



FINAL PROJECT – TI 184833

**DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING WITH
COLUMN GENERATION APPROACH MODEL FOR INVENTORY SHIP
ROUTING PROBLEM**

RAFLY DWI PRIYONO
NRP. 02411640000057

SUPERVISOR:
Nurhadi Siswanto, S.T., M.S.i.E., Ph.D
NIP. 197005231996011001

DEPARTMENT OF INDUSTRIAL SYSTEM AND ENGINEERING
FACULTY OF INDUSTRIAL TECHNOLOGY AND SYSTEMS ENGINEERING
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2020



FINAL PROJECT - TI184833

**DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING
WITH COLUMN GENERATION APPROACH MODEL FOR
INVENTORY SHIP ROUTING PROBLEM**

RAFLY DWI PRIYONO

NRP 02411640000057

Supervisor

Nurhadi Siswanto, S.T., M.S.i.E., Ph.D

NIP. 197005231996011001

DEPARTMENT OF INDUSTRIAL SYSTEM AND ENGINEERING

Faculty of Industrial Technology and Systems Engineering

Institut Teknologi Sepuluh Nopember

Surabaya

2020

This page was intentionally left blank



TUGAS AKHIR - TI184833

**PENGEMBANGAN MODEL *MIXED INTEGER LINEAR PROGRAMMING*
DENGAN PENDEKATAN GENERASI KOLOM UNTUK *INVENTORY*
*SHIP ROUTING PROBLEM***

RAFLY DWI PRIYONO

NRP 02411640000057

Dosen Pembimbing

Nurhadi Siswanto, S.T., M.S.i.E., Ph.D

NIP. 197005231996011001

DEPARTEMEN TEKNIK SISTEM DAN INDUSTRI

Fakultas Teknologi Industri Dan Rekayasa Sistem

Institut Teknologi Sepuluh Nopember

Surabaya

2020

This page was intentionally left blank

APPROVAL SHEET

DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING WITH COLUMN GENERATION APPROACH MODEL FOR INVENTORY SHIP ROUTING PROBLEM FINAL PROJECT


Submitted to Acquire the Requirement of Bachelor Degree at
Undergraduate Program Industrial and System Engineering Department
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya

Author:

RAFLY DWI PRIYONO

NRP 02411640000057

Acknowledge and Approved by,
Final Project Supervisor


Nurhadi Siswanto, S.T., M.S.i.E., Ph.D
NID. 197005231996011001

SURABAYA, AUGUST 2020



This page was intentionally left blank

DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING WITH COLUMN GENERATION APPROACH MODEL FOR INVENTORY SHIP ROUTING PROBLEM

Student Name : Rafly Dwi Priyono
ID Number : 02411640000057
Department : Industrial and System Engineering ITS
Supervisor : Nurhadi Siswanto, M.S.i.E., Ph.D

ABSTRACT

Maritime transport remains the backbone of globalized trade and the manufacturing supply chain and cement industry is one of many industries that rely on maritime transportation to move their raw material or semi-finished product. A large number of ships are required to deliver products between the islands and often times the number of ships exceeds the number of existing packaging plants which causing inefficiencies in ship routing and scheduling. This problem is categorized as Inventory Ship Routing Problem (ISRP) and its complexity increases exponentially with the number of ships and ports especially when using Mixed Integer Linear Programming (MILP). There were approaches that can handle a large number of variables named Column Generation (CG) and had shown the ability to save computation time and memory. The problem researched in this final project is to develop a mathematical model of MILP with CG approach that able to handle ISRP. In this study, a mathematical model for ISRP using MILP with CG approach was successfully developed but the algorithm has not been able to prove its robustness and not yet validated. As an alternative, several pieces of researches about ISRP that evaluate the variation of dataset size were compared. MILP with CG can handle up to 5 ships, 5 ports, and 60 days planning horizon with ease and more than 5 ports if the duration of the planning horizon is reduced. Hence, it's recommended to shorten the planning horizon when dealing with large datasets. To overcome the challenges with the algorithm, it's advised to utilize the built-in function of LINGO 18 called branch-and-price (BNP) solver.

Keyword: Inventory Ship Routing Problem, Mixed Integer Linear Programming, Column Generation

This page was intentionally left blank

**PENGEMBANGAN MODEL *MIXED INTEGER LINEAR*
PROGRAMMING DENGAN PENDEKATAN GENERASI
KOLOM UNTUK *INVENTORY SHIP ROUTING PROBLEM***

Nama : Rafly Dwi Priyono
NRP : 02411640000057
Departemen : Teknik dan Sistem Industri ITS
Dosen Pembimbing : Nurhadi Siswanto, M.S.i.E., Ph.D

ABSTRAK

Transportasi laut tetap menjadi tulang punggung perdagangan global dan rantai pasokan manufaktur serta industri semen merupakan salah satu dari banyak industri yang mengandalkan transportasi laut untuk memindahkan bahan mentah atau produk setengah jadi. Jumlah kapal yang besar dibutuhkan untuk mengirimkan produk antar pulau dan seringkali jumlah kapal melebihi jumlah pabrik pengemasan yang ada yang menyebabkan inefisiensi dalam pengarahannya dan penjadwalan kapal. Masalah ini dikategorikan sebagai *Inventory Ship Routing Problem* (ISRP) dan kompleksitasnya meningkat secara eksponensial dengan jumlah kapal dan pelabuhan terutama bila menggunakan *Mixed Integer Linear Programming* (MILP). Ada pendekatan yang dapat menangani masalah dengan jumlah variabel yang banyak bernama *Column Generation* (CG) dan telah terbukti dapat menghemat waktu komputasi dan memori. Masalah yang diteliti dalam tugas akhir ini adalah mengembangkan model matematis MILP dengan pendekatan CG yang mampu menangani ISRP. Pada penelitian ini telah berhasil dikembangkan model matematika MILP dengan pendekatan CG namun algoritma tersebut belum terbukti ampuh dan belum tervalidasi. Sebagai alternatif, beberapa penelitian tentang ISRP yang mengevaluasi variasi ukuran *dataset* dibandingkan. MILP dengan CG dapat menangani hingga 5 kapal, 5 pelabuhan, dan jangka waktu perencanaan 60 hari dengan mudah dan lebih dari 5 pelabuhan jika durasi jangka waktu perencanaan dikurangi. Oleh karena itu, disarankan untuk mempersingkat jangka waktu perencanaan saat menangani *dataset* yang besar. Untuk mengatasi tantangan dengan algoritma, disarankan untuk memanfaatkan fungsi bawaan LINGO 18 yang disebut *branch-and-price* (BNP) solver.

Kata kunci: *Inventory Ship Routing Problem, Mixed Integer Linear Programming, Column Generation*

This page was intentionally left blank

ACKNOWLEDGEMENT

Praise be to God Almighty because, by His will, the author can complete the final research with the title "Development Of Mixed Integer Linear Programming With Column Generation Approach Model For Inventory Ship Routing Problem" properly. This Final Project research is structured as one of the requirements for completing the Strata-1 study program in Industrial and Systems Engineering, Faculty of Industrial Technology and Systems Engineering, Sepuluh Nopember Institute of Technology, Surabaya. On this occasion the author would like to express his deepest gratitude to several parties as follows:

1. Mr. Nurhadi Siswanto, S.T., MSIE., Ph.D. as a supervisor who always provides direction, knowledge, input, and motivation to the author so that the author can complete the final project properly.
2. Prof. Dr. Ir. Budisantoso Wirjodirdjo, M.E., Prof. Ir. Budi Santosa, MS., Ph.D., Dr. Maria Anityasari, S.T., M.E., Stefanus Eko Wiratno, ST., MT., and Dewanti Anggrahini, ST., MT. as a lecturer examiner for the proposal seminar and final assignment session, for all the suggestions and input given.
3. All lecturers for the knowledge that has been given and the secretariat staff of the System and Industrial Engineering ITS, for the services provided for the last 4 years.
4. My parents who always provide prayer, support, and motivation are countless.
5. M. Afif, Risang Arkaan, Fadhila Putra, and Harnum Yulia for the prayers and moral support given during the process of this final project.
6. Der Administrator of the Laboratory of Quantitative Modeling and Industrial Analysis Policy (Mba Firda, Danis, Stanley, Asta, Nada, Betalia, Revino, Roderick, Satrio, and other friends) for their support.
7. All ADHIGANA - TI 32 friends and other parties that cannot be named one by one who has provided assistance, input, and support to the author.

Surabaya, August 2020

Rafly Dwi Priyono

This page was intentionally left blank

TABLE OF CONTENTS

APPROVAL SHEET	i
ABSTRACT	iii
ABSTRAK	v
ACKNOWLEDGEMENT	vii
TABLE OF CONTENTS	ix
LIST OF FIGURES	xi
LIST OF TABLES	xiii
CHAPTER I INTRODUCTION	1
1.1. Background	1
1.2. Problem Formulation	4
1.3. Objectives	4
1.4. Benefit	5
1.5. Scope of Research	5
1.6. Writing Methodology	6
CHAPTER II LITERATURE REVIEW	9
2.1. Maritime Transportation	9
2.2. Vendor Managed Inventory (VMI)	10
2.3. Inventory Routing Problem (IRP)	11
2.4. Single Depot & Multi Depot ISRP	12
2.5. Column Generation	18
2.6. Research Gap & Positions	21
CHAPTER III RESEARCH METHODOLOGY	23
3.1. Research Methodology	23
3.1.1 Literature Review	23
3.1.2 Model Development	24
3.1.3 Algorithm Development	26
3.1.4 Column Generation Schemes	28
3.1.5 Conclusion and Suggestion	28
CHAPTER IV DEVELOPMENT OF MODEL AND ALGORITHM	29
4.1 Problem Definition	29
4.2 Model Development	31
4.2.1 Reformulate Arc-Flow ISRP Mathematical Model	31

4.2.2	Dantzig–Wolfe Decomposition	34
4.2.3	Master Problem.....	37
4.2.4	Column Generation.....	39
4.2.5	Port Subproblem	40
4.2.6	Ship Subproblem	41
4.3	Algorithm Development	42
4.3.1	Initialize Restricted Master Problem (RMP) and Subproblems	43
4.3.2	Solve the LP-Relaxation of RMP and the Subproblems	45
4.3.3	Check for negative reduced cost in Subproblems solutions	46
4.3.4	Adding new column to the RMP	47
4.3.5	Re-optimize the RMP	48
4.3.6	Obtain optimal solution for Master Problem	49
CHAPTER V COLUMN GENERATION SCHEMES.....		53
5.1	Column Generation on Large Dataset ISRP	53
5.2	Dual Value Analysis and Column Generation Challenges	55
5.3	Intensification and Stabilization	56
5.4	Diversification.....	58
CHAPTER VI CONCLUSION & SUGGESTION.....		61
6.1	Conclusion	61
6.2	Suggestion.....	61
REFERENCES		63
ATTACHMENT.....		67

LIST OF FIGURES

Figure 1. 1 Cement Sales in Indonesia 2011 - 2018	2
Figure 2. 1 General Bulk carrier	10
Figure 2. 2 Example of Inventory Routing Problem Solution	11
Figure 2. 3 Algorithm for Column Generation Approach.....	20
Figure 3. 1 Research Methodology	23
Figure 3. 2 ISRP Model Development	24
Figure 3. 3 ISRP Algorithm Development.....	26
Figure 5. 1 Algorithm Of Diversification By Resolution	59
Figure 5. 2 Algorithm Of Diversification By Selection.....	59

This page was intentionally left blank

LIST OF TABLES

Table 2. 1 ISRP Research Literature.....	22
Table 4. 1 Algorithm Column Generation for ISRP	43
Table 5.1 Time Comparison between MILP and Column Generation.....	54

This page was intentionally left blank

CHAPTER I

INTRODUCTION

This chapter is an introductory part of the research that consists of explanations about the background, problem formulation, objectives, benefits, scope of research, and writing methodology. The comprehensive explanations will be presented through several sub-chapters.

1.1. Background

Maritime transport remains the backbone of globalized trade and the manufacturing supply chain, as more than 80 percent of world merchandise trade by volume is carried by sea, and the percentage is even higher for most developing countries (UNCTAD, 2019). In industrial shipping, the company has to ensure the availability of goods at the point of demand by minimizing costs distribution which consists mainly of operational costs of the ship. To minimize these costs, route selection and scheduling of ships become very important. This problem is then called the vehicle routing problem (VRP) or in sea transportation better known as Ship Routing Problem (SRP). Another variant of VRP is inventory routing problem (IRP). In contrast to VRP where the destination can only be visited once, in IRP the destination can be visited more than once (Siswanto et.al, 2011).

Inventory routing problems began to be known since the emergence of the vendor concept managed inventory (VMI) in the supply chain. The role of VMI is believed to have a positive impact both for suppliers and customers. For the supplier side, they can integrate their distribution and on the other hand customers can cut costs in their inventory management (Sui et al, 2010). The purpose of IRP is to find a route that can minimize total costs (distribution costs and inventory costs) by maintaining inventory at the customer's warehouse at a certain level or no stock out (Hwang, 2007). IRP in maritime transportation is better known as inventory ship routing problem (ISRP).

From this ISRP then another problem develops related to the number of depots or supply ports, which are then referred to as ISRP-single depot and ISRP-

multi depot. Another problem variation regarding on how many products being transported, in this case there are ISRP for single product and multi-product. ISRP for single product (s-ISRP) was discussed by Christiansen, M., & Fagerholt, K. (2007) while for the case of multi-product (m-ISRP) has been discussed by Al-Khayyal & Hwang (2007), and Siswanto et.al (2011). In multi-product shipments new problems arise, namely product compatibility issues. Therefore, ISRP were further divided into two types because of the product compatibility issue, namely dedicated compartment and undedicated compartment.

As one of many industries that rely on maritime transportation to move their raw material or semi-finished product, cement industry in Indonesia has been playing an important role of Indonesia's economic growth and infrastructure development for a long time. Based on statistics of the Indonesian Cement Association (ASI), the growth of cement consumption in Indonesia showed positive upward trend, with average growth of 7.5% per year in the 2011-2018 period.

TAHUN YEAR	PENJUALAN SEMEN DOMESTIK (JUTA TON) DOMESTIC CEMENT SALES (MILLION TON)	PERTUMBUHAN (%) GROWTH (%)
2011	48	20.0
2012	55	14.6
2013	58	5.6
2014	60	2.56
2015	62	2.5
2016	62	0.0
2017	66	7.6
2018	69	4.9

Figure 1. 1 Cement Sales in Indonesia 2011 - 2018
Source: ASI (Indonesian Cement Association)

Take example for a cement industry player, PT Semen Tonasa is the largest cement industry in Eastern Indonesia (KTI) with the main market territory in Eastern Indonesia region. PT Semen Tonasa has five plant located in Biringere Village, Bungoro Subdistrict, Pangkep Regency, South Sulawesi Province. Currently PT Semen Tonasa has one supply port or production plant and several demand or packaging plant (10 packaging plant) covering eastern Indonesia. PT Semen Tonasa using sea transport and utilize several numbers of ships (16 bulk

cement carrier) to deliver their semi-finished product to the packaging plant. Total domestic cement distribution in regions I, II and III in 2018 reached 5.2 million tons. With that condition, PT Semen Tonasa most likely to have ISRP problem (and it may be a large scale one) with single depot and multi-product.

The number of ships used to deliver the bulk cement exceeds the number of existing packaging plant which may resulted in low utilization rate of the ships and also causing inefficiencies in ship routing and scheduling. According to Indonesia National Shipowners Association (INSA), a utilization rate of the offshore shipping armada in 2018 is about 50% which only experienced thin growth from previous year. In addition, there is an imbalance between the ship's carrying capacity and the demands of the packing factory. As a result, there is an excess carrying capacity of the ship in one assignment route but lack of carry capacity in the other. This practice might result in high operational or distribution costs if the ship routing and scheduling is done carelessly. Because of that, the optimization of ship routing and scheduling become important or even necessary.

