 KP=1,NBAYL
545              IF (IBAYL(KP).NE.JRAY) GO TO 880
              NCOL = NCBL(KP)
              DO 870 KC=1,NCOL
                    IF (KOLL(KP,KC).EQ.JCOL) GO TO 890
870          CONTINUE
880          CONTINUE
550      890      HMOD = WTOP(K9,KC)-WGT(JC)+COMP
              DEK = DEKL2(JRAY)
              IF (IS7.EQ.2) DEK = DEKL4(JRAY)
              IF (HMOD.GT.DEK) GO TO 910
              JYP = JYPOS(JC)
555      900      IF (MPID(JC).EQ.1) GO TO 910
              DECODE (E,1128,JYP) JSROW,JUNK
              JSEK = JSROW/100
              IF (LAP.EQ.1.AND.MSEK.NE.JSEK) GO TO 910
              IF (LAP.EQ.2.AND.IARS(MSEK-JSEK).NE.10) GO TO 910
560      WDI = COMP-WGT(JC)
              IF (WDI.LT.WDEF) GO TO 910
              WDEF = WDI
              JCN = JC
              J99 = J99Y
565      JKOL = JCOL
              JTT = JTTIR
              WWT = WGT(JCN)
              CONTINUE
C 910      IF (HMOD.GT.0.0) GO TO 920
              IF (LAP.GE.2) GO TO 930
              LAP = 2
              GO TO 860
570      920      LLOC(IDD) = J99*10000+JKOL*100+JTT
              LLOC(JCN) = IB*10000+ICOL*100+ITT
              WTOP(JBL,KOL) = WTOP(JBL,KOL)-COMP+WWT
              WTOP(K9,KC) = WTOP(K9,KC)-WWT+COMP
              DIFF = WTOP(JBL,KOL)-OLIM
575      PRINT 1150, (IB*100+KOL),IDD,JCN,WTOP(JBL,KOL)
              IF (DIFF.LE.0.5) GO TO 770
580      930      CONTINUE
              PRINT 1160, TCCL,IP
              GO TO 770
C
C 940      PRINT 1170
C 950      DO 960 I=1,NVICL
              IF (JVIOL(I).NE.4) GO TO 960
              IF (KDUN(I).EQ.1) GO TO 960
              KDUN(I) = 1
              I9AD = LBAO(I)
              GO TO 970
590      960      CONTINUE
              RETURN
595      970      JBL = I9AD/100
              KOL = MOD(I9AD,100)
              IB = IBAYL(JBL)
              ICOL = KOLL(JBL,KOL)
600      IS7 = KOIN(JBL,KOL)
              MAX = MHCH(IA,ICOL)
              NHN = MHC(JBL,KOL)
              NOVER = NHN-MA
              KOUNT = 0
605      IT1 = 1
              KEY = KKEY(IA)
              NTIR = MOD(KEY,100)
              980      DO 1010 IT=IT1,NTIR
                    LOKN = IA*10000+ICOL+IT
                    DO 990 IC=1,NCTLD
                          IF (LLOC(IC).EQ.LOKN) GO TO 1000
990          CONTINUE
              GO TO 1010
610      1000      ID = IC
              IF (MPID(ID).EQ.1) GO TO 1010
              IHC = MOD(IFST(ID),10)

```



SUBROUTINE REMEDY

73/172 CPT=0 TRACE PMDMP

FTN 4.8+529

```

        IF (IHC.NE.1) GO TO 1010
        IDO = ID
        ITT = IT
620      GO TO 1020
        1010 CONTINUE
        GO TO 1030
        1020 DO 1030 IG=1,MGROUP
        IF (INSTG(IG).GE.IDO) GO TO 1040
625      1030 CONTINUE
        1040 IGG = IG-1
        NSS = NSTG(IGG)
        NLL = NSS+NIG(IGG)-1
        LAP = 1
630      IYP = JYPOS(ITT)
        DECODE (6,1120,IYP) MSROW,JUNK
        1050 OUST = 500.0
C
C      ---FIND THE CONTAINER IN THE SAME GROUP THAT CAN REPLACE THE
C      ---HIGH CUBE IF THIS COLUMN.
C      -----
        DO 1080 JC=NSS,NLL
        IF (LLOC(JC).EQ.0) GO TO 1080
        IF (JC.EQ.IDO) GO TO 1080
640      JHC = MOD(IST(JC),10)
        IF (JHC.EQ.1) GO TO 1080
        JLOC = LLOC(JC)
        JRAY = JLOC/10000
645      IF (LAP.EQ.2) GO TO 1060
        IF (JRAY.NE.18) GO TO 1080
        1060 JLI = JLOC/100
        JCOL = MOD(JLI,100)
        JTIR = MOD(JLOC,100)
        IF (JCOL.NE.ICOL) GO TO 1070
        IF (LAP.EQ.2.AND.JRAY.EQ.18) GO TO 1080
650      IF (LAP.EQ.1) GO TO 1080
        1070 JYP = JYPOS(JC)
        DECODE (6,1120,JYP) JSROW,JUNK
        DIST = FLOAT(NOIST(MSROW,JSROW))
655      IF (DIST.GE.OUST) GO TO 1080
        IF (MPID(JC).EQ.1) GO TO 1080
        OUST = DIST
        JCN = JC
        JRB = JRAY
        JKOL = JCOL
        JTT = JTIR
        1080 CONTINUE
        IF (OUST.LT.500.0) GO TO 1100
        IF (LAP.EQ.2) GO TO 1090
665      LAP = 2
        GO TO 1050
        1090 PRINT 1190, KCL,18
        GO TO 950
        1100 LLOC(IDO) = JRB*10000+JKOL*100+JTT
        LLOC(JCN) = 18*10000+ICOL*100+ITT
        NHC(JAL,KOL) = NPC(JBL,KOL)-1
        POINT 1150, (18*100+KOL),IDO,JCN
        KOUNT = KOUNT+1
675      IF (KOUNT.LT.NCVER) GO TO 990
        GO TO 950
C
C
        1110 FORMAT (///.69H*****ENTERING REMEDY ROUTINE FOR VERTICAL MOMENT
        1 CORRECTION****.//)
680      1120 FORMAT (I4,R2)
        1130 FORMAT (///.55H*****ENTERING REMEDY ROUTINE FOR TRIM CORRECTION****
        1 * .//)
        1140 FORMAT (///.49H*****ENTERING DECK STRENGTH VIOLATION REMEDY****.//)
685      1150 FORMAT (11H3AD COLUMN=,I5,3X,8HLEAVING=,I5,3X,9HENTERING=,I5,
        113HNEW STACK HT=,F5.1)
        1160 FORMAT (3X,.2HIMPOSSIBLE TO CORRECT DECK ST VIOLATION IN,5X,
        18H COLUMN:,I4,3X,4HJAY:,I4)
        1170 FORMAT (///.55H*****ENTERING REMEDY ROUTINE FOR HIGH CUBE EXCESS**
        1 * .//)
690      1180 FORMAT (53HIMPOSSIBLE TO CORRECT THE HIGH CUBE SITUATION IN COL.,
        1I4,3X,4HJAY:,I4)
        END

```

SUBROUTINE XCHNGE

73/172 CPT=0 TRACE P4DMP

FTN 4.8+528

```

1      C      XCH
          SUBROUTINE XCHNGE (KOL,LOB)
5      COMMON /XCOM/ LOONE(200),IDEX(200),ROI(200),EMT(200),HMT(200),ITER
          COMMON /AFCON/ LLOC(500),NCTLO,NSTGT(32),MPCL(25),MPC(25),MPLAN,
          1K9KEY(40),MRRAY(32),NLCAD(32,30)
          COMMON /GCON/ NBAYL(25),NBAYL,MSTAGE,GMEAN(32),NIG(32),NCEL(40,12)
10     COMMON /JHOLD(40),JGRUP,KTRAN(70)
          COMMON /FSCOM/ CGL2(40),CGL4(40),KOLL(25,12),KOIN(25,12),NCRL(25),
          1MTIRH(15),NHC(25,10),VCG2D(15,4),VCG4D(15,4),VCG2H(15,10),VCG4H(15
          2,10),M10D(25,12),MHCH(40,12),DEKL2(40),JEKL4(40),LOVE,NVIDL,JVIDL
          3(25),L1AD(25),ACVM1,ACVM2,ACLM1,ACLM2,ACVM,ACLM,NCURE(25,10),WCECK
          4(25,10)
15     C      C      C      C      C
          C      ---THIS SUBROUTINE MAKES PAIOWISE EXCHANGES OF CONTAINER ASSIGNMEN
          C      ---USING D/I RATIOS OF SELECTED CANDIDATE PAIRS.  EXCHANGES OCCUR
          C      ---THE SMALLEST RATIO TOWARDS LARGER ONES UNTIL STABILITY CONSTRAI
          C      ---ARE SATISFIED.
          C      -----
20     NTER = ITER-1
          DO 10 NX=1,ITER
              LOONE(NX) = 0
10     CONTINUE
20     COMP = 1000.
          DO 30 NX=1,NTER
              IF (LOONE(NX).GE.1) GO TO 30
              IF (ROI(NX).GT.COMP) GO TO 30
              COMP = ROI(NX)
              NXX = NX
30     CONTINUE
              IF (COMP.EQ.1000.) GO TO 60
              LOONE(NXX) = 1
              IOX1 = IDEX(NXX)/10000
              IOX2 = MOD(IDEX(NXX),10000)
35     LOC1 = LLOC(IOX1)
              LOC2 = LLOC(IOX2)
              LLOC(IOX1) = LOC2
              LLOC(IOX2) = LOC1
              IF (KOL.EQ.1) ACVM = ACVM-EMT(NXX)+HMT(NXX)
              IF (KOL.EQ.2) ACLM = ACLM-EMT(NXX)+HMT(NXX)
              IF (KOL.EQ.2) GO TO 70
              PRINT 100, ITER,ACVM,IOX1,IOX2,ROI(NXX)
              IF (LOB.EQ.2) GO TO 40
              IF (ACVM.LT.ACVM2) GO TO 50
              GO TO 20
40     IF (ACVM.GT.ACVM1) GO TO 50
              GO TO 20
50     PRINT 110
              GO TO 90
60     PRINT 120
              GO TO 90
70     IF (LOB.EQ.2) GO TO 80
              IF (ACLM.GT.ACLM1) GO TO 50
              GO TO 20
55     PRINT 130, ITER,ACLM,IOX1,IOX2,ROI(NXX)
              IF (ACLM.LT.ACLM2) GO TO 50
              GO TO 20
90     RETURN
60     C
100    FORMAT (3X,5HITER=,I3,3HV4=,F11.1,3X,8HLEAVING=,I4,3X,9HENTERING=,
          1I4,3X,10HDI RATIO=,F10.2)
110    FORMAT (/,3X,3CHFEASIBILITY HAS BEEN ATTAINED.)
120    FORMAT (/,3X,3HFEASIBILITY IS NOT POSSIBLE TO ATTAIN,/,53HUNDER T
          1HE PRESENT SITUATION.  IT IS ADVISED TO MODIFY,/,25HTHE GENERAL LO
          2ADING PLAN.)
65    130    FORMAT (3X,5HITER=,I3,3HLM=,F11.1,3X,8HLEAVING=,I4,3X,9HENTERING=,
          1I4,3X,10HDI RATIO=,F10.2)
          END

```

FUNCTION MPID            73/172   CPT=0 TRACE   PMOMP            FTN 4.2+528

```

1      C      MPID
          FUNCTION MPID (NUM)
5      COMMON /AFCON/ LLOC(500),NCTLD,NSTG(32),MPCL(25),MPC(25),MPLAN,
          1K1KEY(40),NR3AY(32),NLOAD(32,30)
          C      --- THIS FUNCTION RETURNS VALUE 0 IF NUM IS NOT A MANUAL PLAN CONTA
          C      --- AND RETURNS VALUE 1 OTHERWISE.
          C      -----
10     MPID = 0
          DO 10 MP=1,MPLAN
              IF (NUM.NE.MPC(MP)) GO TO 10
              MPID = 1
              RETURN
15     CONTINUE
          RETURN
          END

```

FUNCTION LXAMIN           73/172   CPT=0 TRACE   PMOMP            FTN 4.2+528

```

1      C      LXA
          FUNCTION LXAMIN (KOM)
5      COMMON /XCON/ LDONE(200),IDEX(200),ROI(200),EMT(200),HMT(200),ITER
          C      --- THIS FUNCTION EXAMINES TO SEE IF A CONTAINER HAS BEEN
          C      --- INCLUDED IN THE EXCHANGE CANDIDATE LIST PREVIOUSLY.
