UC Berkeley
Earlier Faculty Research

Title
Crane Double Cycling in Container Ports: Algorithms, Evaluation, and Planning

Permalink
https://escholarship.org/uc/item/0nt8t1db

Author
Goodchild, Anne Victoria

Publication Date
2005

eScholarship.org Powered by the California Digital Library
University of California

Crane Double Cycling in Container Ports: Algorithms, Evaluation, and Planning

by

Anne Victoria Goodchild

B.S. (University of California at Davis) 1995
M.S. (University of California at Berkeley) 2002

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Civil and Environmental Engineering

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor Carlos Daganzo, Chair
Professor Candace Yano
Professor Samer Madanat

Fall 2005

UMI Number: 3210596

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI®

UMI Microform 3210596

Copyright 2006 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Crane Double Cycling in Container Ports:

Algorithms, Evaluation, and Planning

Copyright 2005

by

Anne Victoria Goodchild

Abstract

Crane Double Cycling in Container Ports:

Algorithms, Evaluation, and Planning

by

Anne Victoria Goodchild

Doctor of Philosophy in Engineering - Civil and Environmental Engineering

University of California, Berkeley

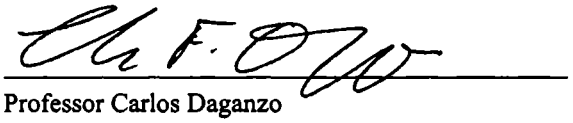
Professor Carlos Daganzo, Chair

Loading ships as they are unloaded (double-cycling) can improve the efficiency of a quay crane and container port. This dissertation describes the double-cycling problem, and presents solution algorithms and simple formulae to estimate benefits. In Chapter 2 we focus on reducing the number of operations necessary to turn around a ship. First an intuitive lower bound is developed. We then present a greedy algorithm that was developed based on the physical properties of the problem and yields a tight upper bound. The formula for an upper bound on the greedy algorithm's performance can be used to accurately predict crane performance. The problem is also formulated as a scheduling problem, which can be solved optimally using Johnson's rule. The problem is extended to include an analysis of double-cycling when ships have deck hatches.

In Chapter 3 we consider at the longer term impact of double cycling on port operations including crane, vessel, and berth productivity. We use another double cycling sequence that is

operationally convenient, easy to model, and nearly optimum. We compare the performance of this sequence to those determined by a greedy algorithm, and Johnson's rule. A framework is developed for analysis, and a simple formula is developed to predict the longer term impact on turn around time. The formula is an accurate predictor of performance. We then show that double cycling can reduce the requirements for landside vehicles and drivers. We also comment on strategies for altering port operations to support double cycling such as segmenting vessel storage, and streamlining traffic flows. We show that double cycling can reduce the amount of time required to complete vessel loading and unloading operations by 10%, improving vessel, crane, and berth productivity. It can reduce by about 20% the requirement for landside vehicles and drivers. Further, for wheeled operations, we suggest a method to reduce the requirement for chassis by about 25%. In Chapter 4 we consider somewhat broader issues, including the interaction of double cycling and security regulations, as well as ship design and routing. We estimate the financial impact of the these benefits, which total approximately $70.00 per container moved, and address obstacles to implementation.

The research demonstrates that double cycling can create significant efficiency gains in vessel, crane, and berth productivity while simplifying some aspects of port operations. We also offer an explanation as to why a method that offers such significant benefits at low cost has not been rapidly adopted. We demonstrate that complementary analytical methods can be used to gain broad insight, and that simple, general models, are often of more value than more complex, detailed models.



Professor Carlos Daganzo
Dissertation Committee Chair

To Bill,

for everything.

Contents

List of Figures iv

List of Tables vi

1 Introduction 1
- 1.1 Literature Review . . . 6
  - 1.1.1 Container Terminals . . . 6
  - 1.1.2 Material Handling . . . 7
  - 1.1.3 Scheduling . . . 9
  - 1.1.4 Maritime Economics . . . 12
- 1.2 State of the Practice . . . 12

2 Double Cycling 15
- 2.1 Modeling Framework . . . 15
- 2.2 Lower Bound . . . 22
- 2.3 Greedy Strategy and an Upper Bound . . . 23
- 2.4 Scheduling Approach . . . 28
  - 2.4.1 Johnson's Rule . . . 31
- 2.5 Deck Hatches . . . 33
- 2.6 Computer Program . . . 39
- 2.7 The Proximal Stack Strategy . . . 41
  - 2.7.1 Comparison of Algorithms on Large Datasets . . . 41

3 Impact on Marine Terminal Operations 45
- 3.1 Evaluation . . . 46
  - 3.1.1 Time-Savings . . . 47
  - 3.1.2 Validation . . . 52
- 3.2 Estimation of the Reduction in Cycles Given a Fleet . . . 53
  - 3.2.1 Expected Number of Cycles Using Single Cycling . . . 54
  - 3.2.2 Expected Number of Cycles Using Double Cycling . . . 54
  - 3.2.3 Comparison of Formula Result and Average of Many Vessels . . . 60

3.2.4 Correspondence between computer program parameters and equation parameters . . . 61
3.3 Impact on Landside Terminal Operations . . . 63
3.3.1 Modeling the Quay Crane . . . 67
3.3.2 Travel Times Between Servers . . . 68
3.3.3 Modeling the Local Storage . . . 68
3.3.4 Estimating the Number of Landside Vehicles Required . . . 69
3.4 Operational Changes to Support Double Cycling . . . 71
3.4.1 Loading Plans . . . 73
3.4.2 Load Sequencing . . . 77
3.4.3 Impact on Storage Equipment . . . 78
3.4.4 Yard Storage Locations . . . 82
3.4.5 Stevedores . . . 84
3.4.6 Container Handling Equipment . . . 84

4 Impact on Maritime Transportation **85**
4.1 Interaction with New Security Regulations . . . 85
4.2 Ship Design and Routing . . . 86
4.2.1 Optimal Number of Hatches . . . 86
4.2.2 Optimal Deck Height . . . 89
4.2.3 Ship Size, Routing and Stopping Frequency . . . 90
4.3 Economic Evaluation . . . 92
4.3.1 Vessel Productivity . . . 94
4.3.2 Opportunity Cost of Crane Time . . . 95
4.3.3 Berth Utilization . . . 96
4.3.4 Landside Vehicles and Drivers . . . 96
4.3.5 Additional Container Handling Equipment . . . 97
4.4 Implementation Obstacles . . . 97

5 Summary **100**

Bibliography **102**

A The Beta Distribution **110**

B Number of Landside Vehicles and Drivers **111**

C List of Acronyms and Abbreviations **114**

List of Figures

1.1 (a) Unloading using single-cycling (b) Unloading and loading with double-cycling. 3

2.1 Plan and side views of a simplified ship (number of containers shown not representative of typical ship size). 16
2.2 Detailed plan for containers to be unloaded and loaded. 18
2.3 Turn around time with different methods: (a) single-cycling with ordering A,B,C,D unloading starts at w = 10, 20 cycles (b) double-cycling with ordering A,B,C,D unloading starts at w = 4, 14 cycles. (c) double-cycling with ordering A,B,C,D unloading starts at w = 3, 14 cycles. (d) double-cycling ordering B,A,C,D unloading starts at w = 3, 13 cycles. 21
2.4 (a) Definition of the greedy strategy (b) Rotated version of part (a). 26
2.5 The greedy strategy is asymptotically optimal. 27
2.6 Performance comparison of the scheduling and greedy algorithms. Each point shows the percentage difference between the greedy solution and the optimal solution. Each point shows the average across 40 generated vessels. 33
2.7 (a) Top view and side view of a ship with hatch coverings (b) Rear view of a simple hatched ship. 35
2.8 Performance comparison of greedy strategy, proximal strategy, and Johnson's rule to single cycling. Each data point shows the percentage savings over single cycling and is the average result for 40 generated vessels. 43
2.9 Comparison of the greedy strategy, proximal strategy, and Johnson's rule to single cycling when double cycling only below deck. Each data point shows the percentage savings over single cycling and is the average result for 40 generated vessels. . . . 44

3.1 Horizontal and vertical motion of the crane. 48
3.2 (a) Example queuing diagram for loading and unloading operations. Notice delay of one cycle to loading operations after completing stack 1. (b) Delay, M_i inserted before any loading operations start to avoid later delay. 57
3.3 (a) Percentage reduction in number of cycles to complete loading and unloading operations below deck using the proximal stack strategy for varying values of D. $C_i = 20$, and $Y = \Lambda$. (b) Percentage reduction, $Y = 0.5\Lambda$. 60

3.4 Squares show the number of moves as calculated by Formula 3.13. Diamonds show the average of 30 simulated vessels with the same characteristics. Error bars show one standard deviation from the mean. 64
3.5 (a) Schematic diagram of landside operations when single cycling. (b) double cycling. 66
3.6 Number of landside vehicle drivers required for single and double cycling, against expected wait time at II. 72
3.7 (a) Vessel storage segregated by origin-destination pairs (labelled O-D). (b) This example vessel calls at four ports labelled by number. The vessel has just left port 4, sailing for port 1. 75
3.8 (a) Vessel storage segregated by origin-destination pairs (labelled O-D). (b) This example vessel calls at four ports labelled by number. The vessel has just left port 1, sailing for port 2. 76
3.9 (a) Traffic flows for straddle carriers when single cycling. (b) double cycling. . . . 78
3.10 (a) Traffic flow pattern for yard tractors single cycling. If imports are stored on-wheels it is necessary for the driver to stop at the empty chassis pool. (b) Landside traffic flow patterns for yard tractors when double cycling. With double cycling, trips to the empty chassis pool are not typically required. 79

4.1 The number of moves using the greedy hatch strategy, and single cycling for a specific vessel. The vessel has 20 stacks, so 1 through 20 hatches are considered. . 88
4.2 Optimal deck height for varying parameters η and δ. 91

List of Tables

2.1 Parameter settings used to generate vessels data. 28
2.2 Example problem data to illustrate Johnson's rule 32
2.3 How computer program parameters can be varied to reflect different market conditions. 40

3.1 Average and standard deviation of benefit over single cycling of three algorithms when double cycling without hatches (or when there is one hatch per stack). 47
3.2 Average and standard deviation of benefit over single cycling of three algorithms when double cycling only below deck. 47
3.3 Parameter values for evaluation, based on Port of Tacoma trial. 52
3.4 Average absolute error in turn around time between estimate and realized value using the greedy strategy. 56
3.5 Correspondence between diffusion parameters and container terminal parameters. . 58
3.6 Parameter values used to generate vessel data in Figure 3.4. 62
3.7 Parameter values used to generate equation estimates in Figure 3.4. 62

4.1 Comparison of the approximate economic benefits of double cycling. 93

Acknowledgments

This work has been funded in part by the National Science Foundation, Grant CMS-0313317, a University of California Graduate Opportunities Fellowship, a University of California Transportation Center Dissertation Fellowship, an FHWA Eisenhower Fellowship, and an American Association of University Women Selected Professions Dissertation Fellowship. I would like to express deep gratitude to my advisor Carlos Daganzo whose guidance and support were crucial for the successful completion of this project, and who has been a patient and wise mentor on many topics outside of this dissertation. I'd also like to gratefully thank Tom Ward of DMJM Harris, and Chris May of the Ben E. Nutter Terminal at the Port of Oakland for their patience and valuable input.

Chapter 1

Introduction

The volume of goods moved by container through the US transportation system has grown dramatically over the last 15 years, but infrastructure has not. In 2004, peak levels of container traffic through major US West Coast ports jumped approximately 15% from the previous year. This caused significant port congestion; for example, containers required an additional week just to be moved from vessels through the marine terminals [58]. It has also caused greater interest in goods movement from governmental agencies as freight traffic is becoming a more significant portion of total traffic flow, adding to regional traffic congestion. Planning for goods movement requires different considerations than passenger traffic, and there is an interest in expediting flow as it benefits national and regional economies. In 2005, inbound container volumes at the ports of Oakland and the Pacific Northwest have grown by about 20% over the previous year, while the volume at the Southern California complex of Los Angeles and Long Beach is up about 5 to 6% over the previous year [59]. There is no reason to believe this growth will not continue, except that our inland transportation infrastructure will not have sufficient capacity. This growth in container volumes

will require additional capacity on the freight transportation network and through ports in particular. Strategies are required that speed the movement of freight through the system, and specifically through terminals. In this research we consider such a strategy. Quay cranes are the most expensive single unit of handling equipment in port container terminals, and because of this, one of the key operational bottlenecks at ports is quay crane availability [14]. By improving quay crane efficiency, ports can reduce ship turn around time, improve port productivity and improve throughput in the freight transportation system. The research presented in this dissertation addresses the key bottleneck to port productivity, quay crane efficiency. In contrast to other measures to increase capacity such as terminal expansion and information technology deployments, double cycling, the method considered here, is a low cost method to increase capacity; it does not require new technology or infrastructure. Although double cycling in the long term will not solve the capacity problem, it can be more quickly implemented than other solutions, and can be used to complement other strategies.

Double-cycling is a technique that can be used to improve the efficiency of quay cranes by eliminating some empty crane moves. Instead of using the current method, where all relevant containers are unloaded before any are loaded (single-cycling), containers are loaded and unloaded simultaneously (see Fig. 1.1). This allows the crane to carry a container while moving from the apron to the ship (one move) as well as from the ship to the apron; doubling the number of containers transported in a cycle (or two moves). This crane efficiency improvement can be used to reduce ship turn around time and therefore improve port throughput, and address the capacity problem.

In their efforts to increase productivity, ports are currently undertaking many varied projects such as renovating and adding terminals, constructing and expanding intermodal facilities, and implementing new IT infrastructure [60]. Because crane productivity is so important, ports have also

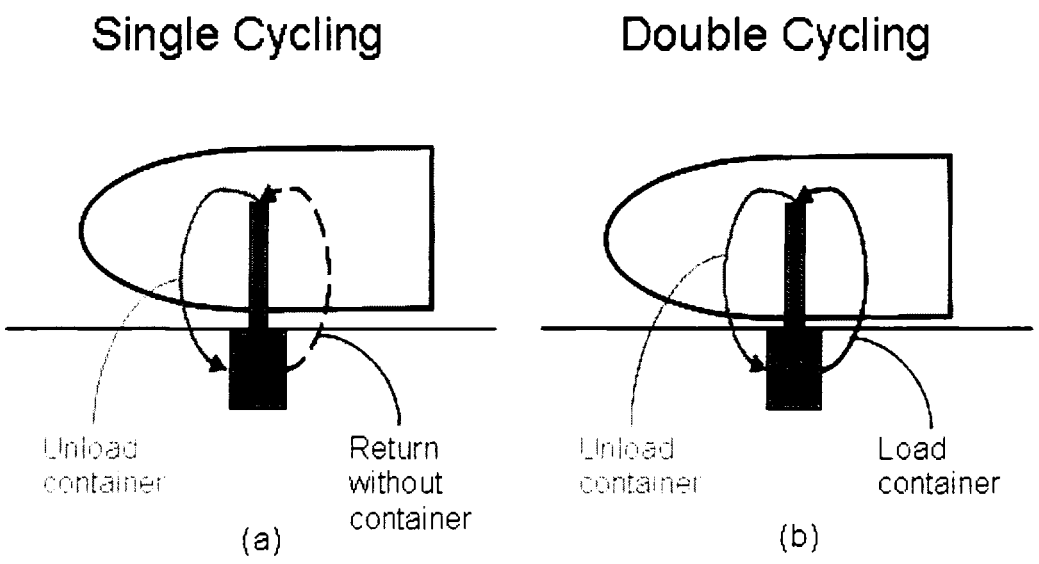


Figure 1.1: (a) Unloading using single-cycling (b) Unloading and loading with double-cycling.

invested in various crane efficiency improvement strategies. For example, dual hoist cranes have been developed that separate the crane's cycle into two subcycles that can be operated largely independently. These are currently in use at many ports, including Rotterdam and Norfolk. Operational changes such as double cycling are of interest because they provide benefits at less cost. The concept of double-cycling is recognized by many in the industry [79], [80], and its potential to improve efficiency is intuitively understood. The technique is used to a limited extent at many ports in the US and abroad, and a research-oriented trial of double cycling took place at the Port of Tacoma in the summer of 2003, but a broad implementation of double cycling has not occurred.

One of the goals of this research is to understand why a broad implementation has not occurred. I believe one reason is that small scale trials have understated the benefits of double cycling which, as we will show, increase with the size of the ship. More importantly the lack of

a rigorous analysis that details the necessary operational changes leave some operators doubting that the benefits will overcome the operational costs. This research aims to address this point by providing a quantitative analysis of the impact of double cycling on crane efficiency and more broadly port productivity.

We initially assume the ship's loading plans are given. Creating vessel loading plans is a complicated process, with many competing objectives. Shipping lines use software tools to create loading plans that accommodate vessel stability requirements, placement constraints on hazardous materials, refrigerated containers, above, and below deck storage, and may include strategies to minimize the number of cycles necessary to unload containers at subsequently visited ports. From these tools, a sequence of operations is generated for the crane operator, foreman (who directs landside operations), and terminal management system.

When initially considering the benefits of double cycling, we assumed that existing planning tools have been used to create a loading plan, as is current practice, and that this loading plan has made no accommodations for double cycling. We therefore consider changes only to the crane's sequence of operations. In practice, this would be determined at a planning stage, and the crane operator would be given a sequence of operations to carry out, in the same way they are when performing single-cycle operations. This way we demonstrate that double-cycling is feasible and beneficial without disrupting current operations. In Section 3.4.1 we consider changes to loading plans that would increase the opportunities to double cycle.

In this research we define and evaluate efficient algorithms for determining a double cycling sequence (for example the greedy strategy, and Johnson's rule), and demonstrate the benefits of double cycling over the status quo (where all containers are unloaded from the vessel before any

are loaded). While one could single cycle (unload or load one container per cycle) some loading operations between unloading operations, we will consider the method currently used in practice, where the entire vessel is unloaded before any containers are loaded. We consider these benefits for vessels with and without deck hatches. We then define a simpler strategy that is more operationally convenient, and focus on this strategy for analyzing longer-term planning. The results of this strategy compare well with the optimal and greedy strategies. Based on this strategy, we develop tools to quantify the impact of double cycling on crane productivity and other aspects of port operations given a fleet of calling vessels. We consider three main operational improvements: reduction in crane operating time per vessel, reduction in the number of landside vehicles and drivers, and reduction in the amount of storage equipment required. We also suggest supporting operational changes for container storage and transportation. We emphasize the relevance of our analysis through comparison with empirical data. We estimate the financial impact of these benefits. Of course, the financial impact will depend on the choices a port makes regarding what to do with the productivity improvements; for example, will an improvement in crane productivity be used to increase vessel throughput, or to reduce labor requirements? These choices depend on the details of each port such as current and expected traffic in peak and off-peak seasons, current labor contracts, and current operational bottlenecks. While we provide estimates as to the magnitude of financial benefits, we stop short of providing a very detailed analysis because this should be done on a case by case basis.

While problems of port design and operation are the subject of much academic research (see Section 1.1) and currently the subject of much political attention (California State Assembly Bill 2650, 2002-03 and 2042, 2003-04), to date no study on double cycling has appeared in a schol-

arly journal. We believe the tools developed in this research will be particularly useful for those considering implementation of double cycling, and performing long-range planning. Modern ports and vessel operators use computer programs to design loading plans, sequence loading and unloading operations, and schedule daily port operations. Double cycling could be easily incorporated into these tools. The ideas presented here are not meant to substitute for detailed terminal and vessel planning programs, which are well suited to managing a specific vessel and terminal configuration, but to provide portable insights into double cycling at a more general level. Results can be used to suggest how a port should be configured and operated when implementing double cycling. These results extend to vessel loading, vessel design, and the relationships between double cycling and trends in maritime transportation.

1.1 Literature Review

The relevant literature crosses many subject areas, I have structured this review into four sections; container terminal literature, material handling literature, scheduling literature, and maritime economics literature. In addition to these topics, the Journal of Commerce and the business section of major newspapers such as The Los Angeles Times cover the more recent trends in terminal operations and maritime transportation.

1.1.1 Container Terminals

The first resource on modern port operations was Atkins' 1983 book "Modern Marine Terminal Operations and Management" [4]. Also of importance is Imakita's early book on marine transportation "A Techno-Economic Analysis of the Port Transport System" which provided

methodological guidance to this thesis. And Erik Rath's Container Systems [72]. Although no academic research has addressed the problem of double cycling, a significant amount of research has addressed other problems of port design and operation. Port research typically focuses on strategic design planning issues such as the number of berths [73], the size of storage space [44], and the number of various pieces of equipment to install [78], and the trade-offs inherent in these choices [63]. Operational planning and control problems include berth scheduling [67], berth assignment [34], quay-crane scheduling [15] and [68], stowage planning and sequencing [19] and [46], storage space planning [11], and dispatching of yard cranes and prime movers [43]. To date, most of this work utilizes queueing theory and stochastic models [16], simulation [51], and classical operations research techniques such as routing, network, and scheduling problems [44].

Outside of the port considerable attention has been paid to container fleet management, as imbalances in imports and exports at most ports require some repositioning or empty balancing of containers. This research has often focused on rail and motor carriers. Recent work includes [24], and [25].

1.1.2 Material Handling

The problem of storage and retrieval has been addressed in the material handling literature, called Automated Storage and Retrieval Systems or AS/RS. Double cycling has some commonalities with AS/RS used in warehousing and manufacturing applications. An important AS/RS system performance measure is the throughput capacity of the system. The throughput capacity for a single aisle is the inverse of the mean transaction time, which is the expected amount of time required for the S/R machine to store and/or retrieve a unit load. The service time for a transaction includes both S/R machine travel time and pickup/deposit time. This time typically depends on the

configuration of the storage rack and the S/R machine specifications. The service time is similar to our cycle time.

The mean transaction time has been estimated by [10] for single command and dual command cycles for randomized storage and retrieval with different I/O configurations. Single command is operationally similar to single cycling, in that one item is stored or retrieved in a cycle and dual command is similar to double cycling in that an item is both stored and retrieved in a cycle. Reducing the mean transaction time is critical for increasing the throughput of the AS/RS, as reducing cycle time, or the number of cycles, is critical for increasing port throughput.

By intelligently sequencing the retrievals, Han et al. [30] improved the throughput capacity of the AS/RS, by reducing unproductive travel between storage location and retrieval location, much as we have done with proximal stack strategy (see Section 2.7. They develop an expression for the maximum possible improvement in throughput if travel between is eliminated for an AS/RS that is throughput bound and operates in dual command mode. This would provide a lower bound on the travel time, but still does not address the variable job time based on previous jobs.

Keserla and Peters [42] analyze an alternative design of the S/R machine that has two shuttles instead of one as in a regular AS/RS. This would be equivalent to a crane that could carry two containers at the same time, and drop or pick-up independently. Cranes of this variety are not currently available, but are under consideration. The new design eliminates the travel between the storage and retrieval points and performs both a storage and a retrieval at the point of retrieval, thereby achieving the maximum throughput increase calculated by [30]. They also provide heuristics for minimizing travel time by sequencing retrievals when the S/R machine operates in dual shuttle mode and give simulation results testing these heuristics.

AS/RS models have also been applied to port operations. In [8], the authors address the movement of containers from the vessel to storage locations. In [6], simulation is used to address three dimensional storage, assuming random demand.

There are many other articles which address dual command operations in AS/RS systems, such as [18]. For an excellent review of AS/RS models see [39]. These provide an interesting comparison and parallels to the double cycling problem, but I have not found any that address the variable job time based on previous jobs (unloading entire stacks where unloading time can depend on the sequence of loading jobs), so the literature is not satisfactory in addressing the double cycling problem. While some of the models that estimate travel time between stacks could apply to the double cycling work presented in this thesis, I have chosen not to model the problem in this level of detail. Empirical evidence suggests that crane cycles are very consistent in their duration, and not as dependent on the distance travelled as the cycles in automated storage and retrieval, where more precise distance models are applicable.

1.1.3 Scheduling

The problem of determining the sequence of operations that provides the smallest number of moves to turn around a vessel can be formulated as a scheduling problem see Section 2.4. This scheduling problem is in the family of well known job-shop scheduling problems [21]. Our problem is formulated with two machines, one to unload stacks of containers, and one to load stacks of containers. It is well known that for two machine problems it is not necessary to consider schedules in which the processing orders on the two machines are not identical [40]. Consequently, we can restrict ourselves to permutation schedules which have identical processing orders on the machines, called flow-shop problems. Our formulation see 2.4, includes a technological constraint: stacks

must be unloaded before they are loaded, but no precedence constraints. We assume all rehandles (containers that must be moved to access another container, but are to stay on the vessel) are loaded back into the stack from which they are unloaded, and that there are no constraints on the order in which a set of stacks is operated on my an individual machine (loading or unloading).

