Optimizing Re-marshalling Operation in Export Container Terminals

Vincent F. Yu†
Department of Industrial Management
National Taiwan University of Science and Technology, Taipei 106, TAIWAN
Email: vincent@mail.ntust.edu.tw

Hsiang-Yu Cheng
Department of Logistics Management
National Kaohsiung First University of Science and Technology, Kaohsiung 811, TAIWAN
Email: u9423825@ccms.nkfust.edu.tw

Hsiu-I Ting
Department of Money and Banking
National Kaohsiung First University of Science and Technology, Kaohsiung 811, TAIWAN
Email: hiting@ccms.nkfust.edu.tw

Abstract. *Container loading and unloading time takes up a large portion of the berthing time of a containership. To reduce the container loading time, the storage locations of containers in the yard must be arranged in a way that export containers can be loaded efficiently and effectively onto a containership. We decompose the re-marshalling planning operation into two stages: "Optimal Container Stacking Location Planning" and the "Optimal Container Moving Sequence Planning" and developed integer programming models for each of them, respectively. In the first stage, we solve an assignment problem to determine the optimal stacking location of each container. Based on the result, we solve a travelling sales problem with precedence constraints (TSPPC) to determine the most cost effective sequence of container moves to convert the initial layout to the optimal layout. AMPL/CPLEX can be used to solve the assignment problem in the first stage efficiently, as shown by the computational study. However, AMPL/CPLEX cannot solve the TSPPC within a reasonable time. Therefore, we use EVELOVER, a genetic algorithm solver, to solve the TSPPC. Computational study indicates that our model is capable of generating quality solutions within the predefined time limit.*

Keywords: *Re-marshalling, Container Yard, Travelling salesman problem with precedence constraints.*

1. INTRODUCTION

The berth time is one of the key factors affecting the shipping company's choice of ports. Loading and unloading operations take up a large portion of the berth time. In general, the faster the loading/unloading rate, the shorter the berth time. Thus, in practice, container terminals frequently perform re-marshalling operations during which containers are sorted, grouped, and moved to an ideal layout that may speed up the loading/unloading operation and shorten the berth time.

We only discuss a few relevant articles on this topic. Cao and Uebe (1993) modeled the problem as a capacitated multi-commodity p-median transportation problem which is solved by a branch and bound method. Kim and Bae (1998) divided the problem into two stages. In the first stage, the best locations for each container are determined. Then the optimal container moving sequence is solved by dynamic programming complimented by the big M method. The goal is to convert the current layout of the container terminal into the layout determined in the first stage in a way that the number of moves and total moving distance are minimized.

2. PROBLEM DEFINITION

There are two stages in the planning of the re-marshalling operation: determining the optimal storage space for containers and optimizing the container moving plan.

Stage 1 determines optimal storage space for containers. Containers are grouped by their destinations. The optimal storage location for each group of containers needs to be determined so that containers can be moved from their current locations to the optimal locations, when a yard crane is not in operation for loading or unloading containers.

Stage 2 optimizes container moving plan. After the optimal location of each container group is determined, the optimal moving sequence for containers that need to be moved should be determined. Note that some containers may stay at their current locations and do not need to be moved.

We adopted the concept of job-shop scheduling problem and treated each container movement as a job. Considering the precedence relationship between container movements, the problem is formulated as a traveling salesman problem with precedence constraints. Each job is denoted by a node in a network. The objective is to find the shortest distance trip for the crane to move from its parking location (origin), go through each job node, and return to the origin. The precedence relationship much be observed. Since the moving distance of each job is fixed, the only distance that need to be considered is the empty trip distance.

3. MATHEMATICAL MODEL

An integer programming model for the re-marshalling problem is formulated. We assume that

a. All containers are identical.

b. There is only one yard crane available for the re-marshalling operation. The yard crane can only move one container at a time.

c. There are enough storage spaces for the containers.

d. All the containers are to be loaded onto the same container ship.

e. All containers are in the container terminal before the re-marshalling operation starts.

f. We only consider the moving distance between bays. Each container's location within the bay is not considered.

g. We only consider the precedence relationship at the container's original location.