          C      -----
10     LXAMIN = 0
          NTER = ITER-1
          DO 10 NT=1,NTER
              IOX1 = IDEX(NT)/10000
              IOX2 = MOD(IDEX(NT),10000)
              IF (KOM.NE.IOX1.AND.KOM.NE.IOX2) GO TO 10
              LXAMIN = 1
              RETURN
15     CONTINUE
          RETURN
          END

```

## APPENDIX E

A Sample Output of COMCL Run1. DATPREP Output (Test #1)

## \*\*\*OPERATOR RECAPITULATION\*\*\*

| LINE   | KOBE |    |    |    |     |    |    |    | NAGOYA |    |    |    |    |    |    |    | SHIMIZU |    |    |    |    |    |    |    | YOKOHAMA |    |    |    |     |    |    |    | TOTALS |
|--------|------|----|----|----|-----|----|----|----|--------|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|----|----------|----|----|----|-----|----|----|----|--------|
|        | 2F   | 2R | 2S | 2M | 4F  | 4R | 4S | 4M | 2F     | 2R | 2S | 2M | 4F | 4R | 4S | 4M | 2F      | 2R | 2S | 2M | 4F | 4R | 4S | 4M | 2F       | 2R | 2S | 2M | 4F  | 4R | 4S | 4M |        |
| 1      | 17   | 0  | 0  | 0  | 21  | 0  | 0  | 0  | 4      | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 8        | 0  | 0  | 0  | 43  | 0  | 0  | 0  | 93     |
| 2      | 0    | 0  | 0  | 0  | 0   | 4  | 0  | 0  | 0      | 0  | 0  | 0  | 2  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 14 | 0  | 0  | 0  | 1        | 0  | 0  | 0  | 16  | 10 | 0  | 0  | 47     |
| 3      | 2    | 0  | 0  | 0  | 2   | 1  | 0  | 0  | 0      | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 5        | 0  | 0  | 0  | 4   | 0  | 0  | 0  | 14     |
| 4      | 6    | 0  | 0  | 0  | 0   | 2  | 0  | 0  | 0      | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0        | 0  | 0  | 0  | 10  | 0  | 0  | 0  | 18     |
| 5      | 5    | 0  | 0  | 0  | 1   | 3  | 0  | 0  | 0      | 0  | 0  | 0  | 1  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0        | 0  | 0  | 0  | 4   | 1  | 0  | 0  | 15     |
| 6      | 38   | 0  | 0  | 0  | 1   | 1  | 0  | 0  | 0      | 0  | 0  | 0  | 4  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 15       | 0  | 0  | 0  | 14  | 1  | 0  | 0  | 74     |
| SUBT   | 68   | 0  | 0  | 0  | 25  | 11 | 0  | 0  | 4      | 0  | 0  | 0  | 7  | 0  | 0  | 0  | 0       | 0  | 0  | 0  | 14 | 0  | 0  | 0  | 29       | 0  | 0  | 0  | 91  | 12 | 0  | 0  |        |
| TOTALS |      |    |    |    | 104 |    |    |    |        |    |    |    | 11 |    |    |    |         |    |    |    | 14 |    |    |    |          |    |    |    | 132 |    |    |    | 261    |

## \*\*\*\*\*TONNAGE BY LINE\*\*\*\*\*

SHIP: J APOLLO

VOYAGE NO: 5

DATE: 8/15/80

20FT AVERAGE WEIGHT: 16.0

40FT AVERAGE WEIGHT: 23.5

| ***20 FT CONTAINERS*** |        |        |       |          |            |         |        |
|------------------------|--------|--------|-------|----------|------------|---------|--------|
| PORT                   | J LINE | K LINE | NISSK | NYK LINE | SHOWA LINE | YS LINE | TOTALS |
| KURE                   | 295.5  | 0.0    | 24.4  | 112.3    | 96.1       | 577.1   | 1105.4 |
| NAGOYA                 | 81.1   | 0.0    | 0.0   | 0.0      | 0.0        | 0.0     | 81.1   |
| SHIMIZU                | 0.0    | 0.0    | 0.0   | 0.0      | 0.0        | 0.0     | 0.0    |
| YOKOHAMA               | 137.0  | 11.3   | 67.8  | 0.0      | 0.0        | 215.9   | 432.0  |
| 20' LOAD               | 513.6  | 11.3   | 92.2  | 112.3    | 96.1       | 793.0   | 1618.5 |

| ***40 FT CONTAINERS*** |        |        |       |          |            |         |        |
|------------------------|--------|--------|-------|----------|------------|---------|--------|
| PORT                   | J LINE | K LINE | NISSK | NYK LINE | SHOWA LINE | YS LINE | TOTALS |
| KURE                   | 494.2  | 91.9   | 69.3  | 46.4     | 85.4       | 48.4    | 835.6  |
| NAGOYA                 | 0.0    | 43.6   | 0.0   | 0.0      | 22.8       | 97.5    | 163.9  |

|             |        |        |       |       |       |        |        |
|-------------|--------|--------|-------|-------|-------|--------|--------|
| SHIMIZU     | 0.0    | 326.1  | 0.0   | 0.0   | 0.0   | 0.0    | 326.1  |
| YOKOHAMA    | 1028.4 | 612.4  | 99.9  | 251.1 | 100.8 | 340.1  | 2432.7 |
| 40' LOAD    | 1522.6 | 1074.0 | 169.2 | 297.5 | 209.0 | 486.0  | 3758.3 |
| 20' EMPTY   | 0.0    | 0.0    | 0.0   | 0.0   | 0.0   | 0.0    | 0.0    |
| 40' EMPTY   | 0.0    | 0.0    | 0.0   | 0.0   | 0.0   | 0.0    | 0.0    |
| GRAND TOTAL | 2036.2 | 1085.3 | 261.4 | 409.8 | 305.1 | 1279.0 | 5376.8 |

## \*\*\*CONTAINER GROUP DATA\*\*\*

TOTAL NO. TO LOAD: 261

TOTAL WEIGHT(KT):5376.8

| SEQ NO. | IPST | YPOS | LCMC | IDC | WGT |
|---------|------|------|------|-----|-----|
| *****   | **** | **** | **** | *** | *** |

| GP NO: | 1 | PORT:KORE | TYPE:20F | AV. WT: 16.3 |
|--------|---|-----------|----------|--------------|
|--------|---|-----------|----------|--------------|

|    |    |        |      |            |      |
|----|----|--------|------|------------|------|
| 1  | 11 | 4201CC | 6010 | YSAA668899 | 12.1 |
| 2  | 11 | 4201DC | 6010 | YSFF674248 | 15.2 |
| 3  | 11 | 4201DB | 6010 | YSFF674147 | 15.7 |
| 4  | 11 | 4203AC | 6010 | YSAA660642 | 14.5 |
| 5  | 11 | 4203BB | 6010 | YSFF674145 | 15.6 |
| 6  | 11 | 4203CC | 6010 | YSFF674423 | 15.7 |
| 7  | 11 | 4209AC | 6100 | YSFF668894 | 14.9 |
| 8  | 11 | 4209AB | 6101 | YSFF674343 | 15.3 |
| 9  | 11 | 4209BB | 6100 | YSAA668795 | 14.7 |
| 10 | 11 | 4209CC | 6100 | YSAA668018 | 14.4 |
| 11 | 11 | 4209CB | 6100 | YSAA668714 | 14.1 |
| 12 | 11 | 4209DB | 6101 | YSFF670555 | 15.4 |
| 13 | 11 | 4209EC | 6100 | YSAA660634 | 13.9 |
| 14 | 11 | 4209EB | 6100 | YSAA660689 | 14.3 |
| 15 | 11 | 4209FC | 6101 | YSFF674355 | 15.3 |
| 16 | 11 | 4209FB | 6101 | YSFF674212 | 15.1 |
| 17 | 11 | 4215AB | 6100 | YSAA668004 | 14.8 |
| 18 | 11 | 4215BC | 6010 | YSAA668764 | 14.6 |
| 19 | 11 | 4215BB | 6100 | YSAA660684 | 14.7 |
| 20 | 11 | 4215CC | 6100 | YSAA660956 | 14.6 |
| 21 | 11 | 4215LB | 6100 | YSAA660931 | 14.6 |
| 22 | 11 | 4215BC | 6100 | YSAA668857 | 15.3 |
| 23 | 11 | 4215BB | 6010 | YSAA668791 | 15.2 |
| 24 | 11 | 4215BC | 6010 | YSAA668812 | 15.7 |
| 25 | 11 | 4215EB | 6010 | YSAA668794 | 14.9 |
| 26 | 11 | 4215FI | 6010 | YSFF668954 | 14.8 |

|    |    |        |      |            |      |
|----|----|--------|------|------------|------|
| 27 | 11 | 4215FL | 6010 | TSAA668818 | 15.2 |
| 28 | 11 | 4215FH | 6010 | TSAA668874 | 16.1 |
| 29 | 11 | 4257CT | 1010 | JLAU265323 | 15.5 |
| 30 | 11 | 4257CC | 1010 | JLAU265382 | 16.4 |
| 31 | 11 | 4257DF | 6010 | TSAA668780 | 15.0 |
| 32 | 11 | 4257DC | 6010 | TSAA668766 | 15.6 |
| 33 | 11 | 4257DB | 6010 | TSAA668728 | 15.7 |
| 34 | 11 | 5203ET | 4011 | NLSU262148 | 20.1 |
| 35 | 11 | 5203EC | 4011 | SCPU228197 | 19.6 |
| 36 | 11 | 5203EB | 4011 | MTKU110155 | 15.6 |
| 37 | 11 | 5203FC | 4010 | MTKEY119   | 19.1 |
| 38 | 11 | 5203FB | 4010 | B20Y8      | 18.2 |
| 39 | 11 | 5205AH | 6010 | TSFF674152 | 12.4 |
| 40 | 11 | 5205BH | 1010 | JLAU265689 | 16.1 |
| 41 | 11 | 5205BB | 1010 | JLAU265310 | 15.0 |
| 42 | 11 | 5205EC | 1010 | JLAU265035 | 14.6 |
| 43 | 11 | 5237CB | 3010 | NULU276455 | 12.1 |
| 44 | 11 | 5237DB | 6010 | JLLU265145 | 15.1 |
| 45 | 11 | 5237EB | 1010 | JLAU265510 | 8.4  |
| 46 | 11 | 5237FB | 3011 | ILCU/85704 | 12.3 |
| 47 | 11 | 5237GB | 5011 | SLBK286000 | 20.2 |
| 48 | 11 | 5237HB | 5101 | SLBK286026 | 20.0 |
| 49 | 11 | 5237IB | 4011 | MTKU172596 | 19.7 |
| 50 | 11 | 5747BB | 6010 | TSAA668711 | 18.5 |
| 51 | 11 | 5751AC | 6010 | TSAA668881 | 14.5 |
| 52 | 11 | 5753AB | 6010 | TSFF674046 | 15.0 |
| 53 | 11 | 5757AB | 1010 | JLLU210145 | 17.9 |
| 54 | 11 | 5757BB | 1010 | NILU343500 | 18.4 |
| 55 | 11 | 5757CB | 1010 | IULU281822 | 18.9 |
| 56 | 11 | 5757DB | 1010 | NICB/8439  | 18.5 |
| 57 | 11 | 5757EB | 1010 | IULU260451 | 18.5 |
| 58 | 11 | 5757FB | 1010 | JLLU261359 | 18.0 |
| 59 | 11 | 5759AB | 1010 | JLLU665071 | 19.1 |
| 60 | 11 | 5759BB | 1010 | TSFF674280 | 19.9 |
| 61 | 11 | 5759CB | 1010 | TSFF668827 | 20.2 |
| 62 | 11 | 5759DB | 1010 | JLLU665054 | 20.4 |
| 63 | 11 | 5759EB | 1010 | JLAU665542 | 19.7 |
| 64 | 11 | 5763AB | 5010 | IULU279989 | 18.7 |
| 65 | 11 | 5763BB | 5010 | IULU281092 | 18.8 |
| 66 | 11 | 5763CB | 6010 | IULU281745 | 18.6 |
| 67 | 11 | 5763DB | 6010 | IULU284285 | 19.1 |
| 68 | 11 | 5763EB | 5010 | IULU288159 | 18.4 |

2 FORT:KUBE TYPE:40F AV. WT: 22.8

|    |    |        |      |            |      |
|----|----|--------|------|------------|------|
| 69 | 15 | 5214AB | 1010 | XTRU880758 | 24.3 |
| 70 | 15 | 5214BC | 1010 | JLAU205231 | 24.8 |
| 71 | 15 | 5214BB | 1010 | XTRU880835 | 24.8 |
| 72 | 15 | 5214CC | 1010 | SSIU205048 | 24.0 |
| 73 | 15 | 5214CB | 1010 | JLLU206196 | 24.8 |

|    |    |        |      |            |      |
|----|----|--------|------|------------|------|
| 74 | 15 | 5214DC | 1010 | XTRU880721 | 23.7 |
| 75 | 15 | 5214DB | 1010 | JLAU202034 | 25.6 |
| 76 | 15 | 5214EC | 1010 | SSIU205076 | 25.9 |
| 77 | 15 | 5214EB | 1010 | JLLU202094 | 25.1 |
| 78 | 15 | 5214FB | 1010 | SSIU202604 | 25.3 |
| 79 | 15 | 5218AB | 1010 | SSIU202545 | 24.9 |
| 80 | 15 | 5218BB | 1010 | SSIU202628 | 25.1 |
| 81 | 15 | 5218CC | 1010 | XTRU880830 | 15.2 |
| 82 | 15 | 5218CB | 1010 | JLLU206211 | 25.0 |
| 83 | 15 | 5218DC | 1010 | NICU110591 | 24.7 |
| 84 | 15 | 5218DB | 1010 | SSIU202712 | 25.1 |
| 85 | 15 | 5218EC | 1010 | SSIU202655 | 24.8 |
| 86 | 15 | 5218EB | 1010 | JLAU204064 | 17.6 |
| 87 | 15 | 5218FC | 1010 | JLLU201175 | 25.0 |
| 88 | 15 | 5218FB | 1010 | JLLU204142 | 24.4 |
| 89 | 15 | 5250BB | 5010 | SSIU220228 | 12.3 |
| 90 | 15 | 5250EB | 1010 | XTRU880774 | 17.1 |
| 91 | 15 | 5260FB | 3010 | NULU205383 | 20.5 |
| 92 | 15 | 5272AB | 3010 | ILU504558  | 23.8 |
| 93 | 15 | 5272BB | 6010 | USLU205046 | 18.9 |

GP NO: 3 FORT:KUBE TYPE:40K AV. WT: 24.2

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 94  | 16 | 5460BB | 6020 | TSLU503009 | 29.5 |
| 95  | 16 | 5464AB | 4020 | MTKU772076 | 23.0 |
| 96  | 16 | 5464BC | 3020 | NULU500045 | 25.0 |
| 97  | 16 | 5464CC | 5020 | SLKF486007 | 24.0 |
| 98  | 16 | 5464CB | 5020 | SLKF486028 | 25.3 |
| 99  | 16 | 5464DC | 5020 | SLKF486014 | 23.8 |
| 100 | 16 | 5464DB | 2020 | ITSU500117 | 22.7 |
| 101 | 16 | 5464EC | 2020 | KKLU504342 | 23.1 |
| 102 | 16 | 5464EB | 2020 | KKLU504256 | 23.1 |
| 103 | 16 | 5464FC | 4020 | MTKU781525 | 23.4 |
