

Mathematical Modelling and Efficient Algorithms for Autonomous Straddle Carriers Planning at Automated Container Terminals

by

Shuai Yuan

**A thesis submitted in fulfilment
of the requirements for the degree of
Doctor of Philosophy**

University of Technology, Sydney
Faculty of Engineering

October 2013

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of a requirement for a degree except as fully acknowledged within the text.

I certify that the thesis has been written by me. Any help that I have received in my research work and preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

(Shuai Yuan)

Sydney, October 2013

Abstract

In the past several decades, automation of handling equipment has been a worldwide trend in seaport container terminals. Increasing automation of yard handling vehicles not only reduces the cost of terminal operation, but also increases the efficiency of container transport. However, the primary loss of performance in the transshipment process is caused by the uncoordinated allocation and scheduling of quay cranes, yard vehicles and land-side operations. Hence, integrating transshipment processes is imperative for a fully automated container terminal. This thesis aims to study an integrated process and develop practically deployable strategies and algorithms, with the practical example of the Patrick AutoStrad container terminal, located in Brisbane, Australia.

The thesis first formulates two mathematical models: The *Comprehensive Model* is an analytical optimisation model which integrates the quay-side, yard and land-side operational sub-problems of the Patrick AutoStrad container terminal. Derived from the comprehensive model, the *Job Scheduling Model* is formulated to focus on the optimisation of job scheduling, as job scheduling plays a more important role than path planning, and resource utilisation and port operation are more dependent on job scheduling.

To solve the *Comprehensive Model*, a job grouping approach is proposed for solving the integrated problem, and experimental results show that the job grouping approach can effectively improve the time related performance of planning container transfers. Solving the *Job Scheduling Model* using a global optimisation approach is expected to provide higher productivity in automated container terminals. Hence, a modified genetic algorithm is proposed for solving the job scheduling problem derived from the integrated mathematical model of container transfers. Moreover, the live testing results show that the proposed algorithm can effectively reduce the overall time-related cost of container transfers at the automated container terminal.

Last but not least, a new crossover approach is proposed in order to further improve the solution quality based on the modified genetic algorithm, and it can also be directly

applied in solving the generic multiple travelling salesmen problem using the two-part chromosome genetic algorithm. The experimental results also show that the proposed crossover approach statistically outperforms the existing approaches when solving the job scheduling problem and the standard multiple travelling salesmen problem.

Acknowledgements

This thesis would never have been written without the support of the people who have enriched me through their wisdom, friendship and love.

I would like to express my deepest appreciation to my supervisors: Professor Dikai Liu and Associate Professor Shoudong Huang. I really appreciate their guidance, encouragement and constructive advice on modelling and algorithms. They have provided me with invaluable help throughout the course of my research.

I would like to thank my Patrick project mentors: Dr Bradley Skinner and Dr Haye Lau for all the hours of collaboration, insightful ideas and research inspiration.

I would like to thank all the other team members from UTS and PTS: Professor Gamini Dissanayake, Dr Binghuang Cai, Daniel Pagac, Timothy Pratley and Andrew Bott for their helpful insight along with their patience and passion for autonomous straddle carriers.

Last, but not the least, I am deeply indebted to my parents, Xijia and Shuhua, for the years of education, encouragement and care; my beloved wife Momo, for her unwavering support; and my dearest son Shengbo for imbuing my life with wonderful expectations.

To the people who helped

List of Abbreviations

ASC:	Autonomous Straddle Carrier
QC:	Quay Crane
TK:	Truck
TC:	Transfer Crane
TIA:	Truck Import Area
B2Y:	Buffer-to-Yard
Y2B:	Yard-to-Buffer
T2Y:	Truck-to-Yard
Y2T:	Yard-to-Truck
Y2Y:	Yard-to-Yard
VRP:	Vehicle Routing Problem
CVRP:	Capacitated Vehicle Routing Problem
PDP:	Pickup and Delivery Problem
QCSP:	Quay Crane Scheduling Problem
QCAP:	Quay Crane Allocation Problem
BAP:	Berth Allocation Problem