Many researcher have sought the solution of ISRP using exact method as done by Christiansen (2005), Ramkumar (2010) and Siswanto et.al (2011) using Mixed Integer Linear Programming model. As shown in Siswanto et.al (2011), the complexity of ISRP increases exponentially with the number of ships and ports especially when using mixed integer linear programming. Considering the excessive computational time in solving a large-scale problem, Siswanto et.al (2011) also present a set of heuristics. The heuristics results show that 67% of problem instances have the same result with the MILP approach with a 1.96% on average gap. The heuristics shows promising result to be an alternative from MILP approach but unfortunately didn't solve the problem with excessive computational time when dealing with large scale dataset.

There were approaches that can handle large number of variables and might be useful to solve large scale dataset named Column Generation algorithm. Column generation algorithms are most useful when dealing with large numbers of variables (Desaulniers, 2005). The Column generation technique was born from the need of improving the performance of the simplex method. Indeed, the purpose of CG was to avoid the full enumeration of columns when solving a

formulation because the number of variables was too large to be dealt with explicitly. Column Generation helps to save computation time and memory. Column generation had been developed recently by some researcher like Dantzig (1960), Christiansen (2005) and Nemhauser (2010) to tackle this large scale ISRP. Column generation methods are currently essential tools for solving many classes of integer programming and combinatorial optimization problems.

Rusdianto (2019) has tried to solve the ship routing and scheduling problem of PT Semen Tonasa using MILP. Because of the large dataset, Rusdianto (2019) only cover cement distribution for Kalimantan Island (reduced number of consumption ports) and successfully solve the problem. Unfortunately, the MILP only able to handle one route, one product and one compartment with reasonable average computing time, beyond that the time needed to solve larger dataset is quite long (up to a dozen hours). Because of that, this research tries to develop mathematical model of MILP using Column Generation approach to handle ISRP with the same characteristics of PT Semen Tonasa case. Mixed Integer Linear Programming (MILP) with Column Generation chosen because its effectiveness when dealing problem with large numbers of variables (Desaulniers, 2005). The MILP model is used to determine the type and the quantity of products to be loaded, the ship routing and delivery schedules, and the type and the quantity of products to be unloaded at the destinations ports simultaneously.

1.2. Problem Formulation

The problem researched in this final project is developing mathematical model of MILP with Column Generation approach that able to handle Inventory Ship Routing Problem.

1.3. Objectives

The objectives of this research are:

1. Developing mathematical model of Mixed Integer Linear Programming with Column Generation approach for Inventory Ship Routing Problem.

2. Applying mathematical model of Mixed Integer Linear Programming with Column Generation approach for Inventory Ship Routing Problem into LINGO optimization software.
3. Discovering the challenges and difficulties in developing and applying mathematical model of Mixed Integer Linear Programming with Column Generation approach for Inventory Ship Routing Problem.

1.4. Benefit

The results of this final project might be useful for further research about Inventory Ship Routing Problem using of Mixed Integer Linear Programming with Column Generation approach.

1.5. Scope of Research

The scope of research of this final project consists of boundaries and assumptions that are used as the approach of the presentation of the actual conditions.

1.5.1. Boundary

The limitation of this final project is presented as follows:

1. The research only focuses on the delivery of product from supply port to demand port.
2. The research does not consider delayed penalty or stock out penalty.
3. The research does not consider the effect of weather on the travel time of each ship.
4. The research does not consider the minimum inventory level in the ship before being allowed to sail.
5. The research does not consider the inventory cost and ship operating cost.

1.5.2. Assumptions

1. Cost components such as travel material costs, port setup and load-unloading cost are fixed throughout the planning horizon.
2. The ships are always available and ready to use.
3. The travel time between ports for all ship are the same and fixed.

4. The port setup time is the same for each port.
5. The production rates of all supply ports are exceeding the consumption rate of all demand ports.
6. The loading and unloading time for all port are fixed.

1.6. Writing Methodology

This final project is arranged in a scientific writing methodology which consists of six chapters. The content overviews of each chapter are presented as follows:

CHAPTER I INTRODUCTION

This chapter will cover up the introductory part of the research conducted in this final project. In this chapter there will be comprehensive explanation about the background, problem formulation, objective, benefit, the scope of research, and the writing methodology of this final project.

CHAPTER II LITERATURE REVIEW

Chapter II contain several literatures that underlie the research conducted in this final project. It consists of several related theories that are obtained from several relevant sources, such as: books, journals, company's annual report, government regulations, and scientific publications.

CHAPTER III RESEARCH METHODOLOGY

In the research methodology, the approach used as well as the structured systematic methods in conducting this research will be comprehensively presented and explained.

CHAPTER IV DEVELOPMENT OF MODELS AND ALGORITHMS

In this chapter there will be a comprehensive explanation of the problem description, model development, algorithm development and algorithm validation.

CHAPTER V COLUMN GENERATION SCHEMES

In this chapter several column generations schemes that can offer several possibilities for improvement will be explained.

CHAPTER VI CONCLUSION AND SUGGESTION

The final part of this final project is the conclusion of the results of the research and the suggestion given after conducting this research that might be beneficial and useful for several parties.

This page was intentionally left blank

CHAPTER II

LITERATURE REVIEW

This chapter contains comprehensive explanation about several theories underlying the research conducted. The theories are including the general information related to maritime transportation, vendor managed inventory, inventory routing problem, single and multiple depot ISRP and integer linear programming, column generation and research gap and position.

2.1. Maritime Transportation

Maritime transport remains the backbone of globalized trade and the manufacturing supply chain, as more than 80 percent of world merchandise trade by volume is carried by sea, and the percentage is even higher for most developing countries (UNCTAD, 2019). Furthermore, according to Hwang (2007) that 70% of the value of goods shipped worldwide is distributed by sea. Large product shipments and the lowest total shipping costs are the advantages of shipping goods by sea compared to other types of transportation. Compared to the cost of shipping goods by land shipping, the costs by sea only amounted about 1.15% - 2.6% of the costs by land based on dollar / ton-kilometer shipped (Sinha, 2019).

World merchandise trade consists of several cargo types i.e. Containers, Dry Cargo, Main Bulks and Tanker Trade that are being traded across continent. Thus, several types of water carriers are used and they can be classified as either barges, or bulk carriers (liquid or dry), or container ships, or special purpose carriers such as car carriers, Liquefied Natural Gas (LNG) carriers (Hwang, 2007). Bulk carriers consists of two types, one for transportation of bulk liquids such as crude oil, etc., and the other for transporting dry bulk such as coal, cement, etc. The example of bulk carriers is shown in figure 2.1 is a General bulk carrier that can transport several types of cargo at the same time, without restrictions on specific cargoes.

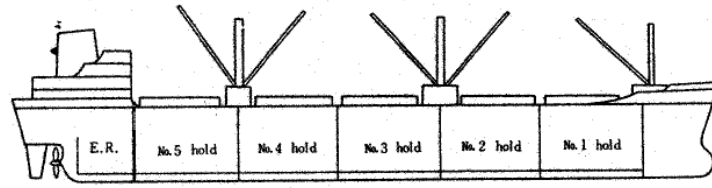


Figure 2. 1 General Bulk carrier (Hwang,2007)

Research on maritime transportation is mainly consists of three parts namely strategic, tactical, and operational. Christiansen et.al. (2007) classifies strategic problems related to maritime transportation including determining the port location, ship design, etc. For tactical issues consisting of ship scheduling, ship routing, etc. While the problems included in operational issues include determining the speed of shipping, loading activity on ships, etc.

2.2. Vendor Managed Inventory (VMI)

Integration and information transparency become the key of success of a company's supply chain nowadays. Therefore, many companies began to implement the concept of Vendor Managed Inventory (VMI), which is a concept in which inventory management policies in customers are managed by vendors. VMI was first popularized by Wall-Mart and Procter Gamble in the late 1980s in the retail industry. Nowadays, VMI was also being implemented by automotive, machinery services, chemicals, packaging, wood and furniture industries, raw materials distribution, etc (Vitgil, 2007).

VMI is often referred to as a win-win situation, where suppliers can save on production costs because they can plan production more accurately according to real time demand data of the customer and can cut distribution costs by coordinating the distribution of goods to several customers in one go, on the other hand customers can also make savings because there is no inventory cost because of excessive inventory. In this case the vendor or supplier must make three important decisions, which are when the customer must be served, how many products must be sent, and how to integrate the customer into a route (Sui et al, 2010).

2.3. Inventory Routing Problem (IRP)

With the emergence of the VMI concept, a new problem emerged called the inventory routing problem (IRP). IRP is a distribution problem in which each customer maintains a local inventory of a product. Some nodes consume a certain amount of product daily, and others produce a certain amount of product each day. The objective is to minimize delivery costs while attempting to ensure that no customer runs out of the commodity, and no producer has to stop production because of limited storage capacity (Hwang, 2007). IRP combines inventory management and vehicle routing planning together to get the optimal total cost. The illustration of IRP solution with one depot, two vehicle and two product are shown in figure 2.2. In maritime transportation, the IRP problem is called the inventory ship routing problem (ISRP) or maritime inventory routing problem (MIRP).

Day1: no delivery in day 1

Day2:

Day3:

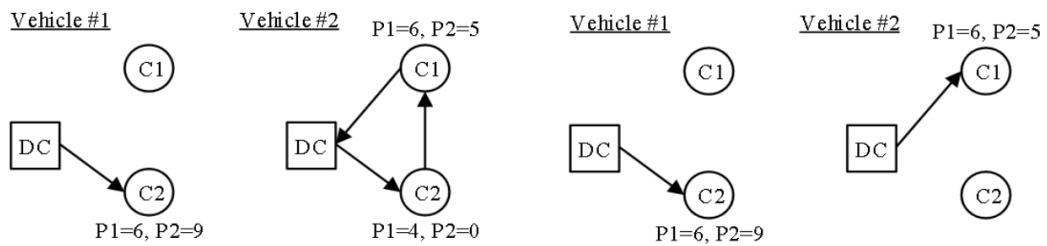


Figure 2. 2 Example of Inventory Routing Problem Solution (Yeh et.al, 2017)

As Bertazzi and Speranza, (2012) describe and structure the main characteristics of an IRP, there are five main characteristics such as the shipping times, planning horizon, the structure of the distribution policy, the objective of the policy and the decision space. The possible shipping times of an IRP can be: Continuous, Continuous with a minimum intershipment time and Discrete Shipments. While the planning horizon over which an optimal policy is looked at may be infinite or finite. The typical examples of structured policies often inspired by practical relevance can take form such as Zero Inventory Ordering (ZIO), Periodic, Frequency-based, Full load Shipments, Direct shipping, etc. The decision space applied in general in IRP can be classified into two namely

Decisions over time only and Decisions over time and space. Different specifications of these characteristics give rise to a different IRP.

Some developments from ISRP research include ISRP single and multi-product, ISRP single and multi-depot, ISRP with time windows, stochastic demand ISRP, homogeneous and heterogeneous fleet ISRP, and so on. One interesting variant of ISRP is related to the loading port of the product. Research on this issue has evolved from a single loading port or supply port (single depot-ISRP) to more from one port (multi-depot-ISRP).

2.4. Single Depot & Multi Depot ISRP

At the ISRP single depot, suppliers distribute goods from one port of loading to some unloading port. Goods distributed can consist of one or several types of goods. The ships used and warehouses at the unloading port have a certain capacity. The location of the unloading port is spread over several areas. Research related to single depot This ISRP was carried out by Christiansen (2009) and Harianto (2015).

In contrast to single depot, in multi-depot ISRP, the amount of loading port can be more than one. Often times a port can be both loading port and an unloading port at the same time. The inventory capacity and initial inventory of each port are different and are known at the beginning of the planning horizon. Each port also has different production and consumption speeds and is often assumed to be constant. With the information available, the objective is to find a route that minimizes transportation costs and ensures the availability of goods at each port at a certain level (Siswanto et.al, 2011).

Siswanto et.al (2011) developed a mathematical model for aforementioned problem with multiple depots, multiple products and undedicated compartment. The notations used in the research model include:

- Parameters:
- CT_{ijv} travelling cost of ship v from port i to port j
- CW_{iv} setup cost of ship v at port i
- CO_{ik} loading or unloading setup cost of product k at port i .

J_{ik} a constant that indicates that a port is a production or demand port. Which equal to +1 if port i is a producer of product k, and equal to -1 if a consumer.

TH planning horizon

TT_{ijv} travelling time of ship v from port i to port j

TW_i setup time at port i

TO_{ik} loading or unloading setup time of product k at port i

TQ_{ik} time needed to load or unload a unit of product k at port i

IS_{ik} initial inventory of product k at port i

SM_{ik} maximum level of product k at port i

SX_{ik} minimum level of product k at port i

R_{ik} production (or consumer) rate of product k at port i.

M_i maximum number visits of ships at port i

- Continuous Variables:

l_{imvkc} quantity of product k onboard loaded in compartment c of ship v after departing from node (i, m);

q_{imvkc} quantity of product k loaded (or unloaded) into (or from) compartment c of ship v at node (i, m). It is assumed that there is no loading or unloading activity at the artificial nodes; $q_{o(v)1vkc} = 0$

t_{im} arrival time of a ship at node (i, m).

S_{imk} level of product k at the time when a ship arrives at node (i, m).

- Binary Variables:

$X_{imjnv} \begin{cases} 1, & \text{if ship v routes from node (i,m) to node (j,n)} \\ 0, & \text{otherwise} \end{cases}$

$Y_{im} \begin{cases} 1, & \text{if node (i,m) is not visited} \\ 0, & \text{otherwise} \end{cases}$

$Z_{im} \begin{cases} 1, & \text{if ship v finishes its route at node (i,m)} \\ 0, & \text{otherwise} \end{cases}$

$O_{imvkc} \begin{cases} 1, & \text{if ship v visits node (i,m) and loads/unloads product k into/from compartment c} \\ 0, & \text{otherwise} \end{cases}$

$W_{imv} \begin{cases} 1, & \text{if ship v visits node (i,m) and performs port setup} \\ 0, & \text{otherwise} \end{cases}$

- Variable Set:

V	a set of ship with index v
N	a set that denotes as all possible positions of ships
N_p	a subset of ship positions at production ports
P_v	a set of products loaded in ship v
C_v	a set of compartment c in ship v
A_v	a set of feasible arcs for ship v
H	a set a set of physical ports
H_v	a subset of physical ports that can be visited by ship v
K_i	a set of product in port i
M_i	a set of arrival number in port i

Following is a mathematical model of the ISRP by Siswanto et.al, 2011 that being adopted in this study.