| 104 | 16 | 5464FB | 2020 | KKLU504020 | 23.0 |

GP NO: 4 FORT:NAUTA TYPE:20F AV. WT: 20.5

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 105 | 21 | 5761AB | 1010 | JLLU661090 | 20.3 |
| 106 | 21 | 5761BB | 1010 | TSFF668897 | 20.2 |
| 107 | 21 | 5761CB | 1010 | JLAU665515 | 20.3 |
| 108 | 21 | 5761DB | 1010 | JLLU665064 | 20.3 |

GP NO: 5 FORT:NAUTA TYPE:40F AV. WT: 23.4

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 109 | 25 | 5264BB | 6010 | TSLU207027 | 22.0 |
| 110 | 25 | 5264CC | 5010 | CIU099382  | 22.8 |
| 111 | 25 | 5264CB | 2010 | KKLU201335 | 21.5 |
| 112 | 25 | 5264FB | 6010 | FECU340133 | 27.0 |

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 113 | 25 | 5264FB | 6010 | YSLU203027 | 21.8 |
| 114 | 25 | 5268BB | 2010 | KKLU204004 | 22.1 |
| 115 | 25 | 5268BB | 6010 | YSLU209257 | 26.7 |

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 156 | 41 | 5755CB | 6010 | YSAA660754 | 15.4 |
| 157 | 41 | 5755DB | 6010 | YSFF674064 | 15.4 |
| 158 | 41 | 5755EB | 6010 | YSAA660672 | 14.2 |

6 PORT:SHIMIZU TYPE:40F AV. WT: 23.3

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 116 | 35 | 5208AC | 2010 | ITSU202306 | 23.3 |
| 117 | 35 | 5208AB | 2010 | KKLU204735 | 23.0 |
| 118 | 35 | 5208BC | 2010 | ITSU202352 | 22.9 |
| 119 | 35 | 5208BB | 2010 | KKLU204941 | 23.5 |
| 120 | 35 | 5208CC | 2010 | ITSU203707 | 22.8 |
| 121 | 35 | 5208CB | 2010 | ITSU202389 | 22.9 |
| 122 | 35 | 5208DC | 2010 | KKLU201128 | 23.3 |
| 123 | 35 | 5208DB | 2010 | CTIU415904 | 24.4 |
| 124 | 35 | 5208ET | 2010 | KKLU201396 | 23.0 |
| 125 | 35 | 5208EC | 2010 | SSIU217439 | 24.5 |
| 126 | 35 | 5208EB | 2010 | KKLU204952 | 23.4 |
| 127 | 35 | 5208FT | 2010 | NYLU201131 | 23.2 |
| 128 | 35 | 5208FC | 2010 | ITSU203319 | 22.9 |
| 129 | 35 | 5208FB | 2010 | NYLU201299 | 23.0 |

7 PORT:YOKOHAMA TYPE:20F AV. WT: 14.9

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 130 | 41 | 4201CB | 6010 | YSAA668717 | 15.0 |
| 131 | 41 | 4203AB | 6010 | YSFF671464 | 15.0 |
| 132 | 41 | 4203BC | 6010 | YSFF640717 | 15.2 |
| 133 | 41 | 4203CB | 6010 | YSFF67431H | 15.8 |
| 134 | 41 | 4257CB | 1010 | JLAU262656 | 16.6 |
| 135 | 41 | 4273BB | 5010 | JLAU263464 | 12.1 |
| 136 | 41 | 4273CB | 1010 | SCXU629864 | 12.2 |
| 137 | 41 | 4273DB | 6010 | YSAA275160 | 20.1 |
| 138 | 41 | 4273FC | 2010 | KKLU261042 | 11.3 |
| 139 | 41 | 4273FB | 6010 | YSAA266714 | 9.5  |
| 140 | 41 | 5205FC | 1010 | JLAU261848 | 18.4 |
| 141 | 41 | 5205FB | 1010 | JLAU665086 | 17.4 |
| 142 | 41 | 5243EC | 3010 | MOLU277849 | 14.7 |
| 143 | 41 | 5243EB | 3010 | MOLU276466 | 15.3 |
| 144 | 41 | 5243FC | 3010 | MOLU277818 | 15.2 |
| 145 | 41 | 5243FB | 1010 | MOLU277131 | 16.2 |
| 146 | 41 | 5749AB | 6010 | YSAA668895 | 7.5  |
| 147 | 41 | 5749CB | 3010 | UFUJ385256 | 6.4  |
| 148 | 41 | 5753AF | 6010 | YSFF674338 | 15.0 |
| 149 | 41 | 5753AC | 1010 | NICU709583 | 15.3 |
| 150 | 41 | 5753AB | 1010 | JLAU261792 | 17.1 |
| 151 | 41 | 5753BF | 6010 | YSFF668940 | 15.5 |
| 152 | 41 | 5753BB | 1010 | JLLU661099 | 20.1 |
| 153 | 41 | 5753CB | 1010 | YSFF674092 | 19.9 |
| 154 | 41 | 5755BF | 6010 | YSAA668725 | 15.1 |
| 155 | 41 | 5755BB | 6010 | YSAA668785 | 15.1 |

8 PORT:YOKOHAMA TYPE:40F AV. WT: 23.7

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 159 | 45 | 4212DC | 6010 | YSLU299028 | 28.8 |
| 160 | 45 | 4218BC | 2010 | ITSU202094 | 23.7 |
| 161 | 45 | 4218BB | 2010 | ITSU202430 | 23.5 |
| 162 | 45 | 4218CC | 2010 | KKLU205177 | 20.3 |
| 163 | 45 | 4218CB | 2010 | KKLU204979 | 24.7 |
| 164 | 45 | 4218DC | 2010 | ITSU202222 | 27.5 |
| 165 | 45 | 4218DB | 2010 | ITSU202112 | 23.4 |
| 166 | 45 | 4218EC | 2010 | KKLU204164 | 23.6 |
| 167 | 45 | 4218EB | 2010 | ITSU202145 | 24.5 |
| 168 | 45 | 4218FC | 2010 | KKLU201381 | 23.9 |
| 169 | 45 | 4218FB | 2010 | ITSU202716 | 23.7 |
| 170 | 45 | 4224AB | 1010 | NICC11540  | 14.8 |
| 171 | 45 | 4224BT | 1010 | JLLU204022 | 24.4 |
| 172 | 45 | 4224BC | 1010 | JLLU204026 | 24.6 |
| 173 | 45 | 4224BB | 1010 | JLLU210157 | 21.2 |
| 174 | 45 | 4224CT | 1010 | SCIU411502 | 25.5 |
| 175 | 45 | 4224CC | 1010 | SSIU224771 | 25.9 |
| 176 | 45 | 4224CB | 1010 | NYLU201209 | 24.8 |
| 177 | 45 | 4224DT | 1010 | JLLU202018 | 24.5 |
| 178 | 45 | 4224DC | 1010 | SSIU205057 | 24.8 |
| 179 | 45 | 4224DB | 1010 | NICC110503 | 25.1 |
| 180 | 45 | 4224ET | 1010 | JLAU202035 | 26.3 |
| 181 | 45 | 4224EC | 1010 | JLLU206240 | 24.8 |
| 182 | 45 | 4224EB | 1010 | JLLU204021 | 24.9 |
| 183 | 45 | 4224FT | 1010 | NICA1361   | 25.0 |
| 184 | 45 | 4224FC | 1010 | XTRU880717 | 24.1 |
| 185 | 45 | 4224FB | 1010 | UFUJ214439 | 24.3 |
| 186 | 45 | 4228AB | 1010 | JLAU202055 | 25.1 |
| 187 | 45 | 4228BT | 1010 | JLLU206251 | 24.9 |
| 188 | 45 | 4228BC | 1010 | XTRU880798 | 24.3 |
| 189 | 45 | 4228BB | 1010 | JLLU202054 | 24.9 |
| 190 | 45 | 4228CT | 1010 | NICC110546 | 24.9 |
| 191 | 45 | 4228CB | 1010 | XTRU880749 | 24.4 |
| 192 | 45 | 4228CF | 1010 | JLAU205251 | 24.7 |
| 193 | 45 | 4228CC | 1010 | SSIU205061 | 24.7 |
| 194 | 45 | 4228CB | 1010 | NICA1476   | 24.7 |
| 195 | 45 | 4228BC | 1010 | JLLU202949 | 25.3 |
| 196 | 45 | 4228BB | 1010 | FECU340345 | 25.6 |
| 197 | 45 | 4228EC | 1010 | JLLU201027 | 25.0 |
| 198 | 45 | 4228EB | 1010 | XTRU880964 | 24.6 |
| 199 | 45 | 4228FT | 1010 | JLLU202133 | 25.3 |
| 200 | 45 | 4228FC | 1010 | NYLU201219 | 24.8 |
| 201 | 45 | 4228FB | 1010 | SSIU205084 | 24.9 |
| 202 | 45 | 4237AB | 1010 | NICC110542 | 21.9 |

|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 203 | 45 | 4232BC | 1010 | SSIU202611 | 25.8 |
| 204 | 45 | 4232BB | 1010 | SSIU202640 | 22.2 |
| 205 | 45 | 4232CC | 1010 | SSIU202784 | 19.1 |
| 206 | 45 | 4232CB | 1010 | NYLU201036 | 22.2 |
| 207 | 45 | 4232DC | 1010 | SSIU204110 | 22.4 |
| 208 | 45 | 4232DB | 1010 | SSIU202007 | 21.7 |
| 209 | 45 | 4232EC | 1010 | SSIU202676 | 21.7 |
| 210 | 45 | 4232EB | 1010 | SSIU202646 | 22.5 |
| 211 | 45 | 4232FC | 1010 | JLAU205265 | 22.4 |
| 212 | 45 | 4232FB | 1010 | JLLU201136 | 23.6 |
| 213 | 45 | 4236AC | 4010 | NYKU680227 | 25.2 |
| 214 | 45 | 4236AB | 4010 | NYKU681921 | 25.0 |
| 215 | 45 | 4236BT | 4010 | NYKE5817   | 24.8 |
| 216 | 45 | 4236BC | 4010 | NYKU6685   | 25.0 |
| 217 | 45 | 4236BB | 4010 | NYKU681457 | 25.6 |
| 218 | 45 | 4236CC | 4010 | NYKU681866 | 25.1 |
| 219 | 45 | 4236CB | 4010 | NYKE4017   | 25.4 |
| 220 | 45 | 4236DC | 4010 | NYKU670340 | 24.6 |
| 221 | 45 | 4236DB | 4010 | NYKU4051   | 25.1 |
| 222 | 45 | 4236EC | 5010 | SHWU200386 | 24.7 |
| 223 | 45 | 4236EB | 4010 | SHWU202034 | 25.3 |
| 224 | 45 | 4236FC | 6010 | FECU340034 | 21.0 |
| 225 | 45 | 4236FB | 6010 | FECU340076 | 24.3 |
| 226 | 45 | 4240AC | 6010 | ICSU225306 | 24.4 |
| 227 | 45 | 4240AB | 5010 | SLDR486060 | 13.6 |
| 228 | 45 | 4240BC | 6010 | YSLU201249 | 22.7 |
| 229 | 45 | 4240BB | 6010 | YSLU201187 | 22.6 |
| 230 | 45 | 4240EC | 6010 | CTIU444064 | 22.1 |
| 231 | 45 | 4240EB | 5010 | CTIU438536 | 23.9 |
| 232 | 45 | 4240FT | 6010 | YSLU209119 | 21.3 |
| 233 | 45 | 4240FC | 6010 | YSLU206146 | 21.4 |
| 234 | 45 | 4240FB | 5010 | SHWU202339 | 13.9 |
| 235 | 45 | 4248DB | 2010 | KKLU205902 | 21.8 |
| 236 | 45 | 4248EC | 2010 | ITSU203755 | 23.9 |
| 237 | 45 | 4248EB | 2010 | KKLU205476 | 24.4 |
| 238 | 45 | 4248FC | 2010 | KKLU204299 | 23.8 |
| 239 | 45 | 4248FB | 2010 | ITSU202650 | 24.4 |
| 240 | 45 | 4260AC | 6010 | ICSU228231 | 22.6 |
| 241 | 45 | 4260AB | 6010 | YSAA204727 | 20.9 |
| 242 | 45 | 4260BC | 6010 | YSLU203081 | 21.0 |
| 243 | 45 | 4260BB | 6010 | YSLU209287 | 20.7 |
| 244 | 45 | 4260CC | 3010 | MULU201846 | 27.6 |
| 245 | 45 | 4260CB | 2010 | NYLU205592 | 23.8 |
| 246 | 45 | 4260DC | 3010 | MULU203552 | 27.6 |
| 247 | 45 | 4260DB | 6010 | ICSU220043 | 23.1 |
| 248 | 45 | 4260EB | 3010 | MULU206296 | 17.5 |
| 249 | 45 | 4260FT | 3010 | MULU201874 | 27.2 |



|     |    |        |      |            |      |
|-----|----|--------|------|------------|------|
| 250 | 46 | 5460AC | 2020 | KKLU500161 | 23.3 |
| 251 | 46 | 5460AB | 2020 | KKLU504427 | 23.1 |
| 252 | 46 | 5460BC | 5020 | SLRF486012 | 25.6 |
| 253 | 46 | 5460BB | 6020 | YSLN503664 | 23.2 |
| 254 | 46 | 5460CC | 2020 | ITSU500240 | 23.1 |
| 255 | 46 | 5460CB | 2020 | ITSU500209 | 23.0 |
| 256 | 46 | 5460DC | 2020 | KKLU504226 | 23.5 |
| 257 | 46 | 5460DB | 2020 | KKLU504237 | 23.1 |
| 258 | 46 | 5460EC | 2020 | KKLU504439 | 23.2 |
| 259 | 46 | 5460EB | 2020 | KKLU504341 | 23.1 |
| 260 | 46 | 5460FC | 2020 | KKLU504080 | 23.0 |
| 261 | 46 | 5460FB | 2020 | KKLU504095 | 23.1 |

1

## \*\*\*DISTRIBUTION OF CONTAINERS IN YARD\*\*\*

| GP NO | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   | 11   | 12   | 13   | 14   | 15   | 16 | 17 |
|-------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|----|----|
| 1     | 4201 | 4203 | 4209 | 4215 | 4257 | 5203 | 5295 | 5237 | 5239 | 5749 | 5751 | 5755 | 5757 | 5759 | 5763 |    |    |
|       | 3    | 3    | 10   | 12   | 5    | 5    | 4    | 4    | 3    | 1    | 1    | 1    | 6    | 5    | 5    |    |    |
| 2     | 5214 | 5218 | 5250 | 5260 | 5272 |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 10   | 10   | 2    | 1    | 2    |      |      |      |      |      |      |      |      |      |      |    |    |
| 3     | 5460 | 5464 |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 1    | 10   |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
| 4     | 5761 |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 4    |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
| 5     | 5264 | 5268 |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 5    | 2    |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
| 6     | 5208 |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 14   |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
| 7     | 4201 | 4203 | 4257 | 4273 | 5205 | 5243 | 5749 | 5753 | 5755 |      |      |      |      |      |      |    |    |
|       | 1    | 3    | 1    | 5    | 2    | 4    | 2    | 6    | 5    |      |      |      |      |      |      |    |    |
| 8     | 4212 | 4218 | 4224 | 4228 | 4232 | 4236 | 4240 | 4248 | 4260 |      |      |      |      |      |      |    |    |
|       | 1    | 10   | 16   | 16   | 11   | 13   | 9    | 5    | 10   |      |      |      |      |      |      |    |    |
| 9     | 5460 |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |
|       | 12   |      |      |      |      |      |      |      |      |      |      |      |      |      |      |    |    |