3.1 Mathematical Model for the Optimal Container Locations

Following Kim and Bae (1998), we assume that each bay can only stack at most *R*=2 container groups. There are M bays in the container terminal and G container groups. The objective is to determine the optimal locations of containers in a way that the total moving distance required is minimized. Define the following parameters and decision variables.

C_{ij}: Distance between bay *i* and bay *j*
G: Set of container groups
M: Set of bays
Mj: Capacity of the *j*th bay
N: A large positive number
R: Maximum number of container groups in a bay. Assume *R*=2.
S_{ik}: Number of group *k* containers currently located at bay *i*
X_{ijk}: Number of group *k* containers to be moved from bay *i* to bay *j*
Z_{jk}: 1, if there are containers to be moved from bay *i* to bay *j*; 0 otherwise

Then the problem can be formulated as follows.

minimize
$$\sum_{i\in M}\sum_{j\in M}\sum_{k\in G} C_{ij}X_{ijk} \qquad (1)$$

subject to
$$\sum_{i\in M}\sum_{k\in G} X_{ijk} \le m_j, \quad \forall j \in M \qquad (2)$$

$$\sum_{j\in M} X_{ijk} = S_{ik}, \quad \forall i \in M, k \in G \qquad (3)$$

$$\sum_{i\in M} X_{ijk} \le N \times z_{jk}, \quad \forall j \in M, k \in G \qquad (4)$$

$$\sum_{i\in M} X_{ijk} \ge z_{jk}, \quad \forall j \in M, k \in G \qquad (5)$$

$$\sum_{k\in G} z_{jk} \le R, \quad \forall j \in M \qquad (6)$$

$$X_{ijk} \ge 0 \text{ integer}, \quad \forall i \in M, j \in M, k \in G \qquad (7)$$

$$z_{jk} \in \{0,1\}, \quad \forall i \in M, j \in M, k \in G \qquad (8)$$

The object (1) is to minimize the total container moving distances. Constraints (2) are the bay capacity constraints. Constraints (3) ensure that containers that are moved out of bay *i* is the same as the containers currently in bay *i* for each container group. Contraints (4) and (5) together make sure that if any $X_{ijk} \ge 1$, then $z_{jk} = 1$, otherwise $z_{jk} = 0$. Constraints (6) state that at most two group of containers can be located at the same bay.

3.2 Mathematical Model for the Optimal Container Moving Sequence

We modified the two commodity network model proposed by Kusiak and Finke (1987) to formulate a network model for finding the optimal container moving sequence. The concept of the two commodity network model is illustrated in Figure 1. In this figure, there are n=6 nodes and the sequence of the tour is 1→2→3→4→5→6. There are two commodities, denoted by p and q. There are n-1 units of supply for p-commodity at node 1 (origin) and 1 unit of demand for p-commodity at all other nodes. On the other hand, there are 1 unit of supply for q-commodity at each node except for node 1, which has n-1 unit of demand for q-commodity.

Define the following parameters and variables. integer programming model for the re-marshalling problem is formulated. We assume that

- n: number of nodes
- D_{ij}: distance between the end location of the ith job and the starting location of the jth job, i.e., the empty trip distance.
- Y_{ij}^p : flow value of p-commodity on arc (i,j)
- Y_{ij}^q : flow value of q-commodity on arc (i,j)
- Y_{ij} : 1 if job i immediately precedes job j, 0 otherwise.

Then the problem can be formulated as the following network model:

minimize
$$\frac{1}{n}\sum_{i=0}^{n}\sum_{j=0}^{n} D_{ij}\left(Y_{ij}^p + Y_{ij}^q\right) \tag{9}$$

subject to
$$\sum_{j=0}^{n} Y_{ij}^p - \sum_{j=0}^{n} Y_{ji}^p = \begin{cases} n & \text{, for } i=0 \\ -1 & \text{, elsewhere} \end{cases} \tag{10}$$

$$\sum_{j=0}^{n} Y_{ij}^q - \sum_{j=0}^{n} Y_{ji}^q = \begin{cases} -n & \text{, for } i=0 \\ 1 & \text{, elsewhere} \end{cases} \tag{11}$$