Minimize

$$\sum_{v \in V} \sum_{(i,m,j,n) \in Av} CT_{ijv} X_{imnjv} + \sum_{v \in V} \sum_{(i,m) \in N} CW_{iv} w_{imv} + \sum_{v \in V} \sum_{(i,m) \in N} \sum_{k \in P_v} \sum_{c \in C_v} CO_{ik} O_{imvkc} \quad (1)$$

Subject To:

$$\sum_{(j,n) \in N} X_{0(v)1jnv} + Z_{0(v)1v} = 1, \quad \forall v \in V \quad (2)$$

$$\sum_{(j,n) \in N \cup \{0(v),1\}} X_{jnimv} - \sum_{(j,n) \in N} X_{imjnv} + Z_{imv} = 0, \quad \forall (v,i,m) \in V \times N, i \neq j \quad (3)$$

$$\sum_{(i,m) \in N} Z_{imv} = 1, \quad \forall v \in V \quad (4)$$

$$\sum_{v \in V} \sum_{(j,n) \in N \cup \{0(v),1\}} X_{jnimv} + y_{im} = 1, \quad \forall (i,m) \in N, i \neq j \quad (5)$$

$$y_{im} - y_{i(m-1)} \geq 0, \quad \forall (i,m) \in N, m \neq 1 \quad (6)$$

$$x_{imjnv} \in \{0,1\}, \quad \forall v \in V, \forall (i,m,j,n) \in A_v \quad (7)$$

$$z_{imv} \in \{0,1\}, \quad \forall v \in V, \forall (i,m) \in N \cup \{0(v),1\} \quad (8)$$

$$y_{im} \in \{0,1\}, \quad \forall (i,m) \in N \cup \{0(v),1\} \quad (9)$$

$$x_{imjnv}(l_{imvkc} + J_{ik}q_{jnvkc} - l_{jnvkc}) = 0, \quad \forall v \in V, \forall (i,m) \in N, (j,n) \in N, (k,c) \in P_v \times C_v, i \neq j \quad (10)$$

$$x_{imjnv}(l_{imvkc}) = 0, \quad \forall v \in V, \forall (i,m) \in N \cup \{0(v),1\}, (j,n) \in N, (k,c) \in P_v \times C_v, i \neq j \quad (11)$$

$$QQ_{vkc} = l_{0(v)1vkc}, \quad \forall v \in V, \forall (k,c) \in P_v \times C_v \quad (12)$$

$$q_{imvkc} \leq \sum_{(j,n) \in N \cup \{0(v),1\}} CM_{vc} X_{jnimv}, \quad \forall (i,m) \in N, \forall v \in V, \forall (k,c) \in P_v \times C_v, i \neq j \quad (13)$$

$$l_{imvkc} \leq \sum_{(j,n) \in N \cup \{0(v),1\}} CM_{vc} X_{jnimv}, \quad \forall (i,m) \in N, \forall v \in V, \forall (k,c) \in P_v \times C_v, i \neq j \quad (14)$$

$$q_{imvkc} \leq CM_{vc} o_{imvkc}, \quad \forall (i,m) \in N \cup \{0(v),1\}, \forall v \in V, \forall (k,c) \in P_v \times C_v \quad (15)$$

$$\sum_{k \in P_v} \sum_{c \in C_v} o_{imvkc} \leq 100w_{imv}, \quad \forall (i,m) \in N \cup \{0(v),1\}, \forall v \in V \quad (16)$$

$$l_{imvkc} \leq CM_{vc}(1 - o_{imvkc}), \quad \forall (i,m) \in N_p, \forall v \in V, \forall (k',k'') \in P_v, \forall c \in C_v, k' \neq k'' \quad (17)$$

$$\sum_{k \in P_v} o_{imvkc} \leq 1, \quad \forall (i,m) \in N \cup \{0(v),1\}, \forall v \in V, \forall c \in C_v \quad (18)$$

$$\sum_{v \in V} W_{imv} \leq 1, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (19)$$

$$O_{imvkc} \in \{0, 1\}, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \forall v \in V \quad (20)$$

$$w_{imv} \in \{0, 1\}, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \forall v \in V \quad (21)$$

$$l_{imvkc} q_{imvkc} \geq 0, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \quad \forall v \in V, \forall (k, c) \in P_v \times C_v \quad (22)$$

$$t_{im} - t_{i(m-1)} \geq 0, \quad \forall (i, m) \in N, m \neq 1 \quad (23)$$

$$t_{im} \leq TH, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (24)$$

$$x_{imjnv} \left[t_{im} + TW_i w_{imv} + \sum_{k \in P_v} \sum_{c \in C_v} \begin{pmatrix} TQ_{ik} q_{imvkc} \\ + TO_{ik} o_{imvkc} \end{pmatrix} + TT_{ijv} - t_{jn} \right] \leq 0, \quad \forall (i, m, j, n) \in A_v, \forall v \in V, i \neq j \quad (25)$$

$$t_{im} \geq 0, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (26)$$

$$q_{imvkc} \leq s_{imk} + J_{ik} R_{ik} [TW_i w_{imv} + \sum_{k \in P_v} \sum_{c \in C_v} (TQ_{ik} q_{imvkc} + TO_{ik} o_{imvkc})], \quad \forall (i, m) \in N_p, \forall v \in V, \forall k \in P_v, \quad \forall c \in C_v \quad (27)$$

$$S_{i1k} = IS_{ik} + J_{ik} R_{ik} t_{i1}, \quad \forall (i, k) \in H \times K_i \quad (28)$$

$$S_{i(m-1)k} - \sum_{v \in V} \sum_{c \in C_v} J_{ik} q_{i(m-1)vk} + J_{ik} R_{ik} (t_{im} - t_{i(m-1)}) - S_{imk} = 0, \quad \forall (i, m, k) \in N \times K_i, m \neq 1 \quad (29)$$

$$SM_{ik} \leq S_{imk} \leq SX_{ik}, \quad \forall (i, m, k) \in N \times K_i \quad (30)$$

$$SM_{ik} \leq S_{imk} - \sum_{v \in V} \sum_{c \in C_v} J_{ik} q_{imvkc} + \sum_{v \in V} J_{ik} R_{ik} \left[TW_i w_{imv} + \sum_{k \in P_v} \sum_{c \in C_v} \begin{pmatrix} TQ_{ik} q_{imvkc} \\ + TO_{ik} o_{imvkc} \end{pmatrix} \right] \leq SX_{ik}, \quad \forall (i, m, k) \in N \times K_i \quad (31)$$

$$SM_{ik} \leq S_{imk} - \sum_{v \in V} \sum_{c \in C_v} J_{ik} q_{imvkc} + J_{ik} R_{ik} (TH - t_{im}) \leq SX_{ik}, \quad \forall (i, m, k) \in N \times K_i, m \neq MLi \quad (32)$$

$$S_{imk} \geq 0, \quad \forall v \in V, \forall (i, m) \in N \quad (33)$$

Objective function (1) is a minimization function of the total operating cost during the planning horizon which consists of travelling costs and setup costs for ship in port and setup cost for loading and unloading the product. Constraints related to routing are from constraint (2) until constraint (9). Constraint (2) ensures that each ship departs from or remains at its initial position. Constraints (3) and (4) enforce that a ship either leaves or finishes its route at node (i,m) . Constraint (5) restricts so that only one ship can occupy node (i,m) . Constraint (6) imposes that the precedence constraint is not violated. Then, constraints (7)–(9) declare the binary values for the variables involved in routing constraints.

Constraints related to loading and unloading activity are from constraint (10) until constraint (22). Constraint (10) tracks the quantity onboard before and after a visit at node (i, m) . Constraint (11) imposes that compartments on board must be empty if ship v goes to production ports. Constraint (12) ensures that the quantity onboard ship v at the beginning of the period is the same as the initial quantity of product k loaded in compartment c . Constraints (13) and (14) restrict the quantity loaded (or unloaded) and the quantity onboard within their limits. Constraints (15) and (16) enforce that if there are loading (or unloading) activities, a loading (or unloading) setup and a port setup must be considered, respectively. Constraint (17) assures that if ship v has a product in its compartment, only the same product can be loaded into there. Constraints (18) and (19) restricts so that only one ship can perform a loading (or unloading) setup and port setup, respectively. Constraint (20) declares the binary variables for the variables involved in loading (or unloading) activity. Then, constraints (21) and (22) state the continuous variables of loading (or unloading) quantity.

Constraint related to time and scheduling are from constraint (23) until constraint (26). Constraint (23) ensures that the time precedence constraints are not violated. Constraint (24) restricts the arrival time within the planning horizon. Constraint (25) tracks the routing time from a node to another node. Then, constraint (26) declares the continuous variables of arrival time.

Constraints related to inventory are from constraint (27) until constraint (33). Constraint (27) imposes that the loading quantity must not exceed the available product in the storage. Constraint (28) tracks the storage level at the time

of first arrival. Constraint (29) tracks the storage level of the current and previous visit. Constraint (30) and (31) guarantee that the storage level will be within its limits at the time a ship arrives at and departs from a port, respectively. Constraint (32) bounds the storage level at the end of the planning horizon. Finally, constraint (33) declares the continuous variables of inventory level.

2.5. Column Generation

Column generation refers to linear programming (LP) algorithms designed to solve problems in which there are a huge number of variables compared to the number of constraints and the simplex algorithm step of determining whether the current basic solution is optimal or finding a variable to enter the basis is done by solving an optimization problem rather than by enumeration (Nemhauser, 2010).

This idea can be generalized to yield an algorithm for solving any LP by partitioning the constraints into a set of master constraints and a set of subproblem constraints and also known as Dantzig–Wolfe decomposition (Dantzig, 1960). Using Dantzig–Wolfe decomposition, a large LP can be decomposed into a master problem with a small number of constraints and an exponential number of variables corresponding to the extreme points of the subproblems, the solution of which represents convex combinations of these extreme points that satisfy the master constraints. Optimal dual solutions of the master problem provide prices to the subproblems, whose solutions yield new extreme point variables for the master (Nemhauser, 2010).

The first column generation work that involved integer variables appears to have been done by Gilmore and Gomory, 1961. They studied the cutting stock problem: given a positive integer number $d(i)$ of items of integer size $a(i)$, determine the minimum number of stock rolls of integer size b needed to pack all of the items. These models appear in many practical applications as well like the one that has received the most attention in the literature is airline crew scheduling, but there are many other applications to all kinds of transportation routing problems, scheduling problems, districting problems, coloring problems, etc (Lübbecke and Desrosiers, 2005).

The problem being solved is split into two problems: the master problem and the subproblem. The master problem is the original problem with only a subset of variables being considered. The subproblem is a new problem created to identify a new variable. Let assume the following linear problem is considered as the master problem (MP). An explicit search of J may be computationally impossible when $|J|$ is huge. In practice, it usually only considers a reasonably small subset $J' \subseteq J$ of columns, and later called it a restricted master problem (RMP).

$$z^* := \min \sum_{j \in J} c_j \lambda_j$$

Subject to:

$$\sum_{j \in J} a_j \lambda_j \geq b$$

where $\lambda_j \geq 0, j \in J$

Given $u \geq 0$ of dual variables, it is expected to find $\text{Arg min } \{c'_j := c_j - u^T a_j | j \in J\}$. Assuming the feasible solution has been found, let λ' and u' be primal and dual optimal solutions of the RMP, respectively. When columns $a_j, j \in J$, are implicitly given as elements of a set $\mathcal{A} \neq \emptyset$, and the cost coefficient c_j can be computed from a_j , then the subproblem returns an answer to the pricing problem.

$$c'^* := \min \{c(a) - u'^T a \mid a \in \mathcal{A}\}$$

If $c'^* \geq 0$, then no reduced cost of coefficient c'_j is negative and λ' optimally solves the master problem as well. If $c'^* < 0$, then add a column derived from the subproblem answer to the RMP and this algorithm called the column generation subproblem (Lübbecke and Desrosiers, 2005).

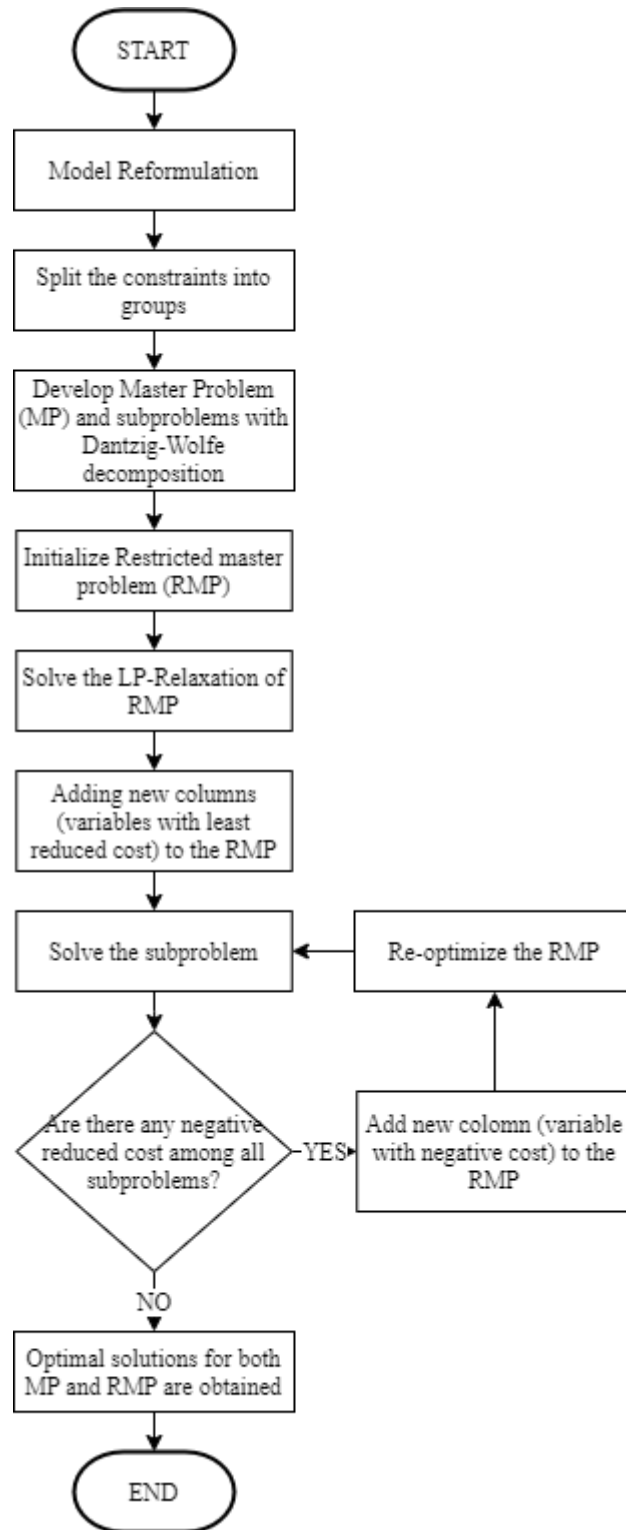


Figure 2. 3 Algorithm for Column Generation Approach

The processes of column generation are shown in the figure 2.3. First, an initial restricted master problem is solved. Then some new columns are added to the restricted master problem. These columns correspond to the variables with

least reduced costs in the solution of the restricted master problem. The dual values from the solution of the restricted master problem are then utilized in the objective function of the subproblem. The subproblem is solved. If the objective value of the subproblem is negative, a variable with negative reduced cost has been identified. This variable is then added to the restricted master problem, and the restricted master problem is re-solved. Re-solving the restricted master problem will generate a new set of dual values, and the process is repeated until no negative reduced cost variables are identified. At that point all the feasible solutions in the original master problem have been implicitly evaluated. A continuous optimal solution is then attained for both the original and the restricted master problem (Christiansen and Nygreen, 2005).

2.6. Research Gap & Positions

The ISRP research literature that had been done before can be mapped into several research quadrants between single-multiple product ISRP, single-multiple depot ISRP and dedicated-undedicated compartment. The mapping can be seen in table 2.1 below.

The research of ISRP with multi product and undedicated compartment has been done several time using exact algorithm like the one done by (Siswanto et.al, 2011) and (Rani, 2010). But both of them are considering multiple depot/loading port in their model. The research involving single depot has been done using several approach like dynamic programming algorithm and column generation by (Christiansen, 2005, 2009), heuristic by (Rahman, 2008) and metaheuristic by (Nurminarsih, 2012) and (Santosa, 2016). With regard of type of compartment, (Hwang, 2007) using dedicated compartment when dealing with multiple products. This means that it is not permissible to assign a product to a compartment that has been used previously by other products. In the other hand, in undedicated compartment the compartments are not bound to specific type of product and can be used to store different products.