4.250 CP SECONDS EXECUTION TIME.

2. CLP Output (Test #1)

## GROUP LOAD PLAN

\*\*\*\*\*

| GROUP | TO LOAD | IN GROUP | RECEIVING BAYS |    |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |  |  |  |  |  |  |  |  |
|-------|---------|----------|----------------|----|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--|--|--|--|--|--|--|--|
|       |         |          | 3              | 4  | 5 | 7  | 8  | 9  | 11 | 12 | 14 | 15 | 18 | 20 | 21 | 22 | 24 | 25 | 28 | 32 |  |  |  |  |  |  |  |  |
| 1     | 68      | 68       | 0              | 2  | 0 | 0  | 0  | 21 | 35 | 0  | 0  | 0  | 0  | 1  | 0  | 4  | 0  | 0  | 0  | 5  |  |  |  |  |  |  |  |  |
| 2     | 25      | 25       | 8              | 10 | 0 | 0  | 0  | 0  | 0  | 2  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 5  |  |  |  |  |  |  |  |  |
| 3     | 11      | 11       | 0              | 0  | 0 | 0  | 0  | 0  | 0  | 4  | 0  | 0  | 4  | 0  | 0  | 0  | 0  | 0  | 3  | 0  |  |  |  |  |  |  |  |  |
| 4     | 4       | 4        | 0              | 0  | 0 | 0  | 0  | 0  | 4  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  |  |  |  |  |  |  |  |  |
| 5     | 7       | 7        | 0              | 0  | 0 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 7  | 0  | 0  | 0  | 0  |  |  |  |  |  |  |  |  |
| 6     | 14      | 14       | 0              | 0  | 0 | 14 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  |  |  |  |  |  |  |  |  |
| 7     | 29      | 29       | 0              | 0  | 6 | 0  | 21 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 2  | 0  |  |  |  |  |  |  |  |  |
| 8     | 91      | 91       | 0              | 0  | 0 | 4  | 0  | 0  | 0  | 4  | 21 | 7  | 0  | 0  | 21 | 6  | 12 | 12 | 4  | 0  |  |  |  |  |  |  |  |  |
| 9     | 12      | 12       | 0              | 0  | 0 | 0  | 0  | 0  | 0  | 0  | 0  | 8  | 0  | 0  | 0  | 0  | 0  | 0  | 4  | 0  |  |  |  |  |  |  |  |  |
| TOTAL | 261     | 261      |                |    |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |  |  |  |  |  |  |  |  |

## WORK SEQUENCE TABLE

\*\*\*\*\*

| STAGE | TRANS | GROUP | HOW MANY | BAY |
|-------|-------|-------|----------|-----|
| 1     | 2     | 8     | 21       | 21  |
| 2     | 2     | 1     | 1        | 20  |
| 3     | 2     | 1     | 4        | 22  |
| 4     | 2     | 8     | 6        | 22  |
| 5     | 2     | 5     | 7        | 22  |
| 6     | 2     | 8     | 12       | 24  |
| 7     | 2     | 8     | 12       | 25  |
| 8     | 2     | 3     | 4        | 18  |
| 9     | 2     | 3     | 3        | 28  |
| 10    | 2     | 9     | 4        | 28  |
| 11    | 2     | 8     | 4        | 28  |
| 12    | 2     | 7     | 2        | 28  |
| 13    | 2     | 1     | 5        | 32  |
| 14    | 2     | 2     | 5        | 32  |
| 15    | 1     | 2     | 8        | 3   |
| 16    | 2     | 8     | 21       | 14  |
| 17    | 1     | 2     | 10       | 4   |
| 18    | 1     | 1     | 2        | 4   |
| 19    | 1     | 7     | 6        | 5   |
| 20    | 1     | 7     | 21       | 8   |
| 21    | 2     | 9     | 8        | 15  |

|    |   |   |    |    |
|----|---|---|----|----|
| 22 | 2 | 8 | 7  | 15 |
| 23 | 1 | 8 | 4  | 7  |
| 24 | 1 | 6 | 14 | 7  |
| 25 | 1 | 1 | 21 | 9  |
| 26 | 1 | 1 | 35 | 11 |
| 27 | 1 | 4 | 4  | 11 |
| 28 | 1 | 3 | 4  | 12 |
| 29 | 1 | 2 | 2  | 12 |
| 30 | 1 | 8 | 4  | 12 |

## \*\*\*\*\*DP SOLUTIONS GENERATED IN STAGES\*\*\*\*\*

| STAGE | TRANS | GROUP | MPICK | SOL NO | SOLUTION            | END ROW | FN VAL   | ROW STATUS ARRAY                                |
|-------|-------|-------|-------|--------|---------------------|---------|----------|---|
| ***** | ***** | ***** | ***** | *****  | *****               | *****   | *****    | *****:*****:*****:*****:*****:*****:*****:***** |
| 1     | 2     | 8     | 21    | 1      | 12 42014224         |         | 10700000 | 0 0 6 16 11 13 9 5 10                           |
|       |       |       |       |        | MOVE DETAILS: 4011  | 1097    |          |   |
| 1     | 2     | 8     | 21    | 2      | 12 42014240         |         | 16300000 | 1 10 16 16 11 13 3 0 0                          |
|       |       |       |       |        | MOVE DETAILS: 4091  | 2017    |          |   |
|       |       |       |       |        | OPT SOLUTION NO: 1  |         |          |   |
| 2     | 2     | 1     | 1     | 3      | 12 42015203         |         | 21200000 | 3 3 10 12 5 4 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 4062  | 1098    |          |   |
| 2     | 2     | 1     | 1     | 4      | 12 42015239         |         | 17600000 | 3 3 10 12 5 5 4 4 2 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 4092  | 2068    |          |   |
| 2     | 2     | 1     | 1     | 5      | 12 42015749         |         | 64900000 | 3 3 10 12 5 5 4 4 3 0 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 4106  | 1158    |          |   |
| 2     | 2     | 1     | 1     | 6      | 12 42015763         |         | 63500000 | 3 3 10 12 5 5 4 4 3 1 1 1 6 5 4                 |
|       |       |       |       |        | MOVE DETAILS: 4156  | 2108    |          |   |
| 2     | 2     | 1     | 1     | 7      | 12 42014201         |         | 13000000 | 2 3 10 12 5 5 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 3017  | 2017    |          |   |
| 2     | 2     | 1     | 1     | 8      | 12 42014203         |         | 12800000 | 3 2 10 12 5 5 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 3027  | 2017    |          |   |
| 2     | 2     | 1     | 1     | 9      | 12 42014209         |         | 12200000 | 3 3 9 12 5 5 4 4 3 1 1 1 6 5 5                  |
|       |       |       |       |        | MOVE DETAILS: 3037  | 2017    |          |   |
| 2     | 2     | 1     | 1     | 10     | 12 42014215         |         | 11600000 | 3 3 10 11 5 5 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 3047  | 2017    |          |   |
| 2     | 2     | 1     | 1     | 11     | 12 42014257         |         | 14000000 | 3 3 10 12 4 5 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 3057  | 2017    |          |   |
|       |       |       |       |        | OPT SOLUTION NO: 10 |         |          |   |
| 3     | 2     | 1     | 4     | 12     | 42 42014203         |         | 22400000 | 0 2 10 12 5 5 4 4 2 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 4011  | 1057    |          |   |
| 3     | 2     | 1     | 4     | 13     | 112 42014257        |         | 14000000 | 3 3 10 12 0 5 4 4 3 1 1 1 6 5 5                 |
|       |       |       |       |        | MOVE DETAILS: 3057  | 2017    |          |   |
| 3     | 2     | 1     | 4     | 14     | 62 42015757         |         | 64100000 | 3 3 10 12 5 5 4 4 3 1 1 1 2 5 4                 |
|       |       |       |       |        | MOVE DETAILS: 3138  | 1158    |          |   |
| 3     | 2     | 1     | 4     | 15     | 62 42015763         |         | 63500000 | 3 3 10 12 5 5 4 4 3 1 1 1 6 5 0                 |



|    |                     |   |   |               |     |          |           |      |                              |
|----|---------------------|---|---|---------------|-----|----------|-----------|------|------------------------------|
| 9  | 2                   | 3 | 3 | 35            | 341 | 42015464 | 75600000  | 1    | 3                            |
|    |                     |   |   | MOVE DETAILS: |     | 2018     |           |      |                              |
|    | OPT SOLUTION NO: 35 |   |   |               |     |          |           |      |                              |
| 10 | 2                   | 9 | 4 | 36            | 351 | 42015460 | 76000000  | 8    |                              |
|    |                     |   |   | MOVE DETAILS: |     | 2018     |           |      |                              |
|    | OPT SOLUTION NO: 36 |   |   |               |     |          |           |      |                              |
| 11 | 2                   | 8 | 4 | 37            | 362 | 42014236 | 130300000 | 0    | 0 0 0 0 0 12 9 5 10          |
|    |                     |   |   | MOVE DETAILS: |     | 4055     | 1097      |      |                              |
| 11 | 2                   | 8 | 4 | 38            | 362 | 42014260 | 132700000 | 0    | 0 0 0 0 3 13 9 5 4           |
|    |                     |   |   | MOVE DETAILS: |     | 4095     | 2057      |      |                              |
|    | OPT SOLUTION NO: 37 |   |   |               |     |          |           |      |                              |
| 12 | 2                   | 7 | 2 | 39            | 372 | 42015205 | 141800000 | 1    | 3 1 5 0 4 2 6 5              |
|    |                     |   |   | MOVE DETAILS: |     | 4052     | 1068      |      |                              |
| 12 | 2                   | 7 | 2 | 40            | 372 | 42015243 | 138000000 | 1    | 3 1 5 2 2 2 6 5              |
|    |                     |   |   | MOVE DETAILS: |     | 4062     | 2058      |      |                              |
| 12 | 2                   | 7 | 2 | 41            | 372 | 42015749 | 185700000 | 1    | 3 1 5 2 4 0 6 5              |
|    |                     |   |   | MOVE DETAILS: |     | 4076     | 1098      |      |                              |
| 12 | 2                   | 7 | 2 | 42            | 372 | 42015755 | 185100000 | 1    | 3 1 5 2 4 2 6 3              |
|    |                     |   |   | MOVE DETAILS: |     | 4096     | 2078      |      |                              |
| 12 | 2                   | 7 | 2 | 43            | 372 | 42014203 | 133600000 | 1    | 1 1 5 2 4 2 6 5              |
|    |                     |   |   | MOVE DETAILS: |     | 3027     | 2017      |      |                              |
| 12 | 2                   | 7 | 2 | 44            | 372 | 42014273 | 134000000 | 1    | 3 1 3 2 4 2 6 5              |
|    |                     |   |   | MOVE DETAILS: |     | 3047     | 2017      |      |                              |
|    | OPT SOLUTION NO: 43 |   |   |               |     |          |           |      |                              |
| 13 | 2                   | 1 | 5 | 45            | 402 | 42014203 | 142400000 | 0    | 1 10 7 5 5 4 4 3 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 4011     | 1057      |      |                              |
| 13 | 2                   | 1 | 5 | 46            | 442 | 42014257 | 135600000 | 3    | 3 10 7 0 5 4 4 3 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3057     | 2017      |      |                              |
| 13 | 2                   | 1 | 5 | 47            | 422 | 42015757 | 185300000 | 3    | 3 10 7 5 5 4 4 3 1 1 0 2 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3128     | 1158      |      |                              |
| 13 | 2                   | 1 | 5 | 48            | 432 | 42015763 | 184300000 | 3    | 3 10 7 5 5 4 4 3 1 1 1 6 5 0 |
|    |                     |   |   | MOVE DETAILS: |     | 4156     | 2108      |      |                              |
| 13 | 2                   | 1 | 5 | 49            | 432 | 42015237 | 138600000 | 3    | 3 10 7 5 5 4 2 0 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 4092     | 2068      |      |                              |
| 13 | 2                   | 1 | 5 | 50            | 392 | 42015203 | 142000000 | 3    | 3 10 7 5 0 4 4 3 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3068     | 1098      |      |                              |