AGV:	Automated Guided Vehicle
ALV:	Automated Lifting Vehicle
ASCSP:	Autonomous Straddle Carrier Scheduling Problem
MTSP:	Multiple Travel Salesmen Problem
GA:	Genetic Algorithm
TCX:	Two-part Chromosome Crossover
ORX:	Ordered Crossover
PMX:	Partially Matched Crossover
CYX:	Cycle Crossover
ORX+A:	Combination of Ordered Crossover and Asexual Crossover
PMX+A:	Combination of Partially Matched Crossover and Asexual Crossover
CYX+A:	Combination of Cycle Crossover and Asexual Crossover

Nomenclature

Indices:

v	index of the ASC.
q	index of the QC.
g	index of the TK.
j	index of the job.
n	index of the positional node which indicates a position on the map.
l	index of the link between positional nodes on the map.
s	index of the schedule. Each schedule contains a list of jobs which need to be finished in order, and each schedule can be assigned to any ASC, but one schedule cannot be shared by multiple ASCs. It is not a complete solution, but it can be a possible sub-solution for some jobs.

Sets:

V	set of all ASCs, $ V $ is the total number of ASCs in the fleet.
Q	set of all QCs, $ Q $ is the total number of QCs.
G	set of all TKs, $ G $ is the total number of TKs.
J	set of all jobs, $ J $ is the total number of jobs.
J^H	set of jobs with high priority for scheduling and these jobs usually are time critical or considered as urgent tasks and $J^H \subset J$.
S	set of all schedules.
J^Y	set of yard-to-yard jobs.
δ_v	planned trajectory of ASC (v) including the travelling nodes and associated times. Please refer to Section 3.4.1 for more detail.
N	set of all positional nodes, $ N $ is the total number of nodes in the map.
L	set of all links, $ L $ is the total number of links on the map.

Vectors:

J_s	vector of jobs in schedule s . That is, a list of jobs that will be done by an ASC in the defined order.
J_q^D	vector of jobs associated with discharging QC q .
J_q^U	vector of jobs associated with uploading QC q .
J_g^E	vector of jobs associated with exporting TK g .
J_g^I	vector of jobs associated with importing TK g .
J_v^G	vector of potential jobs grouped by (v).
$N^{lock}(n)$	vector of locked nodes associated with the node (n) of an ASC to prevent all vehicle collisions.

Parameters:

m_q	the total number of discharging containers for QC q .
n_q	the total number of uploading containers for QC q .
e_g	the total number of containers for exporting TK g .
r_g	the total number of containers for importing TK g .
p_v^I	the initial position of ASC v .
α_{ab}	parameter for defining pick-up job sequence. $\alpha_{ab} = \{0,1\} : \forall(a,b)$. Let $\alpha_{ab} = 1$ if and only if job a needs to be picked up before job b is picked up, else $\alpha_{ab} = 0$.
β_{ab}	parameter for defining set-down job sequence. $\beta_{ab} = \{0,1\} : \forall(a,b)$. Let $\beta_{ab} = 1$ if and only if job a needs to be set-down before job b is set-down, else $\beta_{ab} = 0$.
u_j	the pick-up position of job j . The position is generated by the yard management system.
d_j	the set-down position of job j . The position is generated by the yard management system.

y_{js}	a binary parameter. $y_{js} = 1$ if and only if the candidate schedule s has job j , otherwise $y_{js} = 0$.
t_0	plan starting time.
Δt_{pickup}	time required for an ASC to pick up a container.
$\Delta t_{setdown}$	time required for an ASC to set down a container.
Δt_{acc}	time associated with acceleration of an ASC moving from zero velocity to maximum (constant) velocity.
Δt_{dec}	time associated with deceleration of an ASC from its maximum (constant) velocity to zero velocity.
Δt^{QC}	the turnaround time of each QC unloading/uploading a single container.
Δt^{TK}	the turnaround time of an ASC performing TK importing/exporting for a single container within the TK area.
t_q^D	the starting time of QC q operations (discharging/unloading) which is predefined a priori as part of the QC schedule. For each discharging QC we assume that no container from QC q can be picked up by an ASC at the buffer before the time $(t_q^D + \Delta t^{QC})$.
t_q^U	the starting time of QC q operations (uploading) which is predefined a priori as part of the QC schedule. For each uploading QC we assume that no container can be set-down at buffer by an ASC before the time $(t_q^U + \Delta t^{QC})$.
t_g^E	the starting time of TK g operations (exporting) which is predefined a priori as part of the TK schedule. For each exporting TK we assume that no container from TK g can be picked up by an ASC at the TK gate before the time $(t_g^E + \Delta t^{TK})$.
t_g^I	the starting time of TK g operations (importing) which is predefined a priori as part of the TK schedule. For each importing TK we assume that no container from TK g can be set-down by an ASC at the TK gate before the time $(t_g^I + \Delta t^{TK})$.
w_{oz}	the minimum theoretical travel time between node o and node z . A look-up table built based on Dijkstra's algorithm is used for all positional nodes.