$$\sum_{j=0}^{n}\left(Y_{ij}^p + Y_{ij}^q\right) = n\,,\quad \forall i = 0,\dots,n\,; \quad j = 0,\dots,n \tag{12}$$

$$\sum_{j=0}^{n} Y_{uj}^p - \sum_{j=0}^{n} Y_{vj}^p \ge 1\,,\ \text{if container } u \text{ is on top of container } v \tag{13}$$

$$Y_{ij}^p \ge 0\,,\quad \forall i = 0,\dots,n\,;\quad j = 0,\dots,n \tag{14}$$

$$Y_{ij}^q \ge 0\,,\quad \forall i = 0,\dots,n\,;\quad j = 0,\dots,n \tag{15}$$

$$z_{jk} \in \{0,1\}\,,\quad \forall i \in M, j \in M, k \in G \tag{16}$$

$$Y_{ij} \in \{0,1\}\,,\quad \forall i = 0,\dots,n\,;\quad j = 0,\dots,n \tag{17}$$

The goal is to minimize the objective function (9), which is the total distance of empty trips travelled by the yard crane. Constraints (10) and (11) ensure feasible flow for commodity p and q, respectively. Constraints (12) make sure that the total outflow of any node is n to guarantee a feasible trip. Constraints (13) ensure that the total flow on each arc is n when the arc is used; otherwise it is zero. Constraints (12) and (13) together not only ensure that a total of n units of outflow at each node, but also force p-commodity to flow from node 0 to other nodes, and q-commodity to be moved from all other nodes to node 0. Constraints (10)-(13) and (15)-(17) together forms a complete constraint set for the travelling salesman problem. Constraints (14) are the job precedence constraints which ensure that job u is completed before job v can be started. There are a total of n units of flow in the network, resulting in the 1/n term in the objective function.

4. CASE STUDY

In practice, a container terminal have more than one storage yard for export containers, and other container yards for import containers and transfer containers. There is only limited time available for planning the re-marshalling operation of each export container yard. Therefore, we assume that the re-marshalling operation planning for an export container yard must be done within one minute. The determination of new locations for each container is solved using AMPL/CPLEX, while the job sequencing problem is solved by the EVOLVER, a GA based optimization software. The calculation is terminated regardless of the quality of the current (feasible) solution.

We tested the solution approach using data provided by a shipping company in Taiwan. There are 480 storage spaces (20 bays × 6 rows × 4 tiers = 480 slots) in the export container yard. There are 12 destination ports. Data on the number of containers for each destination port in the test case is given in Table 1. A total of 381 containers are to be shipped to the 12 destination ports. They are randomly placed in the container yard, as shown in Table 2.

Stage 1 determines optimal container locations. We define the distance between neighboring bays to be 1 unit distance. It takes 17.7 seconds of computational time for AMPL/CPLEX to obtain the optimal solution, which 128 containers need to be moved. The shortest moving distance is 219 units. Detail container movement plan is shown in Table 3. Table 4 gives the new container locations after the movements.

We then construct optimal container moving sequence in stage 2. Each container move is treated as a task. The task sequence starts with the left of the first row in Table 3 and ends at left of the last row. Counting the start of crane movement from its parking location, there are 129 tasks to be completed. There are 23 PC constraints. Figure 1 demonstrates the precedence relationship of involved container pairs.

The optimal container moving sequence obtained by the EVOLVER is: 0→4→30→40→36→38→31→100→107→109→89→84→99→102→111→106→112→128→114→115→119→124→122→118→121→127→120→116→117→105→92→83→90→80→53→61→77→75→76→70→62→58→43→51→59→49→55→47→67→78→96→95→82→68→63→74→72→71→66→69→73→1→19→29→33→2→21→79→7→27→8→46→9→48→85→97→91→86→93→87→94→101→98→108→125→123→126→113→110→103→104→88→81→64→57→45→50→35→37→11→60→42→65→56→44→52→54→41→34→39→32→18→12→26→5→24→17→25→10→23→6→14→13→16→28→22→3→15→20→0 with the shortest total empty trip distance being 129.

Table 1: Number of containers for each destination port.

Port	A	B	C	D	E	F	G	H	I	J	K	L
Number of Containers	1	4	45	5	75	2	61	25	14	84	11	54
Total = 381												

Table 2: Original container locations.