| 13 | 2                   | 1 | 5 | 51            | 432 | 42014201 | 133800000 | 1    | 0 10 7 5 5 4 4 3 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3027     | 2017      |      |                              |
| 13 | 2                   | 1 | 5 | 52            | 434 | 42015751 | 185900000 | 0    | 3 10 7 5 5 4 4 3 0 0 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3017     | 2017      | 4106 | 1158                         |
| 13 | 2                   | 1 | 5 | 53            | 434 | 42015239 | 138400000 | 0    | 3 10 7 5 5 4 4 1 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3017     | 2017      | 4092 | 2068                         |
| 13 | 2                   | 1 | 5 | 54            | 432 | 42014209 | 134200000 | 3    | 3 5 7 5 5 4 4 3 1 1 1 6 5 5  |
|    |                     |   |   | MOVE DETAILS: |     | 3037     | 2017      |      |                              |
| 13 | 2                   | 1 | 5 | 55            | 432 | 42014215 | 134800000 | 3    | 3 10 2 5 5 4 4 3 1 1 1 6 5 5 |
|    |                     |   |   | MOVE DETAILS: |     | 3047     | 2017      |      |                              |
|    | OPT SOLUTION NO: 51 |   |   |               |     |          |           |      |                              |

|                     |   |   |    |                    |     |          |               |    |    |    |   |   |
|---------------------|---|---|----|--------------------|-----|----------|---------------|----|----|----|---|---|
| 14                  | 2 | 2 | 5  | 56                 | 512 | 42015214 | 140900000     | 5  | 10 | 2  | 1 | 2 |
|                     |   |   |    | MOVE DETAILS: 4012 |     |          | 1058          |    |    |    |   |   |
| 14                  | 2 | 2 | 5  | 57                 | 512 | 42015250 | 137300000     | 10 | 10 | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4052 |     |          | 2018          |    |    |    |   |   |
| 14                  | 2 | 2 | 5  | 58                 | 531 | 42015272 | 141700000     | 10 | 10 | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 1058 |     |          |               |    |    |    |   |   |
| OPT SOLUTION NO: 57 |   |   |    |                    |     |          |               |    |    |    |   |   |
| 15                  | 1 | 2 | 8  | 59                 | 562 | 52185214 | 670140900000  | 5  | 7  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4052 |     |          | 2018          |    |    |    |   |   |
| 15                  | 1 | 2 | 8  | 60                 | 572 | 52145250 | 710137300000  | 2  | 10 | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4012 |     |          | 1028          |    |    |    |   |   |
| 15                  | 1 | 2 | 8  | 61                 | 572 | 52185250 | 670137300000  | 10 | 2  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4022 |     |          | 2018          |    |    |    |   |   |
| 15                  | 1 | 2 | 8  | 62                 | 582 | 52145272 | 710141700000  | 2  | 10 | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4012 |     |          | 1028          |    |    |    |   |   |
| 15                  | 1 | 2 | 8  | 63                 | 582 | 52185272 | 670141700000  | 10 | 2  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4022 |     |          | 2018          |    |    |    |   |   |
| OPT SOLUTION NO: 61 |   |   |    |                    |     |          |               |    |    |    |   |   |
| 16                  | 2 | 8 | 21 | 64                 | 612 | 52184240 | 670144700000  | 0  | 0  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4061 |     |          | 1097          |    |    |    |   |   |
| 16                  | 2 | 8 | 21 | 65                 | 602 | 52144240 | 710144700000  | 0  | 0  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 4061 |     |          | 1097          |    |    |    |   |   |
| OPT SOLUTION NO: 64 |   |   |    |                    |     |          |               |    |    |    |   |   |
| 17                  | 1 | 2 | 10 | 66                 | 641 | 52144240 | 710144700000  | 2  | 0  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 2018 |     |          |               |    |    |    |   |   |
| 17                  | 1 | 2 | 10 | 67                 | 651 | 52184240 | 750144700000  | 0  | 2  | 0  | 0 | 0 |
|                     |   |   |    | MOVE DETAILS: 1028 |     |          |               |    |    |    |   |   |
| OPT SOLUTION NO: 66 |   |   |    |                    |     |          |               |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 68                 | 662 | 42094240 | 1500144700000 | 0  | 0  | 9  | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 4011 |     |          | 1057          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 69                 | 662 | 42574240 | 1980144700000 | 1  | 0  | 10 | 7 | 3 |
|                     |   |   |    | MOVE DETAILS: 4051 |     |          | 2017          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 70                 | 662 | 57514240 | 6510144700000 | 1  | 0  | 10 | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 4103 |     |          | 1158          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 71                 | 662 | 57634240 | 6350144700000 | 1  | 0  | 10 | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 4153 |     |          | 2108          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 72                 | 662 | 52374240 | 940144700000  | 1  | 0  | 10 | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 3088 |     |          | 1098          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 73                 | 662 | 52054240 | 800144700000  | 1  | 0  | 10 | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 3078 |     |          | 1098          |    |    |    |   |   |
| 18                  | 1 | 1 | 2  | 74                 | 662 | 52034240 | 820144700000  | 1  | 0  | 10 | 7 | 5 |
|                     |   |   |    | MOVE DETAILS: 3068 |     |          | 1098          |    |    |    |   |   |
| OPT SOLUTION NO: 73 |   |   |    |                    |     |          |               |    |    |    |   |   |
| 19                  | 1 | 7 | 6  | 75                 | 732 | 52434240 | 1180144700000 | 1  | 1  | 1  | 5 | 0 |
|                     |   |   |    | MOVE DETAILS: 3058 |     |          | 1068          |    |    |    |   |   |







```

27      1      4      4      119  1131  57614260  21370256100000  0
      MOVE DETAILS: 1018
      OPT SOLUTION NO: 119

28      1      3      4      120  1192  54644260  26610256100000  0 0
      MOVE DETAILS: 4013 1028

28      1      3      4      121  1192  54604260  26370256100000  0 0
      MOVE DETAILS: 4023 2018
      OPT SOLUTION NO: 120

29      1      2      2      122  1202  52144260  32240256100000  0 0 0 0 0
      MOVE DETAILS: 4013 1018
      OPT SOLUTION NO: 122

30      1      8      4      123  1222  42604260  33540256100000  0 0 0 0 0 0 0 0 0
      MOVE DETAILS: 4091 1097
      OPT SOLUTION NO: 123

```

TOTAL NO. OF GENERATED SOLUTIONS= 528

FOR NO. 1 TRANS. INITIAL POSITION: 4201

FOR NO. 2 TRANS. INITIAL POSITION: 5201

TOTAL TRANSTAINER MOVE DISTANCES= 3354 2561

TOTAL DISPLACEMENT FIGURES DO NOT MATCH  
 CHECK DISPLACEMENT ENTRY IN FILE GPLAN. CORRECT VALUE=33426.9 ENTERED VALUE=33470.0  
 1

TRIM AND STABILITY CALCULATION  
 \*\*\*\*\*

SHIP: J APOLLO

VOYAGE NO: 5

DATE: 8/15/80

| BAY # | CONTAINER | VERTICAL | LCG    | TRIM MOMENT |          |
|-------|-----------|----------|--------|-------------|----------|
|       | WEIGHT    | MOMENT   |        | FORE(-)     | AFT(+)   |
| 1     | 1H        | 278.0    | 5357.0 | -75.0       | -20850.0 |
| 2     | 3D        | 56.4     | 1291.0 | -66.8       | -3769.0  |
| 3     | 3H        | 219.8    | 3383.2 | -63.6       | -13987.9 |
| 4     | 4D        | 244.3    | 5751.5 | -63.8       | -15576.6 |
| 4     | 5D        | 101.5    | 2332.6 | -60.0       | -6091.6  |
| 5     | 7H        | 429.3    | 5148.2 | -52.2       | -22396.9 |
| 6     | 7D        | 2.6      | 58.0   | -52.3       | -136.0   |

|    |     |        |         |       |          |
|----|-----|--------|---------|-------|----------|
| 7  | 8D  | 476.3  | 11815.5 | -49.2 | -23443.5 |
| 7  | 9D  | 5.2    | 122.0   | -45.5 | -236.5   |
| 8  | 9H  | 470.7  | 5504.4  | -45.6 | -21463.8 |
| 9  | 11H | 531.8  | 5363.0  | -38.1 | -20277.5 |
| 10 | 11D | 0.0    | 0.0     | -38.3 | 0.0      |
| 11 | 13H | 677.1  | 7570.3  | -31.6 | -21369.3 |
| 12 | 12D | 244.8  | 5830.5  | -35.2 | -3612.1  |
| 12 | 13D | 0.0    | 0.0     | -31.4 | 0.0      |
| 13 | 15D | 272.6  | 6503.0  | -23.9 | -6504.0  |
| 14 | 15H | 1246.8 | 13752.0 | -20.7 | -25771.2 |
| 15 | 16D | 484.3  | 12035.7 | -20.8 | -10069.1 |
| 15 | 17D | 254.2  | 6074.0  | -17.1 | -4334.0  |
| 16 | 19H | 533.4  | 5183.0  | -9.7  | -5174.0  |
| 17 | 19D | 45.6   | 1057.0  | -9.8  | -448.0   |
| 18 | 20D | 97.3   | 2293.6  | -6.8  | -656.8   |
| 18 | 21D | 37.9   | 892.0   | -3.0  | -114.0   |
| 19 | 21H | 694.2  | 6187.0  | -3.1  | -2173.0  |
| 20 | 23D | 327.4  | 7993.8  | 4.6   | 1496.6   |
| 21 | 23H | 1355.0 | 14671.6 | 7.8   | 10514.8  |
| 22 | 24D | 361.1  | 8970.7  | 7.6   | 2758.6   |
| 22 | 25D | 372.5  | 9165.2  | 11.4  | 4238.7   |
| 23 | 27D | 25.2   | 591.0   | 18.6  | 469.0    |
| 24 | 27H | 330.0  | 4248.1  | 21.8  | 7308.1   |
| 24 | 28H | 703.1  | 6934.0  | 21.8  | 15328.0  |
| 25 | 28D | 637.0  | 12112.0 | 21.7  | 28129.5  |
| 25 | 29D | 269.2  | 6327.0  | 25.4  | 5836.0   |
| 26 | 31H | 52.2   | 1189.0  | 61.4  | 3199.0   |
| 27 | 31D | 39.3   | 919.0   | 61.3  | 2676.0   |
| 28 | 32D | 766.9  | 13897.0 | 64.4  | 51195.5  |
| 28 | 33D | 244.1  | 5747.1  | 68.1  | 15820.3  |
| 29 | 33H | 347.3  | 8408.0  | 68.0  | 26662.0  |
| 30 | 35D | 368.1  | 5889.0  | 76.8  | 29430.0  |
| 31 | 35H | 251.0  | 5889.0  | 80.0  | 20979.0  |
| 32 | 36D | 92.6   | 2542.8  | 79.8  | 7393.2   |
| 32 | 37D | 302.8  | 7378.5  | 83.6  | 24448.7  |

|        |         |          |  |         |
|--------|---------|----------|--|---------|
| TOTALS | 14248.9 | 236377.1 |  | 24428.5 |
| 1      |         |          |  |         |

SHIP STABILITY SUMMARY  
\*\*\*\*\*

|                 | WEIGHT(KT) | VERT. MOMENT | LONG. MOMENT | F.S. MOMENT |
|-----------------|------------|--------------|--------------|-------------|
| LIGHT SHIP      | 12184.0    | 149620.0     | 210418.0     |             |
| CONST. & OTHERS | 352.0      | 3900.0       | 13010.0      |             |
| TANK TOTALS     | 6642.0     | 25872.0      | -63349.0     | 1473.0      |
| CONTAINERS      | 14248.9    | 236377.1     | 24428.5      |             |

|   |             |           |              |        |
|---|-------------|-----------|--------------|--------|
| TOTAL   | 33426.9     | 415769.1  | 184507.5     | 14/3.0 |
| TKM= 14.24  | MTC= 442.08 | LCB= 4.08 | DRAFT= 10.17 |        |
| $GM = 14.24 - (415769.1 + 1473.0) / 33426.9 = 1.76 \text{ M}$                     |             |           |              |        |
| $TRI M = (184507.5 - 4.08 \times 33426.9) / (442.08 \times 100) = 1.09 \text{ M}$ |             |           |              |        |
| $FWD \text{ DRAFT} = 9.6 \quad AFT \text{ DRAFT} = 10.6$                          |             |           |              |        |

DO YOU WANT CONTAINER LOAD PLAN PRINTED OUT?  