Table 2. 1 ISRP Research Literature

Researcher	ISRP Characteristics							
	Single Product	Multi Product	Dedicated Compart.	Undedicated Compart.	Single Depot	Multi Depot	Time Windows	Solution
Harianto (2015)	V			V		V		Simple Iterative Mutation Algorithm
Christiansen (2009)	V			V	V		V	Dynamic Programming Algorithm
Christiansen (2005)	V			V	V		V	MILP with Column Generation
Hwang (2007)		V	V			V	V	MILP
Rahman (2008)		V	V		V			Heuristic
Siswanto et.al, (2011)		V		V		V		MILP & multiple heuristic
Rani (2010)		V		V		V		MILP
Nurminarsih (2012)		V		V	V		V	Tabu Search
Santosa (2016)		V		V	V			CEGA & Hybrid Tabu Search
This Research	V			V	V			MILP with Column Generation

In this study, Mixed Integer Linear Programming (MILP) with Column Generation is chosen to handle the combinatorial optimization problem with large number of variables with expectation to achieve faster computation time. This research with column generation approach adds to the variety of ISRP that have been developed with characteristics of single product, undedicated compartment, only single depot and no time windows.

CHAPTER III

RESEARCH METHODOLOGY

The constructed methodology in conducting this research will be presented in this chapter.

3.1. Research Methodology

Research methodology is the basic systematic framework of this final project which in general consists of several parts. Flowcharts regarding the stages of conducting research can be seen in Figure 3.1 and the explanations of each stage are described in the sub-chapters below.

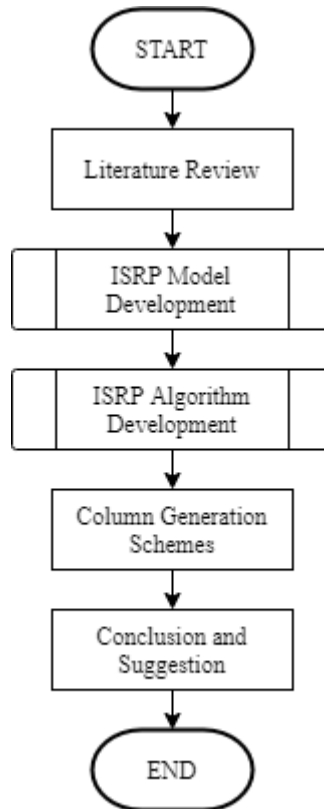


Figure 3. 1 Research Methodology

3.1.1 Literature Review

Several relevant literatures are summarized as the theoretical basis in conducting this research. Exploration of journals and previous research on related issues is needed to better understand the problem and the appropriate solution.

3.1.2 Model Development

At this stage the model development for the problem is carried out. The model was adopted from the multi-product multi depot ship inventory routing problem with undedicated compartment model in the research of Siswanto et.al (2011). The mathematical model from Siswanto et. al will be modified to suit the column generation approach inspired by Christiansen (2005) research.

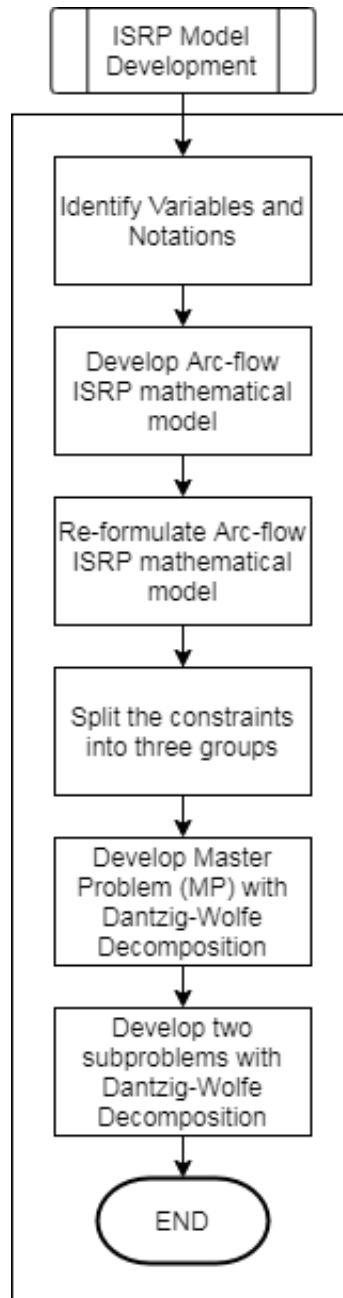


Figure 3. 2 ISRP Model Development

As shown in Figure 3.2, the IRSP model development in this study consisted of several stages and explained as follows (Christiansen, 2005).:

1. Identify variables and notations

Identify ISRP-related variables and notations like planning horizon, port and (un)load setup time, initial position and product quantity onboard of each ships, initial inventory level at each loading and unloading port, product consumption or production rate at each port, travel time between each port, eligibility of each ship to harbor at the different port, etc.

2. Develop arc-flow ISRP mathematical model

Develop arc-flow ISRP mathematical model based on Siswanto, 2011 with modification to suit the problem in this study.

3. Reformulate arc-flow ISRP mathematical model

Reformulate the arc-flow ISRP mathematical model because there are variables which existed in both types of subproblem. This issue is resolved by introducing several new variables which ultimately requires the existed model to be reformulated.

4. Split the constraints

To solves the planning problem with column generation approach it requires to separate the constraints into several groups. In this case, there should be three constraints group which are ship and port dependent variables, ship routing and inventory management.

5. Develop Master Problem (MP) with Dantzig-Wolfe Decomposition

According to the column generation approach, instead of using the original variables from the arc-flow formulation the variables corresponding to ship schedules and port call sequences is used in the master problem. The master problem includes additional coupling constraints for the load quantities and starting times to synchronize the ship schedule and port inventory aspects in addition to the usual visit constraints.

6. Develop Sub problems with Dantzig-Wolfe Decomposition

In the ISRP solved with column generation approach usually had some degree of model adjustment by decomposing the problem into several sub problems. In this study, the problem decomposes into a routing and

scheduling sub problem for each ship and an inventory management sub problem for each port. Each sub problem has its own objective function and the problem reach optimal solution when there is no reduced cost value in any sub problems solution.

3.1.3 Algorithm Development

The algorithm for the single depot single product inventory routing problem with undedicated compartment model will be based on general algorithm of column generation approach as shown in Christiansen (2005) research. The algorithm development in this study is explained in the Figure 3.3.

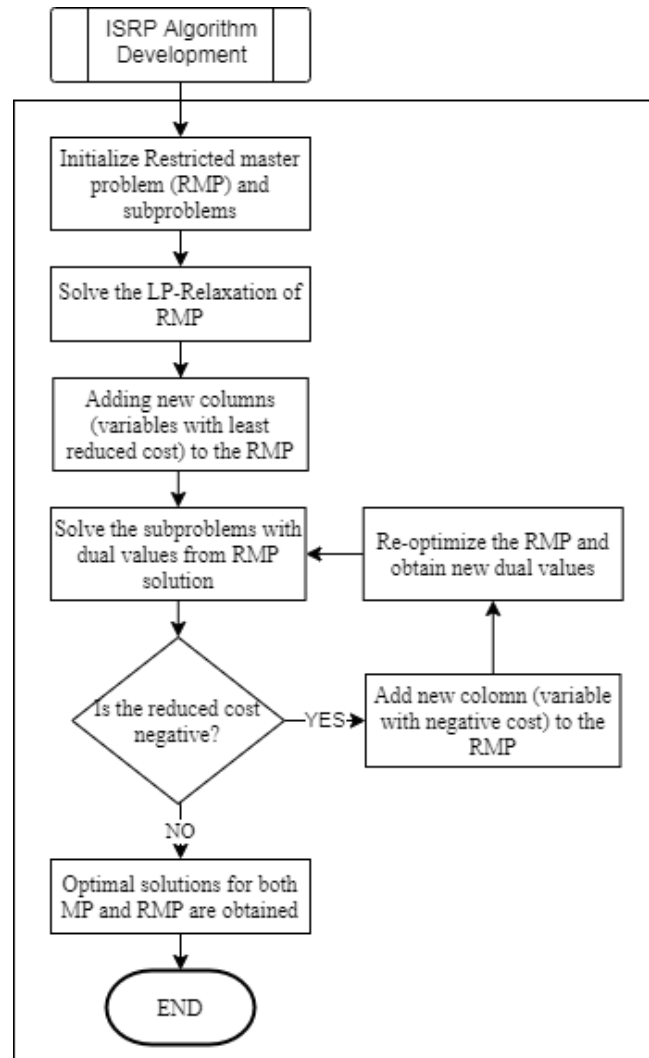


Figure 3. 3 ISRP Algorithm Development

The following is an explanation of the proposed ISRP algorithm with column generation approach:

1. Initialize Restricted Master Problem (RMP) and Subproblems

Since Restricted Master Problem (RMP) only differs from the continuous original master problem by having fewer variables. The variables in question used in the master problem are corresponding to ship schedules and port call sequences.

2. Solve the LP-Relaxation of RMP

By solving the RMP, the information about reduced cost for each variable are known.

3. Adding new columns to the RMP

Then some new columns are added to the restricted master problem. These columns correspond to ship schedules and port call sequences with least reduced costs in the solution of the master problem.

4. Solve the Subproblems

By solving the subproblems, the ship schedules and port call sequences are generated.

5. Check for negative reduced cost in Subproblems solutions

In this step, check for any negative reduced cost value in subproblems solution. If there is any then proceed to step 6, otherwise proceed to step 8.

6. Adding new column to the RMP

Each variable with a negative cost reduction is then translated into a new column added to the RMP.

7. Re-optimize the RMP

The restricted master problem is reoptimized with the added new columns, resulting in new dual values. This means that the dual values from the solution of the new restricted master problem are transferred to the subproblems. Continue to step 4.

8. Obtain optimal solution for Master Problem

This procedure finishes until no columns with negative reduced costs exist, and no improvements can be made. At this point all the feasible solutions in the original master problem have been implicitly evaluated. A

continuous optimal solution is then attained for both the original and the restricted master problem.

3.1.4 Column Generation Schemes

At this stage, the discussion about column generation on large dataset ISRP, dual value analysis, column generation challenges, and column generation improvement schemes such as intensification, diversification, and stabilization will be explained in detail.

3.1.5 Conclusion and Suggestion

At this stage, conclusions and suggestions are drawn. The conclusion will answer the research objectives that were formulated earlier based on the results of this study and suggestions will provide input for further research.

CHAPTER IV

DEVELOPMENT OF MODEL AND ALGORITHM

A comprehensive explanation of the problem description, model development, algorithm development and algorithm validation will be presented in this chapter.

4.1 Problem Definition

The ISRP has a multitude of characteristics derived from the real-world scenario. Different specifications of these characteristics give rise to a different ISRP. The ISRP characteristics that will be modeled in this study try to simulate the cement bulk distribution process with reduced flexibility compared with the real situation for simplicity reason. Cement bulk distribution process involves the design of a set of minimum cost routes for a fleet of bulk ships that service a set of harbors one or more times during the planning period.

In this case, the transporter owns plants located near ports. A single product is produced at sources, and the sources called as loading ports. Similarly, the product is consumed at certain destinations and the corresponding ports are called unloading ports. Because the transporter owns both the internal consumption and production factories, inventory costs will be disregarded in the optimization problem. Inventory storage capacities are given in all ports, and there exist information about the consumption or production rates of the transported product. For all ports, the conditions are relatively the same in term of the facility and accessibility, thus making the ships able to visit all the port as their destination.

The transporter operates a heterogeneous fleet of ships. The ships have different cost structure, load capacity and undedicated compartment as they carry only one product. All the ships are owned by the company, so no time chartered ship and the ship operating cost are included into the travelling cost. The ships transport this product from loading ports to unloading ports and if the product is loaded and unloaded in time, thus there should be no need to stop the production in the cement factories because of missing transportation possibilities. There is no

natural depot for the ships. The initial position of a ship may be at a harbor or a point at sea.

In this study, the transportation cost and the demand for bulk cement is assumed to be constant because of the short planning horizon (14 days). Each ship may visits one or several harbors during the planning horizon, but they only assigned to one route for each ship. There is no requirement for a specified position for any ship at the end of the planning period. It has been arranged so that only one ship is allowed to dock at a port at a time. But the port can be visited more than once, so the arrival time of the ship is important to avoid the ship idling/wait at the port.

In this study, development of a detailed plan which has a planning horizon of 14 days will be developed, whereas a long-term plan may expand up to 3 months. The time will not be discretized and all the port assumed to operate all the time so the port can serve the ship whenever it arrived (no time window). The planning problem is therefore to find routes at least transportation costs given uninterrupted production.

In contrast to most ship scheduling problems, the number of calls at a given port during the planning period is not predetermined, neither is the quantity to be loaded and unloaded in each port call. The production or consumption rate and inventory information at each port, together with ship capacities and the location of the ports, determine the number of possible calls at each port, the time for start of service and the range of feasible load quantities for each port call.

It is more critical to come too late to a port than too early. If the ship comes too late, the storage is full in a production port, and the company has to stop the production. However, if the ship come too early and start loading, the storage gets empty during the loading such that the optimal load quantity cannot be loaded. This comes from the fact that the capacities of the storages are more critical than the capacity of the ships.

In terms of the recent taxonomy introduced by Andersson et al. (2010), the problem considered in this paper is a single-product, finite-horizon IRP with deterministic supply/demand, a heterogeneous fleet, continuous routing (i.e.,

without fixed start and end points), and with a many-to-many distribution/procurement topology.

4.2 Model Development

Due to the complexity of the Siswanto et.al (2011) ISRP model, only small sized data instances can be solved directly to optimality. Therefore, a column generation approach is developed to reduce the computation time required for such complex model. In this subchapter, the model development process will be discussed in further detail including the step by step processes such as reformulation of arc-flow model, reformulation using Dantzig-Wolfe decomposition, development of master problem and its sub problems.

4.2.1 Reformulate Arc-Flow ISRP Mathematical Model

The ISRP model from Siswanto et.al (2011) can't be decompose directly, because it does not separate due to the starting time t_{im} and the load quantity q_{imvk} variables. These variables are needed in both types of subproblem; the ship scheduling and the port subproblems. This issue is resolved by introducing new time and quantity variables, such that the variables for each (i, m, v)-combination (t_{vimv} and q_{vimv}) and each port call (t_{im} and q_{im}) are obtained and introducing coupling constraints to the problem as follows (Christiansen, 2005):

$$t_{im} - \sum_{v \in V} t_{vimv} = 0, \quad \forall i \in N, m \in M_i \quad (34)$$

$$q_{im} - \sum_{v \in V} q_{vimv} = 0, \quad \forall i \in N, m \in M_i \quad (35)$$

$$t_{vimv} \geq 0, \quad \forall v \in V, \forall i \in N, m \in M_i \quad (36)$$

$$q_{im} \geq 0, \quad \forall i \in N, m \in M_i \quad (37)$$

For each harbor arrival (i, m), the load quantity and start time in the chosen ship schedules (q_{vimv} and t_{vimv}) have to be equal to the quantity and time in the chosen harbor visit sequences (q_{im} and t_{im}). This means that coupling constraints are needed for the start times and load quantities to synchronize the harbour inventory and ship schedule aspects, and these are given in constraints

(34) and (35), respectively. With these new variables previous constraints can be updated and to accommodate the problem definition new arc-flow ISRP mathematical model must be defined.

The new arc-flow ISRP model are based on Siswanto et.al (2011) model with modification to number of product and compartment.