We would like to minimize the makespan, or the time is takes to complete all unloading and loading jobs. An efficient algorithm for determining the optimal solution to the unconstrained problem was presented by Johnson in 1954 [40]. This algorithm is presented in Section 2.4. Since Johnson's seminal paper, much work has been done on the two-machine, and more general m-machine problems. The literature is extensive. In this literature review, it is not my goal to completely describe this literature, but to bring to light the most relevant papers in the academic literature, and to this problem. For a more general text on scheduling problems and the literature, see [70].

A few years after Johnson's paper [40], Mitten [56] presented an efficient optimal algorithm for the case in which start and stop lags exist, but this is not necessary to consider for the double cycling problem. Kurisu [49], [50] used Mitten's results to consider parallel-chain precedence constraints. An example of such a constraint would be that stack j must be unloaded after stack i is unloaded. We ignore this issue here as we assume there are no precedence constraints for jobs on the same machines, and assume all containers for rehandling are put back into the same stacks. In his dissertation in 1976, Rinnooy Kan showed the problem with precedence constraints is NP-hard even if only 2 jobs are related via a precedence relation [41]. In 1979, Sidney [74] applied Kurisu's results to develop a polynomial-bounded optimal algorithm for series-parallel precedence structures. Other significant authors which have considered precedence constraints are Hariri and

Potts [31], and Lawler [52] Monma published two seminal papers in 1979 and 1980 [61], [62]. He utilized Lawler's approach to develop an O[nlogn] realization of the algorithm developed by Sidney [74]. Again, we do not consider precedence constraints in this dissertation.

Johnson's paper [40] also initiated research on the general flow-shop problem, where m-machines are considered. In this problem we have n jobs, and again want to find a processing order to minimize the completion time of all jobs. This problem is solved combinatorially with branch and bound methods. Many papers have been written which develop improved solution algorithms, for example, Campbell et al [32]. Lageweg, Lenstra, and Rinnooy Kan [9] develop many bounds to the problem, including the lower bound we generate in Section 2.2. In Lenstra, Rinnooy Kan, and Brucker's 1977 paper [37], they include a review of the existing (at the time) literature on the subject. In a more recent paper, McMahon and Lim [55] extend the results of Lenstra, Rinnooy Kan and Brucker by developing a new branch and bound algorithm for the 2-machine flow shop problem with arbitrary precedence constraints. For a more recent summary of the job-shop scheduling problem see [3].

The flow-shop problem continues to attract attention. In 1994, Gim, Curry, and Deuermeyer [5] again consider the 2-machine problem here with sparse precedence constraints and develop a new branch and bound algorithm for its solution.

Scheduling formulations have been used with other operations problems for example [53]. Kim and Park in [47] develop a similar scheduling model for determining a least cost sequence of operations for a set of container cranes, but they do not consider double cycling.

1.1.4 Maritime Economics

The literature on maritime economics typically addresses larger scale problems such as ship routing and stopping frequency, rather than terminal operations. Double cycling is addressed indirectly in [7], where the authors analyze the productivity gains from hatchless ships. Actually, the hatchless vessel derives much of its benefit from the use of double cycling. Without hatches, the stacks are double cycled and a larger portion of the vessel's containers can be unloaded and loaded simultaneously. In addition the moves required to move the hatch covers are eliminated. The paper assumes that only containers below deck will be double cycled when comparing hatched and hatchless ships.

1.2 State of the Practice

Although double cycling does not appear in the academic literature, it has been a recognized concept in the industry for at least 10 years [45]. Determining the extent to which it has been used or trialed is fairly difficult as it is not publicly available information. But we do know that double cycling, although it is not widely accepted, is used to a limited extent at some ports. In part our research has attempted to understand why the implementation has been so slow.

In June 2003, the CCDoTT (Center for Commercial Deployment of Transportation Technologies), in conjunction with the Port of Tacoma, Transystems Corporation, Washington United Terminals and other agencies, carried out a full-scale trial of the Efficient Marine Terminal concept. Double cycling of container cranes is a key element of this concept and on June 28, one bay of a Hanjin vessel was loaded and unloaded simultaneously using double cycling [13]. Double cycling occurred below deck only. The work is described in a publicly available report, and to my

knowledge is the first work documenting a real-world trial of double cycling [13].

The goal of the CCDoTT trial was to determine the feasibility of developing an Agile Port System (APS) - incorporating an Efficient Marine Terminal (EMT), Intermodal Interface Center (IIC) and a Dedicated Freight Coordinator (DFC) in the Pacific Northwest. More information on each of these concepts can be found at www.ccdott.org. The concept was created by Bill Hubbard, a freight transportation innovator. Another trial of the Agile Port System is planned for the spring of 2006. Bill Hubbard has a patent for the Agile Port System, and has applied for a patent for an even broader concept, the Buffered Magazine Method and System for Loading and Unloading Ships see Hubbard's patent application [33].

The intuition that double cycling could provide turn around time savings is greeted with interest among the maritime community. This is tempered by significant skepticism that the technique will prove beneficial given the complicated nature of port planning and operations, and the numerous parties involved in the transport of containers. The goal of this research is to provide a general analysis of double cycling and its impact on port operations, shipping operations, and the maritime industry to address these concerns.

In the next chapter, the double cycling problem is formulated and analyzed. In Section 2.1 a framework is set for analysis of the problem. The section also includes an example that is used to illustrate the problem's basic properties. A lower bound on the problem is developed in Section 2.2. Section 2.3 presents the greedy algorithm and bounds its results. In Section 2.4 a scheduling formulation, and its solution are presented. The problem is then extended to accommodate ships with deck hatches in Section 2.5. We introduce the data generation and algorithm evaluation program in Section 2.6. In Section 2.7 we define the proximal stack strategy and compare its performance to

those of other double cycling strategies.

In Chapter 3 we broaden the scope to consider the impact of double cycling on marine terminal operations. Section 3.1 reviews the results of simulated vessel loadings and unloadings, tools to convert benefits from cycles to time, and compares time-savings to empirical results. In Section 3.2 we develop a methodology and formula to determine the improvement in crane productivity given a fleet of vessels calling rather than for a single vessel. We compare the results using this formula to the result from generating data for many vessels. In Section 3.3 we consider the impact of double cycling on the requirements for landside vehicles and drivers. Section 3.4 presents operational changes that could be undertaken to support double cycling, including how the number of containers in port during loading and unloading operations is affected by double cycling, and the ramifications for chassis and storage space requirements.

In Chapter 4 we again expand the scope to some aspects of maritime transportation. More specifically, in Section 4.1 we consider the interaction between new security measures and double cycling. Section 4.2 addresses some aspects of ship design, ship routing and stopping frequency. In Section 4.3 we estimate the financial impact of the benefits described in the dissertation.

Chapter 2

Double Cycling

We begin this chapter by developing a framework for analysis of the double cycling problem. We then formulate the problem and develop algorithms to determine sequences to use when carrying out double cycling operations. We compare the performance of these algorithms and quantify the benefits with respect to the number of cycles required to turn around a vessel. We conclude the chapter by extending the analysis to vessels with deck hatches.

2.1 Modeling Framework

The layout of containers on a ship can be modelled as a 3-dimensional matrix. Containers are stacked on top of one another, and arranged in rows. One row stretches across the width of the ship. Large container vessels today typically hold 20 stacks of containers across the width of the ship, and up to 20 stacks along the length of the ship (40-foot equivalent units). Of course, we expect these figures to increase with the penetration of 10,000 TEU (twenty foot equivalent unit)

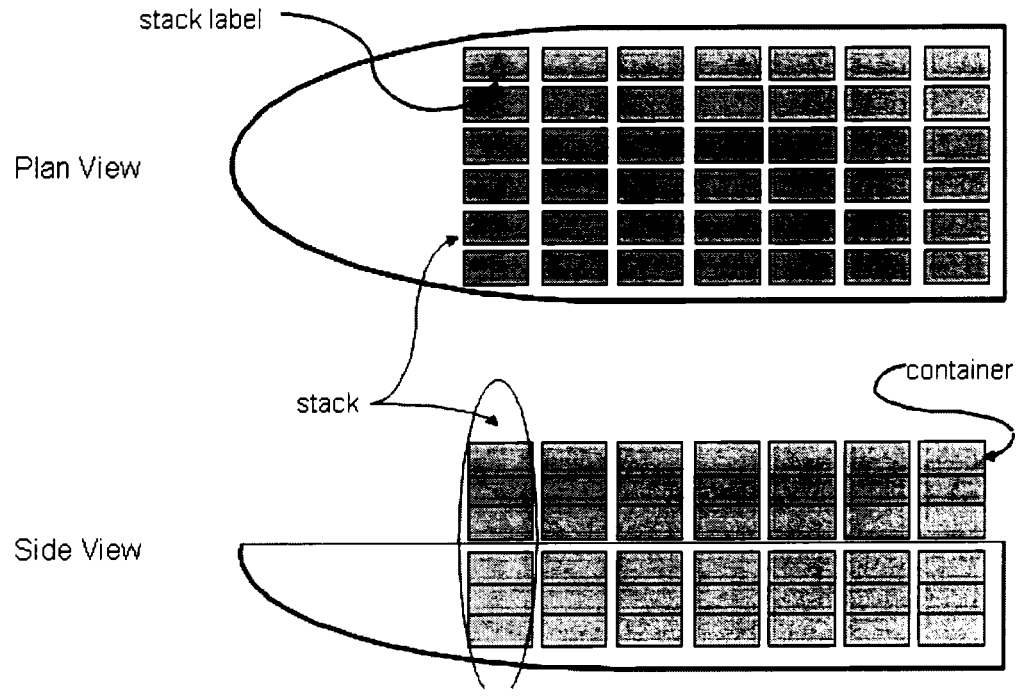


Figure 2.1: Plan and side views of a simplified ship (number of containers shown not representative of typical ship size).

capacity Malacca-max carriers. [1]. Figure 2.1 gives a top and side view of a typical vessel (although the number of container stacks is not representative).

The complete operating cycle of the crane can be broken down into the following components.

1. locking to or unlocking from a container,
2. lengthwise motion along the ship,
3. horizontal motion of the container and/or trolley across the ship,
4. vertical motion of the container and/or trolley.

[1]Most containers in the US are the equivalent of 2 TEUS, or 1 FEU, a forty foot equivalent unit

It is important to point out that the number of locking and unlocking operations is not affected by double cycling. The same number of containers will need to be picked up and put down with single or double cycling. In this research we will assume that dockside containers are ready for loading when required, and containers being unloaded can be quickly removed from the immediate area. In the initial analysis it will be assumed that ships lack hatch-coverings, or doors on the deck that separate above-deck and below-deck storage. This assumption will be relaxed in Section 2.5

Consider the case where a ship arrives in port with a set of containers on board to be unloaded and a loading plan for containers to be loaded. The loading plan indicates the placement of containers on the ship. Given are u_c and l_c, the number of containers to be unloaded and loaded, respectively, in each stack labelled c. Figure 2.2 is an example problem that will be used for illustrative purposes. Notice that in Fig. 2.2, $u_A = 3$ and $l_A = 2$. A rehandle is a container that must be moved to access containers below it, but will then be stowed again before the ship departs. Note that if any rehandles are necessary, we include these in the total number of loads and unloads. For example, if, during unloading, a container must be moved to access a container beneath it, the container moved will be counted as an unload. This container would then also be counted as a load when it is placed back in the original stack. We always assume that rehandles are replaced back into the stack from which they were removed. Note that this will over-estimate the amount of work necessary to unload and load a set of containers because the container is considered moved from the vessel to the shore and back to a location on the vessel. In this work we consider a move to be between the vessel and the apron, but in reality some rehandles may only be moved between locations on the vessel, typically a shorter distance.

The set of stack labels is called S. An ordering of these stacks can be described by a

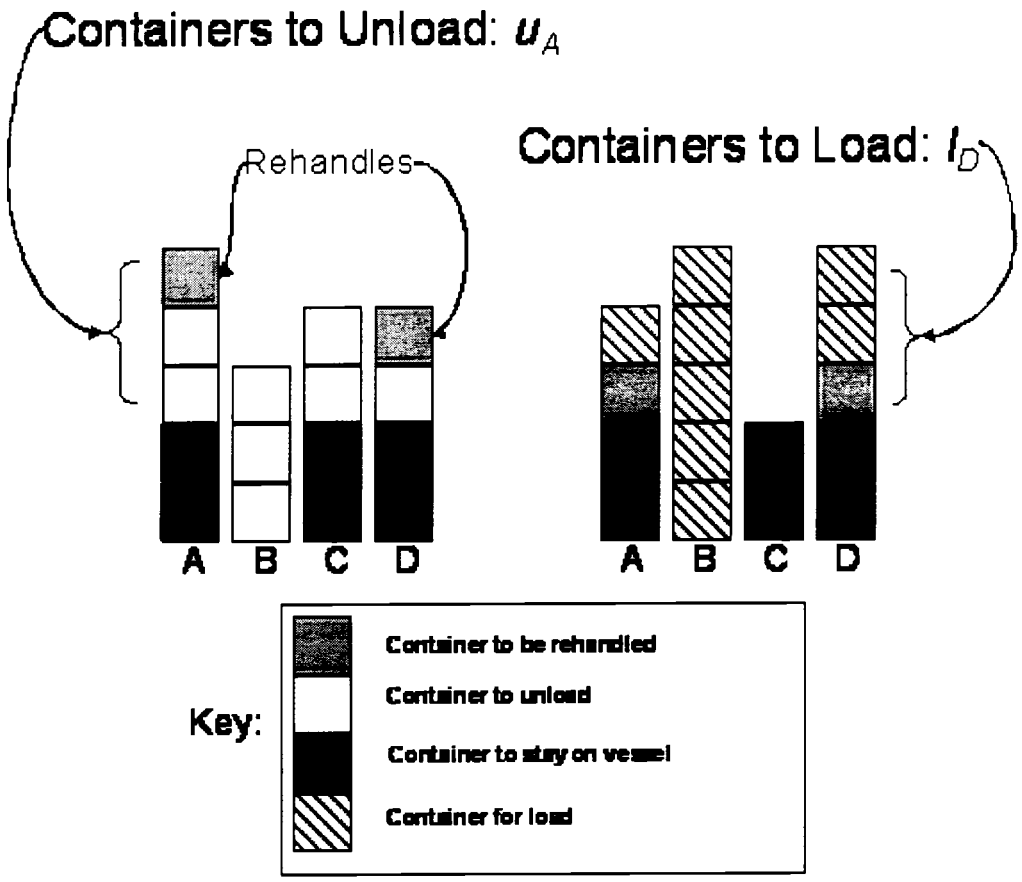


Figure 2.2: Detailed plan for containers to be unloaded and loaded.

permutation function, Π. A permutation is a one-to-one correspondence between the set of $n \in \{1, \ldots, N\}$ and $c \in S$ such that $\Pi(n) = c$, or $n = \Pi^{-1}(c)$. For example, in Fig. 2.2, the set of stack labels is $S = \{A, B, C, D\}$. A permutation of these is $\{B, A, C, D\}$ given by the function Π_e (a permutation for our example) where $\Pi_e(1) = B$, $\Pi_e(2) = A$, $\Pi_e(3) = C$, and $\Pi_e(4) = D$.

If we consider the time it takes to unload and load a ship to be a measure of crane efficiency, then the goal of double cycling is to reduce the total turn around time. A proxy for this is the number of cycles required to unload and load the ship. We will discuss time-savings in Section 3.1. The number of cycles necessary to complete loading and unloading will be represented by the variable w. We will consider double cycling within one row of the ship. Due to the difficulty with which the crane moves laterally along the ship, it is not practical to consider double cycling across two rows. We will complete unloading and loading of one row before moving the crane lengthwise along the ship to the next row. Using this method we will not require the crane to move laterally along the ship within one cycle. We will first determine the number of cycles to complete operations on a row. When we consider time savings in Section 3.1.1, we will consider the time required for the crane to move laterally along the vessel. We will restrict our attention to special cases of the generic double-cycling method described below:

- Choose an unloading permutation, Π'. Unload all containers in the first stack of the permutation, then all containers in the second stack of the permutation, proceed in this fashion until all stacks have been unloaded.

- Choose a loading permutation, Π, and load the stacks in that order. Load all containers in the first stack, then in the second, etc. Loading can start in any stack as soon as it is empty or it

contains just containers that should not be unloaded at this port. Once loading has begun in a stack, continue loading until that stack is complete.

Figure 2.3(a) is a queueing diagram for a single-cycling operation where the stacks of Fig. 2.2 are handled in the order $\{A, B, C, D\}$ both for loading and unloading. Time is expressed in cycles. Note that loading operations must wait until cycle $w = 10$, when unloading is finished. The process requires $w = 20$ cycles. With single cycling, the crane unloads each row of the vessel before loading any containers.

If we double-cycle, we can still plot the unloading curve on the same diagram. Now, using the same sequence for unloading and loading, $\Pi' = \Pi = A, B, C, D$, we can shift the loading curve to the left as far as possible without overlapping the unloading curve. Figure 2.3(b) shows the maximum shift. Loading can start as early as $w = 4$ and the process would require only 14 cycles. The same number of cycles is obviously obtained if we start loading each stack as early as possible, as in Fig. 2.3(c). This introduces some delay as the loading operations must wait one cycle for the unloading operations to be completed in stack B, but does not change the completion time.

With single-cycling one cycle is required for every container. With double-cycling, however, the number of cycles will depend on the sequence. Fig. 2.3(d) shows that if the loading and unloading sequence is B, A, C, D, then the completion time is $w = 13$.

This framework considers the work of one crane, working on individual rows of a vessel. This does not limit our analysis to operations where only one crane works each vessel, as it can be reproduced for each crane, assuming the working areas of the vessel can be segmented by crane.

In the next section we present a lower bound on the number of cycles using any algorithm.

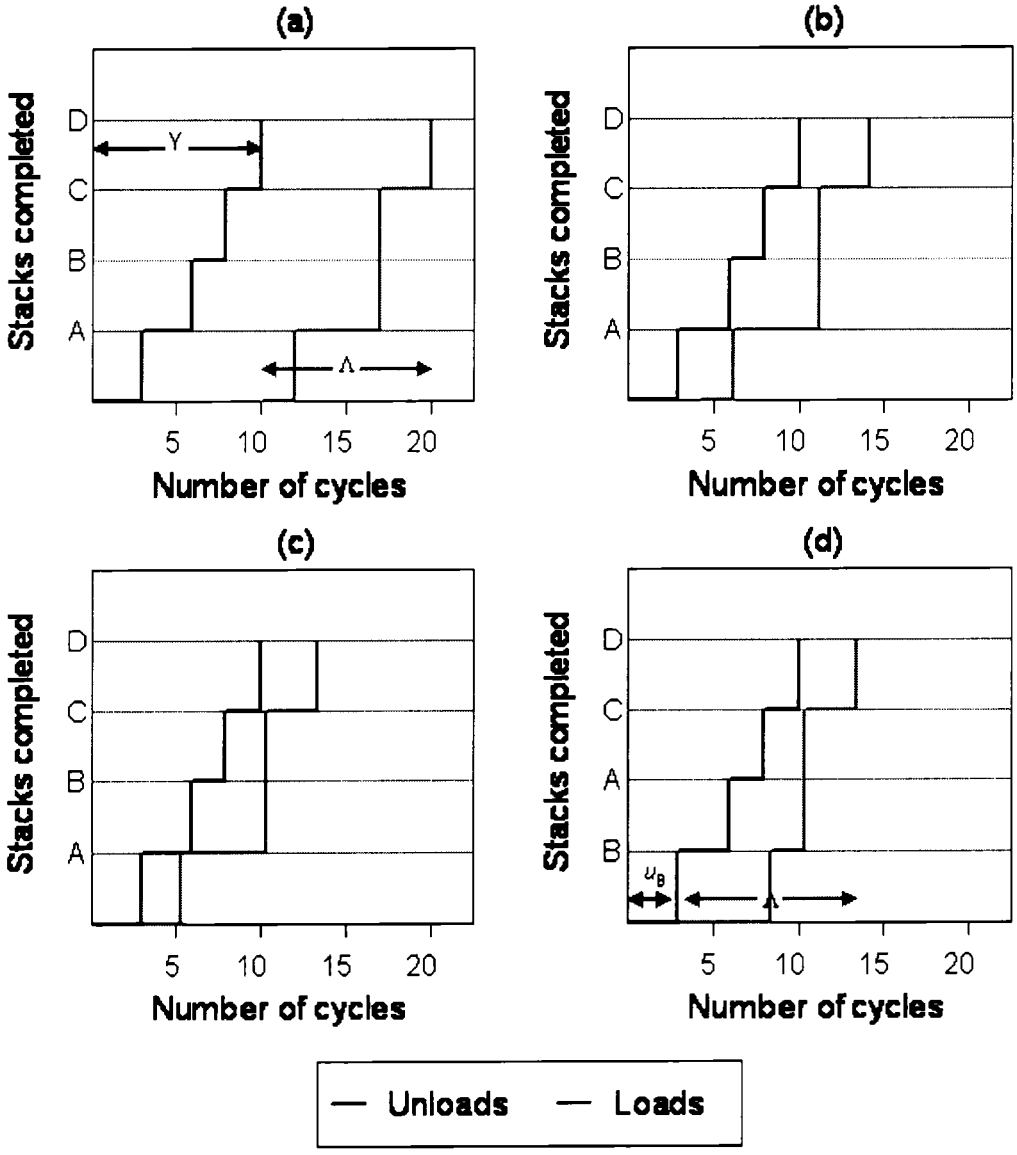


Figure 2.3: Turn around time with different methods: (a) single-cycling with ordering A,B,C,D unloading starts at w = 10, 20 cycles (b) double-cycling with ordering A,B,C,D unloading starts at w = 4, 14 cycles. (c) double-cycling with ordering A,B,C,D unloading starts at w = 3, 14 cycles. (d) double-cycling ordering B,A,C,D unloading starts at w = 3, 13 cycles.

2.2 Lower Bound

Define:

$$Y = \sum_{n=1}^{N} u_{\Pi'(n)} = \sum_{c \in S} u_c, \Lambda = \sum_{n=1}^{N} l_{\Pi(n)} = \sum_{c \in S} l_c. \tag{2.1}$$

Recall from Fig. 2.3(a) that using single-cycling, the number of cycles necessary to complete a row is:

$$Y + \Lambda \tag{2.2}$$

This is intuitive: one cycle for each container. We have assumed the crane must start and finish on the dock.

For double-cycling, with a specific loading permutation Π and unloading permutation Π', assuming $\Lambda \geq Y$, the number of cycles must be at least $\Lambda + u_{\Pi'(1)}$ which satisfies:

$$\Lambda + u_{\Pi'(1)} \geq \Lambda + \min_c(u_c). \tag{2.3}$$

The right-hand side of (2.3) is a general lower bound that applies to all permutations when $\Lambda \geq Y$. It is also a lower bound if we force the same permutation for loads and unloads ($\Pi' = \Pi$). It is well known that there is no loss of optimality by imposing the same sequence on both machines (see Johnson [40]. It is also well known that that if the processing times (number of containers to unload and load in each stack) are interchanged then an equivalent inverse problem results. Therefore, in the case were $Y \geq \Lambda$, the number of cycles must be at least $Y + l_{\Pi(1)}$ which satisfies:

$$Y + l_{\Pi(1)} \geq Y + \min_c(l_c). \tag{2.4}$$

More generally, if w is the number of cycles using any algorithm for double cycling, we can write:

$$w \geq \max\left\{\Lambda + \min_c(u_c), Y + \min_c(l_c)\right\}. \tag{2.5}$$

We will now discuss an algorithm that was developed to keep the loading and unloading operations separate, but would be easy to implement, and would provide mathematical insight. It provides significant operational flexibility, and an upper bound formula for the required number of cycles. The formula can be evaluated without running an algorithm, and thus is useful for planning purposes.

2.3 Greedy Strategy and an Upper Bound

We propose to unload and load each stack as soon as possible, assuming that the loading and unloading sequences are given by the same "greedy" permutation, $\Pi' = \Pi = G$. This greedy permutation is obtained by ordering the stacks in descending order of the variable d_c where:

$$d_c = l_c - u_c \quad when \quad \Lambda \geq Y \tag{2.6}$$

$$d_c = u_c - l_c \quad when \quad Y > \Lambda \tag{2.7}$$

The rationale for 2.6 is that we want the unloading operations to run ahead of the loading operations as much as possible.

We will assume in this section that stacks have been labelled by position in the handling sequence with the greedy strategy. So now, u_j is the number of containers to unload in the jth stack, $j = 1..J$, or equivalently, the number of cycles required to unload the jth stack. Assume there are more loads than unloads, $\Lambda \geq Y$. Let U_j be the cumulative time (in number of cycles) at which the jth stack is finished unloading: $U_j = u_1 + u_2 + ... + u_j$, where the sequence is determined by the greedy strategy. Recall l_j is the number of containers to load in the jth stack, or equivalently the number of cycles required to load the jth stack. Let L_j be the combined operational time (in number of cycles) to load j stacks: $L_j = l_1 + l_2 + ... + l_j$, where the sequence is determined by the greedy strategy.