ANSWER BY ENTERING EITHER 'YES' OR 'NO' HERE.  
? YES  
1

CONTAINER LOAD PLAN  
\*\*\*\*\*

| BAY              | LOCATION | PORT    | ID                     | LINE    | TYPE | WEIGHT | YARD LOCATION | SEQ NO |
|------------------|----------|---------|------------------------|---------|------|--------|---------------|--------|
| FOR NO. 1 TRANS. |          |         | INITIAL POSITION: 4201 |         |      |        |               |        |
| 3                | 3- 2 1   | KOBE    | SSIU202545             | J LINE  | DRY  | 24.9   | 5218AB        | 79     |
| 3                | 3- 2 2   | KOBE    | SSIU202628             | J LINE  | DRY  | 25.1   | 5218BB        | 80     |
| 3                | 3- 3 2   | KOBE    | XTRU880830             | J LINE  | DRY  | 15.2   | 5218CC        | 81     |
| 3                | 3- 2 3   | KOBE    | JLLU206211             | J LINE  | DRY  | 25.0   | 5218CB        | 82     |
| 3                | 3- 3 3   | KOBE    | NICC110591             | J LINE  | DRY  | 24.7   | 5218DC        | 83     |
| 3                | 3- 1 4   | KOBE    | SSIU202712             | J LINE  | DRY  | 25.1   | 5218DB        | 84     |
| 3                | 3- 2 4   | KOBE    | SSIU202655             | J LINE  | DRY  | 24.8   | 5218EC        | 85     |
| 3                | 3- 3 4   | KOBE    | JLAU204064             | J LINE  | DRY  | 17.6   | 5218EB        | 86     |
| 4                | 4- 3 1   | KOBE    | JLLU201175             | J LINE  | DRY  | 25.0   | 5218FC        | 87     |
| 4                | 4- 4 1   | KOBE    | JLLU204142             | J LINE  | DRY  | 24.4   | 5218FB        | 88     |
| 4                | 4- 5 1   | KOBE    | XTRU880758             | J LINE  | DRY  | 24.3   | 5214AB        | 69     |
| 4                | 4- 6 1   | KOBE    | JLAU205231             | J LINE  | DRY  | 24.8   | 5214BC        | 70     |
| 4                | 4- 7 1   | KOBE    | XTRU880835             | J LINE  | DRY  | 24.8   | 5214BB        | 71     |
| 4                | 4- 3 2   | KOBE    | SSIU205048             | J LINE  | DRY  | 24.0   | 5214CC        | 72     |
| 4                | 4- 4 2   | KOBE    | JLLU206196             | J LINE  | DRY  | 24.8   | 5214CB        | 73     |
| 4                | 4- 5 2   | KOBE    | XTRU880721             | J LINE  | DRY  | 23.7   | 5214DB        | 74     |
| 4                | 4- 6 2   | KOBE    | JLAU202034             | J LINE  | DRY  | 25.6   | 5214BB        | 75     |
| 4                | 4- 7 2   | KOBE    | SSIU205076             | J LINE  | DRY  | 22.9   | 5214EC        | 76     |
| 4                | 4- 2 1   | KOBE    | YSFF674152             | YS LINE | DRY  | 12.4   | 5205AB        | 39     |
| 4                | 4- 2 2   | KOBE    | JLAU265689             | J LINE  | DRY  | 16.1   | 5205BB        | 40     |
| 5                | 5- 6 4   | YOKOHAM | JLAU261848             | J LINE  | DRY  | 18.4   | 5205FC        | 140    |
| 5                | 5- 6 5   | YOKOHAM | JLAU665086             | J LINE  | DRY  | 17.4   | 5205FB        | 141    |
| 5                | 5- 6 6   | YOKOHAM | MOLU277869             | MOSK    | DRY  | 14.7   | 5243EC        | 142    |
| 5                | 5- 7 6   | YOKOHAM | MOLU276466             | MOSK    | DRY  | 15.3   | 5243EB        | 143    |
| 5                | 5- 6 7   | YOKOHAM | MOLU277818             | MOSK    | DRY  | 15.2   | 5243FC        | 144    |

|   |        |         |            |          |     |      |        |     |
|---|--------|---------|------------|----------|-----|------|--------|-----|
| 5 | 5- 7 7 | YOKOHAM | HOLU277131 | MOSK     | DRY | 16.2 | 5243FB | 145 |
| 8 | 8- 4 1 | YOKOHAM | YSAA668895 | YS LINE  | DRY | 7.5  | 5749AB | 146 |
| 8 | 8- 4 2 | YOKOHAM | UFCU385256 | MOSK     | DRY | 6.4  | 5749CB | 147 |
| 8 | 8- 5 2 | YOKOHAM | YSFF674338 | YS LINE  | DRY | 15.0 | 5753AT | 148 |
| 8 | 8- 4 3 | YOKOHAM | NICU709583 | J LINE   | DRY | 15.3 | 5753AC | 149 |
| 8 | 8- 5 3 | YOKOHAM | JLAU261792 | J LINE   | DRY | 17.1 | 5753AB | 150 |
| 8 | 8- 6 3 | YOKOHAM | YSFF668940 | YS LINE  | DRY | 15.5 | 5753BT | 151 |
| 8 | 8- 4 4 | YOKOHAM | JLLU661099 | J LINE   | DRY | 20.1 | 5753BB | 152 |
| 8 | 8- 5 4 | YOKOHAM | YSFF674092 | J LINE   | DRY | 19.9 | 5753CB | 153 |
| 8 | 8- 6 4 | YOKOHAM | YSAA668725 | YS LINE  | DRY | 15.1 | 5755BT | 154 |
| 8 | 8- 4 5 | YOKOHAM | YSAA668785 | YS LINE  | DRY | 15.1 | 5755BB | 155 |
| 8 | 8- 5 5 | YOKOHAM | YSAA668754 | YS LINE  | DRY | 15.4 | 5755CB | 156 |
| 8 | 8- 6 5 | YOKOHAM | YSFF674064 | YS LINE  | DRY | 15.4 | 5755DB | 157 |
| 8 | 8- 7 5 | YOKOHAM | YSAA660672 | YS LINE  | DRY | 14.2 | 5755EB | 158 |
| 8 | 8- 4 6 | YOKOHAM | YSAA668717 | YS LINE  | DRY | 15.0 | 4201CB | 130 |
| 8 | 8- 5 6 | YOKOHAM | YSFF674318 | YS LINE  | DRY | 15.8 | 4203CB | 133 |
| 8 | 8- 6 6 | YOKOHAM | JLAU262656 | J LINE   | DRY | 16.6 | 4257CB | 134 |
| 8 | 8- 7 6 | YOKOHAM | JLAU263464 | YS LINE  | DRY | 12.1 | 4273BB | 135 |
| 8 | 8- 4 7 | YOKOHAM | SCXU629864 | J LINE   | DRY | 12.2 | 4273CB | 136 |
| 8 | 8- 5 7 | YOKOHAM | YSAA275160 | YS LINE  | DRY | 20.1 | 4273DB | 137 |
| 8 | 8- 6 7 | YOKOHAM | KKLU261042 | K LINE   | DRY | 11.3 | 4273FC | 138 |
| 8 | 8- 7 7 | YOKOHAM | YSAA266714 | YS LINE  | DRY | 9.5  | 4273FB | 139 |
| 7 | 7- 5 1 | YOKOHAM | YSLU203081 | YS LINE  | DRY | 21.0 | 4260BC | 242 |
| 7 | 7- 6 1 | YOKOHAM | YSLU209287 | YS LINE  | DRY | 20.7 | 4260BB | 243 |
| 7 | 7- 5 2 | YOKOHAM | HOLU201846 | MOSK     | DRY | 27.6 | 4260CC | 244 |
| 7 | 7- 6 2 | YOKOHAM | NYLU205592 | K LINE   | DRY | 23.8 | 4260CB | 245 |
| 7 | 7- 7 1 | SHIMIZU | ITSU202306 | K LINE   | DRY | 23.3 | 5208AC | 116 |
| 7 | 7- 8 1 | SHIMIZU | KKLU204735 | K LINE   | DRY | 23.0 | 5208AB | 117 |
| 7 | 7- 9 1 | SHIMIZU | ITSU202552 | K LINE   | DRY | 22.9 | 5208BC | 118 |
| 7 | 7-10 1 | SHIMIZU | KKLU204961 | K LINE   | DRY | 23.5 | 5208BB | 119 |
| 7 | 7- 7 2 | SHIMIZU | ITSU203707 | K LINE   | DRY | 22.8 | 5208CC | 120 |
| 7 | 7- 8 2 | SHIMIZU | ITSU202389 | K LINE   | DRY | 22.9 | 5208CB | 121 |
| 7 | 7- 9 2 | SHIMIZU | KKLU201128 | K LINE   | DRY | 23.3 | 5208DC | 122 |
| 7 | 7-10 2 | SHIMIZU | CTIU415904 | K LINE   | DRY | 24.4 | 5208DD | 123 |
| 7 | 7- 5 3 | SHIMIZU | KKLU201396 | K LINE   | DRY | 23.0 | 5208ET | 124 |
| 7 | 7- 6 3 | SHIMIZU | SSIU217439 | K LINE   | DRY | 24.5 | 5208EC | 125 |
| 7 | 7- 7 3 | SHIMIZU | KKLU204952 | K LINE   | DRY | 23.4 | 5208EB | 126 |
| 7 | 7- 8 3 | SHIMIZU | NYLU201131 | K LINE   | DRY | 23.2 | 5208FI | 127 |
| 7 | 7- 9 3 | SHIMIZU | ITSU203319 | K LINE   | DRY | 22.9 | 5208FC | 128 |
| 7 | 7-10 3 | SHIMIZU | NYLU201299 | K LINE   | DRY | 23.0 | 5208FB | 129 |
| 9 | 9- 3 1 | KOBE    | NLSU262148 | NYK LINE | DRY | 20.1 | 5203ET | 34  |
| 9 | 9- 4 1 | KOBE    | SCFU228197 | NYK LINE | DRY | 19.6 | 5203EC | 35  |
| 9 | 9- 5 1 | KOBE    | NYKU110155 | NYK LINE | DRY | 15.6 | 5203EB | 36  |
| 9 | 9- 3 2 | KOBE    | NYKE9119   | NYK LINE | DRY | 19.1 | 5203FC | 37  |
| 9 | 9- 4 2 | KOBE    | B2098      | NYK LINE | DRY | 18.2 | 5203FB | 38  |
| 9 | 9- 5 2 | KOBE    | JLAU263107 | J LINE   | DRY | 15.0 | 5205DB | 41  |
| 9 | 9- 3 3 | KOBE    | JLAU263035 | J LINE   | DRY | 14.6 | 5205EC | 42  |
| 9 | 9- 4 3 | KOBE    | HOLU276455 | MOSK     | DRY | 12.1 | 5237CB | 43  |
| 9 | 9- 5 3 | KOBE    | JLLU263145 | YS LINE  | DRY | 15.1 | 5237DB | 44  |
| 9 | 9- 3 4 | KOBE    | JLAU263610 | J LINE   | DRY | 8.4  | 5237EB | 45  |
| 9 | 9- 4 4 | KOBE    | ICCU785904 | MOSK     | DRY | 12.3 | 5237FB | 46  |

|    |         |        |            |            |      |      |        |     |
|----|---------|--------|------------|------------|------|------|--------|-----|
| 9  | 9- 5 4  | KOBE   | SLBK286000 | SHOWA LINE | DRY  | 20.2 | 5239DB | 47  |
| 9  | 9- 3 5  | KOBE   | SLBK286026 | SHOWA LINE | BULK | 20.0 | 5239EB | 48  |
| 9  | 9- 4 5  | KOBE   | NYKU172506 | NYK LINE   | DRY  | 19.7 | 5239FB | 49  |