Minimize

$$\sum_{v \in V} \sum_{(i,m,j,n) \in A_v} CT_{ijv} X_{imn jv} + \sum_{v \in V} \sum_{(i,m) \in N} (CW_{iv} + CO_i) w_{imv} \quad (38)$$

Subject To:

$$\sum_{(j,n) \in N} X_{0(v)1 jnv} + Z_{0(v)1v} = 1, \quad \forall v \in V \quad (39)$$

$$\sum_{(j,n) \in N \cup \{0(v),1\}} X_{jnimv} - \sum_{(j,n) \in N} X_{imjnv} + Z_{imv} = 0, \quad \forall (v,i,m) \in V \times N, i \neq j \quad (40)$$

$$\sum_{(i,m) \in N} Z_{imv} = 1, \quad \forall v \in V \quad (41)$$

$$\sum_{v \in V} \sum_{(j,n) \in N \cup \{0(v),1\}} X_{jnimv} + y_{im} = 1, \quad \forall (i,m) \in N, i \neq j \quad (42)$$

$$x_{imjnv} \in \{0,1\}, \quad \forall v \in V, \forall (i,m,j,n) \in A_v \quad (43)$$

$$Z_{imv} \in \{0,1\}, \quad \forall v \in V, \forall (i,m) \in N \cup \{0(v),1\} \quad (44)$$

$$x_{imjnv} (l_{imv} + J_i q_{Vjnv} - l_{jnv}) = 0, \quad \forall v \in V, \forall (i,m) \in N, (j,n) \in N, i \neq j \quad (45)$$

$$x_{imjnv} (l_{imv}) = 0, \quad \forall v \in V, \forall (i,m) \in N \cup \{0(v),1\}, (j,n) \in N, i \neq j \quad (46)$$

$$QQ_v = l_{0(v)1v}, \quad \forall v \in V, \quad (47)$$

$$q_{Vimv} \leq \sum_{(j,n) \in N \cup \{0(v),1\}} CM_v X_{jnimv}, \quad \forall (i,m) \in N, \forall v \in V, i \neq j \quad (48)$$

$$l_{imv} \leq \sum_{(j,n) \in N \cup \{0(v),1\}} CM_v X_{jnimv}, \quad \forall (i,m) \in N, \forall v \in V, i \neq j \quad (49)$$

$$q_{vimv} \leq CM_v w_{imv}, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \quad \forall v \in V \quad (50)$$

$$\sum_{v \in V} w_{imv} \leq 1, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (51)$$

$$w_{imv} \in \{0, 1\}, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \forall v \in V \quad (52)$$

$$l_{imv}, q_{vimv} \geq 0, \quad \forall (i, m) \in N \cup \{o(v), 1\}, \quad \forall v \in V \quad (53)$$

$$x_{imjnv} [t_{vimv} + (TW_i + TO_i)w_{imv} + TQ_i q_{vimv} + TT_{ijv} - t_{vjnv}] \leq 0, \quad \forall (i, m, j, n) \in Av, \forall v \in V, i \neq j \quad (54)$$

$$y_{im} - y_{i(m-1)} \geq 0, \quad \forall (i, m) \in N, m \neq 1 \quad (55)$$

$$y_{im} \in \{0, 1\}, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (56)$$

$$t_{im} - t_{i(m-1)} \geq 0, \quad \forall (i, m) \in N, m \neq 1 \quad (57)$$

$$t_{im} \leq TH, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (58)$$

$$t_{im} \geq 0, \quad \forall (i, m) \in N \cup \{o(v), 1\} \quad (59)$$

$$q_{vimv} \leq s_{im} + J_i R_i [(TW_i + TO_i)w_{imv} + TQ_i q_{vimv}] \quad \forall (i, m) \in Np, \forall v \in V \quad (60)$$

$$S_{i1} = IS_i + J_i R_i t_{i1}, \quad \forall (i, k) \in H \times Ki \quad (61)$$

$$S_{i(m-1)} - J_i q_{i(m-1)} + J_i R_i (t_{im} - t_{i(m-1)}) - S_{im} = 0, \quad \forall (i, m, k) \in N \times Ki, m \neq 1 \quad (62)$$

$$SM_i \leq S_{im} \leq SX_i, \quad \forall (i, m, k) \in N \times Ki \quad (63)$$

$$SM_i \leq S_{im} - J_i q_{im} + \sum_{v \in V} J_i R_i [(TW_i + TO_i)w_{imv} + TQ_i q_{im}] \leq SX_i, \quad \forall (i, m, k) \in N \times Ki \quad (64)$$

$$SM_i \leq S_{im} - J_i q_{im} + J_i R_i (TH - t_{im}) \leq SX_i \quad \forall (i, m, k) \in N \times Ki, m \neq MLi \quad (65)$$

$$S_{im} \geq 0, \quad \forall v \in V, \forall (i, m) \in N \quad (66)$$

To accommodate the problem definition, several constraints that contain index of k and c are being updated (related to number of product and ship compartment). Another changes can be seen in (54), t_{im} and t_{jn} are transformed to t_{vimv} and t_{vjnv} , while $\sum_{v \in V} q_{imvkc}$ transformed to q_{im} in (62),(64)-(65). In the new mathematical model, the constraint that assures ship v has a product in its

compartment, only the same product can be loaded into there (17) are removed because number of product are reduced and the separation of compartment become unnecessary. Because of that, the use of variable o_{imvkc} in original mathematical formulation can be removed and replaced by w_{imv} . Constraint (45) and (46) are non-linear and can be formulated as equivalent linear constraints as illustrated by Christiansen (1999) and Al-Khayyal and Hwang (2007). Constraints (54) are linearized in the same way as Constraints (45), but, because the original constraints have a \leq sign, just one type of constraints is necessary in the linearized version.

After this reformulation, the constraint set can be split into three groups. This makes it possible to solve the planning problem by a column generation approach. The first constraint group consists of (34)-(37) and (42). These constraints are the common constraints where the ship dependent variables and the port variables are being synchronized. The second constraint group consists of the ship routing constraints (39)-(41) and (43)-(54). The port inventory constraints describe the inventory management for each port because there exists no interaction between the ports, and they are based on the last group of constraints (55)-(66).

4.2.2 Dantzig–Wolfe Decomposition

By use of a Dantzig–Wolfe decomposition approach, the common constraints constitute the master problem. None of the ship constraints include interaction between the ships, so these constraints can be split into a subproblem for each ship. The harbor inventory constraints constitute the harbor subproblem, which can be split and solved individually for each internal harbor because there exists no interaction between the harbors.

Let R_v be the set of ship schedules, indexed by r for ship v . Schedule r includes information about the geographical route, where X_{imjnv} is set equal to 1 if the corresponding variable, x_{imjnv} , in the arc flow model has the value 1 for schedule r . In addition, the following information is given for each (v,r) combination: The number of visits, 0 or 1, at port call (i,m) , A_{imvr} , the load

quantity of each port call, Q_{vimvr} and the starting time of each port call T_{vimvr} . The transportation cost for sailing schedule r by ship v is C_{MVvr} .

The harbors are faced with inventory management problems, and include information about the stock limits and the production pattern. It is important to determine the load quantity and time for start of service at each arrival such that the stock level is within its limits during the entire planning period. Normally, there exist several feasible combinations of load quantities, start times, and number of visits at a harbor during the planning period, and for each of those combinations are called harbor visit sequence s . The values of T_{Pims} and Q_{Pims} represent the start time and load quantity for the harbor arrival (i, m) in sequence s , respectively. The value of W_{ims} is 1 if sequence s is not visiting harbor arrival (i, m) . From W_{ims} , the number of arrivals at harbor i can be known. Because both the harbor and ship subproblems define path structures, its extreme points correspond to paths in the underlying networks. The set of S_{yv} and S_{zi} defines the extreme points for ship v and for internal harbor i respectively. As for the solutions to the ship and harbor subproblems, any solutions that satisfying the harbor inventory constraint and ship constraints can then be expressed as a non-negative convex combination of these extreme points and must consist of binary x_{imnjv} and y_{im} . The new notations that will be used in column generation approach are listed below:

- Continuous Variable:

A_{imvr}	The number of visits, 0 or 1, at port call (i, m)
t_{vimv}	The time ship v visit port call (i, m)
q_{im}	the amount (un)loaded in port call (i, m)
Q_{vimvr}	the load quantity of each port call (i, m) in route r
T_{vimvr}	the starting time of each port call (i, m) in route r
Q_{Pims}	the start time for the harbor arrival (i, m) in sequence s
T_{Pims}	the load quantity for the harbor arrival (i, m) in sequence s
C_{MVvr}	The transportation cost for sailing schedule r by ship v
C_{MPis}	the setup cost for both ship and loading/unloading process for each call-in sequence s for port i

RCS_{vr} reduced cost for y_{vr}

RCP_{is} reduced cost for z_{is}

$F_{Sjnv}(t)$ the least cost from start position for ship v to arrival node at (j,n) and ready to serve this node within time t

$F_{Pims}(t)$ the minimum costs of port i up to arrival at node m if the service of this node can start within time t

- Binary Variable:

X_{imjnv} $\begin{cases} 1, & \text{if the corresponding variable, } x_{imjnv}, \text{ in the arc flow model has the value 1 for schedule } r \\ 0, & \text{otherwise} \end{cases}$

y_{vr} $\begin{cases} 1, & \text{if ship } v \text{ chooses to sail route } r \\ 0, & \text{otherwise} \end{cases}$

W_{ims} $\begin{cases} 1, & \text{if sequence } s \text{ is not visiting harbor arrival } (i,m) \\ 0, & \text{otherwise} \end{cases}$

z_{is} $\begin{cases} 1, & \text{if harbor } i \text{ chooses sequence } s \\ 0, & \text{otherwise} \end{cases}$

- Parameter:

CO_i loading or unloading setup cost for port i

π_i the cost to serve a ship in harbor i at arrival m which are the sum of setup cost of ship v and loading/unloading setup cost at port i

μ_{ij} fixed cost for travelling on every arc $(i,j) \in A_v$

- Variable Set:

N the set of ports

V the set of available ships

M_i the set of possible calls at port i

R_v the set of ship schedules for ship v

A_v the set of all feasible arcs for ship v

S_i the set of sequences for port i

Detailed Dantzig-Wolfe decomposition are shown below as illustrated by Christiansen (1999).

$$x_{imjnv} = \sum_{r \in S_{yv}} X_{imjnv} y_{vr}, \quad \forall v \in V, (i, m, j, n) \in A_v \quad (67)$$

$$x_{imjnv} \in \{0, 1\}, \quad \forall v \in V, (i, m, j, n) \in A_v \quad (68)$$

$$q_{vimv} = \sum_{r \in S_{yv}} Q_{vimvr} y_{vr}, \quad \forall v \in V, i \in N, m \in M_i \quad (69)$$

$$t_{vimv} = \sum_{r \in S_{yv}} T_{vimvr} y_{vr}, \quad \forall v \in V, i \in N, m \in M_i \quad (70)$$

$$\sum_{r \in S_{yv}} y_{vr} = 1, \quad \forall v \in V, \quad (71)$$

$$y_{vr} \geq 0, \quad \forall v \in V, r \in R_v, \quad (72)$$

$$y_{im} = \sum_{s \in S_{zi}} W_{ims} z_{is}, \quad \forall i \in N, m \in M_i \quad (73)$$

$$y_{im} \in \{0, 1\}, \quad \forall i \in N, m \in M_i \quad (74)$$

$$q_{im} = \sum_{s \in S_{zi}} Q_{pims} z_{is}, \quad \forall i \in N, m \in M_i \quad (75)$$

$$t_{im} = \sum_{s \in S_{zi}} T_{pims} z_{is}, \quad \forall i \in N, m \in M_i \quad (76)$$

$$\sum_{s \in S_{zi}} z_{is} = 1, \quad \forall i \in N \quad (77)$$

$$z_{is} \geq 0, \quad \forall i \in N, s \in S_i, \quad (78)$$

The ship paths give values to the x_{imjnv} , q_{vimv} and t_{vimv} variables. The new variables y_{vr} are said to be the route variables, and equal 1 if ship v chooses to sail route r . The number of visits at harbor arrival (i, m) on route r by ship v is given by $A_{imvr} = \sum_{j \in H_v, n \in M_{jv}} X_{jnimvr}$. Values 0 and 1 for A_{imvr} give a feasible route. While the harbor paths include information of the values on y_{im} , q_{im} and t_{im} . Variable $z_{is} = 1$ if harbor i chooses sequence s . The discussed integer requirements, together with the coupling constraints, imply that all positive harbor visit sequences for a harbor give the same number of visits at each harbor.

4.2.3 Master Problem

According to the column generation approach, variables used in the master problem corresponding to ship schedules and port call sequences instead of using

the original variables from the arc-flow formulation. The common constraints, (34) – (37) and (42) are defined by the arc flow variables, but this context using an alternative equivalent formulation consisting of path variables is preferred. This path flow model consists of finding the optimal combination of feasible ship schedules and harbor visit sequences. This path flow model corresponds to a formulation based on the Dantzig–Wolfe decomposition approach, and the common constraints given by the path flow variables constitute the master problem.

Substituting (67)-(73) and (75)-(78) in (34), (35), (38) and (42), the integer master problem is transformed into

$$\min \sum_{v \in V} \sum_{r \in R_v} C_{MVvr} y_{vr} + \sum_{i \in N} \sum_{s \in S_i} C_{MPis} z_{is} \quad (79)$$

Subject to:

$$\sum_{v \in V} \sum_{r \in R_v} A_{imvr} y_{vr} + \sum_{s \in S_i} W_{ims} z_{is} = 1, \quad \forall i \in N, m \in M_i \quad (80)$$

$$\sum_{v \in V} \sum_{r \in R_v} Q_{Vimvr} y_{vr} - \sum_{s \in S_i} Q_{Pims} z_{is} = 0, \quad \forall i \in N, m \in M_i \quad (81)$$

$$\sum_{v \in V} \sum_{r \in R_v} T_{Vimvr} y_{vr} - \sum_{s \in S_i} T_{Pims} z_{is} = 0, \quad \forall i \in N, m \in M_i \quad (82)$$

$$\sum_{r \in S_{yv}} y_{vr} = 1, \quad \forall v \in V, \quad (83)$$

$$\sum_{s \in S_{zi}} z_{is} = 1, \quad \forall i \in N \quad (84)$$

$$y_{vr} \geq 0, \quad \forall v \in V, r \in R_v, \quad (85)$$

$$z_{is} \geq 0, \quad \forall i \in N, s \in S_i, \quad (86)$$

$$\sum_{r \in R_v} X_{imjnv} y_{vr} \in \{0,1\}, \quad \forall v \in V, (i, m, j, n) \in A_v \quad (87)$$

The objective function (79) minimizes the transportation costs and setup cost of ship and loading/unloading process. The master problem includes additional coupling constraints for the load quantities and starting times to synchronize the ship schedule and port inventory aspects in addition to the usual visit constraints (80). These coupling constraints are given in (81) and (82). The convexity rows for the ships and ports are given in constraints (83) and (84). The

integer requirements are defined by (87) and correspond to declaring the original flow variables as binary variables. Here, C_{MVvr} are the transportation costs for sailing route r by ship v . While C_{MPis} are the setup cost for both ship and loading/unloading process for each call-in sequence s for port i . These costs can be written as:

$$C_{MVvr} = \sum_{i,m,j,n \in A_v} CT_{ijv}(A_{imvr}, X_{imjnv}) \quad (88)$$

$$C_{MPis} = \sum_{m \in M_{Ti}} (CW_{iv} + CO_i)W_{ims} \quad (89)$$

4.2.4 Column Generation

According to the Dantzig–Wolfe column generation approach, the generated columns must be the most promising only. These columns are the harbor visit sequences and ship routes with least reduced costs in the master problem. However, for real instances of the ship planning problem it is time consuming to generate all these schedules and sequences, and the number of such schedules and sequences would result in too many columns when solving the models. To avoid such problem, the LP-relaxation version of the restricted master problem is used which only differs from the continuous original master problem by having fewer variables. First, an initial restricted master problem is solved. Then some new columns are added to the restricted master problem. These columns correspond to ship schedules and port call sequences with least reduced costs in the solution of the master problem. By doing this, the dual values from the solution of the restricted master problem are transferred to the subproblems. Then the subproblems are solved, and new ship schedules and port call sequences are generated. The restricted master problem is reoptimized with the added new columns, resulting in new dual values. This procedure continues until no columns with negative reduced costs exist, or when the gap between the two bounds is small. At that point all the feasible solutions in the original master problem have been implicitly evaluated. A continuous optimal solution is then attained for both the original and the restricted master problem.