Given $\Lambda - Y \geq 0$, define $d_j = l_j - u_j$ and $D_j = L_j - U_j = d_1 + d_2 + ... + d_j$. Notice that $d_{j-1} \geq d_j$ because our strategy is greedy. Notice also that $D_J = \sum_{c \in S} d_c = \Lambda - Y \geq 0$ where because there are more (or equal) loads than unloads.

Lemma 2.3.0.1 *If there are more (or equal) loads than unloads then* $D_j \geq 0 \; \forall j$.

Proof Since there are more (or equal) loads than unloads, $D_J \geq 0$. Assume now that $D_j < 0$ for some j. For D_j to be negative, one of its components, d_k , must be negative for some $k \leq j$. This, however, would mean that $d_r < 0 \; \forall \; r \geq k$, and also for $r \geq j$ (since the d_j sequence is decreasing for the greedy strategy). Hence $D_J = D_j + d_{j+1} + ... + d_J$ is a sum of negative terms and would be negative. But this is a contradiction. Thus, there cannot be a $D_j < 0$ for some j. ∎

Lemma 2.3.0.2 *If there are more (or equal) loads than unloads, the number of cycles to complete operations with the greedy strategy,* w_G, *satisfies* $w_G \leq \Lambda + \max_c u_c$.

Proof Introduce s as the time (number of cycles), between the origin, and the beginning of the loading operation. If there are no intermediate delays, then the time to begin loading the jth stack,

B_j, is $B_j = s + L_{j-1}$, ie the shift plus the time to load $j-1$ stacks (define $L_0 = 0$). Clearly now, if $B_j - U_j \geq 0 \forall j$ then there will be no intermediate delays to loading, and we have found a strategy with at most $s + \Lambda$ cycles. We now show that this is true. Note that $\{B_j - U_j\} = [s - u_j] + [L_{j-1} - U_{j-1}]$ and that the first term on the right side is non-negative if we choose $s = \max_{c \in S} u_c$. We also see that the second term is non-negative by Lemma 2.3.0.1. Thus, $B_j - U_j \geq 0 \forall j$ as claimed, and there are no intermediate delays if $s = \max_{c \in S} u_c$. Therefore, $w_G \leq \Lambda + \max_{c \in S} u_c$. ∎

Lemma 2.3.0.3 *If* $\Lambda \leq Y$, *then* $w_G \leq Y + \max_c(l_c)$.

Proof G in this case is defined by 2.7. That Lemma 2.3.0.3 is true should be obvious by symmetry. If one were to record the process of unloading and loading the row, and then play this recording in reverse, the reversed movie would display a sequence of operations with the same total time as for a problem in which the role of loads and unloads is switched. Thus, for every problem instance with loads greater than unloads, there is a dual instance where unloads are greater than loads. Figure 2.4(b) shows the reverse movie of Fig. 2.4(a). Note that the greedy strategy continues to be greedy in reverse, and that everything said up to this point, including the bounds, continues to hold with time running backward. Note that in Fig. 2.4(b) the greedy strategy with time running backward implies a reverse ordering of $\{d_c\}$, as specified in 2.7, thus Lemma 2.3.0.3 holds. ∎

Define $u' = \min_c(u_c)$, $l' = \min_c(l_c)$, $u^* = \max_c(u_c)$, and $l^* = \max_c(l_c)$. We can now state the following theorem:

Theorem 2.3.0.4 $\max(\Lambda + u', Y + l') \leq w^* \leq w_G \leq \max(\Lambda + u^*, Y + l^*)$.

Proof See the proofs for Lemmas 2.3.0.1, 2.3.0.2 and 2.3.0.3.

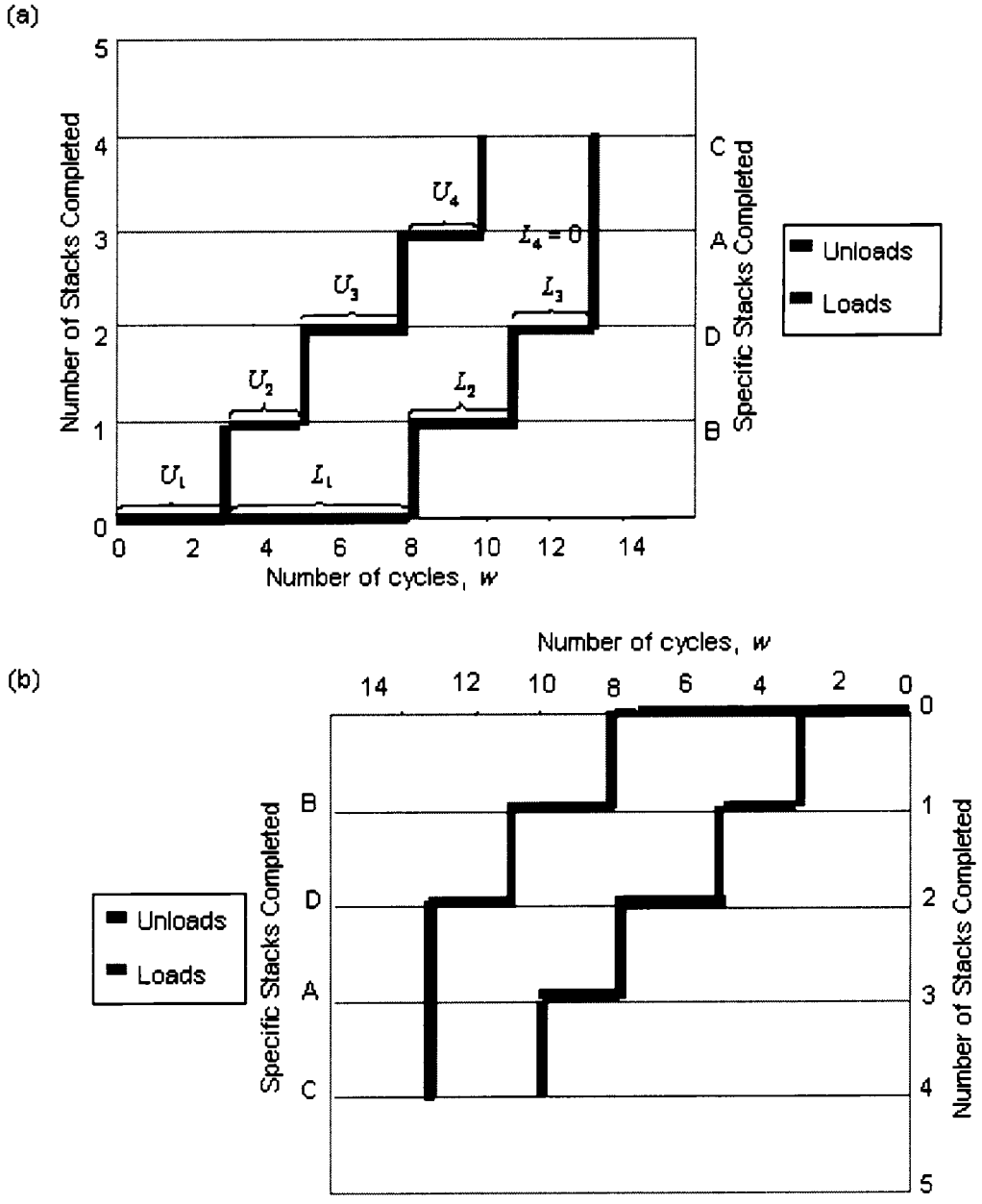


Figure 2.4: (a) Definition of the greedy strategy (b) Rotated version of part (a).

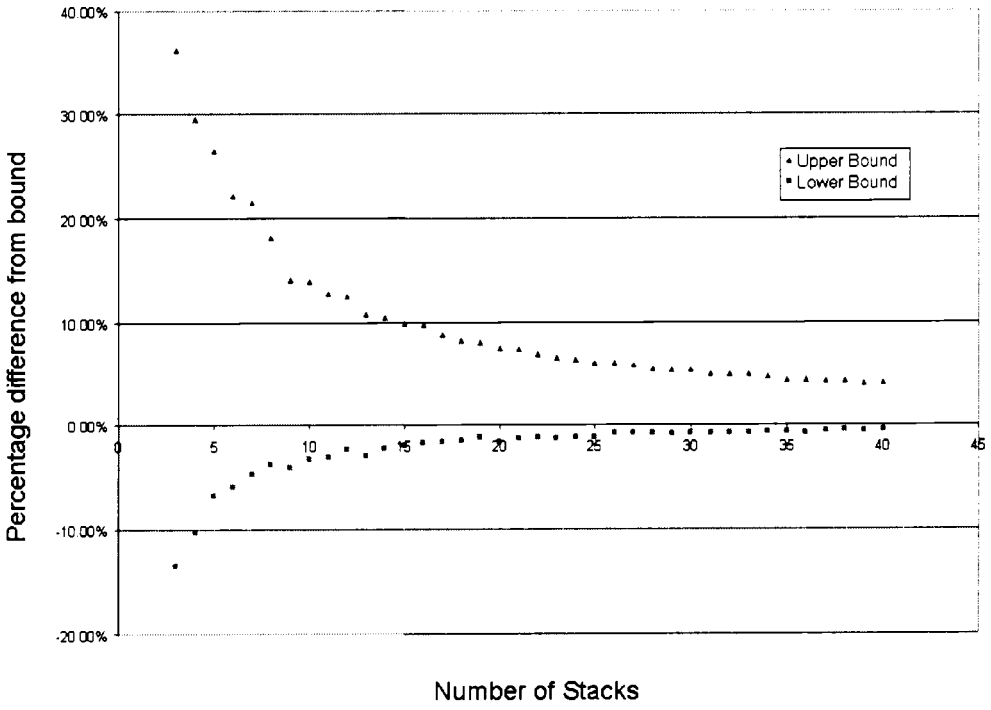


Figure 2.5: The greedy strategy is asymptotically optimal.

For rows of large ships where $\{\Lambda, Y\} >> (u^*, l^*)$, the greedy strategy is very close to both the upper and lower bounds since all the members of 2.3.0.4 are relatively close to $\max(\Lambda, Y)$. More specifically, note that if u_c and l_c are bounded by a constant (stack size) then the percent optimality gap vanishes as the number of stacks (problem size) tends to infinity. Thus the greedy strategy is asymptotically optimal. This is visually represented in Figure 2.5. The parameter values used to generate the vessel data are described in Table 2.1.

If we apply the greedy algorithm to the example problem used in Section 2.1, we obtain the sequence B, D, A, C. This sequence would require 13 cycles to complete unloading and loading

Parameter	Setting
Number of stacks	varies (as indicated in the figure)
Beta Distribution Parameters	5 runs with $p_i = 1, q_i = 1, p_e = 1, q_e = 2$ 5 runs with $p_i = 1, q_i = 1, p_e = 2, q_e = 1$ 10 runs with $p_i = 1, q_i = 1, p_e = 2, q_e = 2$ 5 runs with $p_i = 2, q_i = 2, p_e = 1, q_e = 1$ 10 runs with $p_i = 2, q_i = 2, p_e = 2, q_e = 2$ 5 runs with $p_i = 2, q_i = 2, p_e = 2, q_e = 1$
Maximum number of imports in one stack	20
Maximum number of exports in one stack	20

Table 2.1: Parameter settings used to generate vessels data.

operations. In the next section we formulate the problem of determining Π' and Π as a mixed integer program and describe an optimal solution algorithm.

2.4 Scheduling Approach

The problem is formulated as a two-machine flow shop scheduling problem. There is one job corresponding to each stack, and each job has two operations: an unloading operation that must be completed first, and a subsequent unloading operation. In the scheduling representation, there is a machine for unloading and a separate machine for loading, but in actual implementation, the crane performs both types of operations. One cycle corresponds to one time unit in the scheduling problem. In a single cycle, the crane loads or unloads a single container. In a double cycle, the crane loads a container in one job (i.e., in one stack) and unloads a container in another job.

The problem is to determine the best sequence of unloading and of loading, to minimize the maximum completion time (makespan). Start times are identified from the known job durations. We use the following notation:

- u_c - number of containers to unload in stack $c \in S$
- l_c - number of containers to load in stack $c \in S$
- FU_c - completion time of unloading $c \in S$
- FL_c - completion time of loading $c \in S$
- w - maximum completion time
- X_{kj} - binary variable for ordering of unloading jobs (1 if $j \in S$ is unloaded after $k \in S$ and 0 otherwise)
- Y_{kj} - binary variable for ordering of loading jobs (1 if $j \in S$ is loaded after $k \in S$ and 0 otherwise)
- M - a large number

The scheduling problem (SP) is to minimize the maximum completion time of all jobs subject to constraints. The result is to uniquely identify the permutations Π and Π', and a feasible set of job start and end times. It is assumed that the process starts at time zero. The formulation is:

$$(SP) \qquad minimize \quad w \tag{2.8a}$$

$$subject \quad to \quad w \geq FL_c \quad \forall c \in S, \tag{2.8b}$$

$$FL_c - FU_c \geq l_c \quad \forall c \in S, \tag{2.8c}$$

$$FU_k - FU_j + MX_{kj} \geq u_k \quad \forall j,k \in S, \tag{2.8d}$$

$$FU_j - FU_k + M(1 - X_{kj}) \geq u_j \quad \forall j,k \in S, \tag{2.8e}$$

$$FL_k - FL_j + MY_{kj} \geq l_k \quad \forall j,k \in S, \tag{2.8f}$$

$$FL_j - FL_k + M(1 - Y_{kj}) \geq l_j \quad \forall j,k \in S, \tag{2.8g}$$

$$FU_c \geq u_c \quad \forall c \in S, \tag{2.8h}$$

$$X_{kj}, Y_{kj} = 1,0 \quad \forall j,k \in S. \tag{2.8i}$$

These constraints completely define the double-cycling problem. Constraints 2.8b ensure that the makespan is greater than or equal to the completion of loading of all stacks. Constraints 2.8c ensure that stacks are only loaded after all necessary stacks have been unloaded. Constraints 2.8d, 2.8e and 2.8i ensure that every stack is unloaded after the previous one in Π' has been unloaded. This is achieved by specifying for every pair of stacks (j,k) that either stack k is unloaded before stack j (if $X_{kj} = 1$) or the reverse (if $X_{kj} = 0$), and that the time difference between the two events is large enough to unload the second of the two stacks. Constraints 2.8f, 2.8g and 2.8i are equivalent to 2.8d, 2.8e and 2.8i but for loading jobs. Constraints 2.8h ensure that each unloading completion time allows for enough time to at least unload that stack.

It should be noted that the assumption of continuous loading of a stack and continuous unloading of a stack is without loss of optimality. In other words, preemption cannot improve the

solution. Johnson [40] developed an optimal solution algorithm for the 2-machine, unconstrained problem, that can be used to solve the formulation above. It is well known that for two machine problems it is sufficient to consider schedules in which the processing orders on the two machines are identical. It is also well known that if the processing times are interchanged then an equivalent inverse problem results.

2.4.1 Johnson's Rule

The more formal theorem and proof of the following working rule is described in Johnson's paper [40]. Here we describe the working rule used to find the optimal solution:

1. List the u_c's and l_c's in two vertical columns.
2. Scan all stacks for the smallest value.
3. If it is for the first machine (i.e., a u_c), place the corresponding item first.
4. If it is for the second machine (i.e., an l_c), place the corresponding item last.
5. Cross off both u_c and l_c for that stack.
6. Repeat the steps on the reduced set of $2n - 2$ time intervals, etc. Thus we work from both ends toward the middle.
7. In case of ties, for the sake of definiteness, order the item with the smallest subscript first. In case of a tie between u_c and l_c, order the item according to the u_c.

To illustrate the method, the optimal sequence for the example shown in Section 2.1 is derived. The data are shown in Table 2.2.

c	u_c	l_c
A	3	2
B	3	5
C	2	0
D	2	3

Table 2.2: Example problem data to illustrate Johnson's rule

The smallest value is l_C, and because these are containers for loading, we place C last in the sequence. Of the remaining values, the smallest is 2, with a tie between l_A and u_D. In this case Johnson's rule indicates we should order the item with the smallest subscript first, so here we choose to place A second to last in the operational sequence. Then we place D first in the sequence, since the two containers in D are for unloading. To complete the sequence we place B second. We thus develop the following sequence: $\{D, B, A, C\}$. Notice with this sequence, which is different from the greedy sequence, we require 12 cycles to complete the unloading and loading operation. This is fewer than the 13 cycles required with the greedy algorithm.

Figure 2.6 compares the performance of the scheduling and greedy algorithms on a larger set of data. The parameter values used to generate the vessels represented in this figure are shown in Table 2.1, using the program described in Section 2.6. The percentage difference between the solution using the greedy strategy, and Johnson's rule, is shown. Each data point shows the average difference across 40 vessels. The difference is small, less than 2.5% even for 3 stacks. As expected, the greedy algorithm's performance improves with row size. For 20 stacks, a typical row size, the difference is between 0.5 and 1.0% for the data simulated.

Although we can solve the problem optimally with Johnson's rule, the greedy algorithm provides simple and tight bounds which can be used to estimate the turn around time, and an intu-

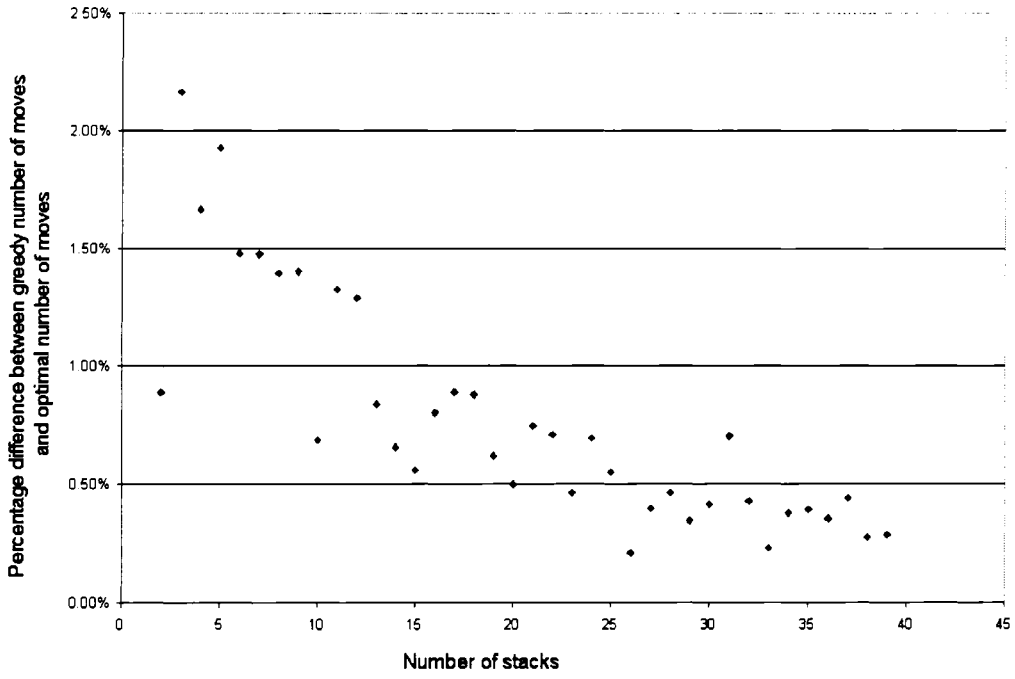


Figure 2.6: Performance comparison of the scheduling and greedy algorithms. Each point shows the percentage difference between the greedy solution and the optimal solution. Each point shows the average across 40 generated vessels.

ition for this result.

In the next section, we extend the analysis to vessels with deck hatches.

2.5 Deck Hatches

Most container ships currently have hatch coverings, as shown in Figure 2.7. These are large steel plates that separate above-deck and below-deck storage. In Fig. 2.7(a) we have shown a vessel two hatches wide, with three stacks of containers above and below each hatch. In Fig. 2.7(b) we have shown a vessel three hatches wide, with five stacks above and below each hatch. Typical ships are three hatches wide, with each hatch containing 5 or 6 stacks of containers. Hatches change

the nature of the problem already addressed because the stacks are no longer independent. To access the containers below a hatch all containers must be unloaded from above the hatch, and before loading containers atop a hatch all containers below the hatch must be loaded.

We propose the following algorithm (the greedy hatch strategy). As in the hatchless case, assume that each stack on the ship is given an initial label. To carry out the strategy it is necessary to:

1. Order the *hatches* using a greedy strategy using the same method as for the hatchless case. Treat the hatches as stacks, considering only the containers atop the hatches.
2. Order the *stacks* within each hatch using a greedy strategy, considering only the containers below deck.

The algorithm is then as follows:

1. Apply the greedy strategy to the containers above deck, treating hatches as stacks, pausing each time all containers above hatch h have been removed.
2. During the hth pause, unload and load the containers below the hth hatch using the greedy strategy for double-cycling.

As with the hatchless case, this method may not provide the fewest cycles to complete loading and unloading operations on the row. For example while loading the last stack below a hatch we could double cycle by unloading atop the next hatch. In addition, we can always treat many hatches as one hatch. Depending on the distribution of containers, this may be advantageous. The value of the greedy hatch strategy is its simplicity; parts of the unloading and loading process are equivalent to the hatchless ship problem already addressed. Each piece below a hatch is equivalent to

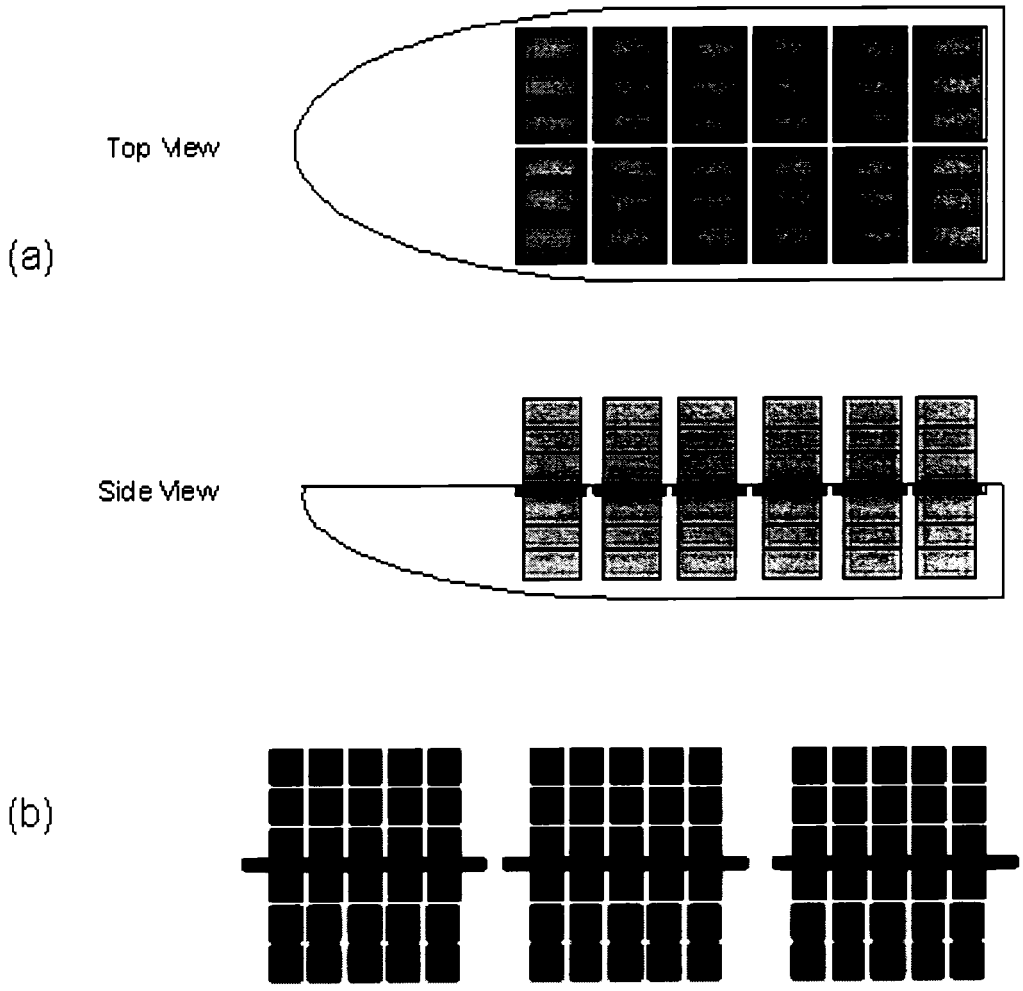


Figure 2.7: (a) Top view and side view of a ship with hatch coverings (b) Rear view of a simple hatched ship.

a hatchless ship, and the containers above a hatch to a stack of a hatchless ship. These relationships will allow us to use the analysis of the hatchless ship to develop bounds for the hatched case. First it is necessary to define some notation.