σ_j^R	the predefined parameter which specifies the container door alignment at the setdown node (d_j) for job (j). Let $\sigma_j^R=1$ if the container door is aligned, otherwise $\sigma_j^R=0$. Here, $\sigma_j^R \in \{0,1\}$, depending on the actual job type.
σ_j^S	the predefined parameter which specifies the container door's initial alignment at the pickup node (u_j) for job (j). Let $\sigma_j^S=1$ if the container door is aligned, otherwise $\sigma_j^S=0$. Here, $\sigma_j^S \in \{0,1\}$, depending on the actual job type.
f_{ij}	a flip flag of each physical link which relates the container alignment during traversal of the link. Let $f_{ij}=1$ if container alignment is changed when an ASC carrying a container from node ($i \in N$) to node ($j \in N$), otherwise $f_{ij}=0$. The flip flag of each link is encoded within the map.
ψ_{abc}	Patrick yard predefined traversal information between ($a \rightarrow b \rightarrow c$). Let $\psi_{abc}=1$ if an ASC needs to decelerate and adjust direction and then accelerate for traversing from node a to node c via node b , otherwise, $\psi_{abc}=0$. This information is encoded within the map.
ε_j	the minimum theoretical processing time based on Dijkstra's algorithm (from pickup to setdown) for job (j).

Decision variables:

X_{vs}	binary decision variable. $X_{vs}=1$ if and only if the candidate schedule s is selected for ASC v , otherwise $X_{vs}=0$.
T_j^S	the planned start time of job j .
T_j^F	the planned finish time of job j .
T_q^{DS}	the planned time to pick-up the final container for discharging QC q .
T_q^{UF}	the planned time to set-down the final container for uploading QC q .
T_g^{ES}	the planned time to pick-up the final container for exporting TK g .
T_g^{IF}	the planned time to set-down the final container for importing TK g .

Dependent variables

$P_v(t)$	position of ASC (v) at time ($t \geq t_0$), according to the trajectory planning.
σ_j^F	dependent variable which is the final box alignment associated with the trajectory planning of job (j), and $\sigma_j^F \in \{0,1\}$.

Table of Contents

CERTIFICATE OF AUTHORSHIP/ORIGINALITY.....	2
Abstract	3
Acknowledgements.....	5
List of Abbreviations.....	6
Nomenclature.....	8
Table of Contents	13
List of Figures.....	16
List of Tables	18
1. Introduction.....	20
1.1 Background.....	21
1.2 Motivation	22
1.3 Aims	24
1.4 Contributions	25
1.5 Publications	25
1.6 Thesis outline.....	27
2. Literature Review	30
2.1 Container Handling in Transshipment.....	30
2.1.1 Quay-side Container Handling Schemes	30
2.1.2 Yard Container Handling Schemes.....	32
2.1.3 Land-Side Container Handling Schemes	34
2.1.4 Integrated Container Handling Schemes.....	35
2.1.5 Scheduling of Automated Vehicles	37
2.2 Relevant Problem Models for Scheduling	38
2.2.1 The Multiple Travelling Salesman Problem	38
2.2.2 Pick-up and Delivery Problem	40
2.2.3 Capacitated Vehicle Routing Problem.....	42
2.2.4 Discussion	43
2.3 Optimisation with Genetic Algorithms	44
2.4 Summary	47
3. Mathematical Modelling for Automated Container Transfers.....	49
3.1 Introduction	49
3.2 Overview of the Patrick AutoStrad Container Terminal.....	49
3.3 Job Definition	51