In order to calculate the reduced costs of the path flow formulation corresponding to constraints (34), (35), (38) and (42) dual variables of the constraints must be defined. The following dual variables D_{Vim} , D_{Qim} , D_{Tim} , D_{Yv} , and D_{Zi} are defined for constraints (80)-(84).

D_{Vim} dual variable of visit row (80) for harbor arrival (i, m);

D_{Qim} dual variable of quantity coupling constraint (81) for harbor arrival (i, m);

D_{Tim} dual variable of time coupling constraint (82) for harbor arrival (i, m).

D_{Yv} dual variable for constraint (83)

D_{Zi} dual variable for constraint (84)

Now, the reduced cost for y_{vr} and z_{is} can be write down as illustrated in Christiansen (2005).

$$RCS_{vr} = C_{MVvr} - \sum_{i \in I} \sum_{m \in Mi} A_{imvr} D_{Vim} - \sum_{i \in I} \sum_{m \in Mi} Q_{vimvr} D_{Qim} - \sum_{i \in I} \sum_{m \in Mi} T_{vimvr} D_{Tim} - D_{Yv}, \quad (90)$$

$$RCP_{is} = C_{MPis} - \sum_{m \in Mi} W_{ims} D_{Vim} + \sum_{m \in Mi} Q_{pims} D_{Qim} + \sum_{m \in Mi} T_{pims} D_{Tim} - D_{Zi} \quad (91)$$

4.2.5 Port Subproblem

The inventory management problem (or port subproblem) can be separated into a problem for each port and that the costs related to the ports are a sum of costs for each port. The reduced costs for the port subproblems are defined in constraint (91). To simply notation, a linear cost function for harbor subproblem sequences is introduced. π_i as the cost to serve a ship in harbor i at arrival m which are the sum of setup cost of ship v and loading/unloading setup cost at port i ($\pi_i = CW_{iv} + CO_i$). Given that port i can be visited at least twice, the general recursive function $F_{Pims}(t)$ as the minimum costs of port i up to arrival at node m if the service of this node can start within time t and $m \geq 2$ become (Christiansen and Nygreen, 1998b). Therefore, the objective function of port subproblem can be written as:

$$\min \sum_{m \in M_i} [(\pi_i - D_{vim})y_{im} + D_{Qim}q_{im} + D_{Tim}t_{im}] \quad (92)$$

$$F_{Pins}(t) = \min_{0 \leq t' \leq t} \left\{ \left(\sum_{m'=m+1}^{n-1} \pi_{im'} - D_{vim'} \right) + D_{Qim} |Q_{Pins} - Q_{Pims}| \right. \\ \left. + D_{Tim}t + F_{Pims}(t') \right\} \quad (93)$$

$$\min F_{PiEs}(TH) \quad (94)$$

For each port i , there exists one node at level $m = 1$. The value of the function at this node is 0, $F_{Pi1}(t) = 0$. The recursive formulas for the port subproblems are defined for the nodes representing real arrivals. A predecessor node to a real node (i, n) represents the previous arrival, $m = n - 1$, at the port. The start time t^* and load quantity $|Q_{Pins} - Q_{Pims}|$ for the last node visited are obtained. The quantity loaded at $(i, m-1)$ is equal to $|Q_{Pims} - Q_{Pi(m-1)s}|$. While, the optimal start time t at $(i, m-1)$ is given by

$$t \in \arg \min_{t^*} \{ D_{Qi(m-1)} |Q_{Pims} - Q_{Pi(m-1)s}| + D_{Ti(m-1)} t^* \\ + F_{Pi(m-1)s}(t^*) \} \quad (94)$$

The backtracking continues until level $m = 1$ reached.

4.2.6 Ship Subproblem

The subproblems are formulated as shortest path problems and solved by specific dynamic programming algorithms on generated networks for each ship and each port. In Christiansen (2005), a discretized load quantity interval is used to obtain easier structure of the subproblem. It is possible because of homogenous vehicle with same vehicle capacity is used. So it allows these variables to have a few discrete values. Meanwhile, it is not applicable here because of the heterogeneous vehicle with different vehicle capacity. The objective functions for the ship subproblem therefore become

$$\min \sum_{(i,m,j,n,v)} [(\mu_{ij} - D_{Vjn})x_{imjnv} - D_{Qim}Q_{vimv} - D_{Tjn}t_{Vjnv}] \quad (95)$$

The recursive function to minimize reduced cost (90) is used. If the quantity on board a ship at arrival (i, m) is Q_{vimv} and at arrival (j, n) is Q_{vjnv} , then the quantity loaded or discharged at (i, m) is $|Q_{vjnv} - Q_{vimv}|$ and the quantity on board increased or decreased from one visit to the next. The recursive functions for the ship subproblems therefore become: (Spliet, 2010)

$$F_{Sjnv}(t) = \min_{i,m} \left\{ \min_{0 \leq t' \leq t} \left\{ \mu_{ij} - D_{Vjn} - D_{Qim}|Q_{vjnv} - Q_{vimv}| \right. \right. \\ \left. \left. - D_{Tim}t' + F_{Simv}(t') \right\} \right\} \quad (96)$$

The first four terms in (95) represent the contribution at node (j, n, v), and the last term represents the recursive function for the previous node (i, m, v). With $F_{SEv}(t) = 0$ if (j, n) is the initial position of the ship. μ_{ij} is a fixed cost for travelling on every arc $(i,j) \in A_v$. The ship subproblem is now defined as:

$$\min_{(j,n)} F_{Sjnv}(TH) \quad (97)$$

4.3 Algorithm Development

The mathematical model that have been developed in subchapter 4.2 then transformed into programming language to be able to run in LINGO 18 optimization software. In this subchapter the algorithm of column generation for ISRP will be explained in detail and the programming code of this algorithm is attached in the attachment. Step by step of the algorithm are listed in the table below and the detailed information about each step will be further explained by using sample data.

Table 4. 1 Algorithm Column Generation for ISRP

Algorithm Column Generation for ISRP	
1	Create initial column for Restricted Master Problem
2	Repeat
3	Solve Restricted Master Problem
4	Copy dual prices to Ship and Port Subproblem
5	Solve Ship Subproblem
6	Update reduced cost for ship subproblem (RCS)
7	Solve Port Subproblem
8	Update reduced cost for port subproblem (RCP)
9	If RCS < 0 then add column to restricted master problem by ...
10	... updating $A(i,m,v,r)$, $Qv(i,m,v,r)$, $Tv(i,m,v,r)$ and cost $CMV(v,r)$
11	If RCP < 0 then add column to restricted master problem by ...
12	... updating $W(i,m,s)$, $Qp(i,m,s)$, $Tp(i,m,s)$ and cost $CMP(i,s)$
13	Until no negative RCS and RCP
14	Solve restricted master problem with integral requirement

4.3.1 Initialize Restricted Master Problem (RMP) and Subproblems

The first step is to initiate restricted master problem with initial column using two ships (ship 1 and 2) both of them assigned to route 1 ($r=1$) and two ports (port 1 and 2) are being visited for port sequence 1 ($s=1$). Voyage from port 3 to port 1 for ship 1 and from port 4 to port 2 for ship 2 obtained from initial column in restricted master problem ($X(3,1,1,1,1)$ and $X(4,1,2,1,2)$ equal to 1). This initial column is arbitrary may not be feasible. Because of that, Cmv and Cmp for initial column are made so that it has a very high cost to avoid being chosen in later iteration. Maximum number of route and port sequence being evaluated are both 2. Initial position of ship 1 and 2 are port 3 and port 4 respectively. The restricted master problem, ship subproblem, port subproblem and their extended version are shown below:

Restricted Master Problem

Minimize

$$\sum Cmv(v,r) * Y(v,r) + \sum Cmp(i,s) * Z(i,s)$$

Subject to:

$$\begin{aligned} \sum A(i,m,v,r) * Y(v,r) + \sum W(i,m,s) * Z(i,s) &= 1 \\ \sum Qv(i,m,v,r) * Y(v,r) - \sum Qp(i,m,s) * Z(i,s) &= 0 \\ \sum Tv(i,m,v,r) * Y(v,r) - \sum Tp(i,m,s) * Z(i,s) &= 0 \\ \sum Y(v,r) &= 1 \end{aligned}$$

$$\sum Z(i,s) = 1$$

Extended version:

Minimize

$$[Cmv(1,1)*Y(1,1) + Cmv(2,1)*Y(2,1)] + [Cmp(1,1)*Z(1,1) + Cmp(2,1)*Z(2,1)]$$

Subject to:

$$[A(1,1,1,1)*Y(1,1) + W(1,1,1)*Z(1,1)] = 1$$

$$[A(2,1,2,1)*Y(2,1) + W(2,1,1)*Z(2,1)] = 1$$

$$[Qv(1,1,1,1)*Y(1,1) - Qp(1,1,1)*Z(1,1)] = 0$$

$$[Qv(2,1,2,1)*Y(2,1) - Qp(2,1,1)*Z(2,1)] = 0$$

$$[Tv(1,1,1,1)*Y(1,1) - Tp(1,1,1)*Z(1,1)] = 0$$

$$[Tv(2,1,2,1)*Y(2,1) - Tp(2,1,1)*Z(2,1)] = 0$$

$$Y(1,1) = 1$$

$$Y(2,1) = 1$$

$$Z(1,1) = 1$$

$$Z(2,1) = 1$$

Initial Ship Subproblem given $Dv = 1$, $Dt = 1$ and $Dq = 1$

Minimize

$$\sum [\mu(i,j,v) - Dv(i,m)] * X(i,m,j,n,v) - Dq(i,m) * qv(i,m,v) + Dt(i,m) * tv(i,m,v)$$

Subject to:

Ship constraint (45)-(54)

Ship Dynamic Programming (96)-(97)

Extended Version:

$$[(\mu(3,j,1) - 1) * X(3,1,j,n,1) + (\mu(4,j,2) - 1) * X(4,1,j,n,2)] \\ + [-1 * qv(3,1,1) - 1 * qv(4,1,2)] + [-1 * tv(j,n,1) - 1 * tv(j,n,2)]$$

Subject to:

Ship constraint (45)-(54)

Ship Dynamic Programming (96)-(97)

The node (j,n) will be obtained after the ship subproblem has been solved.

Initial Port Subproblem given $Dv = 1$, $Dt = 1$, $Dq = 1$

Minimize

$$\sum [\pi(i,m) - Dv(i,m)] * yp(i,m) + Dq(i,m) * qp(i,m) + Dt(i,m) * tp(i,m)$$

Subject to:

Port constraint (34)-(35), (37), (42), (55)-(66)

Port Dynamic Programming (93)-(94)

Extended Version:

$$[(\pi(3) - 1) * Yp(3,1) + (\pi(4) - 1) * Yp(4,1)] + [(1 * Qp(3,1) + 1 * Qp(4,1)] \\ + [(1 * Tp(3,1) + 1 * Tp(4,1)]$$

Subject to:

Port constraint (34)-(35), (37), (42), (55)-(66)

Port Dynamic Programming (93)-(94)

4.3.2 Solve the LP-Relaxation of RMP and the Subproblems

The purpose of setting arbitrary initial column is to generate new dual value. By solving the initial restricted master problem, new value for Dv, Dq and Dt are obtained from dual value of constraint (80)-(82) respectively. New value for Dv, Dq and Dt are 2, -1, -1 respectively. New dual value of Dy and Dz are also obtained from constraint (83)-(84) with value of -1 and 1.

Ship Subproblem with New Value of Dv = 2 , Dq = -1 and Dt = -1**Minimize**

$$\sum [\mu(i,j) - Dv(i,m)] * X(i,m,j,n,v) - Dq(i,m) * qv(i,m,v) + Dt(i,m) * tv(i,m,v)$$

Extended Version:

$$[(\mu(3,j,1) - 2) * X(3,1,j,n,1) + (\mu(4,j,2) - 2) * X(4,1,j,n,2)] \\ + [1 * Qv(3,1,1) + 1 * Qv(4,1,2)] + [1 * Tv(j,n,1) + 1 * Tv(j,n,2)]$$

Subject to:

Ship constraint (45)-(54)

Ship Dynamic Programming (96)-(97)

The output of ship subproblem are $X(i,m,j,n,v)$, $qv(i,m,v)$ and $tv(j,n,v)$. The output consist of new voyage/arc for every ship and how many product quantity are being loaded/unloaded from port (i,m) for every ship and at what time every ship will arrive at the next port. Voyage from port 3 to port 2 for ship 1 and from port 4 to port 1 for ship 2 obtained from solving the ship subproblem ($X(3,1,2,1,1)$ and $X(4,1,1,1,2)$ equal to 1).

Port Subproblem with New Value of $Dv = 2$, $Dq = -1$ And $Dt = -1$

Minimize

$$\sum [\pi(i,m) - Dv(i,m)] * Yp(i,m) + Dq(i,m) * Qp(i,m) + Dt(i,m) * Tp(i,m)$$

Extended Version:

$$[(\pi(3) - 2) * Yp(3,1) + (\pi(4) - 2) * Yp(4,1)] + [(-1) * Qp(3,1) - 1 * Qp(4,1)] \\ + [(-1) * Tp(3,1) - 1 * Tp(4,1)]$$

Subject to:

Port constraint (34)-(35), (37), (42), (55)-(66)

Port Dynamic Programming (93)-(94)

The output of port subproblem are $Yp(i,m)$, $Qp(i,m)$ and $Tp(i,m)$. The output consist of a binary variable which equals 1 if port (i, m) is visited, the total product quantity being loaded/unloaded from port (i,m) from every ship and at what time port i being visited at m-th visit.

4.3.3 Check for negative reduced cost in Subproblems solutions

With updated values of Dt , Dq , Dv , Dy and Dz on every iteration (when solving restricted master problem), the values of all RCS and RCP will also be updated. The calculation of RCS and RCP from constraint (90)-(91) are shown below:

Reduced Cost Ship

$$\begin{aligned} \text{RCS}(v,r) = & \text{Cmv}(v,r) - \sum A(i,m,v,r)*Dv(i,m) - \sum Qv(i,m,v,r)*Dq(i,m) \\ & - \sum Tv(i,m,v,r)*Dt(i,m) - Dy(v) \end{aligned}$$

Extended Version:

$$\begin{aligned} \text{RCS}(1,1) = & \text{Cmv}(1,1) - \sum A(i,m,1,1)*2 - \sum Qv(i,m,1,1)*(-1) \\ & - \sum Tv(i,m,1,1)*(-1) - (-1) \end{aligned}$$

$$\begin{aligned} \text{RCS}(2,1) = & \text{Cmv}(2,1) - \sum A(i,m,2,1)*2 - \sum Qv(i,m,2,1)*(-1) \\ & - \sum Tv(i,m,2,1)*(-1) - (-1) \end{aligned}$$

Reduced Cost Port

$$\begin{aligned} \text{RCP}(i,s) = & \text{Cmp}(i,s) - \sum W(i,m,s)*Dv(i,m) - \sum Qp(i,m,s)*Dq(i,m) \\ & - \sum Tp(i,m,s)*Dt(i,m) - Dz(i) \end{aligned}$$

Extended Version:

$$\text{RCP}(3,1) = \text{Cmp}(3,1) - \sum W(i,m,1)*2 - \sum Qp(i,m,1)*(-1) - \sum Tp(i,m,1)*(-1) - 1$$

$$\text{RCP}(4,1) = \text{Cmp}(4,1) - \sum W(i,m,1)*2 - \sum Qp(i,m,1)*(-1) - \sum Tp(i,m,1)*(-1) - 1$$

If there is still any negative value of RCS and RCP, the iteration (looping) will continue. Any negative value of RCS and RCP indicate that there is still room for improvement (reducing master problem objective function if minimization case).