- h - hatch index
- S_h - the set of stacks for hatch h
- N_h - the number of stacks above or below hatch h
- H - the set of hatches
- F - the number of hatches
- $\overline{u}_{hc}$ - the number of containers to unload below hatch h in stack $c \in S_h$
- $\underline{u}_{hc}$ - the number of containers to unload above hatch h in stack $c \in S_h$
- $\overline{l}_{hc}$ - the number of containers to load below hatch h in stack $c \in S_h$
- $\underline{l}_{hc}$ - the number of containers to load above hatch h in stack $c \in S_h$
- $\overline{u}_h = \sum_{c \in S_h} \overline{u}_{hc}$ - containers to unload below hatch h
- $\underline{u}_h = \sum_{c \in S_h} \underline{u}_{hc}$ - containers to unload above hatch h
- $\overline{l}_h = \sum_{c \in S_h} \overline{l}_{hc}$ - containers to load below hatch h
- $\underline{l}_h = \sum_{c \in S_h} \underline{l}_{hc}$ - containers to load above hatch h
- $\overline{Y} = \sum_{h \in H} \sum_{c \in S_h} \overline{u}_{hc}$ - containers for unloading below deck
- $\underline{Y} = \sum_{h \in H} \sum_{c \in S_h} \underline{u}_{hc}$ - containers for unloading above deck

- $\overline{\Lambda} = \sum_{h \in H} \sum_{c \in S_h} \overline{l}_{hc}$ - containers for loading below deck
- $\underline{\Lambda} = \sum_{h \in H} \sum_{c \in S_h} \underline{l}_{hc}$ - containers for loading above deck
- w_A - the number of cycles above deck
- w_B - the number of cycles below deck
- $w_{B,h}$ - the number of cycles below hatch h
- P - the greedy permutation for hatches
- $G(\cdot|h)$ - the greedy permutation for containers below hatch h; i.e. $G(n|h) = c \in S_h, G^{-1}(c|h) = n$ for $c \in S_h$, $n = \{1...N_h\}$

Theorem 2.5.0.1 *An upper bound on the optimum number of cycles for the hatched case is* $\sum_{h \in H} \max\left\{\overline{u}_h, \overline{l}_h\right\} + \max\left\{\underline{\Lambda}, \underline{Y}\right\} + \max_h \left\{\underline{u}_h, \underline{l}_h\right\} + \max_{c \in S_h} \left\{\overline{u}_{hc}, \overline{l}_{hc}\right\}$.

Proof The number of cycles for a row of F stacks with data given by $\{\underline{u}_h, \underline{l}_h\}$ is w_A. We know from (2.3.0.4) that

$$w_A \leq \max\left\{\underline{\Lambda}, \underline{Y}\right\} + \max_h \left\{\underline{u}_h, \underline{l}_h\right\}. \tag{2.9}$$

Likewise the number of cycles for a row of N_h stacks with data given by $\left\{\overline{u}_{hc}, \overline{l}_{hc}\right\}$ is w_B. We know that

$$w_B = \sum_{h \in H} w_{B,h} \leq \sum_{h \in H} \max\left\{\overline{u}_h, \overline{l}_h\right\} + \max_{c \in S_h} \left\{\overline{u}_{hc}, \overline{l}_{hc}\right\}. \tag{2.10}$$

Obviously then, the total number of cycles with the algorithm satisfies

$$w_A + w_B \leq \sum_{h \in H} \max\left\{\overline{u}_h, \overline{l}_h\right\} + \max\left\{\underline{\Lambda}, \underline{Y}\right\} + \max_h \left\{\underline{u}_h, \underline{l}_h\right\} + \max_{c \in S_h} \left\{\overline{u}_{hc}, \overline{l}_{hc}\right\}. \quad \blacksquare \quad (2.11)$$

Theorem 2.5.0.2 *A lower bound on the optimum number of cycles for the hatched case is*

$\sum_{h \in H} \max\left\{\overline{u}_h, \overline{l}_h\right\} + \max\left\{\underline{\Lambda}, \underline{Y}\right\} + \min_h \left\{\underline{u}_h, \underline{l}_h\right\} + \min_{c \in S_h} \left\{\overline{u}_{hc}, \overline{l}_{hc}\right\}$.

Proof We know from (2.5), but treating each hatch as a stack, that the lower bound on the number of cycles above deck is

$$\max\{\underline{\Lambda}, \underline{Y}\} + \max_h\{\underline{u}_h, \underline{l}_h\}. \quad (2.12)$$

We also know from (2.5), but treating each hatch as a vessel, that the lower bound on the number of cycles below deck is

$$\sum_{h \in H} \max\{\overline{u}_h, \overline{l}_h\} + \max_{c \in S_h}\{\overline{u}_{hc}, \overline{l}_{hc}\}. \quad \blacksquare \quad (2.13)$$

Clearly, for rows where $\sum_{h \in H} \max\{\overline{u}_h, \overline{l}_h\} + \max\{\underline{\Lambda}, \underline{Y}\} >> \max_h\{\underline{u}_h, \underline{l}_h\} + \max_{c \in S_h}\{\overline{u}_{hc}, \overline{l}_{hc}\}$, both the upper and lower bounds are close to the solution provided by the greedy strategy and $\sum_{h \in H} \max\{\overline{u}_h, \overline{l}_h\} + \max\{\underline{\Lambda}, \underline{Y}\}$ provides a reasonable estimate of the total unloading and loading time for one row. As with the hatchless case, the gap between the upper and lower bound is quite small, and decreases with the size of the row. Thus, the simple greedy algorithm is reasonably efficient.

If one double-cycles only below deck, as is current practice, the benefits of double-cycling will be reduced by roughly the ratio of containers unloaded and loaded above deck to containers unloaded and loaded both above and below deck. In the next section we introduce the details of

a computer program, used to compare strategies for double cycling. Results comparing double cycling without hatches (or one for each stack), to only below deck, are shown in Section 2.7.1.

2.6 Computer Program

A computer program was developed that generates ship data, and calculates the number of moves to turn around the ship using various strategies. The purpose of the program was to provide large sets of data upon which to evaluate algorithms. The program provides turn around times (in number of moves) using single cycling, the greedy algorithm, Johnson's rule, and the proximal stack strategy (described Section 2.7). The program can generate data for many different ship designs and market conditions (imports and exports, origins and destinations, and number of stacks). Table 2.3 describes how different parameter values can be used to simulate different market environments.

Necessary inputs into the simulator are:

- number of stacks
- h_i, maximum stack height (in containers) for imports
- h_e, maximum stack height (in containers) for exports
- p_i and q_i, parameters for the beta distribution, applied to imports
- p_e and q_e parameters for the beta distribution, applied to exports

The number of containers to unload and load in each stack, u_c and l_c, are assumed to be independent and generated by sampling from a beta distribution. According to the input parameters p_i, q_i, p_e, and q_e, the beta distribution is sampled twice for each stack, obtaining a value between 0

Description	Parameter Values
Shallow (River Boat)	Maximum stack height ≤ 10
Deep (Ocean Vessel)	Maximum stack height > 10
Small Vessel	Number of stacks in row ≤ 10
Medium Vessel	Number of stacks in row $> 10, \leq 15$
Large Vessel	Number of stacks in row > 15
Balanced Load (similar demand for imports and exports)	maximum stack height, p and q same for imports and exports
Unbalanced Load (different demand for imports and exports)	maximum stack height, p and q different for imports and exports
Varying Demand and Loading	Vary parameters p and q

Table 2.3: How computer program parameters can be varied to reflect different market conditions.

and 1, with a different distribution for imports and exports. Each sampled value is then multiplied by the maximum stack height, then rounded down to the nearest integer. For more information on the beta distribution, see Appendix A. Notice some stacks will have 0 containers.

The program counts the number of cycles required to complete loading and unloading operations on a vessel for the following algorithms:

- Single cycling: this algorithm is described in Section 2.1
- Greedy strategy: this algorithm is described in Section 2.3
- Proximal strategy: this algorithm is described in Section 2.7
- Johnson's rule: this algorithm is described in Section 2.4

In the next section we introduce another strategy, the proximal stack strategy. This strategy is operationally more convenient that using the sequence determined by Johnson's rule or the greedy strategy, and allows for simpler mathematical analysis. It turns out the result is almost optimal.

2.7 The Proximal Stack Strategy

The proximal stack strategy is based on the strategy typically used in current operations. Let R be the number of rows on a vessel, and let C_i be the number of stacks in row i. Rows are numbered $i = 1..R$ starting from the bow. Let $\{1...C_i\}$ be the set of stack numbers, c, in one row of a vessel. The first number, $c = 1$ is the stack closest to the shore, and C_i the stack nearest the water.

Definition 2.7.0.3 (Proximal Stack Strategy) The crane processes rows one at a time in order of increasing i. For each row it:

1. Unloads all containers in the stack closest to the shore, $c = 1$, then all containers in stacks $c = 2, 3$, etc. until all stacks in the row have been unloaded.

2. Loads the stacks using the same ordering. Loading can start in a stack as soon as it is empty or contains just containers that should not be unloaded at this port. Once loading has begun in a stack it is continuously loaded until complete.

We choose to focus on this strategy because it is operationally and mathematically convenient, and the strategy used in practice. For example, this is the method that was used in the real-world trial at the Port of Tacoma (see Section 1.2). We again assume double cycling only takes place within one row. By operating on each row individually we do not require the crane to move laterally along the ship within one cycle.

2.7.1 Comparison of Algorithms on Large Datasets

We use the computer program to generate problem instances and calculate the number of moves for each algorithm, to compare the benefits of the proximal stack strategy to single cycling

and to those of Johnson's rule and the greedy strategy, both when double cycling above deck and below deck, or only below deck. Our comparisons always consider double cycling within one row of the ship. Ship data were generated using the computer program which is described in Section 2.6, and parameter settings shown in Table 2.1. The results comparing the strategies are shown in Figures 2.8 and 2.9. Each data point represents the average of 40 generated vessels.

As expected, the benefits using the proximal stack strategy are smaller than the benefits using the greedy or optimal strategies. For a row of 20 stacks, there is a 45% reduction in number of moves over single cycling for the optimal strategy, a 44% reduction using the greedy strategy, and about a 40% savings for the proximal stack strategy. Notice that the savings range in the figures has been reduced to allow closer comparison of the values, that benefits above 35% are commonplace, and that the benefits of using any of the three strategies are significant. These results assume there are no deck hatches (or one for each stack).

If double cycling only occurs below deck, the results from the three strategies are essentially equivalent. Figure 2.9 shows the percentage savings over single cycling of using Johnson's rule, the greedy, and proximal stack strategies to double cycle only below deck. Notice again the scale of the axis has been adjusted for closer comparison of the strategies. For a vessel with 20 stacks per row, the benefits of all three strategies are between 20 and 22%.

In the next chapter, we expand the scope of consideration to include marine terminal operations.

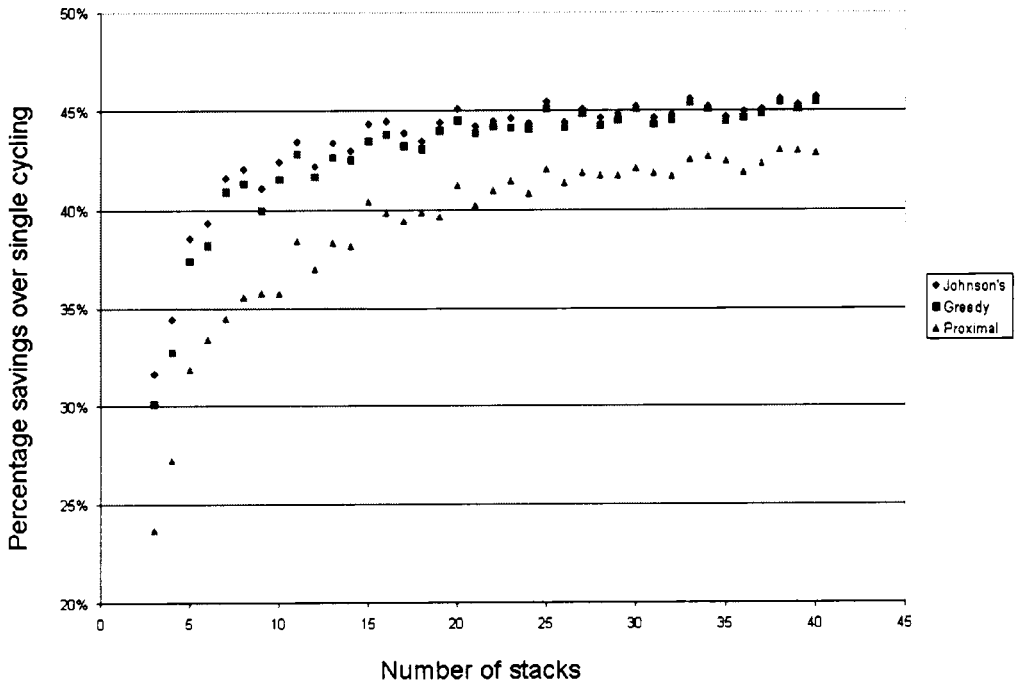


Figure 2.8: Performance comparison of greedy strategy, proximal strategy, and Johnson's rule to single cycling. Each data point shows the percentage savings over single cycling and is the average result for 40 generated vessels.

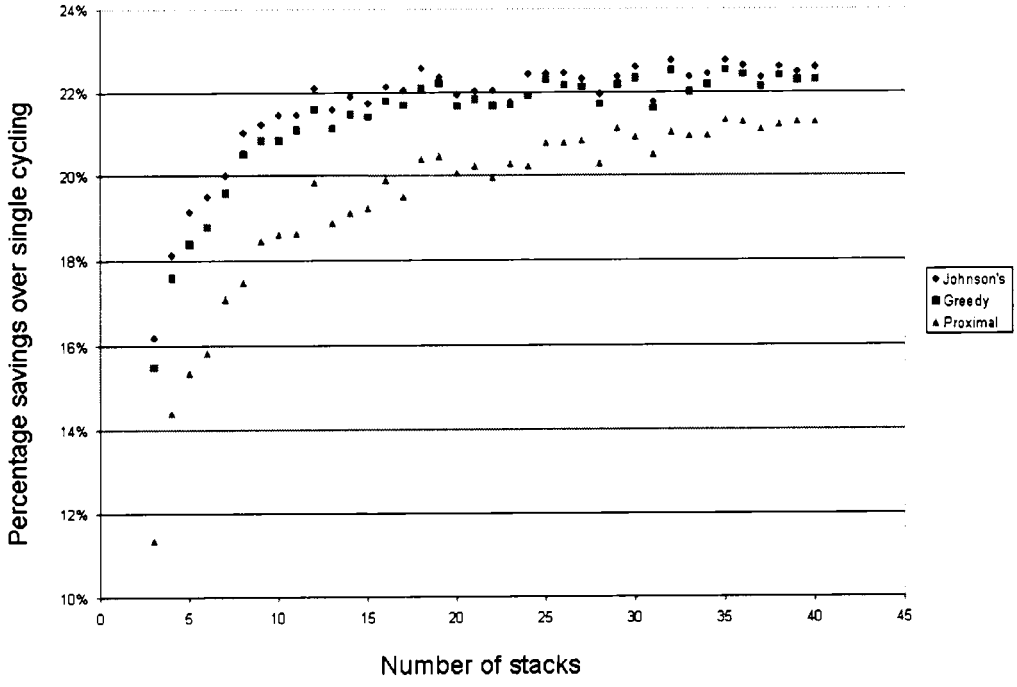


Figure 2.9: Comparison of the greedy strategy, proximal strategy, and Johnson's rule to single cycling when double cycling only below deck. Each data point shows the percentage savings over single cycling and is the average result for 40 generated vessels.

Chapter 3

Impact on Marine Terminal Operations

This chapter expands the scope of the previous chapter, and examines the impact of double cycling landside on marine terminal operations. We evaluate the time savings of a particular reduction in cycles, and compare our analysis to empirical data. We also extend the analysis from that for an individual vessel, to an entire fleet, so that ports can begin planning their operations for double cycling. In Section 3.1 we review results for vessels generated by a computer program (described in Section 2.6), develop a method to convert the benefits from cycles to an amount of time, and validate the results by comparison with empirical data. In Section 3.2, we develop a formula to estimate the benefits for an entire calling fleet. We compare these results to those of vessels generated by the computer program and find the formula to be a very good estimator. Section 3.3 presents the impact of double cycling on landside operations within the port. We conclude with Section 3.4, where we consider some necessary and complementary changes to landside operations, including the impact of double cycling on landside storage space and equipment.

3.1 Evaluation

In this section we review results from the computer program (described in Section 2.6). Using the program we generate a set of ship data representing possible vessel loadings and calculate the number of moves to complete operations with various strategies. We do so in an attempt to understand the benefits for an entire fleet calling, rather than for an individual vessel. We will develop a more robust analysis for fleets in Section 3.2. Later in the section we present tools to convert benefits from number of cycles to an amount of time, and compare these tools to data collected in a real-world trial of double cycling.

The data shown in Figures 2.8 and 2.9 in Section 2.7.1, can provide an overall impression of the benefits of double cycling on a larger scale. If these generated vessels are an exact replica of the universe of future vessels, the error between this predicted benefit, and that which would be realized is 0. But, we expect that these generated vessels are merely a representative sample of the universe served, given the future population of vessels is unknown. The average and standard deviation of the percentage reduction for the three strategies are shown in Tables 3.1 and 3.2. The number of cycles using double cycling is always less than single cycling. Of course, if there are no containers to load, or unload, then double cycling provides no benefit. Similarly, there exist vessel loadings where double cycling will cut the number of cycles almost by half (if we were to have no containers above deck). But, on average, as shown in Table 3.2 when double cycling below deck, there is a 21% reduction in the number of cycles when using the greedy strategy, a 22% reduction with Johnson's rule, and a 19% reduction with the proximal stack strategy. The results when double cycling without deck hatches (or when there is one hatch per stack) are shown in Table 3.1. Of course, this percentage depends on the specific set of vessels analyzed.

Strategy	Average	Standard Deviation
Johnson's Rule	43%	2.94%
Greedy	43%	3.28%
Proximal	39%	4.30%

Table 3.1: Average and standard deviation of benefit over single cycling of three algorithms when double cycling without hatches (or when there is one hatch per stack).

Strategy	Average	Standard Deviation
Johnson's Rule	22%	1.36%
Greedy	21%	1.48%
Proximal	19%	2.16%

Table 3.2: Average and standard deviation of benefit over single cycling of three algorithms when double cycling only below deck.

Each time we replace two single cycles with one double cycle, we reduce the distance travelled by the crane, and save some time. We address time-savings in the next section.

3.1.1 Time-Savings

We now address the benefits of double cycling with respect to reducing operating time. While the number of cycles required to turn around the vessel is a relevant metric, the real benefit comes from reducing operational time consumed by the unloading and loading process. We will use the following notation. Please refer to Figure 3.1.

- W - average time saved for each replacement of two single cycles by one double cycle
- S_r - number of cycles required to turn around row $r \in \{1, \ldots, R\}$ using single-cycling
- D_r - number of cycles required to turn around row $r \in \{1, \ldots, R\}$ using double-cycling
- d_r - number of cycles moving two containers while operating on row $r \in \{1, \ldots, R\}$

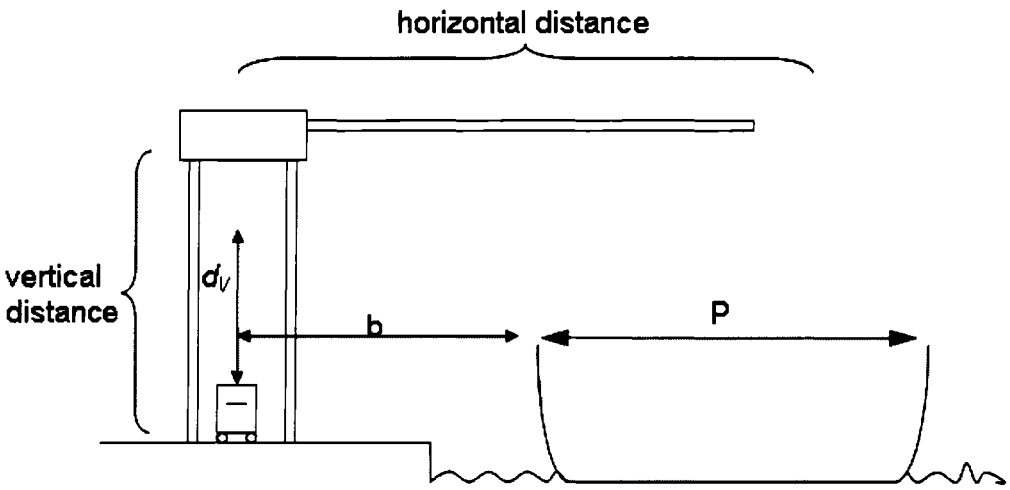


Figure 3.1: Horizontal and vertical motion of the crane.

- V_h - hoist speed of the trolley when not moving a container
- V_l - speed of the crane when moving lengthwise along the vessel
- V_t - horizontal travel speed of the trolley when not moving a container
- d_V - vertical distance from the apron to the maximum height a container can reach
- d_L - lateral distance between two rows of the vessel
- b - horizontal distance from landside vehicle to the edge of the vessel
- P - width of the vessel
- T_r - time required to position landside vehicle after departure of previous vehicle

Consider the time taken by the same two containers with single and double cycling. The crane's tasks in each are described below:

1. Single cycle (unload)

   (a) Lock onto container waterside,

   (b) lift container vertically, distance h_1,

   (c) move container horizontally, distance UD_1 to landside,

   (d) drop container, distance h_2, onto landside vehicle,

   (e) unlock,

   (f) lift trolley, distance h_3,

   (g) move trolley horizontally, distance UD_1, to waterside,

   (h) drop trolley, distance h_4, to next container for unload.

2. Single cycle (load)

   (a) Lock onto container landside,

   (b) lift container vertically, distance h_5,

   (c) move container horizontally, distance UD_2 to waterside,

   (d) drop container onto vessel, distance h_6,

   (e) unlock,

   (f) lift trolley, distance h_7,

   (g) move trolley horizontally distance UD_2 to landside,

   (h) drop trolley to next container for load distance h_8.

3. Double cycle (unload and load)

(a) Lock onto container waterside (equivalent to 1a),

(b) lift container vertically, distance h_1 (equivalent to 1b),

(c) move container horizontally, distance UD_1 to landside (equivalent to 1c),

(d) drop container distance h_2, onto landside vehicle (equivalent to 1d),

(e) unlock (equivalent to 1e),

(f) wait for container for container for loading to be positioned below crane,

(g) lock onto container landside (equivalent to 2a),

(h) lift container vertically distance h_5 (equivalent to 2b),

(i) move container horizontally, distance UD_2, to waterside (equivalent to 2c),

(j) drop container distance h_6 onto vessel (equivalent to 2d),

(k) unlock (equivalent to 2e),

(l) lift trolley distance h_7 (equivalent to 2f),

(m) move trolley horizontally distance $UD_2 - UD_1$ to location of next container to unload,

(n) drop trolley distance $h_4$to next container for unload (equivalent to 1h).

For each double-cycle we save some empty-crane travel relative to the two corresponding single cycles, but we also experience a slight landside-repositioning penalty. The time penalty, T_r, is incurred because after dropping a container for unloading onto a landside vehicle, the crane must wait for a container for loading to be positioned below the crane. With single cycling, this can be done simultaneously with other crane operations. Any minor vertical motion necessary between dropping a container on the landside vehicle and picking up the next container (to remove the trolley from the work area) will also be accounted for in this penalty.

The total distance travelled by the crane is reduced by one complete empty cycle between the apron and the position above either the container to load or the container to unload, whichever is closer. Of course, some horizontal and vertical motions will take place simultaneously. An upper bound on the time saved per cycle, is provided by summing the horizontal and vertical travel times, the lower bound by taking the maximum of the two. Therefore, on average we have:

$$2\left[\frac{d_V}{V_h}+\frac{b}{V_t}+\frac{\left(1/2\right)P}{V_t}\right]-T_r > W > 2\left[\max\left(\frac{d_V}{V_h},\frac{b}{V_t}\right)+\frac{\left(1/3\right)P}{V_t}\right]-T_r \qquad (3.1)$$

For each double cycle the vertical distance is reduced by $2d_V$. The average horizontal distance is reduced by twice the distance between the apron and the closer of the two containers. This is optimistically considered to be $2(b+\frac{1}{2}P)$ for the upper bound (as if the two containers were always next to each other), and pessimistically assumed to be $2(b+\frac{1}{3}P)$ for the lower bound (assuming a uniform distribution of locations). If double cycling only below deck, the maximum height reached while double cycling will be the clearance height of the vessel. The number of cycles saved by double-cycling in each row is $d_r = S_r - D_r$.