3.3.1	Quayside Transfers	51
3.3.2	Yard Transfers	53
3.3.3	Land-side Transfers	54
3.4	Model 1: Comprehensive Model	55
3.4.1	Description of Container Alignment and ASC Trajectory	55
3.4.2	Objective Function.....	57
3.4.3	Constraints.....	59
3.5	Model 2: Job Scheduling Model.....	65
3.6	Matlab Seaport Simulation	67
3.6.1	Matlab Seaport Simulator.....	67
3.6.2	Replanning in the Seaport Simulator	68
3.6.3	Model Validation within the Seaport Simulator.....	69
3.7	Summary	70
4.	Job Grouping Approach for Planning Container Transfers	72
4.1	Motivation	72
4.2	Job Grouping Approach	73
4.2.1	Guiding Function of Job Grouping.....	75
4.2.2	Grouping Algorithm.....	77
4.3	Sequential Job Planning Approach.....	79
4.4	Experimental Results	80
4.5	Summary	85
5.	Modified Genetic Algorithm for Scheduling Optimisation	86
5.1	The Modified GA Approach	86
5.1.1	Two-part Chromosome Representation	87
5.1.2	Chromosome Validation and Repair Operation	88
5.1.3	Fitness Calculation Strategy	89
5.1.4	Selection and Crossover.....	93
5.1.5	Mutation	94
5.1.6	Replacement	94
5.2	Experimental Results	94
5.2.1	Simulation Testings	95
5.2.2	Live Testing Results at Patrick AutoStrad Terminal	99
5.3	Summary	100
6.	A New GA Crossover Approach for Further Improvement.....	102
6.1	Overview and Analysis of the Existing Strategy	102

6.1.1	The Two-part Chromosome Encoding Technique	102
6.1.2	Existing Crossover Method for Two-part Chromosomes	103
6.1.3	Limitations of the Existing Method	104
6.2	A New Crossover Approach.....	106
6.3	Experiments for Job Scheduling Optimisation.....	110
6.4	Computational Testing Methodology for the MTSP	113
6.5	Computational Results for the MTSP	115
6.5.1	Experiments for Minimising Total Travel Distance	116
6.5.2	Experiments for Minimising Longest Tour.....	118
6.5.3	The Robustness of the TCX	120
6.5.4	Optimality.....	122
6.5.5	Examination of TCX Operator with an Additional Dataset.....	123
6.6	Summary	124
7.	Conclusions and Future Research	126
7.1	Summary	126
7.1.1	Two Optimisation Models of Container Transfers by ASCs	126
7.1.2	A Job Grouping Approach for the Comprehensive Model	126
7.1.3	A Modified Genetic Algorithm for Optimising Job Scheduling	127
7.1.4	A Novel Crossover Method.....	127
7.2	Future Work.....	128
Appendix: Methodology of <i>t</i>-test		130
Bibliography		132

List of Figures

Figure 1-1: Satellite view of the Patrick AutoStrad container terminal within the Port of Brisbane at Fisherman Islands Australia [Google Earth].	20
Figure 1-2: ASC transporting a 40-foot container from the quay side as part of a buffer-to-yard task.	21
Figure 1-3: The structure of the thesis	27
Figure 2-1: Typical QCs employed at the container Terminal in Brisbane, Australia.	31
Figure 2-2: ASC servicing a TK in the TIA at the Patrick AutoStrad container terminal located in Brisbane, Australia.	34
Figure 3-1: Schematic diagram of the static seaport environment showing berth, QCs, bays, special nodes (QC and TK gates), TIA, and nodes in the yard.	50
Figure 3-2: Quay Side Operations. A trajectory for the QC jib can be calculated from the associated nodes and links between the ship and wharf. (TLR refers to twist lock release.)	52
Figure 3-3: TK Area Operations showing associated nodes and links between the yard and TK area. TKs can import or export any combination of 20-ft and 40-ft containers (up to a total of 4TEU (Twenty-foot Equivalent Unit) in each direction) into the seaport, as shown by the different combinations.	54
Figure 3-4: Container orientations (a,b) and flip movements (c).	56
Figure 3-5: Typical event timings and timeline for an ASC performing a job.	57
Figure 3-6: A planned path from pickup node to setdown node consisting of 56 nodes. A 3-point turn occurs at the 16 th node (nodes are not uniformly spaced in the map).	64
Figure 3-7: Example of a motion profile, which is applied to a planning path for an ASC (nodes are not uniformly spaced in the map).	64
Figure 3-8: Classification of a 3-point turn (case 1) and regular turning motion (case 2) for motion planning.	65
Figure 3-9: Flow diagramming for sequential job allocation and job injection approach based on a greedy nearest-vehicle-first heuristic.	68
Figure 3-10: Flow diagramming which illustrates the replanning processes within the seaport simulator.	69
Figure 3-11: Model Validation Method – Hierarchical assertion checking within the Matlab Seaport Simulator ensures model constraints are not violated.	70
Figure 4-1: An example of job grouping.	74