4.3.4 Adding new column to the RMP

The output of ship subproblem and port subproblem are added into restricted master problem as new column of route (r) and port sequence(s). The new column r=2 and s=2 are established and related variables that being updated are listed below:

$$X(i,m,j,n,v,2) = X(i,m,j,n,v)$$

Extended version:

$$X(3,1,2,1,1,2) = X(3,1,2,1,1) = 1$$

$$X(4,1,1,1,2,2) = X(4,1,1,1,2) = 1$$

$$A(j,n,v,2) = X(i,m,j,n,v)$$

Extended version:

$$A(2,1,1,2) = X(3,1,2,1,1) = 1$$

$$A(1,1,2,2) = X(4,1,1,1,2) = 1$$

$$Qv(i,m,v,2) = qv(i,m,v)$$

$$Tv(i,m,v,2) = tv(i,m,v)$$

$$W(i,m,2) = yp(i,m)$$

$$Qp(i,m,2) = qp(i,m)$$

$$Tp(i,m,2) = tp(i,m)$$

The cost of new route and new port sequence, $Cmv(v,2)$ and $Cmp(i,2)$, are being updated as well. Cms represent the transportation cost for sailing schedule r by ship v . While Cmp represent the total cost associated with port i (setup and loading/unloading cost) while being visited on port sequence s .

4.3.5 Re-optimize the RMP

The newly updated RMP with added column mark the beginning of new iteration in the looping mechanism. The newly re-optimized RMP will give new value to Dt , Dv , Dq , Dy and Dz that will be used for solving both ship and port subproblem. The extended version of newly updated RMP is shown below:

Updated Restricted Master Problem Extended Version:

Minimize

$$[Cmv(1,1)*Y(1,1) + Cmv(2,1)*Y(2,1) + Cmv(1,2)*Y(1,2) + Cmv(2,2)*Y(2,2)] \\ + [Cmp(1,1)*Z(1,1) + Cmp(2,1)*Z(2,1) + Cmp(1,2)*Z(1,2) + Cmp(2,2)*Z(2,2)]$$

Subject to:

$$[A(1,1,1,1)*Y(1,1) + A(1,1,2,2)*Y(2,2)] + [W(1,1,1)*Z(1,1) + W(1,1,2)*Z(1,2)] = 1 \\ [A(2,1,2,1)*Y(2,1) + A(2,1,1,2)*Y(1,2)] + [W(2,1,1)*Z(2,1) + W(2,1,2)*Z(2,2)] = 1 \\ [Qv(1,1,1,1)*Y(1,1) + Qv(1,1,2,2)*Y(2,2)] - [Qp(1,1,1)*Z(1,1) + Qp(1,1,2)*Y(1,2)] = 0 \\ [Qv(2,1,2,1)*Y(2,1) + Qv(2,1,1,2)*Y(1,2)] - [Qp(2,1,1)*Z(2,1) + Qp(2,1,2)*Z(2,2)] = 0 \\ [Tv(1,1,1,1)*Y(1,1) + Tv(1,1,2,2)*Y(2,2)] - [Tp(1,1,1)*Z(1,1) + Tp(1,1,2)*Y(1,2)] = 0 \\ [Tv(2,1,2,1)*Y(2,1) + Tv(2,1,1,2)*Y(1,2)] - [Tp(2,1,1)*Z(2,1) + Tp(2,1,2)*Z(2,2)] = 0$$

$$Y(1,1) + Y(1,2) = 1$$

$$Y(2,1) + Y(2,2) = 1$$

$$Z(1,1) + Z(1,2) = 1$$

$$Z(2,1) + Z(2,2) = 1$$

After the RMP being updated, the algorithm will proceed to the next step which is solve the ship and port subproblem. But because the number of route and port sequence already reach maximum number, the looping mechanism will break and proceed to solve the master problem.

4.3.6 Obtain optimal solution for Master Problem

The algorithm will continue until there are no negative RCS and RCP values left to reduce the cost or the number of route and port sequence already reach maximum number, then the algorithm will exit the loop and solve the master problem instead of restricted master problem. Master problem will find which route (r) and port sequence (s) that led to least total cost. Because there are only two route and port sequence and the first one is intentionally avoided (high cost), the solution of master problem will be to choose second route (r=2) and port sequence (s=2). Considering $X(3,1,2,1,1)$ and $X(4,1,1,1,2)$ are both equal to 1, the port that are being visited are port 2 and port 1. The result of this example consists of several variables are shown below:

Objective function value

$$= [Cmv(1,1)*Y(1,1) + Cmv(2,1)*Y(2,1) + Cmv(1,2)*Y(1,2) + Cmv(2,2)*Y(2,2)]$$

$$+ [Cmp(1,1)*Z(1,1) + Cmp(2,1)*Z(2,1) + Cmp(1,2)*Z(1,2) + Cmp(2,2)*Z(2,2)]$$

$$= [Cmv(1,1)*0 + Cmv(2,1)*0 + Cmv(1,2)*1 + Cmv(2,2)*1]$$

$$+ [Cmp(1,1)*0 + Cmp(2,1)*0 + Cmp(1,2)*1 + Cmp(2,2)*1]$$

Constraint Testing

(C83a)

$$Y(1,1) + Y(1,2) = 1 \quad Y(1,1) = 0 ; Y(1,2) = 1$$

(C83b)

$$Y(2,1) + Y(2,2) = 1 \quad Y(2,1) = 0 ; Y(2,2) = 1$$

(C84a)

$$Z(1,1) + Z(1,2) = 1 \quad Z(1,1) = 0 ; Z(1,2) = 1$$

(C84b)

$$Z(2,1) + Z(2,2) = 1 \quad Z(2,1) = 0 ; Z(2,2) = 1$$

(C80a)

$$[A(1,1,1,1)*Y(1,1) + A(1,1,2,2)*Y(2,2)] + [W(1,1,1)*Z(1,1) + W(1,1,2)*Z(1,2)] = 1$$
$$A(2,1,1,2) = 1 ; W(1,1,2) = 0$$

(C80b)

$$[A(2,1,2,1)*Y(2,1) + A(2,1,1,2)*Y(1,2)] + [W(2,1,1)*Z(2,1) + W(2,1,2)*Z(2,2)] = 1$$
$$A(2,1,1,2) = 1 ; W(2,1,2) = 0$$

(C81a)

$$[Q_v(1,1,1,1)*Y(1,1) + Q_v(1,1,2,2)*Y(2,2)] - [Q_p(1,1,1)*Z(1,1) + Q_p(1,1,2)*Y(1,2)] = 0$$
$$Q_v(1,1,2,2) = Q_p(1,1,2)$$

(C81b)

$$[Q_v(2,1,2,1)*Y(2,1) + Q_v(2,1,1,2)*Y(1,2)] - [Q_p(2,1,1)*Z(2,1) + Q_p(2,1,2)*Z(2,2)] = 0$$
$$Q_v(2,1,1,2) = Q_p(2,1,2)$$

(C82a)

$$[T_v(1,1,1,1)*Y(1,1) + T_v(1,1,2,2)*Y(2,2)] - [T_p(1,1,1)*Z(1,1) + T_p(1,1,2)*Y(1,2)] = 0$$
$$T_v(1,1,2,2) = T_p(1,1,2)$$

(C82b)

$$[T_v(2,1,2,1)*Y(2,1) + T_v(2,1,1,2)*Y(1,2)] - [T_p(2,1,1)*Z(2,1) + T_p(2,1,2)*Z(2,2)] = 0$$
$$T_v(2,1,1,2) = T_p(2,1,2)$$

From this, the solution from master problem is obtained. But this solution is in the form of path flow model which contain the information of route and port sequence. To convert the solution into the original arc flow variable, the solution can be substituted into constraint (67), (69), (70), (73), (75) and (76) from Dantzig-Wolfe decomposition.

(C67)

$$X(i,m,j,n,v) = \sum X(i,m,j,n,v,r)*Y(v,r)$$
$$X(3,1,2,1,1) = X(3,1,2,1,1,2)*Y(1,2) = 1$$
$$X(4,1,1,1,2) = X(4,1,1,1,2,2)*Y(2,2) = 1$$

(C69)

$$qv(i,m,v) = \sum Qv(i,m,v,r) * Y(v,r)$$

$$qv(2,1,1) = Qv(2,1,1,2) * Y(1,2)$$

$$qv(1,1,2) = Qv(1,1,2,2) * Y(2,2)$$

(C70)

$$tv(i,m,v) = \sum Tv(i,m,v,r) * Y(v,r)$$

$$tv(2,1,1) = Tv(2,1,1,2) * Y(1,2)$$

$$tv(1,1,2) = Tv(1,1,2,2) * Y(2,2)$$

(C73)

$$y(i,m) = \sum W(i,m,s) * Z(i,s)$$

$$y(1,1) = W(1,1,2) * Z(1,2) = 0$$

$$y(2,1) = W(2,1,2) * Z(2,2) = 0$$

(C75)

$$qp(i,m) = \sum Qp(i,m,s) * Z(i,s)$$

$$qp(1,1) = Qp(1,1,2) * Z(1,2)$$

$$qp(2,1) = Qp(2,1,2) * Z(2,2)$$

(C76)

$$tp(i,m) = \sum Tp(i,m,s) * Z(i,s)$$

$$tp(1,1) = Tp(1,1,2) * Z(1,2)$$

$$tp(2,1) = Tp(2,1,2) * Z(2,2)$$

Because of constraint (34) and (35), qp(i,m) will equal to qv(i,m,v) from all ship that visit port i at arrival m-th and because only ship 1 that visit port 2 at 1st arrival then qp(2,1) = qv(2,1,1). The same thing also applies to tp(i,m) and tv(i,mv).

(C34)

$$tp(I,m) - \sum tv = 0$$

$$tp(1,1) = tv(1,1,2)$$

$$tp(2,1) = tv(2,1,1)$$

(C35)

$$qp(I,m) - \sum qv = 0$$

$$qp(1,1) = qv(1,1,2)$$

$$qp(2,1) = qv(2,1,1)$$

After being substitute, the results are made into its original variables. The optimal solution can be interpreted as:

- 1) Ship 1 sailing from port 3 at 1st arrival to port 2 at 1st arrival and arrived at day $tp(2,1)$ to unload product as much as $qp(2,1)$.
- 2) Ship 2 sailing from port 4 at 1st arrival to port 1 at 1st arrival and arrived at day $tp(1,1)$ to unload product as much as $qp(1,1)$.
- 3) With total cost equal to $[Cmv(1,2) + Cmv(2,2)] + [Cmp(1,2) + Cmp(2,2)]$.

CHAPTER V

COLUMN GENERATION SCHEMES

In this chapter will be explained about dual value analysis, column generation challenges and column generation improvement schemes such as intensification, diversification, and stabilization.

5.1 Column Generation on Large Dataset ISRP

Due to one reason or another, the MILP with column generation approach algorithm developed in this study has not been able prove its robustness and not yet validated. Therefore, this study wasn't able to show numerical test for ISRP dataset and compare the computational result from of different types of dataset. As an alternative to check whether column generation approach has advantages when dealing with large dataset in terms of computing speed, several pieces of researches about ISRP that evaluate the variation of dataset size were compared. As shown in Table 5.1, the average computing time for each dataset is different.

The decision expected from inventory ship routing optimization are considered for operational level. Because of that, it is assumed that acceptable average computing time for operational level decision maker is no more than two hour with consideration of small dataset and less than a month of planning horizon. In Table 5.1, dataset variation that need more than two hours to solve were highlighted and it is assumed that the approach being used is not preferred to solve that dataset. From Table 5.1 the information on how large a dataset for MILP can handle to produce acceptable average computing time can be devised. It shows that MILP are suitable for ISRP with dataset of not more than 2 products, 3 ships, and 6 ports. Beyond that, MILP with column generation is preferred especially when dealing with long planning horizon.

Table 5.1 Time Comparison between MILP and Column Generation

Author	Approach	P	S	Port		PH (days)	Computin g Time (s)	Computin g Time (hour)
				D	S			
Ramkumar (2012)	MILP	1	2	3	1	21	0.1	0.0
		2	2	3	2	21	3142.0	0.9
		3	3	5	3	21	28800.0	8.0
		4	3	6	4	21	28800.0	8.0
Yongheng (2015)		1	2	2	1	60	55.0	0.0
		1	3	4	2	70	1144.0	0.3
		1	3	2	4	70	4053.0	1.1
		1	6	7	3	80	36000.0	10.0
		1	3	4	2	70	15084.0	4.2
		1	3	2	4	70	17452.0	4.8
Siswanto et.al (2011),		2	2	2		15	3139.0	0.9
		2	3	4		15	7049.0	2.0
Nemhauser (2010)		MILP with Column Generatio n	1	4	2	2	60	11.0
	1		5	3	2	60	908.0	0.3
	1		5	2	3	60	1199.0	0.3
	1		5	3	3	60	22092.0	6.1
	1		6	4	3	60	31844.0	8.8
	1		6	3	4	60	36000.0	10.0
	1		6	4	4	60	36000.0	10.0
	1		6	6	4	60	28722.0	8.0
	1		6	4	6	60	32404.0	9.0
Christiansen (1998a)	1		3	11		36	993	0.27
Christiansen (1998b)	1	5	16		36	5455.0	1.5	

The MILP with column generation research that were being compared are those who shared similar ISRP characteristic such as single product and no compartment. From Table 5.1, it can be seen that the duration of planning horizon also affecting the average computing time. With planning horizon of 60 days, MILP with column generation can handle up to 5 ships and 5 ports with ease, but start to take significantly more time when dealing with more than 5 ships and 5 ports. But it doesn't mean that MILP with column generation can't handle large dataset. It can be seen from Christiansen (1998a) and (1998b) MILP with column generation can handle more than 5 ports if the duration of planning horizon is

reduced. Because of that, it is advised to shorten the planning horizon when dealing with large dataset to achieve acceptable average computing time.

5.2 Dual Value Analysis and Column Generation Challenges

In column generation, dual value is one of the most important elements with dual value alone the performance of column generation can be both hugely improved or worse endured slow convergence. Because of that, several improvements for column generation focused on how to deal with dual value to avoid its biggest drawback and even improving its performance which will be discussed in detail in the next subchapter.

Dual value itself directly related to master problem, ship subproblem, port subproblem and dynamic programming (for the shortest route) in this study. Therefore, understanding how changes in dual value may affect the other component of column generation is also important. In this subchapter, the analysis and insight about dual value will be discussed.