Port	A	B	C	D	E	F	G	H	I	J	K	L
BAY1	0	0	0	0	10	0	7	0	0	4	0	0
BAY2	1	0	3	1	1	0	1	5	6	0	0	6
BAY3	0	0	1	0	1	0	1	0	0	5	0	5
BAY4	0	0	0	0	4	0	2	0	0	4	0	2
BAY5	0	1	2	0	0	1	0	6	0	13	0	1
BAY6	0	0	0	0	0	0	4	2	0	18	0	0
BAY7	0	1	0	0	2	0	5	0	2	0	0	9
BAY8	0	0	0	0	0	0	0	3	0	9	0	8
BAY9	0	0	1	0	4	1	12	0	1	2	0	1
BAY10	0	0	0	0	0	0	15	0	1	8	0	0
BAY11	0	0	3	0	6	0	0	6	0	0	0	9
BAY12	0	0	2	0	1	0	0	0	3	8	0	10
BAY13	0	0	0	0	12	0	0	1	0	0	0	0
BAY14	0	0	11	4	5	0	3	0	1	0	0	0
BAY15	0	0	4	0	4	0	4	0	0	2	0	0
BAY16	0	0	1	0	6	0	0	2	0	0	0	1
BAY17	0	1	0	0	1	0	5	0	0	10	0	2
BAY18	0	0	8	0	6	0	1	0	0	0	0	0
BAY19	0	0	4	0	5	0	0	0	0	1	5	0
BAY20	0	1	5	0	7	0	1	0	0	0	6	0

Table 3: Original container locations.

Orig. Bay	Port	Number of Containers	Dest. Bay	Dist.	Orig. Bay	Port	Number of Containers	Dest. Bay	Dist.	Orig. Bay	Port	Number of Containers	Dest. Bay	Dist.
BAY 2	E	1	BAY 1	1	BAY 5	L	1	BAY 7	2	BAY 11	C	3	BAY 14	3
BAY 2	G	1	BAY 1	1	BAY 5	B	1	BAY 7	2	BAY 12	C	2	BAY 14	2
BAY 3	E	1	BAY 1	2	BAY 8	L	8	BAY 7	1	BAY 15	C	3	BAY 14	1
BAY 4	E	4	BAY 1	3	BAY 5	F	1	BAY 8	3	BAY 14	E	5	BAY 15	1
BAY 7	I	2	BAY 2	5	BAY 9	J	2	BAY 8	1	BAY 14	G	3	BAY 15	1
BAY 2	L	6	BAY 3	1	BAY 9	F	1	BAY 8	1	BAY 16	E	6	BAY 15	1
BAY 4	J	2	BAY 3	1	BAY 7	E	2	BAY 9	2	BAY 17	L	2	BAY 16	1
BAY 4	L	2	BAY 3	1	BAY 7	G	5	BAY 9	2	BAY 15	J	2	BAY 17	2
BAY 1	J	4	BAY 3	2	BAY 11	E	1	BAY 9	2	BAY 18	G	1	BAY 17	1
BAY 2	A	1	BAY 4	2	BAY 8	H	3	BAY 11	3	BAY 19	J	1	BAY 17	2
BAY 2	C	3	BAY 4	2	BAY 9	L	1	BAY 11	2	BAY 20	G	1	BAY 17	3
BAY 3	C	1	BAY 4	1	BAY 13	H	1	BAY 11	2	BAY 15	C	1	BAY 18	3
BAY 5	C	2	BAY 4	1	BAY 9	I	1	BAY 13	4	BAY 16	C	1	BAY 18	2
BAY 9	C	1	BAY 4	5	BAY 10	I	1	BAY 13	3	BAY 17	E	1	BAY 18	1
BAY 4	J	2	BAY 5	1	BAY 11	E	5	BAY 13	2	BAY 19	E	5	BAY 18	1
BAY 6	J	1	BAY 5	1	BAY 12	E	1	BAY 13	1	BAY 17	B	1	BAY 19	2
BAY 6	H	2	BAY 5	1	BAY 12	I	3	BAY 13	1	BAY 20	B	1	BAY 19	1
BAY 3	G	1	BAY 6	3	BAY 14	I	1	BAY 13	1	BAY 20	C	5	BAY 19	1
BAY 4	G	2	BAY 6	2	BAY 2	D	1	BAY 14	12	BAY 19	K	5	BAY 20	1
Total: 133 containers														

Regarding the parameter settings for GA, we adopt most of the default parameters except for the initial population, crossover rate, and mutation rate. Computational study indicates that setting the initial population to be 200, crossover rate 0.3, and letting the EVOLVER control the mutation rate tends to give better solutions. Figure 2 exhibits the relationship between initial population and solution value based on the results obtained by EVOLVER within one minute of computational time.