After completing loading and unloading operations on a row the crane moves along the vessel to the next row. If there are R rows on a vessel, the time consumed with lateral motion is $2(R-1)\frac{d_L}{V_l}$ with single cycling, and $(R-1)\frac{d_L}{V_l}$, exactly half, with double cycling. Recall that with the current method of single cycling, all relevant containers are unloaded before any containers are loaded. Thus the crane travels the length of the vessel twice. With double cycling, each row is unloaded *and* loaded before the crane moves on to the next row, therefore, the crane travels the length of the vessel only once.

Parameter	Value
V_h	300 feet per minute
V_t	500 feet per minute
P	130 feet
d_V	75 feet
b	60 feet
T_r	15 seconds [79]

Table 3.3: Parameter values for evaluation, based on Port of Tacoma trial.

We can now compare the results of this analysis to empirical data, collected during a double cycling trial at the Port of Tacoma. For more information on the trial please see Section 1.2.

3.1.2 Validation

During the Efficient Marine Terminal trial at the Port of Tacoma, the adjusted average time for a single cycle was 1 minute and 45 seconds, and for a double cycle, 2 minutes and 50 seconds. We now compare the difference in these empirical cycle times to the differences obtained using the expressions developed above. Parameter values based on the trial at Tacoma are given in Table 3.3 [22].

The lower bound is 25.4 seconds. The upper bound, 45 seconds. The empirical difference was 40 seconds. The upper bound is close to the empirical value. It is expected that the empirical double cycle times for will decrease as operators gain more experience with the new method of operating [22].

Clearly the specific results depend on the parameters of each crane, vessel, and container arrangement, but we have demonstrated that our tools provide results that match empirical data. A 21% reduction in the number of cycles reduces operating time by approximately 8%. A 35%

reduction in cycles would reduce operating time by 13%.

We have formulated the problem of minimizing the number of cycles required to turn around the vessel. From these results we have developed a method for converting benefits from number of cycles to an amount of time, assuming constant cycle times for single and double cycling. We have found the use of double cycling would provide significant reductions in operational time. Given constant cycle times, the problems of minimizing cycles and minimizing operational time are equivalent. If cycle times are not constant, we can bound from below, our results. From our analysis we believe the cycle time reduction will be at least 25 seconds, for every single cycle we are able to replace by a double cycle. We now develop a more robust methodology for estimating the reduction in cycles given a calling fleet.

3.2 Estimation of the Reduction in Cycles Given a Fleet

Before adopting double cycling, terminal operators will need to understand its impact on requirements for land-side vehicles, quay cranes, stevedores, etc. Instead of understanding the expected turn around for an individual vessel, they would like to base estimates on the entire fleet calling, and expecting to call, at their terminal for some relevant time horizon. In this analysis we assume knowledge only of the distribution of the number of containers to load and unload in each stack, and the number of stacks on the vessel. We now develop formulae for the expected number of cycles using double cycling given just this distributional information.

Definition 3.2.0.1 (Demand) Introduce random variables $u_{c,i}$ to denote the number of containers to unload in stack $c \in \{1..C_i\}$ of row $i \in \{1..R\}$, and $l_{c,i}$ the number of containers to load in stack c

of row i. [1] The random variables in sets $\{l_{c,i}\}$, and $\{u_{c,i}\}$ are assumed to be mutually independent. The variables in set $\{u_{c,i}\}$ are identically distributed with mean μ_u and variance $\sigma_u^2$and so are the random variables in $\{l_{c,i}\}$ with mean μ_l and variance σ_l^2.

Definition 3.2.0.2 (Cumulative Demand) Define $Y_i = \sum_{c=1}^{C_i} u_{c,i}$ the total number of containers to unload in row i and $\Lambda_i = \sum_{c=1}^{C_i} l_{c,i}$, the total number of containers to load in row i.

3.2.1 Expected Number of Cycles Using Single Cycling

We require one cycle for each container, so the expected number of cycles to unload and load in row i using single cycling is equal to the expected number of containers:

$$E[\Lambda_i] + E[Y_i] \tag{3.2}$$

These expectations are: $E[Y_i] = E[\sum_{c=1}^{C_i} u_{c,i}] = \sum_{c=1}^{C_i} E[u_{c,i}] = C_i\mu_u$, $E[\Lambda_i] = E[\sum_{c=1}^{C_i} l_{c,i}] = \sum_{c=1}^{C_i} E[l_{c,i}] = C_i\mu_l$. It should be noted that while mathematically it is satisfactory to consider the number of cycles necessary to unload and load a row, in current operations, all containers from *the vessel* are unloaded before any containers are loaded onto the vessel.

3.2.2 Expected Number of Cycles Using Double Cycling

In this section three formulas are presented to estimate the expected number of cycles necessary to turn around a ship based on the parameters of a fleet. Such formulas would be particularly useful for planning purposes. Initially we introduce two formulas to estimate the expected number of cycles to complete a (random) row using any of our strategies, which are referred to as the simple

[1] If a container on the vessel needs to be moved to access another container, but it is not to be unloaded at this port, this move will be considered an unload, and a load if it is handled again, for simplicity.

estimate and the alternate estimate. Then we use a diffusion approximation to develop an improved estimate for the proximal stack strategy. The goal is to estimate the number of cycles necessary to turn around a row with double cycling, using knowledge of the distribution of containers rather than specific information about the number of imports and exports on each ship.

General Formulas

Given the asymptotic behavior of the problem, where $max\{\Lambda_i, Y_i\}$ is a tight lower bound, we can use the following estimate for the number of cycles:

Definition 3.2.2.1 (Simple Estimate) $max\{E[Y_i], E[\Lambda_i]\}$

Where $E[Y_i]$, is the expected number of imports a row i, and $E[\Lambda_i]$, the expected number of exports in a row i. For large rows this is a very good estimate of the number of cycles. We also developed an alternate estimate, that is more accurate.

Let Y_i and Λ_i be independent and normally distributed with means $E[Y_i]$ and $E[\Lambda_i]$ and variances $\sigma^2_{Y_i}$ and $\sigma^2_{\Lambda_i}$.

Definition 3.2.2.2 (Alternate Estimate) $E[max\{Y, \Lambda\}] = E[Y]\Phi(\alpha) + E[\Lambda]\Phi(-\alpha) + a\varphi(\alpha)$ where $\varphi(x) = \sqrt{2\pi}e^{\frac{-x^2}{2}}$, $\Phi(x) = \int_{-\infty}^{x} \varphi(t)dt$, $\alpha = E[Y] - \frac{E[\Lambda]}{a}$, and $a^2 = \sigma^2_Y + \sigma^2_\Lambda$.

For ships with many more imports than exports, or vice-versa ("lopsided"), the alternate and simple estimates are essentially equivalent (and are equivalent for the cases summarized in Table 3.4. For evenly balanced ships, the alternate estimate is always slightly higher than the simple estimate. Both will underestimate the true average. In the cases tested, the alternate estimate is somewhat closer to the actual value using the greedy strategy, but both provide good estimates

Average Absolute Error in Turn Around Time	Alternate Estimate	Simple Estimate
Lopsided Ships	1.90 (-0.70%)	1.90 (-0.70%)
Evenly Balanced Ships	45.50 (1.85%)	48.66 (2.04%)

Table 3.4: Average absolute error in turn around time between estimate and realized value using the greedy strategy.

for the number of cycles. The absolute error is given in Table 3.4. As ship size increases, both estimates are closer to the actual value. For small vessels, more significant differences exist between the estimates and the actual value. These estimates provide port planners with easy-to-use tools to understand the impact of double cycling on vessel turn around time.

Improved Formulas for the Proximal Stack Strategy

Figure 3.2(a) shows a queuing diagram for the loading and unloading processes for an example problem; one row with four stacks. In this example, $u_{1,1} = u_{2,1} = 3$, $u_{3,1} = u_{4,1} = 2$, $l_{1,1} = 2$, $l_{2,1} = 5$, $l_{3,1} = 0$, and $l_{4,1} = 3$. The diagram shows two curves; one that documents the loading process, and one that documents the unloading process. The figure shows the number of stacks completed for any number of cycles completed. When both loading and unloading, we operate on the stacks in the order $c = 1, 2, 3, 4$. The loading operations begin on stack 1 as soon as the unloading operations are complete in stack 1. Note the loading operations on stack 2 are delayed by one cycle, as unloading operations on stack 2 are not yet complete. Define:

$$M_i(c) = \sum_{2 \le j \le c} \{u_{j,i} - l_{j-1,i}\} \; \forall c \in \{2..C_i\} \tag{3.3}$$

Also define $M_i(1) = 0$. Then,

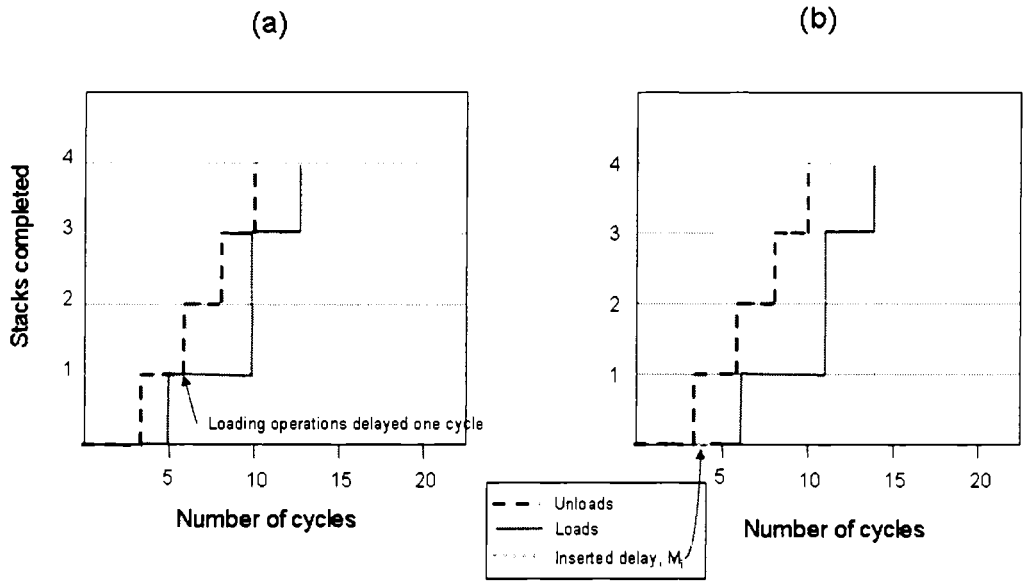


Figure 3.2: (a) Example queuing diagram for loading and unloading operations. Notice delay of one cycle to loading operations after completing stack 1. (b) Delay, M_i inserted before any loading operations start to avoid later delay.

$$M_i = \max_{c=\{1...C_i\}} \{M_i(c)\} \tag{3.4}$$

If we delay the loading of stack 1 by M_i time units (see figure 3.2(b)), we eliminate all future delay caused by waiting and loading operations are completed at time $\{u_{1,i} + M_i\} + \Lambda_i$. The number of cycles required to unload and load the row using the proximal stack strategy is thus:

$$u_{1,i} + \Lambda_i + M_i \tag{3.5}$$

Then on average:

$$E[u_{1,i}] + E[\Lambda_i] + E[M_i] \tag{3.6}$$

The first two terms are easy to estimate; the expected number of containers to unload in

Parameter	Diffusion Units	Container Terminal Units
t	time	number of stacks
x	quantity	number of containers
M	net quantity	number of containers
D	$quantity^2$/time	$number\ of\ containers^2$/number of stacks
d, drift	quantity/time	number of customers/number of stacks

Table 3.5: Correspondence between diffusion parameters and container terminal parameters.

the first stack, and the expected number of containers to load in row i. In fact, a very rough estimate for the number of cycles is $E[\Lambda_i]$, but we will seek a more accurate value. Since the $u_{j,i} - l_{j-1,i}$ are independent, identically distributed random variables, $M_i(c)$ is a diffusion process with the stacks completed, c, as time. Table 3.5 shows the correspondence between diffusion parameters and container terminal parameters. Thus, we can use diffusion formulae for $E[M_i]$. The drift is:

$$d = \frac{E[M_i(c)]}{c} = \mu_u - \mu_l \tag{3.7}$$

and the variance rate is:

$$D = \frac{var(M_i(c))}{c} = \sigma_u^2 + \sigma_l^2 \tag{3.8}$$

Definition 3.2.2.3 (First Passage Time) Let $T_i(z)$ be the time (number of stacks completed) at which $M_i(c)$ first reaches z cycles, assuming the process starts from $z = 0$.

According to [20] the formula for the probability density function of $T_i(z)$ evaluated at c is:

$$f(c|z) = \frac{z}{\sqrt{2\pi Dc^3}} e^{\frac{-(z+dc)^2}{2Dc}} \tag{3.9}$$

The cumulative distribution function is therefore:

$$\Pr\{T(z) \leq c\} = F(c|z) = \int_0^c \frac{z dy}{\sqrt{2\pi D y^3}} e^{\frac{-(z+dy)^2}{(2Dy)}}. \tag{3.10}$$

We are interested in $F(C_i|z)$ where C_i is the number of stacks in row i. Note, however, that

$$Pr\{T(z) \leq C_i\} \equiv Pr\{M_i > z\}. \tag{3.11}$$

The expectation of a non-negative random variable is obtained by integrating the complementary cumulative distribution function. Hence, the expected maximal excursion is;

$$E[M_i] = \int_0^\infty dz \int_0^{C_i} dy \left\{ \frac{z}{\sqrt{2\pi D y^3}} e^{\frac{-(z+dy)^2}{(2Dy)}} \right\} = \frac{2D}{d}[\Phi(\frac{d\sqrt{C_i}}{\sqrt{D}}) - \frac{1}{2} + \int_{\frac{-d\sqrt{C_i}}{\sqrt{D}}}^0 y\Phi(y)dy] \tag{3.12}$$

where $\Phi(x) = \int_{-\infty}^x \frac{dw}{\sqrt{2\pi}} e^{\frac{-w^2}{2}}$. An estimate for the number of cycles to unload and load a row is now given.

$$\mu_u + C_i\mu_l + \frac{2D}{d}[\Phi(\frac{d\sqrt{C_i}}{\sqrt{D}}) - \frac{1}{2} + \int_{\frac{-d\sqrt{C_i}}{\sqrt{D}}}^0 y\Phi(y)dy]. \tag{3.13}$$

$E[u_{c,i}] = E[u_{1,i}] = \mu_u$ because $\{u_{c,i}\}$ all identically distributed, and $\Phi(x) = \int_{-\infty}^x \frac{dw}{\sqrt{2\pi}} e^{\frac{-w^2}{2}}$. Notice that for $d = 0$, the value of M_i is undefined. We expect that in these cases, the value of M_i would be small. In fact, if we define $x = \frac{d\sqrt{C_i}}{\sqrt{D}}$ then the reader can verify from equation (11) that $\lim_{x\to 0} E[M_i] = 0$. The formula allows us to estimate the number of cycles to insert before beginning loading operations so we can avoid future delay and thus estimate the total number of cycles to complete loading and unloading operations.

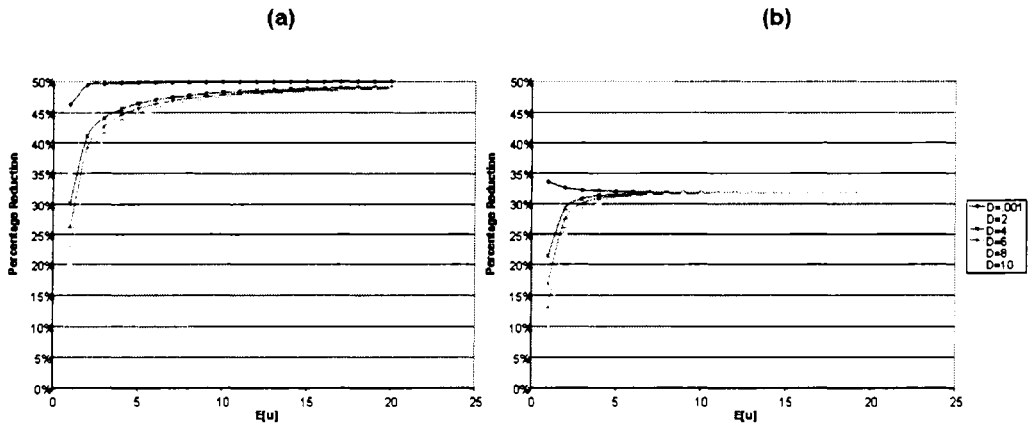


Figure 3.3: (a) Percentage reduction in number of cycles to complete loading and unloading operations below deck using the proximal stack strategy for varying values of D. $C_i = 20$, and $Y = \Lambda$. (b) Percentage reduction, $Y = 0.5\Lambda$.

Formula 3.13 tells us that only μ_u, μ_l, C_i and $D = \sigma_u^2 + \sigma_l^2$ influence crane time; all other ship configuration data are irrelevant. One could not get this without completing the analytical analysis. Figure 3.3(a) shows the percentage reduction in the number of cycles required to complete loading and unloading operations below deck on a row for different values of D. In this example $C = 20$, and $\mu_u = \mu_l$. Figure 3.3(b) shows the same information, but for a vessel where $\mu_l = 0.5\mu_u$. As we expect, benefits decrease with increasing variance (values of D). Given that C_i is fixed, increasing values of μ_u imply a larger total number of containers, and so as a percentage, benefits are greater with increasing μ_u.

3.2.3 Comparison of Formula Result and Average of Many Vessels

From the central limit theorem, we know Formula 3.13 will provide a good estimate for the number of cycles for a large number of stacks, but it is unclear how well the expression will match reality for realistic numbers (up to 20 per row in current ships). We therefore compare

the estimated values using the formula to the average for a set of generated vessels. A computer program was used to generate ship data according to a distribution of imports and exports, and count the number of cycles required to turn around each row. For details of the computer program see Section 2.6. The inputs to the program are parameters p and q of the beta distribution for both the number of imports and number of exports in one stack, the number of stacks, the maximum height of an import stack and the maximum height of an export stack. Inputs to the formula, d, D, C_i, μ_u, and μ_l can be determined uniquely from the computer program inputs.

3.2.4 Correspondence between computer program parameters and equation parameters

The required inputs to the computer program are described below:

- p_i – beta distribution parameter p for imports in one stack
- q_i – beta distribution parameter q for imports in one stack
- p_e – beta distribution parameter p for exports in one stack
- q_e– beta distribution parameter q for exports in one stack
- x – number of stacks on the vessel
- i – maximum stack height for imports
- e – maximum stack height for exports

The required formula inputs are:

- c – drift of diffusion process

Parameter	Value
x	varies (as indicated in Figure 3.4)
p_i	1
q_i	1
p_e	2
q_e	3
i	10
e	10

Table 3.6: Parameter values used to generate vessel data in Figure 3.4.

Parameter	Value
t	varies (as indicated in Figure 3.4)
c	1
D	1.23

Table 3.7: Parameter values used to generate equation estimates in Figure 3.4.

- D – coefficient of diffusion
- t – number of columns on the vessel

The program and formula inputs are related as follows:

$$t = x$$

$$c = \frac{ip_i}{p_i+q_i} - \frac{ep_e}{p_e+q_e}$$

$$D = \frac{ep_eq_e}{(p_e+q_e)^2(p_e+q_e+1)} + \frac{ip_iq_i}{(p_i+q_i)^2(p_i+q_i+1)}$$

Tables 3.6 and 3.7 show the parameter values used to generate the results in Figure 3.4. The number of stacks varies, as shown in the figure.

Figure 3.4 shows the number of moves necessary to turn around the vessel versus the number of stacks on the vessel. Each square represents the estimated number of cycles using Formula 3.13, with input parameters as described in Table 3.7. The result is equivalent to the predicted

average assuming we had an infinite number of simulations. Each diamond shows the average result for 30 generated vessels. The error bars show *just one* standard deviations from the mean, using an estimate of the population standard deviation. We see the estimate and average are very close even for small numbers of stacks. The estimated value from the equation is always within the error bars, and so we have demonstrated that the estimate and the average are not significantly different. The average difference between the estimate and the average was 1.13%. As we might expect, the estimate is worst for the case of 5 stacks, where there is difference of 5.31%. We have thus developed a formula to estimate long-run crane and vessel productivity improvements as a result of double cycling.

In the next section we consider the impact of double cycling on landside terminal operations.

3.3 Impact on Landside Terminal Operations

In addition to improving crane and vessel productivity, the use of double cycling provides an opportunity to increase the productivity of landside equipment. In a container terminal, container handling equipment is used to transport the containers between the apron and the local storage facility. Loading and unloading the vessel simultaneously means that after a vehicle delivers a container to the apron, it can then carry a container from the apron to local storage, instead of returning to the local storage without transporting a container. Container handling equipment is also used to raise or lower containers e.g. from a chassis onto the ground, or from the ground onto a stack of containers. There are many different methods by which containers can be transferred from the apron to the local storage area, and then placed in local storage. We are interested in two

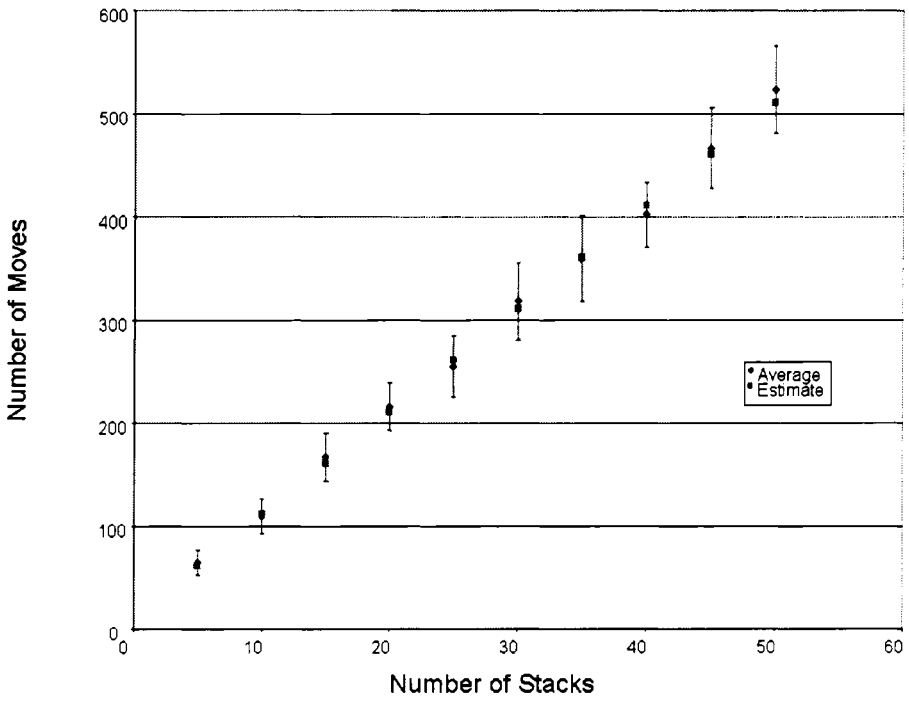


Figure 3.4: Squares show the number of moves as calculated by Formula 3.13. Diamonds show the average of 30 simulated vessels with the same characteristics. Error bars show one standard deviation from the mean.

varieties:

1. Transfer methods that require another piece of equipment to remove the container from the vehicle (or place a container on the vehicle) at the local storage area before the vehicle can move on to serve another container. In one such method, containers are transferred by yard tractors on chassis, but stored on the ground. In this case another machine, such as a top-pick, must be used to unload the container from the chassis before the chassis can be used to service another container.

2. Transfer methods that do *not* require another piece of equipment to remove the container from the vehicle (or place a container on the vehicle) at the local storage before the vehicle can move on to serve another container. For example, containers may be transferred by yard tractors on chassis and stored on the chassis, or containers can be transferred by straddle carriers, which can then place the containers on the ground and move on to serve another container.

The analysis here is appropriate for the second case; methods that do not require another piece of equipment to remove the container from the vehicle. These operations are common.