| 9  | 9- 5 5  | KOBE   | YSFF674147 | YS LINE    | DRY  | 15.7 | 4201DB | 3   |
| 9  | 9- 3 6  | KOBE   | YSFF668894 | YS LINE    | BULK | 14.9 | 4209AC | 7   |
| 9  | 9- 4 6  | KOBE   | YSFF674343 | YS LINE    | BULK | 15.3 | 4209AB | 8   |
| 9  | 9- 5 6  | KOBE   | YSAA668795 | YS LINE    | BULK | 14.7 | 4209BB | 9   |
| 9  | 9- 3 7  | KOBE   | YSAA668018 | YS LINE    | BULK | 14.4 | 4209CC | 10  |
| 9  | 9- 4 7  | KOBE   | YSAA668714 | YS LINE    | BULK | 14.1 | 4209CB | 11  |
| 9  | 9- 5 7  | KOBE   | YSFF670355 | YS LINE    | BULK | 15.4 | 4209DB | 12  |
| 11 | 11- 4 1 | KOBE   | JLAU265323 | J LINE     | DRY  | 15.5 | 4257CT | 29  |
| 11 | 11- 5 1 | KOBE   | JLAU265382 | J LINE     | DRY  | 16.4 | 4257CC | 30  |
| 11 | 11- 6 1 | KOBE   | YSAA668780 | YS LINE    | DRY  | 15.0 | 4257DT | 31  |
| 11 | 11- 3 2 | KOBE   | YSAA668766 | YS LINE    | DRY  | 15.6 | 4257DC | 32  |
| 11 | 11- 4 2 | KOBE   | YSAA668728 | YS LINE    | DRY  | 15.7 | 4257DB | 33  |
| 11 | 11- 5 2 | KOBE   | YSAA668857 | YS LINE    | BULK | 15.3 | 4215DC | 22  |
| 11 | 11- 6 2 | KOBE   | YSAA668791 | YS LINE    | DRY  | 15.2 | 4215DB | 23  |
| 11 | 11- 2 3 | KOBE   | YSAA668812 | YS LINE    | DRY  | 15.7 | 4215EC | 24  |
| 11 | 11- 3 3 | KOBE   | YSAA668794 | YS LINE    | DRY  | 14.9 | 4215EB | 25  |
| 11 | 11- 4 3 | KOBE   | YSFF668954 | YS LINE    | DRY  | 14.8 | 4215FT | 26  |
| 11 | 11- 5 3 | KOBE   | YSAA668818 | YS LINE    | DRY  | 15.2 | 4215FC | 27  |
| 11 | 11- 6 3 | KOBE   | YSAA668874 | YS LINE    | DRY  | 16.1 | 4215FB | 28  |
| 11 | 11- 2 4 | KOBE   | YSAA660634 | YS LINE    | BULK | 13.9 | 4209EC | 13  |
| 11 | 11- 3 4 | KOBE   | YSAA660689 | YS LINE    | BULK | 14.3 | 4209EB | 14  |
| 11 | 11- 4 4 | KOBE   | YSFF674355 | YS LINE    | BULK | 15.3 | 4209FC | 15  |
| 11 | 11- 5 4 | KOBE   | YSFF674212 | YS LINE    | BULK | 15.1 | 4209FB | 16  |
| 11 | 11- 6 4 | KOBE   | TOLU279989 | SHOWA LINE | DRY  | 18.7 | 5763AB | 64  |
| 11 | 11- 1 5 | KOBE   | TOLU283092 | SHOWA LINE | DRY  | 18.8 | 5763BB | 65  |
| 11 | 11- 2 5 | KOBE   | TOLU281745 | YS LINE    | DRY  | 18.6 | 5763CB | 66  |
| 11 | 11- 3 5 | KOBE   | TOLU284255 | YS LINE    | DRY  | 19.1 | 5763DB | 67  |
| 11 | 11- 4 5 | KOBE   | TOLU268139 | SHOWA LINE | DRY  | 18.4 | 5763EB | 68  |
| 11 | 11- 5 5 | KOBE   | JLLU665071 | J LINE     | DRY  | 19.1 | 5759AB | 59  |
| 11 | 11- 6 5 | KOBE   | YSFF674280 | J LINE     | DRY  | 19.9 | 5759BB | 60  |
| 11 | 11- 1 6 | KOBE   | YSFF668827 | J LINE     | DRY  | 20.2 | 5759CB | 61  |
| 11 | 11- 2 6 | KOBE   | JLLU665054 | J LINE     | DRY  | 20.4 | 5759DB | 62  |
| 11 | 11- 3 6 | KOBE   | JLAU665542 | J LINE     | DRY  | 19.7 | 5759EB | 63  |
| 11 | 11- 4 6 | KOBE   | JLLU210146 | J LINE     | DRY  | 17.9 | 5757AB | 53  |
| 11 | 11- 5 6 | KOBE   | NICCS43500 | J LINE     | DRY  | 18.4 | 5757BB | 54  |
| 11 | 11- 6 6 | KOBE   | TOLU283822 | J LINE     | DRY  | 18.9 | 5757CB | 55  |
| 11 | 11- 1 7 | KOBE   | NICB78439  | J LINE     | DRY  | 18.5 | 5757DB | 56  |
| 11 | 11- 2 7 | KOBE   | TOLU260431 | J LINE     | DRY  | 18.5 | 5757EB | 57  |
| 11 | 11- 3 7 | KOBE   | JLLU261359 | J LINE     | DRY  | 18.0 | 5757FB | 58  |
| 11 | 11- 4 7 | KOBE   | YSFF674046 | YS LINE    | DRY  | 15.8 | 5759AB | 52  |
| 11 | 11- 5 7 | KOBE   | YSAA660881 | YS LINE    | DRY  | 14.5 | 5751AC | 51  |
| 11 | 11- 6 7 | KOBE   | YSAA668711 | YS LINE    | DRY  | 18.6 | 5749BB | 50  |
| 11 | 11- 7 2 | NAGOYA | JLLU661090 | J LINE     | DRY  | 20.3 | 5761AB | 105 |
| 11 | 11- 7 3 | NAGOYA | YSFF668897 | J LINE     | DRY  | 20.2 | 5761BB | 106 |
| 11 | 11- 7 4 | NAGOYA | JLAU665515 | J LINE     | DRY  | 20.3 | 5761CB | 107 |
| 11 | 11- 7 5 | NAGOYA | JLLU665064 | J LINE     | DRY  | 20.3 | 5761DB | 108 |
| 12 | 12- 6 1 | KOBE   | YSLD503009 | YS LINE    | RFR  | 29.5 | 5460BB | 94  |
| 12 | 12- 7 1 | KOBE   | KKLU504256 | K LINE     | RFR  | 23.1 | 5464EB | 102 |

|    |         |         |            |          |     |      |        |     |
|----|---------|---------|------------|----------|-----|------|--------|-----|
| 12 | 12- 6 2 | KORE    | MYKU781525 | NYK LINE | RFR | 23.4 | 5464FC | 103 |
| 12 | 12- 7 2 | KORE    | KKLU504020 | K LINE   | RFR | 23.0 | 5464FB | 104 |
| 12 | 12- 6 3 | KORE    | JLLY202094 | J LINE   | DRY | 25.1 | 5214EB | 77  |
| 12 | 12- 7 3 | KORE    | SSIU202604 | J LINE   | DRY | 25.3 | 5214FB | 79  |
| 12 | 12- 1 1 | YOKOHAM | MOLU202552 | MOSK     | DRY | 27.6 | 4260DC | 246 |
| 12 | 12- 2 1 | YOKOHAM | ICSU220043 | YS LINE  | DRY | 23.1 | 4260DB | 247 |
| 12 | 12- 3 1 | YOKOHAM | MOLU206296 | MOSK     | DRY | 17.5 | 4260EB | 248 |
| 12 | 12- 8 1 | YOKOHAM | MOLU201874 | MOSK     | DRY | 27.2 | 4260FT | 249 |

## FOR NO. 2 TRANS.

INITIAL POSITION: 5201

|    |         |         |            |            |      |       |        |     |
|----|---------|---------|------------|------------|------|-------|--------|-----|
| 21 | 21- 4 1 | YOKOHAM | YSLU209028 | YS LINE    | DRY  | 29.13 | 4212DC | 159 |
| 21 | 21- 5 1 | YOKOHAM | ITSU202044 | K LINE     | DRY  | 23.7  | 4218BC | 160 |
| 21 | 21- 6 1 | YOKOHAM | ITSU202430 | K LINE     | DRY  | 23.5  | 4218BB | 161 |
| 21 | 21- 4 2 | YOKOHAM | KKLU205177 | K LINE     | DRY  | 20.3  | 4219CC | 162 |
| 21 | 21- 5 2 | YOKOHAM | KKLU204979 | K LINE     | DRY  | 24.7  | 4218CB | 163 |
| 21 | 21- 6 2 | YOKOHAM | ITSU202222 | K LINE     | DRY  | 27.5  | 4218DC | 164 |
| 21 | 21- 4 3 | YOKOHAM | ITSU202112 | K LINE     | DRY  | 23.4  | 4219DB | 165 |
| 21 | 21- 5 3 | YOKOHAM | KKLU204164 | K LINE     | DRY  | 23.6  | 4218EC | 166 |
| 21 | 21- 6 3 | YOKOHAM | ITSU202145 | K LINE     | DRY  | 24.5  | 4218EB | 167 |
| 21 | 21- 4 4 | YOKOHAM | KKLU201381 | K LINE     | DRY  | 23.9  | 4219FC | 168 |
| 21 | 21- 5 4 | YOKOHAM | ITSU202716 | K LINE     | DRY  | 23.7  | 4218FB | 169 |
| 21 | 21- 6 4 | YOKOHAM | NICC11540  | J LINE     | DRY  | 14.8  | 4224AB | 170 |
| 21 | 21- 4 5 | YOKOHAM | JLLU204022 | J LINE     | DRY  | 24.4  | 4224BT | 171 |
| 21 | 21- 5 5 | YOKOHAM | JLLU204026 | J LINE     | DRY  | 24.6  | 4224BC | 172 |
| 21 | 21- 6 5 | YOKOHAM | JLLU210157 | J LINE     | DRY  | 21.2  | 4224BB | 173 |
| 21 | 21- 4 6 | YOKOHAM | SCIU411502 | J LINE     | DRY  | 25.5  | 4224CT | 174 |
| 21 | 21- 5 6 | YOKOHAM | SSIU224771 | J LINE     | DRY  | 25.9  | 4224CC | 175 |
| 21 | 21- 6 6 | YOKOHAM | NYLU201209 | J LINE     | DRY  | 24.8  | 4224CB | 176 |
| 21 | 21- 4 7 | YOKOHAM | JLLU202018 | J LINE     | DRY  | 24.5  | 4224DT | 177 |
| 21 | 21- 5 7 | YOKOHAM | SSIU205057 | J LINE     | DRY  | 24.8  | 4224DC | 178 |
| 21 | 21- 6 7 | YOKOHAM | NICC110503 | J LINE     | DRY  | 25.1  | 4224DB | 179 |
| 22 | 22- 4 3 | KORE    | YSAA668004 | YS LINE    | BULK | 14.8  | 4215AB | 17  |
| 22 | 22- 2 3 | KORE    | YSAA668764 | YS LINE    | DRY  | 14.6  | 4215BC | 18  |
| 22 | 22- 4 3 | KORE    | YSAA660684 | YS LINE    | BULK | 14.7  | 4215BB | 19  |
| 22 | 22-10 3 | KORE    | YSAA660956 | YS LINE    | BULK | 14.6  | 4215CC | 20  |
| 22 | 22-11 3 | KORE    | YSAA660931 | YS LINE    | BULK | 14.6  | 4215CB | 21  |
| 22 | 22- 6 1 | YOKOHAM | JLAU202035 | J LINE     | DRY  | 26.3  | 4224ET | 180 |
| 22 | 22- 7 1 | YOKOHAM | JLLU206240 | J LINE     | DRY  | 24.8  | 4224EC | 181 |
| 22 | 22- 6 2 | YOKOHAM | JLLU204021 | J LINE     | DRY  | 24.9  | 4224EB | 182 |
| 22 | 22- 7 2 | YOKOHAM | NICA1361   | J LINE     | DRY  | 25.0  | 4224FT | 183 |
| 22 | 22- 6 3 | YOKOHAM | XTRU880717 | J LINE     | DRY  | 24.1  | 4224FC | 184 |
| 22 | 22- 7 3 | YOKOHAM | UFCU214439 | J LINE     | DRY  | 24.3  | 4224FB | 185 |
| 22 | 22- 5 1 | NAGOYA  | YSLU207027 | YS LINE    | DRY  | 22.0  | 5264BB | 109 |
| 22 | 22- 8 1 | NAGOYA  | CTIU090382 | SHOWA LINE | DRY  | 22.8  | 5264CC | 110 |
| 22 | 22- 5 2 | NAGOYA  | KKLU201335 | K LINE     | DRY  | 21.5  | 5264CB | 111 |