5. CONCLUSIONS

This paper discussed the optimization problem involved in the re-marshalling operations of an export container terminal. The problem is dealt with in two stages: determining the optimal storage space for containers and

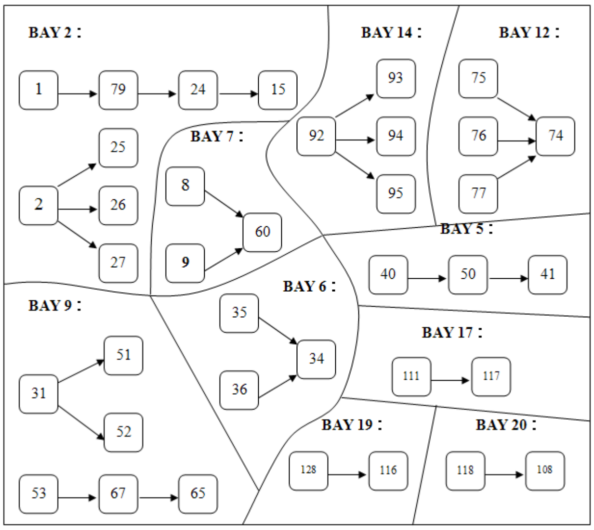

Figure 1: Container moving sequence.

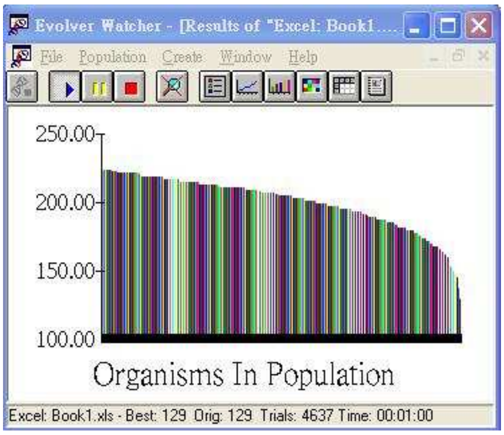

Figure 2: Relationship between solution values and population size.

Table 4: Container locations determined in the first stage.

Port	A	B	C	D	E	F	G	H	I	J	K	L	Total
BAY1					16		8						24
BAY2								5	8				13
BAY3										11		13	24
BAY4	1		7										8
BAY5								8		16			24
BAY6							7			17			24
BAY7		2										18	20
BAY8						2				11			13
BAY9					7		17						24
BAY10							15			8			23
BAY11								10				10	20
BAY12										8		10	18
BAY13					18				6				24
BAY14			19	5									24
BAY15					15		7						22
BAY16								2				3	5
BAY17							7			13			20
BAY18			10		12								22
BAY19		2	9										11
BAY20					7						11		18

optimizing the container moving plan. The first stage problem is modeled as a travelling salesman problem with precedence constraints and solved by AMPL/CPLEX. The problem in the second stage is formulated as a two commodity network model and solved by a GA-based software EVOLVER. The exact location of each container within a bay is not considered in this study. Future research may develop a more detailed model so that it can be applied to practical situation.

REFERENCES

Cao, B., Uebe, G. (1993) An algorithm for solving capacitated multi-commodity p-median transportation problems, *Journal of The Operational Research Society*, **44**, 259-269.

Kim, K. H., Bae, J. W. (1998) Re-marshaling export containers in port container terminals, *Computers & Industrial Engineering*, **35**, 655-658.

Kusiak, A., Finke, G. (1987) Modeling and solving the flexible forging module scheduling problem, *Engineering Optimization*, **12**, 1-12.