Figure 3.5(a) shows a schematic of the landside transportation system when single cycling. In this figure we have shown only one crane operating on the vessel, but the analysis is also applicable to the case where we have many cranes operating per vessel if we assign cranes to non-overlapping segments of the vessel. The diagram shows, with arrows, the flow of landside vehicles with containers in solid lines, and vehicles without containers with dashed lines. We will refer to these trips without containers as empty trips, and would like to reduce their number. Figure 3.5(b) shows a schematic for double cycling, where the two empty trips of single cycling have been re-

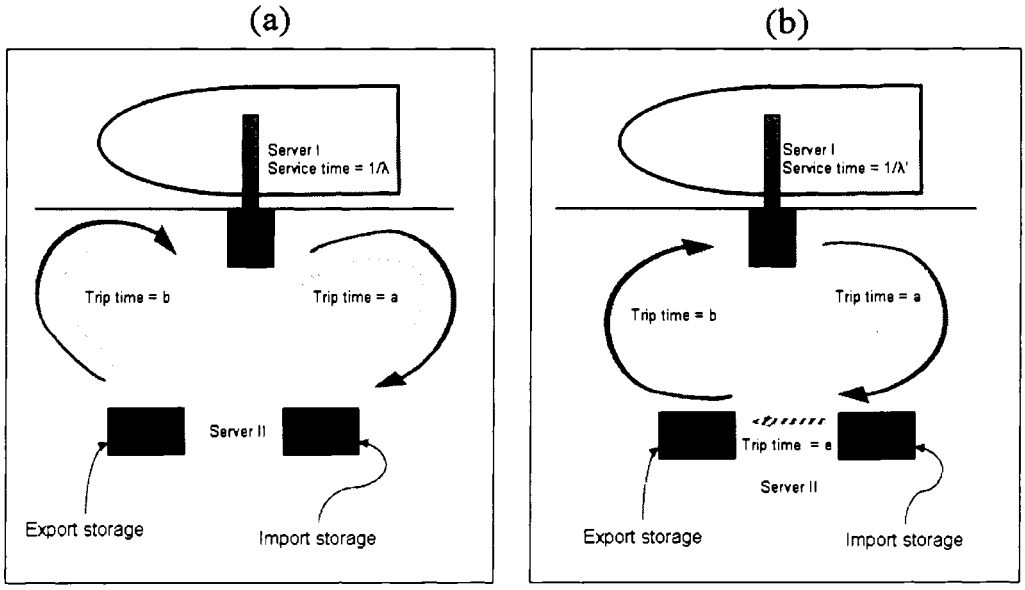


Figure 3.5: (a) Schematic diagram of landside operations when single cycling. (b) double cycling.

placed by one empty trip between the storage locations of import and export containers. These schematics are not meant to fully describe the flow of traffic, but to capture the main elements common to most systems.

In both the single and double cycling cases, we model the flow of landside vehicles as a closed queueing system, with two server stations and many customers. The server stations are 1) the quay crane and 2) landside vehicle drivers at the local storage areas (again, we consider only systems which do not require another piece of equipment to remove the container from the vehicle). The customers are landside vehicles, such as yard tractors or straddle carriers. The system is closed because the landside vehicles shuttle between server stations, visiting the quay crane and then the local storage facilities, until the vessel unloading and loading operations are complete. We assume the queueing system has reached steady state after some time.

Our goal is to develop tools to aide port planners. Here we develop a simple expression for the number of landside vehicles required with double cycling. We assume vehicle travel times are independent, so do not capture any congestion effects between vehicles.

3.3.1 Modeling the Quay Crane

The quay crane is modelled as a single server with deterministic service times. This server station is labelled I in figure 3.5. The crane serves its customers, landside vehicles. In the case of single cycling the following tasks make up service by the quay crane:

- While loading: pick up a container from a landside vehicle or the ground beneath the crane, carry it to the vessel, drop the container, and return to the apron.
- While unloading: drop a container onto a landside vehicle or the ground, return to the vessel, pick up a container, return to the apron.

In the case of double cycling, service by the quay crane requires the crane to pick up a container from a landside vehicle or the ground, carry it to the desired location on the vessel, drop the container, move to the location of the container to unload, pick-up the container, carry it to the apron, and drop the container onto a landside vehicle or the ground. Notice that after a container for loading has been unloaded from a vehicle, the vehicle must wait for the crane to return with a container for unloading.

Service times are $1/\lambda$ and $1/\lambda'$ minutes per customer for the single and double cycling cases, respectively. In practical operations quay crane cycle times are very consistent, so we assume no variance in these service times. Our assumption that cycle times are different for double cycling than for single cycling, is validated in practice [13].

3.3.2 Travel Times Between Servers

After being served by the quay crane, the landside vehicle will travel to either the import or export storage facility. We assume the travel time between the apron and the local import storage is a minutes, with $var(a) = 0$. We assume the travel time between the local export storage and the apron is b minutes, with $var(b) = 0$. We assume the travel time between the local import and local export facilities is e minutes, with $var(e) = 0$.

3.3.3 Modeling the Local Storage

Service at the local storage is the placement of containers in the appropriate storage location or retrieval from the appropriate storage location. Service is provided by the landside vehicle driver. With single cycling, service is defined by the driver finding the storage location, and either attaching or detaching the chassis. If straddle carriers are used the driver must find the correct drop location and drop the container. With double cycling, the driver must find the location of the import, detach the chassis, find the location of the export, and attach the chassis. If straddle carriers are being used the driver must find the location of the import drop, drop the container, find the location of the export, and pick-up the container. The container may then be moved again by a reach stacker or gantry crane, but this is not considered part of the service time as the straddle carrier driver can move on to another task while this operation takes place. For both the single and double cycling cases, we assume there are infinitely many servers at the local storage facility. This is reasonable in the case where landside vehicle drivers provide service, as, in operations, the driver would never have to wait for service. Once the driver arrives at the local storage facility he/she can always be looking for the storage location or dropping/picking-up a container, but the amount of time it takes to complete

this operation varies; the driver may have difficulty locating the position or dropping/picking-up the container. We will label the local storage location II.

Although in reality with double cycling the landside vehicles pick-up and drop-off containers at different instants in time, and at different locations, we consider these two events part of the same mathematical server. With infinite servers this is mathematically equivalent to considering them as two servers. Call the service time at II with single cycling w, and the service time at II with double cycling, w'. With single cycling, we consider picking up or dropping off a container to have the same service time. We assume $E[w] = E[w^u] = E[w^l]$ where $E[w^l]$ is the service time during loading (picking up) and $E[w^u]$ is the service time for unloading (dropping-off). With double cycling, the driver must drop-off and pick-up a container, so $E[w'] = E[w^u] + E[w^l]$. It is therefore reasonable to estimate that $E[w'] = 2E[w]$. To calculate the number of landside vehicles required, we will assume distributional information is available about w and w'.

3.3.4 Estimating the Number of Landside Vehicles Required

Given the framework described above, we can estimate the number of customers required in the system such that a server is almost never waiting for a customer. In practical terms we would like to supply just enough landside vehicles for each crane in service, such that the crane is never idle. If we have excess vehicles, we assume they are in queue at I. Given the nature of our system, the rate at which customers, or landside vehicles arrive at I is determined by the quay crane's service time (assuming we have enough vehicles). The expected arrival rate for single cycling is λ, and for double cycling, λ'. An estimate for the number of landside vehicles required per crane *when single cycling* is:

$$\lambda(2a+2b)+1+\lambda E[w]+2\sqrt{\lambda E[w]\left\{1-\frac{E[min(w_1,w_2)]}{E[w]}\right\}} \tag{3.14}$$

The first three terms are derived using Little's formula. The first two terms, $\lambda(2a+2b)$, reflect the number of vehicles in transit between the apron and the local storage at any instant in time. The third term, 1, is the number of landside vehicles in service at I. The remaining terms are the expected number of landside vehicles in service at II, plus two standard deviations. We use the formula for an ∞-server, ∞-customer system proposed in [65] and proved for more general conditions in Diez-Roux and Daganzo [17]. The expression for the standard deviation is taken from Diez-Roux and Daganzo [17].

Similarly, an estimate for the number of landside vehicles required per crane when double cycling is:

$$\lambda'(a+b+e)+1+\lambda' E[w']+2\sqrt{\lambda' E[w']\left\{1-\frac{E[min(w_1',w_2')]}{E[w']}\right\}} \tag{3.15}$$

Here the first two terms, $\lambda'(a+b)$, are the number of vehicles in transit between the apron and local storage, and the third term, $\lambda' e$, is the number of landside vehicles in transit between the local storage facilities.

Evaluation

It would not be difficult for the port to determine reasonable values of the required parameters, in order to evaluate these expressions. We require only ballpark estimates of the parameters as one of the strengths of our method is that sensitivity analysis could be easily carried out.

While terminals vary widely in their size, crane service rates do not. For single cycling an average cycle time of 1 minute and 45 seconds was recorded at the Efficient Marine Terminal Trial at the Port of Tacoma in 2003. The average double cycle took 2 minutes and 50 seconds. For this data, $\lambda = .57$ landside vehicles per minute and $\lambda' = .35$. See reference [13] for further information. Values of $a = 1$ minute, $b = 2$ minutes, and $e = 1$ minute were estimated based on the current storage plan for, and the dimensions of, the Ben E. Nutter terminal at the Port of Oakland and a yard tractor speed of 15 miles per hour.

If w follows an exponential distribution $E[min(w_1, w_2)]/E[w] = 1/2$,

and $E[min(w'_1, w'_2)]/E[w'] \geq 1/2$ so choosing $1/2$ for this value is conservative. Figure 3.6 shows the number of drivers required for different values of $E[w']$, where $E[w'] = 2E[w]$. For example with $E[w'] = 5$ minutes we estimate 7.5 yard tractors required per crane with single cycling but only 6 required with double cycling. The main contributor to the reduction is the increased cycle time with double cycling. Not only do we reduce the number required, but we improve their productivity per mile travelled or hour driven. The Ben E. Nutter terminal at the Port of Oakland typically deploys 7 yard tractors per crane. At this level, double cycling could offer a reduction of about 1.5 drivers and tractors required, a reduction of about 22%.

In the next section we consider operational changes to support double cycling.

3.4 Operational Changes to Support Double Cycling

While many of the following strategies will not measurably impact the benefits of double cycling, their implementation will simplify double cycling planning and implementation.

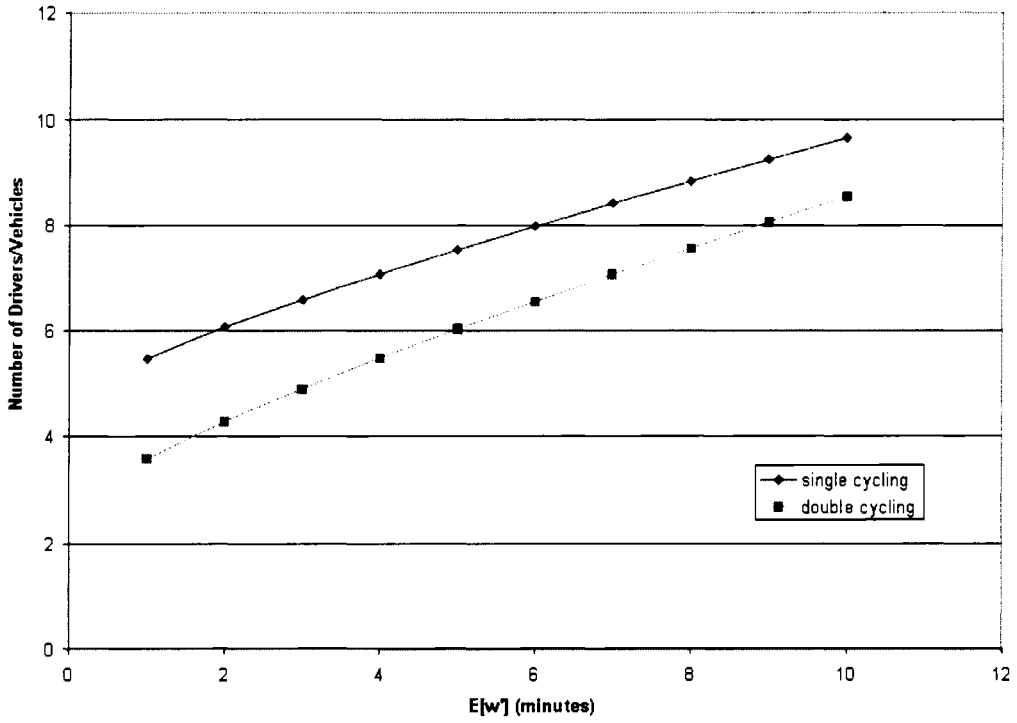


Figure 3.6: Number of landside vehicle drivers required for single and double cycling, against expected wait time at II.

3.4.1 Loading Plans

In this section we discuss changes to existing load planning that may create more significant opportunities to benefit from double cycling. Our goal is not to provide a software tool for determining a loading plan for a specific set of data (as these tools already exist), but to provide general principles upon which loading plans that support double cycling should be based. When developing a ship loading plan there are many considerations; weight balancing, storage of hazardous materials, storage of refrigerated containers, and storage of both 20 and 40 foot units. These considerations are relevant for both single and double cycling, so will be ignored. We assume the total number of imports and exports at each port is determined by market demand. The following four principles should be observed.

Principle 1: Smooth the difference between loads and unloads across the stacks

To reduce delay caused by the loading operations waiting for the unloading operations, we would like to spread the difference between the total number of loads and unloads evenly across all stacks. If possible, we should try to make $\{l_{j,i} - u_{j,i}\} = \frac{\Lambda_i - Y_i}{c}\ \forall j$, but in the case the difference is not a multiple of the number of stacks, or for some other reason infeasible, we should at least ensure that $\{l_{j,i} - u_{j,i}\} \leq \{\Lambda_i - Y_i\}\ \forall j$.

Principle 2: Put stacks for one destination in proximal stacks, in as few rows as possible

With the containers in as few rows as possible, the stacks will be necessarily closer together. This will reduce the time required to unload and load the vessel by keeping the distance between stacks upon which we operate within one cycle small. It will also reduce the number of

times we pay the initial unloading stack penalty, thereby reducing total crane moves.

Principle 3: Segment space on the vessel by origin-destination pairs

Assume vessel stops can be enumerated, as shown in Figure 3.7. This can represent any tour with n stops including shuttle services ($n = 2$), as well as more typical routes with $n = 3$ or more. The example in Figure 3.7 has $n = 4$ stops. The vessel has just called at port 4 and is sailing for port 1. The vessel is divided into 6 segments, one for each of the 6 OD pairs represented on the ship. Each segment is labelled with an OD pair, for example $4 - 1$ indicates containers loaded at port 4 and destined for port 1. Notice there is no segment labelled $3 - 4$, $2 - 4$, or $1 - 4$ as these containers would have just been unloaded at port 4. Similarly, notice there are no segments labelled $2 - 3$, $1 - 3$, or $1 - 2$ as these containers would only be on board on the segments between ports 1 and 3. At any one time, the number of OD pairs with containers on board the ship, N, is at most one-half of all OD pairs $\left\{N = \frac{n(n-1)}{2}\right\}$. Note that at any given time if the vessel is carrying containers from i to j it is not carrying containers from j to i.

Let m be the number of stacks on the vessel below deck. If the demand for service is equivalent for all OD pairs, then we require m/N stacks for each OD pair on the vessel. At any sailing time we have $(n-1)m/N$ cells for the next destination, $(n-2)m/N$ for the second destination, etc. Although desirable, it is not possible to ensure that all segments for a single destination are in co-located segments. With even demand, we divide the ship up into evenly sized segments for each OD pair on the vessel. With N segments each has $1/N$ of the total number of stacks. With uneven demand, the percentage of stacks allocated to an OD pair on the vessel should equal the percentage of total demand that appears for that OD pair.

The example in Figure 3.7 shows an example vessel loading after leaving port 4. Note

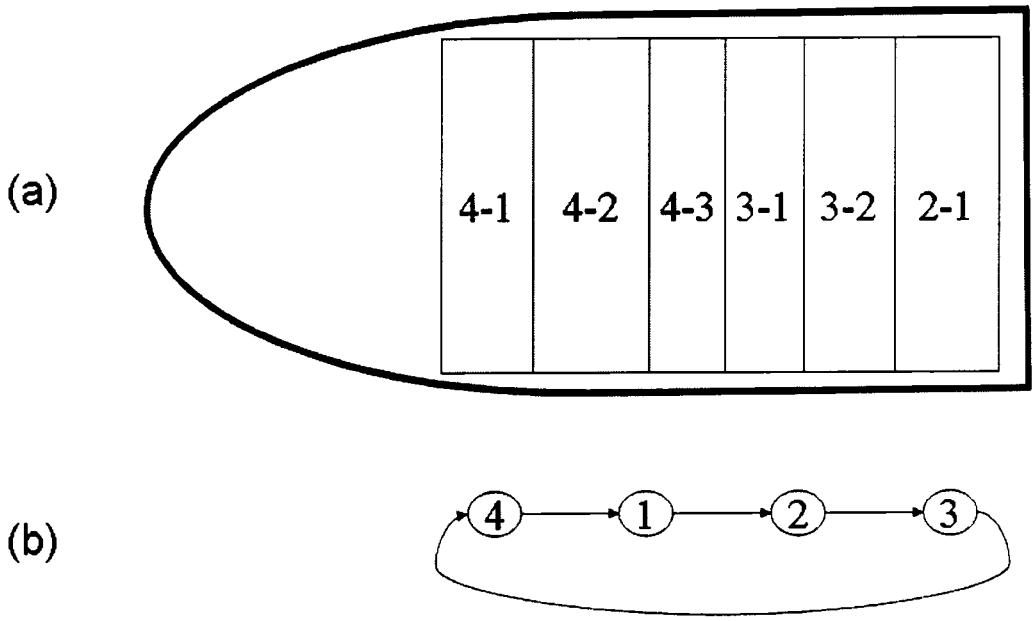


Figure 3.7: (a) Vessel storage segregated by origin-destination pairs (labelled O-D). (b) This example vessel calls at four ports labelled by number. The vessel has just left port 4, sailing for port 1.

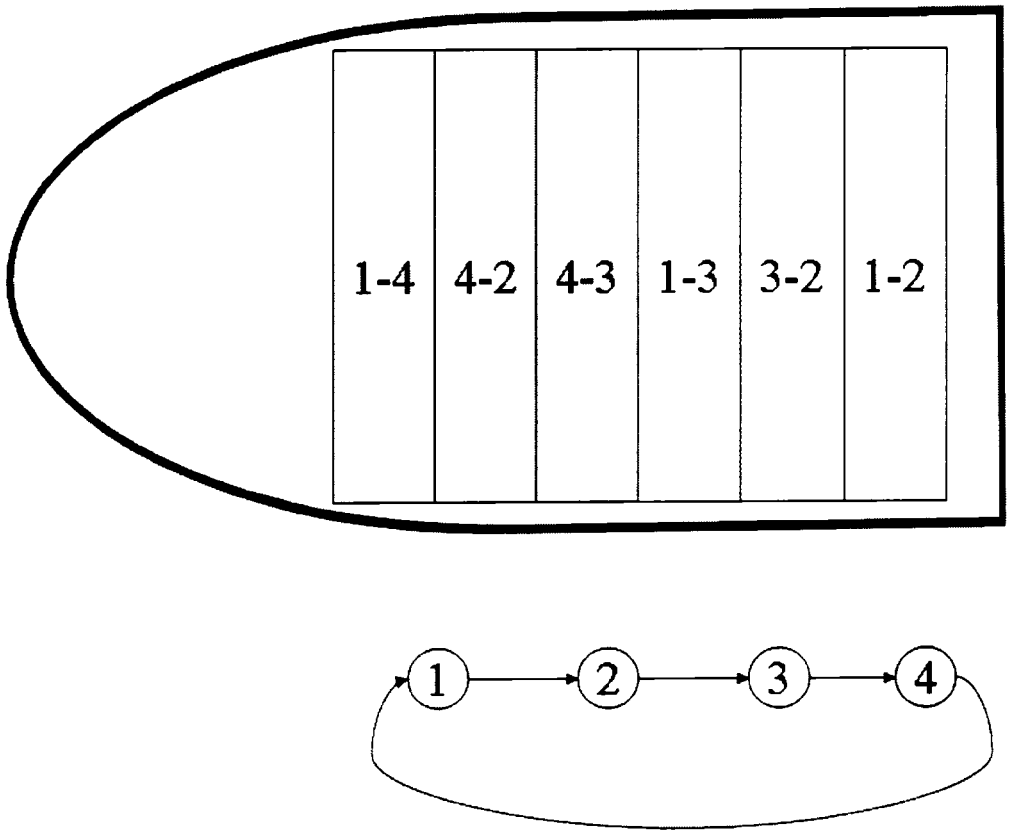


Figure 3.8: (a) Vessel storage segregated by origin-destination pairs (labelled O-D). (b) This example vessel calls at four ports labelled by number. The vessel has just left port 1, sailing for port 2.

that after visiting port 1 the vessel could be loaded as shown in Figure 3.8. Notice that visiting a port flips the pairs on which the port number appears. A part of the ship is always used by containers for $i-j$ or $j-i$ and can be reserved for this port pair. If the trade is balanced, including empties, there would never be any empty space on the vessel.

Generally, demand to move freight is not well balanced across a vessel's route; Asian ports operate as export terminals, while US ports are essentially import terminals. The vast majority of trans-Pacific containers are loaded onto the vessel at an Asian port and discharged at a US terminal.

That said, empty containers must be returned to Asia, so over time the flow of containers out of a terminal almost balances the flow in. In this case you can divide the ship into uneven parts reserved for specific trades (e.g. $i - j$).

Principle 4: For any OD pair with stacks in a row, leave one stack empty

By doing this we can avoid waiting to begin the loading operations, since there will always be space available to begin loading as soon as we begin unloading. Unfortunately this reduces the utilization of the vessel so is not practical.

3.4.2 Load Sequencing

In modern port operations, information is given to landside vehicle drivers regarding the destination of their load, or which load to pick-up, from a terminal operating system and delivered through a mobile data device. Terminal operation software programs decide which container should be served next, and where the container should be stored, based on the information it has received from other drivers, and port planners. These programs could easily be changed to accommodate the new sequencing rules of double cycling, thus continuing to provide drivers with clear instructions.

A further benefit of double cycling is that internal port traffic flows can be significantly simplified. Figures 3.9(a) and 3.10(a) show typical traffic flows for straddle carriers and yard tractors operating single cycling. Figures 3.9(b) and 3.10(b) show typical traffic flows for straddle carriers and yard tractors operating double cycling. These schematics are not meant to fully describe the traffic flows, but to provide a general description of the system. Before implementing double cycling, a port will need to reconsider traffic flow patterns. Each port will make modifications based on the specifics of the port. Although it may initially seem traffic flows would be complicated by

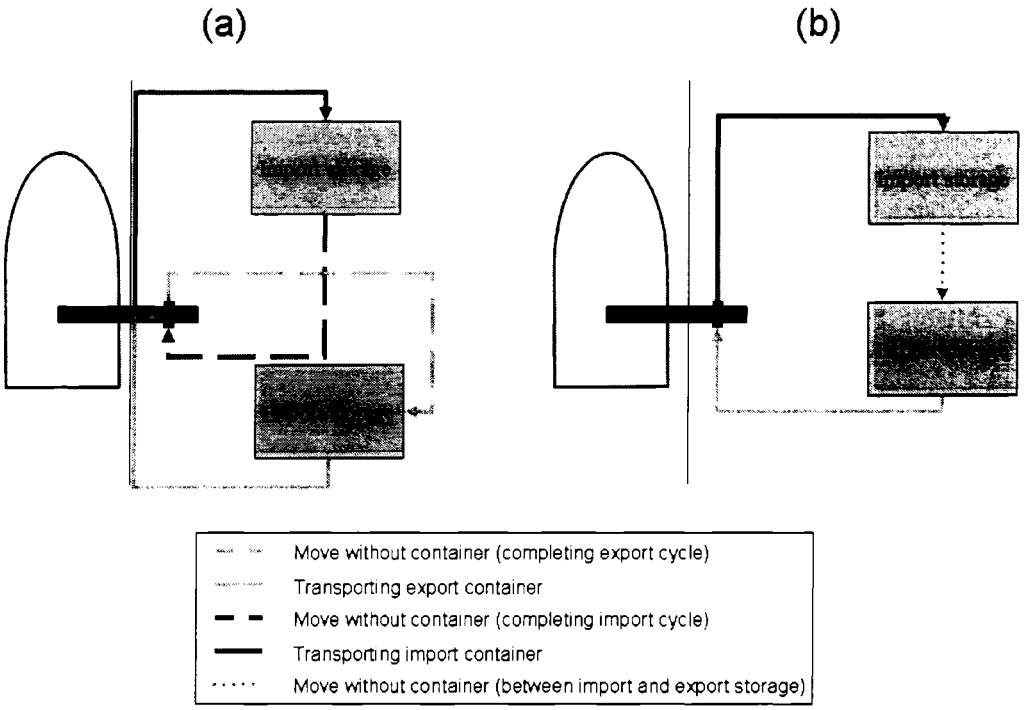


Figure 3.9: (a) Traffic flows for straddle carriers when single cycling. (b) double cycling.

double cycling, these figures demonstrate double cycling's simplifying nature.