| 22 | 22- 8 2 | NAGOYA  | FECU340133 | YS LINE    | DRY  | 27.0  | 5264FC | 112 |
| 22 | 22- 5 3 | NAGOYA  | YSLU203027 | YS LINE    | DRY  | 21.8  | 5264FB | 113 |
| 22 | 22- 8 3 | NAGOYA  | KKLU204004 | K LINE     | DRY  | 22.1  | 5268BB | 114 |
| 22 | 22- 9 3 | NAGOYA  | YSLU209257 | YS LINE    | DRY  | 26.7  | 5268DB | 115 |
| 24 | 24- 6 1 | YOKOHAM | JLAU202055 | J LINE     | DRY  | 25.1  | 4224AB | 186 |

|    |         |         |            |            |     |      |        |     |
|----|---------|---------|------------|------------|-----|------|--------|-----|
| 24 | 24- 6 2 | YOKOHAM | JLLU206251 | J LINE     | DRY | 24.8 | 4228BF | 187 |
| 24 | 24- 5 3 | YOKOHAM | XTRU880788 | J LINE     | DRY | 24.3 | 4228BC | 188 |
| 24 | 24- 6 3 | YOKOHAM | JLLU202054 | J LINE     | DRY | 24.9 | 4228BB | 189 |
| 24 | 24- 5 4 | YOKOHAM | NICC110546 | J LINE     | DRY | 24.9 | 4228CF | 190 |
| 24 | 24- 6 4 | YOKOHAM | XTRU880748 | J LINE     | DRY | 24.4 | 4228CT | 191 |
| 24 | 24- 5 5 | YOKOHAM | JLAU205251 | J LINE     | DRY | 24.7 | 4228CT | 192 |
| 24 | 24- 6 5 | YOKOHAM | SSIU205061 | J LINE     | DRY | 24.7 | 4228CC | 193 |
| 24 | 24- 5 6 | YOKOHAM | NICA1476   | J LINE     | DRY | 24.7 | 4228CB | 194 |
| 24 | 24- 6 6 | YOKOHAM | JLLU202049 | J LINE     | DRY | 25.3 | 4228DC | 195 |
| 24 | 24- 5 7 | YOKOHAM | FECU340345 | J LINE     | DRY | 25.6 | 4228DB | 196 |
| 24 | 24- 6 7 | YOKOHAM | JLLU201027 | J LINE     | DRY | 25.0 | 4228EC | 197 |
| 25 | 25- 6 1 | YOKOHAM | XTRU880864 | J LINE     | DRY | 24.6 | 4228EB | 198 |
| 25 | 25- 7 1 | YOKOHAM | JLLU202133 | J LINE     | DRY | 25.3 | 4228FT | 199 |
| 25 | 25- 8 1 | YOKOHAM | NYLU201219 | J LINE     | DRY | 24.8 | 4228FC | 200 |
| 25 | 25- 6 2 | YOKOHAM | SSIU205084 | J LINE     | DRY | 24.8 | 4228FB | 201 |
| 25 | 25- 7 2 | YOKOHAM | NICC110542 | J LINE     | DRY | 21.9 | 4232AB | 202 |
| 25 | 25- 8 2 | YOKOHAM | SSIU202611 | J LINE     | DRY | 25.8 | 4232BC | 203 |
| 25 | 25- 6 3 | YOKOHAM | SSIU202640 | J LINE     | DRY | 22.2 | 4232BB | 204 |
| 25 | 25- 7 3 | YOKOHAM | SSIU202784 | J LINE     | DRY | 19.1 | 4232CC | 205 |
| 25 | 25- 8 3 | YOKOHAM | NYLU201036 | J LINE     | DRY | 22.2 | 4232CB | 206 |
| 25 | 25- 9 3 | YOKOHAM | SSIU204110 | J LINE     | DRY | 22.4 | 4232DC | 207 |
| 25 | 25-10 3 | YOKOHAM | SSIU202007 | J LINE     | DRY | 21.7 | 4232DB | 208 |
| 25 | 25-11 3 | YOKOHAM | SSIU202676 | J LINE     | DRY | 21.7 | 4232EC | 209 |
| 18 | 18- 6 1 | KOBE    | NYKU772076 | NYK LINE   | RFR | 23.0 | 5464AB | 95  |
| 18 | 18- 7 1 | KOBE    | MOLU500065 | MOSK       | RFR | 25.0 | 5464BC | 96  |
| 18 | 18- 6 2 | KOBE    | SLRF486007 | SHOWA LINE | RFR | 24.0 | 5464CC | 97  |
| 18 | 18- 7 2 | KOBE    | SLRF486028 | SHOWA LINE | RFR | 25.3 | 5464CB | 98  |
| 28 | 28- 5 1 | KOBE    | SLRF486014 | SHOWA LINE | RFR | 23.8 | 5464DC | 99  |
| 28 | 28- 8 1 | KOBE    | ITSU500117 | K LINE     | RFR | 22.7 | 5464DB | 100 |
| 28 | 28- 8 2 | KOBE    | KKLU504342 | K LINE     | RFR | 23.1 | 5464EC | 101 |
| 28 | 28- 9 1 | YOKOHAM | KKLU500161 | K LINE     | RFR | 23.3 | 5460AC | 250 |
| 28 | 28-11 1 | YOKOHAM | KKLU504427 | K LINE     | RFR | 23.1 | 5460AB | 251 |
| 28 | 28- 9 2 | YOKOHAM | SLRF486012 | SHOWA LINE | RFR | 25.6 | 5460BC | 252 |
| 28 | 28-10 2 | YOKOHAM | YSLH503664 | YS LINE    | RFR | 23.2 | 5460BB | 253 |
| 28 | 28-11 2 | YOKOHAM | SSIU202646 | J LINE     | DRY | 22.5 | 4232EB | 210 |
| 28 | 28- 9 3 | YOKOHAM | JLAU205265 | J LINE     | DRY | 22.4 | 4232FC | 211 |
| 28 | 28-10 3 | YOKOHAM | JLLU201136 | J LINE     | DRY | 23.6 | 4232FB | 212 |
| 28 | 28-11 3 | YOKOHAM | NYKU680227 | NYK LINE   | DRY | 25.2 | 4236AC | 213 |
| 28 | 28-12 1 | YOKOHAM | YSFF674464 | YS LINE    | DRY | 15.0 | 4203AB | 131 |
| 28 | 28-12 2 | YOKOHAM | YSFF640717 | YS LINE    | DRY | 15.2 | 4203BC | 132 |
| 32 | 32- 2 3 | KOBE    | YSAA660642 | YS LINE    | DRY | 14.5 | 4203AC | 4   |
| 32 | 32- 3 3 | KOBE    | YSFF674145 | YS LINE    | DRY | 15.6 | 4203BB | 5   |
| 32 | 32- 9 3 | KOBE    | YSFF674423 | YS LINE    | DRY | 15.7 | 4203CC | 6   |
| 32 | 32-10 3 | KOBE    | YSAA660899 | YS LINE    | DRY | 12.1 | 4201CC | 1   |
| 32 | 32-11 3 | KOBE    | YSFF674248 | YS LINE    | DRY | 15.2 | 4201BC | 2   |
| 32 | 32- 4 3 | KOBE    | ITLU504558 | MOSK       | DRY | 23.8 | 5272AB | 92  |
| 32 | 32- 5 3 | KOBE    | USLU205046 | YS LINE    | DRY | 18.9 | 5272BB | 93  |
| 32 | 32- 6 3 | KOBE    | MOLU205383 | MOSK       | DRY | 20.5 | 5260FB | 91  |
| 32 | 32- 7 3 | KOBE    | SSIU220228 | SHOWA LINE | DRY | 12.3 | 5250DB | 89  |
| 32 | 32- 8 3 | KOBE    | XTRU880774 | J LINE     | DRY | 17.1 | 5250EB | 90  |
| 14 | 14- 4 1 | YOKOHAM | NYKU681921 | NYK LINE   | DRY | 25.0 | 4236AB | 214 |

|    |         |         |            |            |     |      |        |     |
|----|---------|---------|------------|------------|-----|------|--------|-----|
| 14 | 14- 5 1 | YOKOHAM | NYKE5817   | NYK LINE   | DRY | 24.8 | 4236B1 | 215 |
| 14 | 14- 6 1 | YOKOHAM | NYKU6685   | NYK LINE   | DRY | 25.0 | 4236B0 | 216 |
| 14 | 14- 4 2 | YOKOHAM | NYKU681457 | NYK LINE   | DRY | 25.6 | 4236B8 | 217 |
| 14 | 14- 5 2 | YOKOHAM | NYKU681866 | NYK LINE   | DRY | 25.1 | 4236C0 | 218 |
| 14 | 14- 6 2 | YOKOHAM | NYKE4017   | NYK LINE   | DRY | 25.4 | 4236C8 | 219 |
| 14 | 14- 4 3 | YOKOHAM | NYKU670340 | NYK LINE   | DRY | 24.6 | 4236D0 | 220 |
| 14 | 14- 5 3 | YOKOHAM | NYKU4051   | NYK LINE   | DRY | 25.1 | 4236D8 | 221 |
| 14 | 14- 6 3 | YOKOHAM | SHWU200386 | SHOWA LINE | DRY | 24.7 | 4236E0 | 222 |
| 14 | 14- 4 4 | YOKOHAM | SHWU202034 | NYK LINE   | DRY | 25.3 | 4236E8 | 223 |
| 14 | 14- 5 4 | YOKOHAM | FECU340034 | YS LINE    | DRY | 21.0 | 4236F0 | 224 |
| 14 | 14- 6 4 | YOKOHAM | FECU340076 | YS LINE    | DRY | 24.3 | 4236F8 | 225 |
| 14 | 14- 4 5 | YOKOHAM | ICSU225306 | YS LINE    | DRY | 24.4 | 4240A0 | 226 |
| 14 | 14- 5 5 | YOKOHAM | SLDR486060 | SHOWA LINE | DRY | 13.6 | 4240A8 | 227 |
| 14 | 14- 6 5 | YOKOHAM | YSLU201249 | YS LINE    | DRY | 22.7 | 4240B0 | 228 |
| 14 | 14- 4 6 | YOKOHAM | YSLU201189 | YS LINE    | DRY | 22.6 | 4240B8 | 229 |
| 14 | 14- 5 6 | YOKOHAM | CTIU444064 | YS LINE    | DRY | 22.1 | 4240E0 | 230 |
| 14 | 14- 6 6 | YOKOHAM | CTIU438536 | SHOWA LINE | DRY | 23.0 | 4240E8 | 231 |
| 14 | 14- 4 7 | YOKOHAM | YSLU209119 | YS LINE    | DRY | 21.3 | 4240F0 | 232 |
| 14 | 14- 5 7 | YOKOHAM | YSLU206146 | YS LINE    | DRY | 21.4 | 4240F8 | 233 |
| 14 | 14- 6 7 | YOKOHAM | SHWU202339 | SHOWA LINE | DRY | 13.9 | 4240F0 | 234 |
| 15 | 15- 5 1 | YOKOHAM | ITSU500240 | K LINE     | RFR | 23.1 | 5460C0 | 254 |
| 15 | 15- 6 1 | YOKOHAM | ITSU500209 | K LINE     | RFR | 23.0 | 5460C8 | 255 |
| 15 | 15- 7 1 | YOKOHAM | KKLU504226 | K LINE     | RFR | 23.5 | 5460D0 | 256 |
| 15 | 15- 8 1 | YOKOHAM | KKLU504237 | K LINE     | RFR | 23.1 | 5460D8 | 257 |
| 15 | 15- 5 2 | YOKOHAM | KKLU504439 | K LINE     | RFR | 23.2 | 5460E0 | 258 |
| 15 | 15- 6 2 | YOKOHAM | KKLU504341 | K LINE     | RFR | 23.1 | 5460E8 | 259 |
| 15 | 15- 7 2 | YOKOHAM | KKLU504080 | K LINE     | RFR | 23.0 | 5460F0 | 260 |
| 15 | 15- 8 2 | YOKOHAM | KKLU504095 | K LINE     | RFR | 23.1 | 5460F8 | 261 |
| 15 | 15- 5 3 | YOKOHAM | KKLU205902 | K LINE     | DRY | 21.8 | 4248D0 | 235 |
| 15 | 15- 6 3 | YOKOHAM | ITSU203755 | K LINE     | DRY | 23.9 | 4248E0 | 236 |
| 15 | 15- 7 3 | YOKOHAM | KKLU205476 | K LINE     | DRY | 24.4 | 4248E8 | 237 |
| 15 | 15- 8 3 | YOKOHAM | KKLU204298 | K LINE     | DRY | 23.8 | 4248F0 | 238 |
| 15 | 15- 9 3 | YOKOHAM | ITSU202650 | K LINE     | DRY | 24.4 | 4248F8 | 239 |
| 15 | 15-10 3 | YOKOHAM | ICSU228231 | YS LINE    | DRY | 22.6 | 4260A0 | 240 |
| 15 | 15-11 3 | YOKOHAM | YSAA204727 | YS LINE    | DRY | 20.9 | 4260A8 | 241 |

TOTAL TRANSFAINER MOVE DISTANCES= 3354 2561

DO YOU WANT TO MAKE ANY ADJUSTMENT TO THE GENERATED LOAD PLAN?  
ANSWER EITHER 'YES' OR 'NO.'

? NO

12.674 CP SECONDS EXECUTION TIME.

/