Figure 4-2: Plot of Eq(4.2) for job starting times (sec).....	76
Figure 4-3: Plot of Eq(4.2) for job finishing times (sec).....	76
Figure 4-4: Flowchart of job grouping method	77
Figure 4-5: Flowchart of sequential job allocation method	80
Figure 4-6: Performance comparison for the high QC load scenario.	82
Figure 4-7: Performance comparison for the medium QC load scenario.	82
Figure 4-8: Performance comparison for the low QC load scenario.	83
Figure 5-1: Example of two-part chromosome representation for a 9-job schedule with 3 ASCs.....	88
Figure 5-2: An example of calculating ASC and QC waiting times and updating related timings.....	91
Figure 5-3: Ordered crossover (ORX) method for first part chromosome	93
Figure 5-4: Single point asexual crossover for second part chromosome.....	93
Figure 5-5: Comparison on scheduling for 24 mixed jobs with 8 ASCs	97
Figure 5-6: Comparison on scheduling for 80 mixed jobs with 20 ASCs	98
Figure 6-1: Example of using the ORX+A crossover method (Carter and Ragsdale, 2006) for two-part chromosomes.....	103
Figure 6-2: The modified GA flowchart for solving the job scheduling problem.....	110
Figure 6-3: Comparison on scheduling for 24 mixed jobs with 8 ASCs	111
Figure 6-4: Comparison on scheduling for 80 mixed jobs with 20 ASCs	112

List of Tables

Table 4-1: Parametric Configuration	81
Table 4-2: Number of Jobs for Three Different Scenarios.....	81
Table 4-3: Comparison of the Job Grouping Algorithm VS Sequential Job Allocation Algorithm	83
Table 4-4: Performance of the Job Grouping Algorithm Relative to the Sequential Job Allocation Algorithm	84
Table 5-1: Parametric configuration for scheduling 24 mixed jobs	95
Table 5-2: Parametric configurations for scheduling 80 mixed jobs	97
Table 5-3: Comparison of GA vs Sequential Job Scheduling Algorithm for Total Cost	98
Table 5-4: GA settings	99
Table 5-5: Various testing results from Patrick AutoStrad scheduling system.....	100
Table 6-1: The number of fundamental combinations in the second part of the chromosome for each job scheduling problem (number of jobs_number of ASCs) showing the exponentially increasing number of combinations with increasing m	106
Table 6-2: Limited search space of the existing method for the second part of the chromosome.....	106
Table 6-3: Comparison of ORX GA vs TCX GA for Total Cost.....	113
Table 6-4: Computational test conditions	114
Table 6-5: Parametric configuration for GA	114
Table 6-6: Experimental results for minimising total travel distance.....	116
Table 6-7: t -test results of TCX versus (ORX, CYX and PMX) crossover approaches for minimising total travel distance.	117
Table 6-8: Comparison of the mean results between two approaches using GA with seeding enabled for minimising total travel distance.....	117
Table 6-9: Experimental results for minimising longest tour.....	118
Table 6-10: t -test results of TCX versus (ORX, CYX and PMX) crossover approaches for minimising longest tour.	119

Table 6-11: Comparison of crossovers with GA seeding for minimising longest tour.	119
Table 6-12: TCX Robustness to varying GA parameters for minimising total travel distance.....	120
Table 6-13: TCX Robustness to varying GA parameters for minimising longest tour	122
Table 6-14: Optimality Gap for small-sized symmetric MTSP problems for minimising total travel distance	123
Table 6-15: Optimality Gap for small-sized symmetric MTSP problems for minimising longest tour	123
Table 6-16: Comparison of crossover methods for the sgb128 MTSP test problem for minimising the total travel distance	124
Table 6-17: Comparison of crossover methods for the sgb128 MTSP test problem for minimising longest tour.....	124