3.4.3 Impact on Storage Equipment

We will consider the need for chassis in wheeled operations (where containers are stored on chassis while in the terminal, rather than of the ground or on top of another container). While wheeled operations are becoming less common due to the need for greater land utilization, many terminals still have some, or a significant portion of their containers stored temporarily on chassis. With the current method of single cycling, just after the unloading operations have been completed on a vessel, all containers to load and unload on that vessel are sitting in the terminal, because all

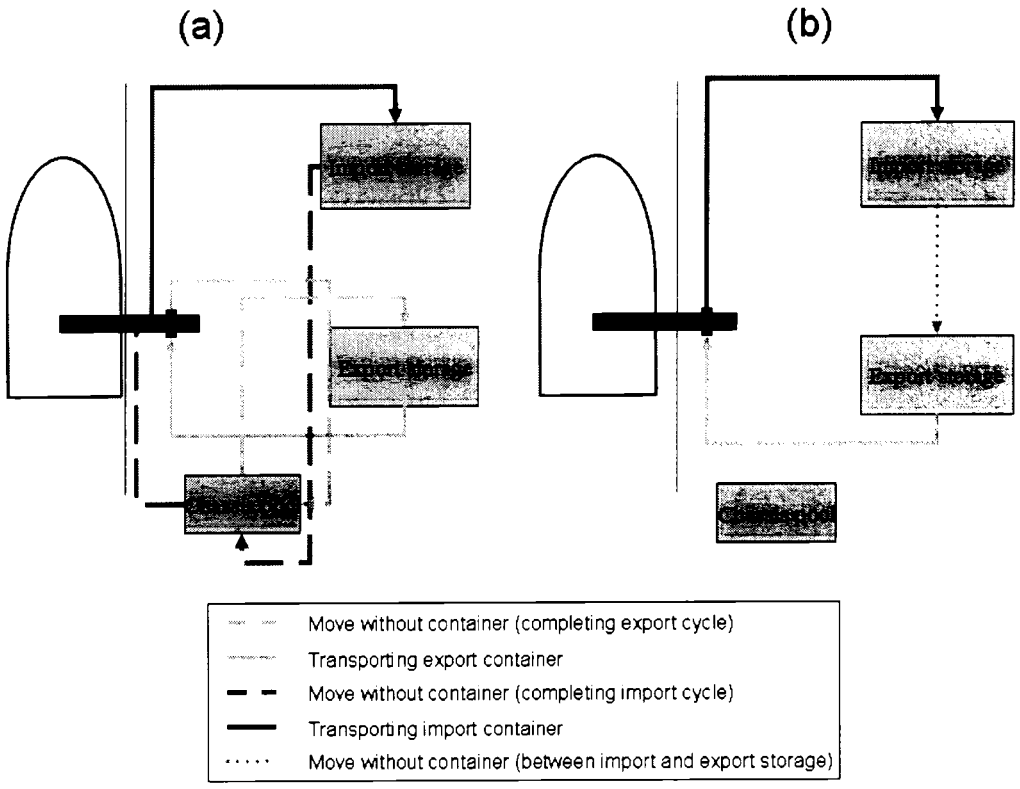


Figure 3.10: (a) Traffic flow pattern for yard tractors single cycling. If imports are stored on-wheels it is necessary for the driver to stop at the empty chassis pool. (b) Landside traffic flow patterns for yard tractors when double cycling. With double cycling, trips to the empty chassis pool are not typically required.

containers are unloaded from a vessel before any are loaded. It is possible to operate in another way, where each stack is first unloaded and then loaded with single cycling, but that is not the method that is currently used, so it is not considered here. With double cycling, we never have all loads and unloads sitting in the terminal because almost as soon as we start unloading the vessel, we begin loading the vessel. Because chassis can be used for both import containers and export containers, double cycling provides an opportunity to reduce the number of chassis required, over the status quo.

We would like to accommodate the entire fleet of vessels calling at a terminal, so should plan to provide enough chassis for the largest vessel. When a container is in the terminal, it requires a chassis, therefore, we can think of the number of chassis required as equal to the total number of containers to load and unload on the vessel, less the minimum number of containers on the ship at any time during the loading and unloading operations. The number of containers in port per vessel at any point in time during the loading and unloading operations can be seen in Figure 3.2(a). Interpret the curves of unloads and loads as the cumulative number of containers added to and removed from the terminal. The picture is just a conventional queueing diagram for the landside of the terminal.

With single cycling, the minimum number of containers on the ship at any time during the loading and unloading operations is 0 (when unloading is complete), so with wheeled operations the number of chassis required (k_s) equals the number of containers:

$$k_s = \sum_{i=1}^{R} \Lambda_i + \sum_{i=1}^{R} Y_i \tag{3.16}$$

So on average, for the largest ship that repeatedly calls at the berth,

$$E[k_s] = E[\sum_{i=1}^{R} \Lambda_i + \sum_{i=1}^{R} Y_i] = \sum_{i=1}^{R} C_i \{\mu_u + \mu_l\} \tag{3.17}$$

and,

$$var(k_s) = var(\sum_{i=1}^{R} \Lambda_i + \sum_{i=1}^{R} Y_i) = \sum_{i=1}^{R} C_i \{var(u_{c,i}) + var(l_{c,i})\} \tag{3.18}$$

If we plan to almost always have enough chassis, the number required is given by:

$$E[k_s] + 2\sqrt{var(k_s)} = \sum_{i=1}^{R} C_i \{\mu_u + \mu_l\} + 2\sqrt{\sum_{i=1}^{R} C_i \{var(u_{c,i}) + var(l_{c,i})\}} \tag{3.19}$$

We can significantly reduce the number of chassis required by choosing to single cycle within one row. Instead of unloading the entire vessel before loading, unload just one row, and load that row, before moving on to the next row on the vessel. The number of containers on the ship at any one time can be approximated by a lower bound:

$$(R-1) \min_{i \in 1..R} \{\min \{E[Y_i], E[\Lambda_i]\}\} \tag{3.20}$$

We estimate the number of chassis required when using the single cycle by row an upper bound:

$$\sum_{i=1}^{R} C_i \{\mu_u + \mu_l\} - (R-1) \min_{i \in 1..R} \{\min \{E[Y_i], E[\Lambda_i]\}\} \tag{3.21}$$

The main reduction in the number of chassis required comes not from double cycling over single cycling, but by loading and unloading simultaneously, rather than unloading the vessel before loading it. Therefore, while we consider the economic benefits here, we do not consider them as a

benefit of double cycling, and therefore not in section 4.3. Requirements for provision of chassis will be reduced by single cycling on each row. The chassis provider could therefore utilize these chassis in another way. Assume a yard chassis costs about $20000. Amortized over 20 years, the chassis costs $0.12 per hour (excluding maintenance and management costs). On a large vessel with 1500 containers to move we remove the need for chassis for 5 hours, and it would not be atypical reduce our chassis requirement by 25% for the 45 hours during which the vessel was being loaded and unloaded. This could provide a savings of almost $2.00 per container moved. Notice that for operations that are "on wheels", where containers are stored on chassis rather than on the ground, there is an additional $4.00 benefit per container moved using double cycling, over operations that are "grounded".

Any additional reduction in number of chassis required when double cycling is small, and as we have seen these benefits are of minor economic significance. In an evenly loaded vessel (where $E[\Lambda_i] = E[Y_i] \forall i$), we could reduce the number of chassis required by approximately $(R-1)E[\Lambda_i]$.

3.4.4 Yard Storage Locations

A storage space is a location within the terminal where an import or export container can be stored for a period of time. For an import container this time period includes some time after the vessel departs. For an export container this includes some time before the vessel arrives.

Containers for import and export are typically stored in different locations on the terminal. In this case, the use of double cycling will not reduce storage space requirements. There will be fewer containers in port during loading and unloading operations, and this may be of some benefit as space is freed for maneuvering, but this benefit is difficult to quantify without empirical data.

Here we discuss the efficiencies of sharing import and export storage. Figures 3.9 and

3.10 show hypothetical terminal traffic patterns and storage locations for export containers, import containers, and empty chassis. These figures clearly simplify reality, where, for example, empty chassis may be segregated by carrier, and imports and exports may be stored in several different locations, but the general results still hold. While the introduction of double cycling has removed the wasted trips between the storage facilities and the apron, it has added the wasted trip between the import and export storage facilities. To minimize the distance of this trip, containers for import and export should be placed as close together as possible, or co-located. The ability of a port to do so will depend on existing storage strategies. Typically, terminals consider the shipping line, storage mode, destination, outbound transportation mode, weight, importer and/or size of the container when determining the storage location. Of course, the terminal would need to consider how any changes impacted the time to locate a container and reposition containers to access it.

In fact, when using chassis, more significant benefits can be generated by not segregating imports and exports at all, and using the same storage locations for either import containers or export containers. If containers are being stacked in local storage, the cost of additional labor required to retrieve containers would need to be compared to the benefits of reducing the storage footprint. This has two benefits:

- Reducing the distance and thus the travel time between the local export and import storage (e in equation 3.15). This reduces the requirements for landside vehicles and drivers.

- Achieving higher utilization of the storage facility by pooling the storage facilities and smoothing demand. When segregating storage we need to provide $E[\sum_{i=1}^{R} Y_i] + 2\sqrt{var(\sum_{i=1}^{R} Y_i)}$ locations for imports and $E[\sum_{i=1}^{R} \Lambda_i] + 2\sqrt{var(\sum_{i=1}^{R} \Lambda_i)}$ locations for exports. When pooling storage

facilities we only require, $E[\sum_{i=1}^{R} Y_i] + E[\sum_{i=1}^{R} \Lambda_i] + 2\sqrt{var(\sum_{i=1}^{R} Y_i) + var(\sum_{i=1}^{R} \Lambda_i)}$. By getting higher utilization out of the storage facility the port can reduce its land requirements for the same throughput.

3.4.5 Stevedores

Stevedores are present on the vessel when containers are being loaded and unloaded. They do not guide containers into position, but ensure that containers are lashed down correctly and stored in the correct positions. Given this, we do not expect that additional staff will be required when carrying out double cycling. The same gang of Stevedores will be asked to monitor containers being placed on and retrieved from the vessel. Of course, when turn around time is reduced, less labor may be required.

3.4.6 Container Handling Equipment

With double cycling, containers are being placed in short-term storage and retrieved from short-term storage, simultaneously. With the current method of vessel loading and unloading, containers are placed in local storage before any containers are retrieved. Depending on how the port operates, additional container handling equipment, for example, top picks or reach stackers, may be required to move containers between the landside vehicle and the correct storage position, so that machines are available to both retrieve and store containers simultaneously.

Chapter 4

Impact on Maritime Transportation

In this chapter we again expand our perspective to consider some aspects of the maritime transportation system that may interact with the decision to double cycle. These include security regulations, and ship design and routing. We also consider the economic costs and benefits of double cycling.

4.1 Interaction with New Security Regulations

Over the last three years much attention has been paid to port security, and the risks presented by the transport of containerized cargo. The Coast Guard, the Department of Homeland Security, the US Bureau of Customs and Border Protection (Customs), and the Transportation Security Administration (TSA) have all introduced initiatives to tighten transportation security. These initiatives include the Coast Guard's 72-hour rule and Maritime Transportation Security Act, Customs' 24-hour rule, Container Security Initiative, and Customs Trade Partnership Against Terrorism, and TSA's Operation Safe Commerce, and the Central American Free Trade Agreement.

Based on observations of the Ben E. Nutter Terminal at the Port of Oakland, these security initiatives have required changes in the flow of exit traffic from the port, and in the processing of high-risk containers that arrive at the port. Processing and inspection of high-risk containers has required allocation of space near the vessel for storage of these containers on chassis. Re-routing of traffic past customs officials has required additional space at the exit gate for officials, and an area for trucks to wait if any further inspection is necessary. Neither of these changes will affect the ability to reduce operational time with double cycling.

4.2 Ship Design and Routing

4.2.1 Optimal Number of Hatches

Theorems 2.5.0.1 and 2.5.0.2 suggest a trade-off between the height of the stacks above and below deck, and the optimal number of hatches (with respect to maximizing the opportunities to double cycle). Of course, the optimal number of hatches is one for each stack, or none, as in either case we can operate on each stack individually, but assuming this is not feasible, the optimal number can be determined using the following analysis. We assume the vessel has hatch coverings that cover more than one stack, and that the greedy hatch strategy is used (as described in Section ??). Recall the following notation:

- F - number of hatches per row
- C_i - number of stacks per hatch
- $\overline{u}_{hc}$ - number of containers to unload in one stack below deck
- $\underline{u}_{hc}$ - number of containers to unload in one stack above deck

We would like to determine the optimal number of hatches, F^*, the number that minimizes the number of times we are only able to single cycle. We must single cycle while unloading all containers atop the first hatch, and we must single cycle for one stack below each hatch.

$$F^* = argmin_F \left\{ \underline{u}_{hc} \frac{C_i}{F} + \overline{u}_{hc} F \right\} \quad s.t. \quad F > 0 \tag{4.1}$$

Hatchless vessels currently exist. In their 1996 paper [7], Bendall and Stent present the benefits of the hatchless vessel, a significant portion of which come from the reduced turn around time due to double cycling. But, in the last 10 years, these vessels have not made any significant progress in the market. Given this, it is useful to consider the optimal number of hatches, but removing one per stack (or hatchless vessels) from consideration. Assuming $F^* < C_i$ and $F^* \neq 0$ the optimal number of hatches is given by:

$$F^* = \sqrt{\frac{\underline{u}_{hc} C_i}{\overline{u}_{hc}}}, for \quad F < C_i, F \neq 0 \tag{4.2}$$

For the same total number of stacks per row, by making the hatches larger, we increase the number of containers atop each hatch, similarly, by making the hatches smaller, we decrease the number of containers atop each hatch. With smaller hatches we can do more double cycling above deck, because we reduce the size of the initial penalty (not being able to double cycle atop the first hatch). With larger hatches we can do more double cycling below deck because we reduce the number of times we must pay the penalty of unloading the first stack under each hatch. We assume the vessels will be fully utilized and attempt to minimize the number of cycles during which we cannot double cycle.

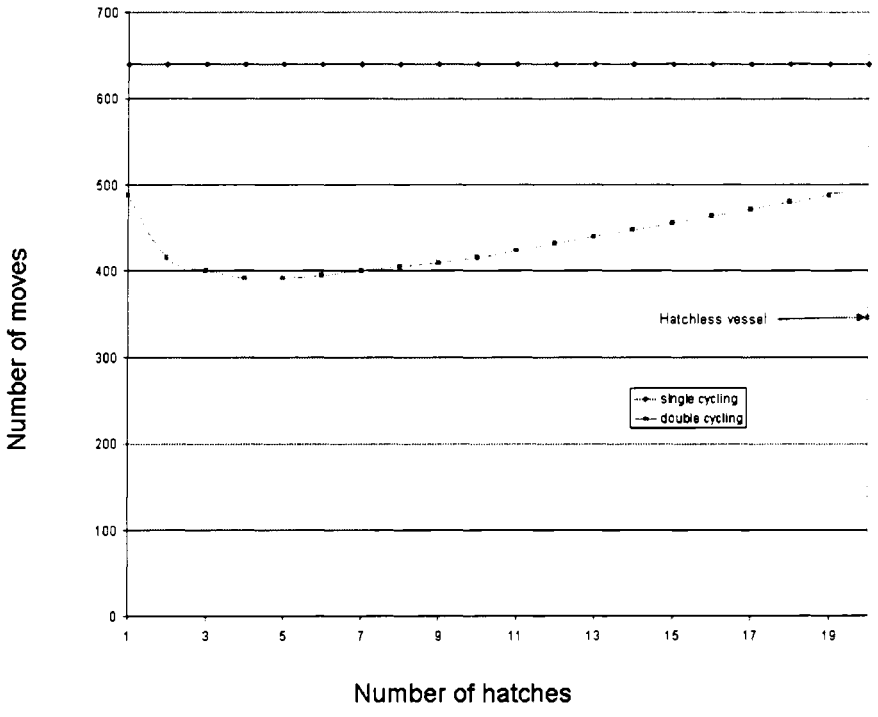


Figure 4.1: The number of moves using the greedy hatch strategy, and single cycling for a specific vessel. The vessel has 20 stacks, so 1 through 20 hatches are considered.

Figure 4.1 shows an example. The number of moves using single cycling and the greedy hatch strategy are shown for an increasing number of hatches. In this example there are 20 stacks, with 8 containers to unload and load above deck and below deck in each stack. Again, note the optimal number of hatches is 20 (which is equivalent to none), one for each stack, so that we can operate on each stack independently, as in the hatchless case, but this is not practical. Excluding this, the optimal number of hatches is approximately 5. In the next section we consider a vessel's optimal deck height, in light of double cycling.

4.2.2 Optimal Deck Height

Theorems 2.5.0.1 and 2.5.0.2 also suggest a trade-off between the height of the stacks above and below deck. In this section we will develop a formula for determining the optimal deck height; one that will provide the most opportunities to double cycle. Recall the following notation:

- F - number of hatches per row
- C_i - number of stacks per hatch
- $\overline{u}_h c$ - number of containers to unload in one stack above deck
- $\underline{u}_h c$ - the number of containers to unload in one stack below deck

Also define the following dimensionless parameters:

- $\eta = \frac{F}{\sqrt{C_i}}$, a measure of hatch size
- $\delta = \frac{E[\overline{u}_{hc}]}{E[\overline{u}_{hc}]+E[\underline{u}_{hc}]}$, the ratio of containers below deck to the total number (above and below deck) in a stack

To determine the optimal deck height, δ^*, we again minimize the number of times we can only single cycle, and it is sufficient to solve:

$$\delta^* = \min\left\{\frac{1-\delta}{\eta} + \eta\delta\right\} \tag{4.3}$$

Figure 4.2 shows the trade-off between η and δ for various C_i. For this analysis we assume double cycling occurs above and below deck using the greedy hatch strategy. Notice the asymptote at $\eta = 1$. The figure shows some subtleties regarding the interaction of the parameters, but more broadly the results are intuitive. For example, with fewer containers below deck we want more hatches, and with containers only below deck we want few hatches. Of course, ideally we would have no deck (the same results as one hatch per stack with the deck at any height) as this allows us to always double cycle after unloading the first stack. This makes sense, as we know from intuition that it would be most efficient to use a hatchless vessel.

4.2.3 Ship Size, Routing and Stopping Frequency

The benefits of double cycling increase with ship size as it is preferable to have continuous blocks of containers upon which we can double cycle. There are distinct economies of scale with maritime transportation, and for that reason container vessels continue to increase in size. These ships will likely make fewer stops due to the large cost of idle vessels, and feeder ships will continue to be used to visit smaller ports. In addition, these large vessels will be unable to visit many ports due to insufficient water depth at the ports. In this future scenario, double cycling can provide more benefit, as rows of the larger vessel provide larger blocks of containers in which we can double cycle. In addition, fewer stops for the vessel means larger blocks of containers for the same OD

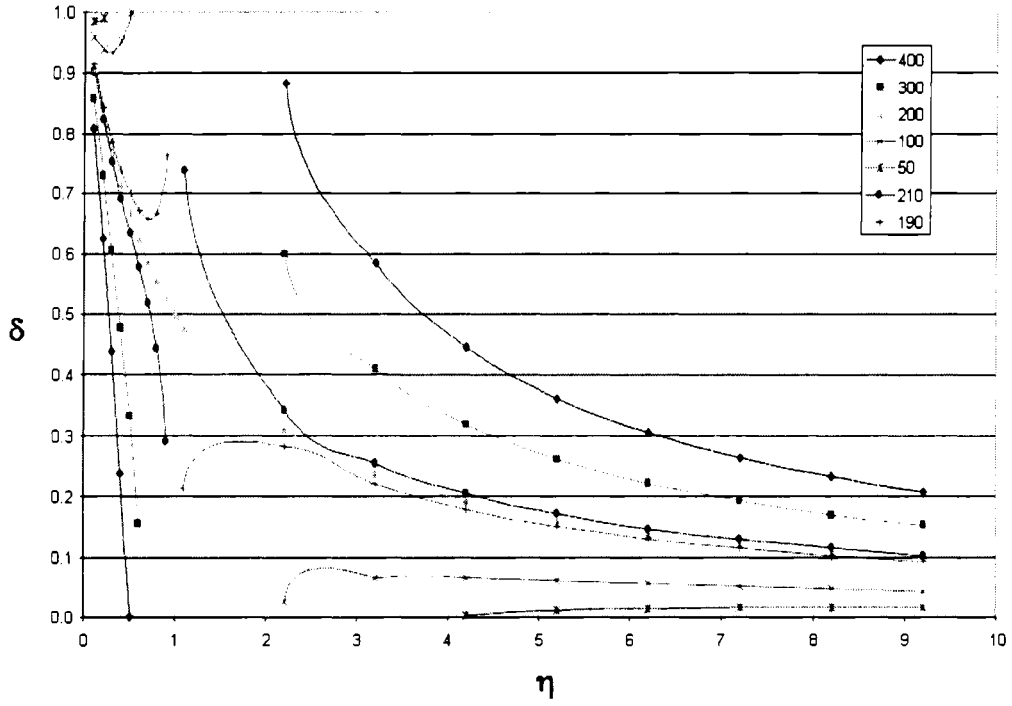


Figure 4.2: Optimal deck height for varying parameters η and δ.

pair, and therefore larger blocks of containers in which we can double cycle. This is especially true if, at some point in the future, we can use cranes to double cycle within more than one row.

4.3 Economic Evaluation

In this section we provide estimates as to the order of magnitude of the financial impact of double cycling. As mentioned, ports vary distinctly in their ownership and fee structures, so a more detailed analysis should be carried out on a case by case basis.

We provide a very rough estimate of the opportunity costs assuming that the freed resources can be usefully employed. In practical terms this means the released capacity can be used to move additional containers. This assumption reflects current market conditions, where demand is expected to exceed capacity during the peak season.

Using equation 3.1 we can convert the benefits from a number of cycles to an amount of time. Using the same parameter values as given in Table 3.3 we would expect a reduction by 25% in the number of moves to equate to a time reduction of 9.5%. In the following subsections we will consider the economic impact of a 10% reduction in operating time. We assume a typical vessel, capable of carrying 6000 TEUs, unloads and loads 1500 containers in 50 hours using single cycling and 45 hours using double cycling. This equates to moving 30 containers per hour with single cycling, and 33 with the proximal stack strategy. We compare all of the benefits in dollars per container moved in Table 4.1.

In Table 4.1, under savings we list the approximate value, in dollars per container moved, of employing the resources that are freed with double cycling. Detailed assumptions for each resource are described in the sections below. In the case of landside vehicles the savings would be

Category	Proximal Stack Strategy	Savings ($/container moved)	Extra Cost ($/container moved)	Beneficiary
Waterside	Vessel Productivity	$40.00	$13.00	Vessel Operator
	Crane Productivity	$22.00	$0.00	Terminal Operator
	Single Berth Utilization - Warfage Fees	$22.00	$0.00	Port Authority
	Single Berth Utilization - Dockage Fees	$1.60	$0.00	Vessel Operator
Landside	Container Handling Equipment (per vehicle and driver if grounded)	$0.00	$2.00	Terminal Operator
	Landside Vehicles (per vehicle and driver)	$0.20	$0.00	Terminal Operator

Table 4.1: Comparison of the approximate economic benefits of double cycling.

obtained by renting out or selling off unused resources. In the case of berth utilization - dockage fees, this benefit comes from reduced expenses. In the case of vessel productivity, crane productivity and berth utilization, these benefits come from utilizing the resources freed by double cycling. Under extra cost, we list the cost of using the resource for the time it is freed up by double cycling. In the case of vessel productivity this cost is incurred by consumption of fuel, in the case of container handling equipment this includes the equipment, operator, fuel and maintenance cost of the new machine.

Notice the order of magnitude difference in the net benefits on the waterside (on the order of $10.00) as compared to the benefits landside (on the order of $1.00). Notice also the benefit from reducing requirements for landside vehicles and drivers are essentially negligible. The increased requirement for container handling equipment comes at a cost on the order of $1.00 per container moved. The greatest benefits come from higher utilization of the vessel, crane, and berth. A significant portion of the benefits of double cycling are experienced by parties who are not responsible

for implementing the operation. These results highlight the distributed nature of the benefits of double cycling. If a larger portion of the benefits were experienced by those responsible for its implementation, we may see more widespread use of the technique.

Major shipping lines APL and P&O Nedloyd experienced average revenue of almost $3000 per FEU in 2004 [2], [64]. If double cycling can save the vessel operator approximately $25.00 per FEU at each port this would reduce transportation cost, and increase profit by about 1.5%, a significant benefit.

4.3.1 Vessel Productivity

Given a reduction in idle time (while unloading and loading), a vessel has the opportunity to increase productive time (transporting containers). We assume this productive time can generate revenue by transporting containers.

Revenue

Using freight rates from [66], we estimate the additional revenue generated for an average vessel at $54000 (assuming a 6000 TEU vessel can travel 200 km in the 5 hours saved). This is a value of $36.00 per container moved (because we assume a total of 1500 containers are loaded and unloaded).

Cost

We only incur the additional cost of operating the vessel as the cost of ownership and staffing are not increased. We consider only the cost of fuel, and ignore the increase in maintenance cost as this value is much smaller. We estimate the cost of fuel at $100 per km for a 6000 TEU

vessel and again assume the vessel could travel 200 km in the 5 hours saved. This would cost the vessel operator $20000 in fuel, or $13.33 per container moved.

Inventory Cost

By reducing operational time, we reduce the travel time for all containers, and thus reduce inventory cost. The average value of a container, for low value goods such as those entering the US from Asia, is $25,000 [76]. We assume the vessel is loaded with 6,000 containers. With an interest rate of 8% we reduce the inventory cost by $6850 per vessel, or about $4.50 per container moved. Note that this benefit increases with ship size.

The net benefit of $27.24 per container moved or about $750 an hour compares well with the benefits when calculated using the profit per TEU experienced in the industry in 2004. These profits fall within the range of $100 to $500 per FEU or $11.00 to $55.00 per container moved [2],[64].

4.3.2 Opportunity Cost of Crane Time

We have ignored any additional cost increase due to increased fuel consumption or maintenance caused by a greater percentage of the crane's time spent moving containers versus moving empty. Terminals are typically paid for each container moved, and although these cost structures are complicated, average rates are on the order of $200 per container [54]. In the 5 hours of crane time freed for a typical vessel we could generate $33,000, or $22.00 per container moved.

4.3.3 Berth Utilization

Again we do not incur any additional cost, but there is the opportunity to increase revenue if additional capacity is used to move containers. Shipping lines typically pay both wharfage and dockage fees to the terminal or the port authority, depending on the particular port's structure. Dockage fees are paid by the vessel operator to the port authority on a daily or hourly basis. Therefore, revenue to the port authority will not increase but cost per vessel will decrease. Warfage rates are also paid by the vessel operator to the port authority, but on a container basis. Therefore, rates for the same vessel will not decrease, but total revenue to the port authority will increase.

Dockage Fees

Dockage fees at the Port of Oakland are $12000 per day for the largest vessels (more than 1200 feet long) and $3000 per day for smaller vessels (640 - 700 feet). A vessel that transfers 1500 containers during a 50 hour port visit, currently pays about $24000 in dockage fees. This could be reduced by $2400 using double cycling, or $1.60 per container move.

Warfage Fees

Warfage rates are typically paid by the container, and are on the order of $200 per container. Income would increase from wharfage fees to the port authority or terminal operator by about $22.00 per container moved.

4.3.4 Landside Vehicles and Drivers

Requirements for provision of landside vehicles and drivers are reduced, so there is an opportunity to reduce staffing and sell or rent the vehicles. Yard tractor or straddle carrier operators

are typically unionized and earn close to $100000 annually [38]. A new yard tractor costs about $50000. We amortize this cost over 20 years, operating 40 hours a week, 52 weeks a year. We assume $10.00 per hour for fuel and maintenance. This provides a benefit of $.18 per container moved. In a typical operation we might be able to save 3 drivers and tractors per crane.

4.3.5 Additional Container Handling Equipment

With double cycling, we will be storing containers for import, and retrieving containers for export, simultaneously. If the operations at the terminal require top picks, or other container handling equipment for storage and retrieval, the use of double cycling may increase the need for this equipment (depending on the method of operation it could be at most doubled). The number of handlers required depends on the layout of storage in the terminal. At a cost of about $400000, amortized over 20 years, operating 40 hours a week, 52 weeks a year, each handler costs almost $9.62 per hour. We expect maintenance and fuel costs are approximately $10.00 per hour. In addition, vehicle operators will be required at approximately $50.00 per hour. The additional cost of container handling equipment is thus about $2.00 per container moved.

4.4 Implementation Obstacles

The previous section demonstrates a major obstacle to implementing double cycling; the distributed nature of the benefits. Many lay people believe it is organized labor's resistance to change, or a general resistance to change in the industry. I don't agree. After study of the topic for many years and many conversations with those in industry I believe the reasons double cycling has not already been widely adopted are the following:

- Distribution of economic benefits
- Larger issues occupying the minds of managers
- Ad hoc nature with which much of the operations are carried out
- Experience with double cycling to date has been with small scale implementations

Since the lockout of International Longshore and Warehouse Union (ILWU) workers on the West Coast in 2002, terminal operators have been rushing to implement new technologies before renegotiation of the contract in 2006. This means that during the period of most congestion, terminal operators minds have been on technological solutions rather than operational ones. Before this time, capacity was not so constrained that terminal operators were feeling the pressure to make changes.

After spending a significant amount of time at both the Ports of Oakland and Long Beach, my impression of terminal operations is that many decisions made on a day to day basis are made to simplify operations, not necessarily make them more efficient. For example storage strategies (the storage of import and export containers separately). Most terminal managers are not trained in operations research techniques, but have significant experience running terminals, and aim to do so smoothly rather than most efficiently.

Those with experience with double cycling have typically used it to unload containers from one bay of a vessel. Small scale implementations understate the benefits of double cycling since, as we have seen, the benefits of double cycling increase with the number of stacks. Further, a small test example does not show the shipping line the benefits they could achieve since these would come from sustained operations (where they could adjust their schedule, etc.).

In future research I plan to understand more fully the relationship between economic benefits and the implementation of efficiency improvements, for different port ownership structures. For example, if shipping lines share the substantial benefits they can expect from double cycling with terminal operators, terminal operators will be more motivated to implement this efficiency improvement.

Chapter 5

Summary

Through this research we have developed an understanding of the impact of double-cycling on loading and unloading operations both with respect to the number of cycles, and amount of time. We have provided port planners with tools and insights as to the impact of double cycling on requirements for port resources. The results are general enough to allow for broad conclusions, but also allow us to quantify the benefits for specific cases. For example our results obtained are applicable to a wide range of scenarios including large and small vessels, those with many or few port visits, and a varying number of cranes operating per ship. An alternative approach that focused on providing more accurate information for specific cases would require detailed information on each container. Our approach relies on only a small number of parameters, and sensitivities to these parameters are easily measured. Favorable comparisons to empirical results support our approach.

We have also considered double-cycling landside equipment, where, for example, vehicles used to deliver containers to the apron then carry an export container to local storage instead of returning empty. We have provided tools to understand the impact on all relevant aspects of landside

operations, including vehicle traffic, container handling equipment, and container storage. We have made suggestions to streamline traffic flows and integrate double cycling into existing operations. Outside of double cycling we have made suggestions for improving storage yard utilization and reducing chassis requirements.

We have quantified the key operational and financial benefits of double cycling; increased crane productivity, berth utilization, and vessel utilization. We have shown the net benefit of double cycling is on the order of $70.00 per container moved, a significant value. Double cycling does not require significant capital investment beyond additional container handling equipment, only additional planning and modifications to the terminal operating system. The most sizable piece of these benefits goes to vessel operators, who are not responsible for port operations, and therefore cannot independently implement double cycling. We believe this in part explains the slow implementation of the practice.

While double cycling will not eliminate current port congestion, it can be implemented quickly and, in conjunction with other measures, can ease congestion before more long-term infrastructure projects come on line. The next step in our research is to understand how other port resources, such as gate time and rail capacity can accommodate this additional traffic. We will also continue to work on understanding the distributed nature of the benefits of double cycling and its affect on implementation. We will examine remuneration schemes and other cost models, as tools to encourage implementation of operational improvements for different port management paradigms.

Bibliography

[1] H.M. Abdel-Wahab and T. Kameda. Scheduling to minimize maximum cummulative cost subject to series-parallel precedence constraints. *Operations Research*, 26:141–158, 1978.

[2] APL. Annual report. Report available at www.apl.com, 2004.

[3] D. Applegate and W. Cook. A computational study of the job-shop scheduling problem. *ORSA Journal on Computing*.

[4] W.H. Atkins. Modern Marine Terminal Operations and Management. The Compage Company, San Francisco, 1983.

[5] G.L. Curry B. Gim and B.L. Deuermeyer.

[6] K. Behrokh and A. Asef-Vaziri. 3D virtual and physical simulation of automated container terminal and analysis of impact on inland transportation. METRANS report, 2000.

[7] H.B. Bendall and A.F. Stent. Hatchcoverless container ships: Productivity gains from a new technology. *Maritime Policy and Management*, 23(2):187–199, 1996.

[8] E.K. Bish, T. Leong, C. Li, J.W.C. Ng, and D. Simchi-Levi. Analysis of a new vehicle scheduling and location problem. *Naval Research Logistics*, 48(5):363–385, 2001.

[9] J.K. Lenstra B.J. Lageweg and A.H.G. Rinnooy Kan. A general bounding scheme for the permutation flow-shop problem. *Operations Research*, 26:53–67, 1978.

[10] Y.A. Bozer and J.A. White. Travel-time models for automated storage/retrieval systems. *IIE Transactions*, 16(4):329–338, 1984.

[11] B. D. Castilho and C. F. Daganzo. Handling strategies for import containers at marine terminals. *Transportation Research B:Methodology*, 27:151–166, 1993.

[12] C.E. Clark. The greatest of a finite set of random variables. *Operations Research*, 9(2):145–162, 1961.

[13] TranSystems Corporation. Efficient marine terminal full scale demonstration. Report available at www.ccdott.org, 2003.

[14] T. G. Crainic and K. H. Kim. Transportation, chapter title Intermodal Transportation. Handbooks in Operations Research and Management Science. North-Holland, Amsterdam, The Netherlands, 2005. Forthcoming.

[15] C. F. Daganzo. The crane scheduling probem. *Transportation Research B: Methodology*, 23:159–175, 1989.

[16] C. F. Daganzo. The productivity of multipurpose seaport terminals. *Transportation Science*, 24:205–216, 1990.

[17] E. Diez-Roux and C.F. Daganzo. The variance of the number of customers in an infinite server queueing system with an arbitrarily correlated arrival process. *Queueing Systems*, 19:289–299, 1995.

[18] E.A. Elsayed and O.I Unal. Order batching algorithms and travel time estimation for automated storage/retrieval systems. *International Journal of Production Research*, 27(7):1097–1114, 1989.

[19] M. Christiansen et al. Transportation, chapter title Maritime Transportation. Handbooks in Operations Research and Management Science. North-Holland, Amsterdam, The Netherlands, 2005. Forthcoming.

[20] William Feller. Probability Theory and its Applications. John Wiley & Sons, London, 1950.

[21] S. French. Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop. Halsted Press: a division of John Wiley and Sons, New York, 1982.

[22] B. Garcia. Personal communications. Transystems Corportation, 2003-2005.

[23] M. Garey and D. Johnson. Analysis and Design of Algorithms in Combinatorial Optimization, chapter title Approximation algorithms for bin packing problems-A survey. Springer-Verlag.

[24] G.A. Godfrey and W.B. Powell. An adaptive dynamic programming algorithm for dynamic fleet management i: Single period travel times. *Transportation Science*, 36(1):21–39, 2002.

[25] G.A. Godfrey and W.B. Powell. An adaptive dynamic programming algorithm for dynamic fleet management ii: Multiperiod travel times. *Transportation Science*, 36(1):40–54, 2002.

[26] A. V. Goodchild. *Crane Double Cycling in Container Ports: Algorithms, Evaluation, and Planning*. PhD thesis, University of California at Berkeley, Department of Civil and Environmental Engineering, 2005.

[27] A. V. Goodchild. Performance comparison of double cycling strategies. The Institute for Transportation Studies Working Paper UCB-ITS-WP-2005-2, August 2005.

[28] A.V. Goodchild and C. F. Daganzo. Crane double cycling in container ports: Effect on ship dwell time. The Institute for Transportation Studies Report UCB-ITS-RR-2005-5, 2005.

[29] D. Gusfield. Bounds for naive multiple machine scheduling with release times and deadlines. *Journal of Algorithms*, 5:1–6, 1984.

[30] M.H. Han, J.S. McGinnis, J.S Shieh, and J.A. White. On sequencing retrivals in an automated storage/retrieval system. *IIE Transactions*, 19(1):56–66, 1987.

[31] A.M.A. Hariri and C.N. Potts. Algorithms for two-machine flow-shop sequencing with precedence constraints. *European Journal of Operational Research*, 17(2):238–248, 1984.

[32] R.A. Dudek H.G. Campbell and M.L. Smith. A heuristic algorithm for the n job, m machine sequencing problem. *Management Science*, 16:B630–637, 1970.

[33] W.B. Hubbard. Buffered magazine method and system for loading and unloading ships. Application for Patent, 2002.

[34] A. Imai, E. Nishimura, and S. Papadimitriou. The dynamic berth allocation problem for a container port. *Transportation Research B: Methodology*, 35:401–417, 2001.

[35] A. Imai, E. Nishimura, and S. Papadimitriou. Berth allocation with service priority. *Transportation Research B: Methodology*, 37:437–457, 2003.

[36] Junichi Imakita. A Techno-Economic Analysis of the Port Transport System. Praeger Publishers, New York, 1978.

[37] A.H.G. Rinnooy Kan J.K. Lenstra and P. Brucker. Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 1:343–362, 1977.

[38] E. Johnson. L.B. and L.A. port wages top $100k.

[39] M.E. Johnson and M.L Brandeau. Stochastic modeling for automated material handling system design and control. *Transportation Science*, 30(4):330–350, 1996.

[40] S.M. Johnson. Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1:61–68, 1954.

[41] A.H.G. Rinnooy Kan. *Machine Scheduling Problems: Classification, Complexity and Computations*. PhD thesis, University of Amsterdam, 1976),.

[42] A. Keserla and B.A. Peters. Analysis of dual shuttle automated storage/retrieval systems. *Journal of Manufacturing Systems*, 13(6):424–434, 1994.

[43] K. H. Kim and J. W. Bae. A dispatching method for automated guided vehicles to minimize delay of containership operations. *International Journal of Managment Science*, 5:1–25, 1999.

[44] K. H. Kim and H. B. Kim. The optimal sizing of the storage space and handling facilities for import containers. *Transportation Research B: Methodology*, 36:821–835, 2002.

[45] K.H. Kim. Personal communications. Pusan National University, Korea, 2004.

[46] K.H. Kim, J.S. Kang, and K.R. Ryu. A beam search algorithm for the load sequencing of outbound containers in port container terminals. *OR Spectrum*, 26(10):93–116, 2004.

[47] K.H. Kim and Y.M. Park. A crane scheduling method for port container terminals. *European Journal of Operational Research*, 156:752–768, 2004.

[48] S. Kirkpatrick, C.D. Gerlatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[49] T. Kurisu. Two-machine scheduling under required precedence among jobs, 1976.

[50] T. Kurisu. Two-machine scheduling under arbitrary precedence constraints. *Journal of the Operations Research Society of Japan*, 20:113–131, 1977.

[51] K. K. Lai and J. Leung. Analysis of gate house operations in a container terminal. *International Journal of Modeling and Simulation*, 20:89–94, 2000.

[52] E.L. Lawler. Sequencing jobs to minimize total weighted completion time subject to precedence constraints. *Annals of Descrete Mathematics*.

[53] C.L. Li and J. Ou. Machine scheduling with pickup and delivery. *Naval Research Logistics*, Forthcoming. Available online as of July 15, 2005.

[54] C. May. Personal communications. Ben E. Nutter Terminal, Port of Oakland, 2005.

[55] G.B. McMahon and C.J. Lim. The 2-machine flow-shop problem with arbitrary precedence relations. *European Journal of Operational Research*, 64(2):249–257, 1993.

[56] L.G. Mitten. Sequencing n jobs on two machines with arbitrary time lags. *Managment Science*, 5:293–298, 1959.

[57] K. Monaco. Trucking: Issues and policy challenges. The Transportation, Landuse, Environment Connection, Linking Goods Movement to Economic Prosperity and Environmental Quality, Oct. 24-26, 2004.

[58] B. Mongelluzzo. L.A. won't see peak repeat. *The Journal of Commerce*, 6 (24):12, 2005.

[59] B. Mongelluzzo. A peek at the peak. *The Journal of Commerce*, 6 (26):16, 2005.

[60] B. Mongelluzzo. Ports rush to keep up. *The Journal of Commerce*, 6:26–30, 2005.

[61] C.L. Monma. The two-machine maximum flow time problem with series-parallel precedence constraints: an algorithm and extensions. *Operations Research*, 27:792–798, 1979.

[62] C.L. Monma. Sequencing to minimize the maximum job cost. *Operations Research*, 28:942–951, 1980.

[63] M. Taleb-Ibrahimi nad B. Castilho and C. F. Daganzo. Storage space vs handling work in container terminals. *Transportation Research B: Methodology*, 27:13–32, 1993.

[64] P&O Nedloyd. Annual report. Report available at www.ponl.com, 2004.

[65] G. F. Newell. Approximate Stochastic Behavior of n-Server Service Systems with Large n. Springer-Verlag, Berlin, Heidelberg, New York, 1987.

[66] T. Notteboom. Container shipping and ports: An overview. *Review of Network Economics*, 3(2):86–106, 2004.

[67] Y. M. Park and K. H. Kim. A scheduling method for berth and quay cranes. *OR Spectrum*, 25:1–23, 2003.

[68] R. I. Peterofsky and C. F. Daganzo. A branch and bound solution method for the crane scheduling problem. *Transportation Research B: Methodology*, 24:159–172, 1990.

[69] H. Peters. Personal communications. International Finance Corporation, 2003.

[70] M. Pinedo. Scheduling: Theory, Algorithms, and Systems. Prentice Hall, Upper Saddle River, N.J., 2002.

[71] W.H. Press, B. Flannery, S. Tevkolsky, and W. Vettering. Numerical Recipes in C. Cambridge University Press, New York, 1986.

[72] E. Rath. Container Systems. Wiley.

[73] P. Schonfeld and O. Sharafeldien. Optimal berth and crane combinations. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 111:1060–1072, 1985.

[74] J.B. Sidney. The two-machine maximum flow time problem with series-parallel precedence relations. *Operations Research*, 27:782–791, 1979.

[75] B. Slack. Transformation of port terminal operations: from the local to the global. *Transport Reviews*, 25 (1):117–130, 2005.

[76] GE Insurance Solutions. Information available at www.gesolutions.com, 2005.

[77] P.J.M. van Laarhoven and E.H.L Aarts. Simulated Annealing: Theory and Applications. Kluwer Academic Publishers, Dordrecht, 1987.

[78] I. F. A. Vis, R. de Koster, and K. J. Roodbergen. Determination of the number of automoated guided vehicles required at a semi-automated container terminal. *Journal of the Operational Research Society*, 52:409–417, 2001.

[79] T. Ward. Personal communications. JWD Group, 2003-2005.

[80] V. Yau. Personal communications. American President Lines, 2003.

Appendix A

The Beta Distribution

Definition A.0.0.1 (Probability Function of the Beta Distribution) $P(x) = \frac{(1-x)^{q-1}x^{p-1}}{B(p,q)}$. Where B is the beta function, $B(p,q) = \frac{(p-1)!(q-1)!}{(p-q-1)!}$. The mean and variance of the beta distribution are $\mu = \frac{p}{p+q}$, and $\sigma^2 = \frac{pq}{(p+q)^2(p+q+1)}$.

Appendix B

Number of Landside Vehicles and Drivers

- $a(w)$ = the number of landside vehicles to arrive at II in the period $\{w, w + dw\}$
- $\tilde{F}(w)$ = the probability that a container is still at II at time $t = 0$ given it arrived at time $w = -t$
- λ = arrival rate of customers at II
- I = index of dispersion of the arrival rate at II
- α = all previous arrivals at all intervals at II
- $\bar{w} = E(w) = \sum_{w} dw\tilde{F}(w)$

Then the number of arrivals in the period $\{w, w + dw\}$ still at II at $t = 0$ has a binomial distribution with parameters, $B(a(w), \tilde{F}(w))$. Given this information, we would like to determine

the mean and variance of the number of customers in service at a point in time. This will tell us the required number of chassis. The number of customers in service at time t is demoted by $Q(0)$.

$[Q(0)/\alpha]$ is the number of customers in service at time t given α.

$[Q(0)/\alpha] = \sum_w B(a(w), \tilde{F}(w))$. We can then calculate the expected value and variance, conditional on the arrival history. $E[Q(0)/\alpha] = \sum_w a(w)\tilde{F}(w)$, and $var[Q(0)/\alpha] = \sum_w a(w)\tilde{F}(w)[1 - \tilde{F}(w)]$. Taking the expectation across all α, we obtain $E[Q] = E_\alpha(E(Q/\alpha)) = \sum_w E(a(w))\tilde{F}(w) = \lambda \sum_w \tilde{F}(w)dw = \lambda\bar{w}$. $var[Q] = var_\alpha(E(Q/\alpha)) + E_\alpha(var(Q/\alpha)) = \sum_w \tilde{F}(w)^2 var(a(w)) + \sum_w E(a(w))\tilde{F}(w)(1 - \tilde{F}(w)) = \sum_w \tilde{F}(w)^2 \lambda I dw + \sum_w \lambda dw \tilde{F}(w)(1 - \tilde{F}(w)) = \lambda \sum_w dw\tilde{F}(w) + (\lambda I - \lambda)\sum_w dw(\tilde{F}(w))^2 = \lambda\bar{w} + \lambda(I-1)\sum_w dw(\tilde{F}(w))^2 = \lambda\bar{w} + \lambda(I-1)E[min(w_1, w_2)]$. That $\sum_w dw(\tilde{F}(w))^2 = E[min(w_1, w_2)]$ is shown in proof 1 below.

Definition B.0.0.2 (Complement W) $\tilde{F}_W(y) = 1 - F_W(y)$

Definition B.0.0.3 (Complement Y) $\tilde{F}_Y(y) = 1 - F_Y(y)$

Proof (Proof 1)Let $y = min(w_1, w_2)$. As shown in proof 2 below:

$$F_Y(y) = 1 - (1 - F_W(y))^2 \tag{B.1}$$

As shown with proof 3 below:

$$E[y] = \sum_y dy\tilde{F}_Y(y) \tag{B.2}$$

Therefore,

$$\tilde{F}_Y(y) = 1 - Y_Y(y) = (1 - F_W(y))^2 \tag{B.3}$$

$E[min(w_1, w_2)] = E[y] = \sum_y dy\tilde{F}_Y(y)$ by B.2. $\sum_y dy\tilde{F}_Y(y) = \sum_y dy(1 - F_W(y))^2$ by B.3. $\sum_y dy(1 - F_W(y))^2 = \sum_y dy(\tilde{F}_W(y))^2$ by B.0.0.2, so we can write $E[min(W_1, w_2)] = \sum_w dw\tilde{F}_W(w)^2$ replacing variable y with w as $F_W(y) = PrW \leq y = PrW \leq w = F_W(w)$ if $w = y$. ∎

Proof (Proof 2)$F_Y(y) = Pr\{Y \leq y\} = 1 - Pr\{Y > y\} = 1 - Pr\{min(w_1, w_2) > y\} = 1 - Pr\{W_1 > y, w_2 > y\}$. Given that w_1 and w_2 are independent this is equivalent to: $1 - Pr\{w_1 > y, w_2 > y\} = 1 - \prod_{n=1}^{2} Pr\{w_n > y\} = 1 - \prod_{n=1}^{2}(1 - Pr\{w_n \leq y\}) = 1 - \prod_{n=1}^{2}(1 - F_W(y))$. Therefore when w_1 and w_2 are independent, identically distributed, $F_Y(y) = 1 - (1 - F_W(y))^2$. ∎

Proof (Proof 3)By definition, $E[y] = \int_{-\infty}^{\infty} yf(y)dy$. When there are only positive values of y this is equivalent to $\int_0^{\infty} yf(y)dy$. Using integration by parts this is equivalent to: $yF(y)\,|_0^{\infty} - \int_0^{\infty} F(y)dy = \int_0^{\infty} dy - \int_0^{\infty} F(y)dy = \int_0^{\infty}[1 - F(y)]dy = \int_0^{\infty} \tilde{F}(y)dy$. So in the discrete case this gives us $E[y] = \sum_y dy\tilde{F}_Y(y)$. ∎

The number of landside vehicles provided to service II can be estimated by the mean number of vehicles in service plus two standard deviations.

$$E[Q] + 2\sqrt{var(Q)} = \lambda\bar{w} + 2\sqrt{\lambda\bar{w} + \lambda(I-1)\overline{E[min(w_1, w_2)]}} \tag{B.4}$$

Note in our case $I = 0$. So the expression can be simplified to:

$$\lambda\bar{w} + 2\sqrt{\lambda\{\bar{w} - E[min(w_1, w_2)]\}} \tag{B.5}$$

Appendix C

List of Acronyms and Abbreviations

- APS - Agile Port System
- AS/RS - Automated Storage and Retrieval System
- CCDoTT - The Center for Commercial Development of Transportation Technologies
- CPU - Central Processing Unit
- DFC - Dedicated Freight Coordinator
- EMT - Efficient Marine Terminal
- FEU - Forty Foot Equivalent Unit
- IT - Information Technology
- S/R - Storage and Retrieval
- TEU - Twenty Foot Equivalent Unit