In dynamic programming for both ship and port, one of the outputs is about what time the ship departs or arrived at a certain port. Among the dual values, dual value of D_t (from constraint 82) has the most significant effect on the result related to ship departure or arrival at a certain port. The insight gained in this study, if D_t value of a certain port is negative then dynamic programming for port suggest to visit that port as late as possible. If there is any time window on port, the negative value of D_t makes the visit at near the end of time window more attractive (has the least RC). Whereas it applies to the opposite, if D_t value is positive then dynamic programming for ship suggest to visit that port as soon as possible (least RC). The effect are opposite for both dynamic programming for ship and port if the D_t value changes sign.

The other output of dynamic programming is how to choose the next port to be visited. Dual value that affects this decision is both D_v and D_q . The interesting part is they has exactly polar opposite effect. The higher (positive) the value of D_v on a port make the other ports less attractive to visit (their reduced cost are worsen). But the smaller (negative) the value of D_v make other ports

more attractive (their reduced costs are improving). Despite that, the change of D_v value has absolutely no effect on the reduced cost of the said port (no changing).

And for the dual value of D_q , the effect is exactly the same with D_v but in reverse. But important side note for dual value of D_q , the effect of D_q value will be amplified by the quantity of product that are being loaded (or unloaded) into (or from) of ship v at node (i, m) . To be more specific are the value of $q(I, m)$ and $q_v(I, m, v)$. And also, if the quantity uploaded / loaded on the destination port is greater than that of the origin port then it strengthens the effect of D_q proportionally.

And for the challenges and difficulties to develop column generation approach in this study can be listed down below:

- 1) There are little to no information on the internet on what is the constraints for each ship and port subproblem. This may happen due to the very specific problem specification and information on each published journal on column generation (problem specific column generation algorithm).
- 2) The restricted master problem that have been developed can't generate dual value (D_t , D_v , D_q) that are required to proceed to next process (solving the subproblem and dynamic programming). This may happen due to the dimension of dual value in this study. Meanwhile the algorithm which is the reference for coding the restricted master problem, its dual value has only one dimension.
- 3) The lack of understanding on how the dynamic programming for both ship and port can contribute to the ship and port subproblem. When the output of dynamic programming are put into subproblem's objective function, the subproblem became infeasible. This may happen due to the lack of understanding on how to develop mathematical model for both dynamic programming because the references being used offers many variation of dynamic programming model.

5.3 Intensification and Stabilization

Column generation is known to suffer from tailing off , that is, there is only incremental progress per iteration as it get closer to the optimum, in

particular, for large and degenerate problems. The main reason lies in the unstable behavior of the dual variables. A dual solution may be far apart from the previous one called bang-bang effect (Lübbecke (2011)). Stabilization of the dual variables tries to reduce this effect and aim at computing “good” dual solutions, close to the best current one. In the case that stabilization is successful, regardless of the method employed, one typically observes a reduction in the number of column generation iterations. The downside of it is that the pricing problems become harder to solve on an average. However, among more sophisticated implementation techniques, stabilization may provide the largest performance gains. The popular methods which operate on dual space and aim at stabilizing dual variable behavior are interior point stabilization and bundle method.

In Interior Point Stabilization, a simplex-method-based approach to obtain a solution in the interior of the dual-optimal face is taken. It works in two steps and exploits the extremity of basic solutions. First, the RMP is solved and the objective function value is fixed to the optimum by adding an additional constraint. Then, several random objective functions c are chosen (and also the opposite direction $-c$), each of which produces an extreme point of the optimal face. The final dual solution is a convex combination of all extreme points obtained. This approach is computationally expensive but easy to implement (L’etocart, 2010).

The other method is Bundle Methods also known as Quadratic Penalty Term. The aim of bundle method is to encourage a dual solution to stay close to the stability center; so the penalty is larger the further away it goes thus named penalty term (Lübbecke (2011)). Pictorially, a quadratic penalty function can achieve this goal better than a piecewise linear penalty, and bundle methods do precisely this: penalizing the Euclidean distance to the stability center.

Another more conventional way to reduce the number of iterations in practice is adding several columns to the RMP at each iteration, corresponding to solutions with negative reduced cost (minimization problem case), also including not-optimal solutions. This method, intensification, allows the Master Problem (MP) approximation to be improved, an optimal basis to be characterized more quickly, and hence to decrease the number of iterations (L’etocart, 2010).

Unfortunately, this can considerably expand the RMP, when the final optimal base contains a very restricted number of the generated columns. The set of columns of negative reduced cost to be added to the master problem can deeply affect the overall number of generated variables and the computing time required to find an optimal solution.

To avoid the rapid and needless increase in the master problem size, limiting the generation of the k best solutions is needed, k being a parameter to be determined and expressed in %. This procedure called $k\%$ Intensified Column Generation (L'etocart, 2010). The column number is reduced when k decreases, whereas the iteration number and the resolution time increase in most instances. The addition of columns with good reduced cost is not enough to improve the computing time required to find an optimal basis. On average, this percentage increases when k increases, so suboptimal solutions with good reduced cost contribute less to the final optimal basis than those with a worse reduced cost.

5.4 Diversification

When stabilization aims at computing a good dual function local approximation around the best dual solution found, diversification aims at constructing a good dual function global approximation. The diversification procedure consists in inserting a set of 0-neighbor columns into the master problem at each column generation iteration. These procedures can be more efficient on the first iterations to quickly characterize a good approximation and useless on the last ones. Thus, diversification may be applied only on the first iterations. The following two procedures are for computing 0-neighbor columns at an iteration of column generation from L'etocart (2010).

1) Diversification by Resolution (CGDR)

At each column generation iteration with the same master problem dual variables, Diversification by Resolution (CGDR) consists of iteratively computing a 0-neighbor solution compared to all generated columns with optimal reduced cost. The picture below showing the algorithm of diversification by resolution.

Algorithm 1. Schematic iteration of CG with diversification by resolution

```

 $\mathcal{S} \leftarrow$  set of all pricing problem's feasible solutions
 $\mathcal{C} \leftarrow \emptyset$  {set of columns to add to the MP}, next  $\leftarrow$  true
repeat
   $x^k \leftarrow$  a solution of minimum cost in  $\mathcal{S}$ 
  if  $x^k$  has negative cost then
     $\mathcal{C} \leftarrow \mathcal{C} \cup \{x^k\}$ 
     $\mathcal{S} \leftarrow \mathcal{S} \setminus \{\text{solutions contributing to the same constraints than } x^k\}$ 
    if  $\mathcal{S} \neq \emptyset$  then  $k \leftarrow k + 1$ ,
    else next  $\leftarrow$  false
    end if
  else
    next  $\leftarrow$  false.
  end if
until (next = false)

```

Figure 5. 1 Algorithm of Diversification By Resolution

2) Diversification by Selection (CGDS)

Diversification by Selection (CGDS) is consist in selecting 0-neighbor columns among all the negative reduced cost solutions computed at each iteration of column generation. This technique can lead to the computation of very few columns at each iteration, which include many poor reduced cost ones compared to those generated using the CGDR approach. The picture below showing the algorithm of diversification by selection.

Algorithm 2. Schematic iteration of CG with diversification by selection

```

 $\mathcal{C} \leftarrow \emptyset$  {set of columns to add to the MP}
Solve the pricing problem
 $X \leftarrow$  all solutions with negative reduced cost
while  $X \neq \emptyset$  do
   $\hat{x} \leftarrow$  minimal cost solution in  $X$ ,  $\mathcal{C} \leftarrow \mathcal{C} \cup \{\hat{x}\}$ 
   $X \leftarrow X \setminus \{\text{solutions contributing to at least one same constraint than } \hat{x}\}$ 
end while

```

Figure 5. 2 Algorithm of Diversification By Selection

This page was intentionally left blank

CHAPTER VI

CONCLUSION & SUGGESTION

This chapter contains the conclusions of the research results and suggestions relating to further research.

6.1 Conclusion

From the results of the research and analysis that has been carried out, several conclusions can be drawn as follows:

1. In this study, a mathematical model for inventory ship routing problem using mixed integer linear programming with column generation approach was successfully developed.
2. The algorithm for mixed integer linear programming using column generation approach using LINGO optimization software was not been able to be applied to solve inventory ship routing problem.
3. There were several challenges and difficulties faced during the development of both mathematical model and software algorithm in which may useful or insightful for further research about related topic.

6.2 Suggestion

Suggestions that can be given to further research are as follows:

1. To use and utilize the built-in function of LINGO 18 called branch-and-price (BNP) solver. The BNP solver is a mixed integer programming solver for solving linear models with block structure. Based on the decomposition structure, the solver divides the original problem into several subproblems and solves them (almost) independently, exploiting parallel processing if multiple cores are available.
2. Search for journals that explained or have examples for algorithm about restricted master problem that has two-dimension dual values.
3. Search or build a robust dynamic programming module for both ship and port subproblems.

4. To try some package that provides an efficient simplex algorithm. There are plenty of packages available, both commercial and open-source, such as CLP, GLPK , and Soplex. The situation is a little different when doing branch-and-price, but there are several frameworks that support its implementation (and thus in particular column generation) such as ABACUS, BCP, SCIP, and SYMPHONY.

REFERENCES

- A. Vigtil, "Information exchange in vendor managed inventory," *Int. J. Phys. Distrib. Logist. Manag.*, 2007
- Andersson, H., A. Hoff, M. Christiansen, G. Hasle, A. Løkketangen. 2010. Industrial aspects and literature survey: Combined inventory management and routing. *Comput. Oper. Res.* 37(9) 1515–1536.
- Bertazzi, L. and Speranza, G., 2012. *Inventory Routing Problems: An Introduction*. Springer-Verlag Berlin Heidelberg and EURO - The Association of European Operational Research Societies 2012.
- Budi Santosa, Rita Damayanti & Biswajit Sarkar (2016) Solving multi-product inventory ship routing with a heterogeneous fleet model using a hybrid cross entropy-genetic algorithm: a case study in Indonesia, *Production & Manufacturing Research*, 4:1,90-113,
- Christiansen, M. (1999). Decomposition of a combined inventory and time constrained ship routing problem. *Transportation Science*, 33(1):3-16.
- Christiansen, M. and Nygreen, B., 1998a. A method for solving ship routing problems with inventory constraints. *Annals of Operations Research*, 81, pp.357-378.
- Christiansen, M. and Nygreen. B. (1998b). Modeling path flows for a combined ship routing and inventory management problem. *Annals of Operations Research*, 82:391-412.
- Christiansen, M. and Nygreen, B., 2005. Robust Inventory Ship Routing by Column Generation. *Column Generation*, pp.197-224.
- Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D., 2007, Maritime Transportation. In: C. Barhart, & G. Laporte (Eds.), *Handbook in OR & MS* (Vol. 14, pp. 189-284). Berlin: Elsevier.
- Christiansen, M., & Fagerholt, K. (2009). Maritime inventory routing problems. In C. A. Floudas & P. M. Pardalos (Eds.), *Encyclopedia of optimization* (pp. 1947–1955). US: Springer.
- Dantzig G.B. and Wolfe P. (1960). Decomposition principle for linear programs. *Operations Research* 8, 101–111.

- Desaulniers, G., Desrosiers, J., Solomon, M. (2005). : Column Generation. Springer, Berlin
- Hariato, Ketut Hendra (2015) Pengembangan Simple Iterative Mutation Algorithm (SIM-A) Untuk Menyelesaikan Permasalahan Ship Scheduling And Assignment (Studi Kasus : Distribusi Semen Curah Pada PT. X). Undergraduate thesis, Institut Technology Sepuluh Nopember.
- Hwang, S. J. (2007). Discrete optimization – Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk, Part i: Applications and model. *European Journal of Operational Research*, 176, 106–130.
- INSA, 2019. Pelayaran Nasional Belum Tumbuh Cemerlang. [online] insa.or.id. Available at: <<https://insa.or.id/pelayaran-nasional-belum-tumbuh-cemerlang/>>.
- Jiang, Y. and Grossmann, I., 2015. Alternative mixed-integer linear programming models of a maritime inventory routing problem. *Computers & Chemical Engineering*, 77, pp.147-161.
- L'etocart, L., 2010. Solutions diversification in a column generation algorithm. *Algorithmic Operations Research*, 5(86–95).
- Lübbecke, M. and Desrosiers, J., 2005. Selected Topics in Column Generation. *Operations Research*, 53(6), pp.1007-1023.
- Lübbecke, M., 2011. Column Generation. *Wiley Encyclopedia of Operations Research and Management Science*,.
- Munari, P. and Gondzio, J., 2015. Column generation and branch-and-price with interior point methods. *Proceeding Series of the Brazilian Society of Applied and Computational Mathematics*, 3.
- N. Touati Moun gla, L. L'etocart, and A. Nagih, 2010. Solutions diversification in a column generation algorithm. *Algorithmic Operations Research*, vol. 5, no. 2, pp. 86–95.
- Nemhauser, G., 2010. Column Generation for Linear and Integer Programming. *Documenta Mathematica · Extra Volume ISMP (2012)* 65–73,.
- P.C. Gilmore and R.E. Gomory (1961). A linear programming approach to the cutting-stock problem. *Operations Research* 9, 849–859.

- PT Semen Tonasa, 2019. Annual Report 2018 - Turning Cost Into Higher Profit. [online] Available at: <<http://www.sementonasa.co.id/dokumen/AR%20Semen%20Tonasa%202018-merged.pdf>>
- Ramkumar, N., 2012. Mixed integer linear programming model for multi-commodity multi-depot inventory routing problem. *OPSEARCH*, 49(4), pp.413-429.
- Rani, Fitri Karunia.(2010). Model Multi Product Inventory Routing Problem Kapal Tanker Dengan Mempertimbangkan Faktor Batasan Kompatibilitas Dalam Pemuatan Produk.Thesis. Institut Teknologi Sepuluh Nopember.
- Rani, Fitri Karunia.(2010). Model Multi Product Inventory Routing Problem Kapal Tanker Dengan Mempertimbangkan Faktor Batasan Kompatibilitas Dalam Pemuatan Produk.Thesis. Institut Teknologi Sepuluh Nopember.
- Rusdianto, M., 2019. OPTIMASI PENJADWALAN DAN PENENTUAN RUTE TRANSPORTASI LAUT SEMEN CURAH (STUDI KASUS: PT. SEMEN INDONESIA REGIONAL III). Surabaya.
- Semen Indonesia Group, 2019. Annual Report 2018 - Accelerated Transformation. [online] Available at: <<http://semenindonesia.com/wp-content/uploads/2020/02/Annual-Report-2018.pdf>>

- Sinha, D., 2019. *Global Supply Chains And Multimodal Logistics: Emerging Research And Opportunities*. p.171.
- Siswanto, N., Essam, D., & Sarker, R. (2011). Solving the Ship Inventory Routing and Scheduling Problem with Undedicated Compartments. *Computers & Industrial Engineering* .
- Spliet, R., 2010. A Branch-and-Price Approach for a Ship Routing Problem with Multiple Products and Inventory Constraints. *Econometric Institute Report EI2010-05*,.
- UNCTAD, 2019, United Nations Conference on Trade And Development, Review of maritime transport 2019.
- World Bank, 2016. Evaluating The Shift In Incoterms For Indonesian Export Products. Surabaya: PT. ITS Kemitraan.
- Yeh, Y.L. and Low, C.Y. (2017) Mathematical Modelling for a Multi-Product Inventory Routing Problem with Split Delivery. *Journal of Applied Mathematics and Physics*, 5, 1607-1612.
- Z. Sui, A. Gosavi, and L. Lin, “A reinforcement learning approach for inventory replenishment in vendor-managed inventory systems with consignment inventory,” *EMJ - Eng. Manag. J.*, 2010.

ATTACHMENT

LINGO CODE

```
MODEL:
SETS:
Ship/1..2/:CAP,QQ,Dy;
Port/1..4/:Dz,Je,Ri,ST,TQ,IL,SMX,SC;
ArrivalNo/1..3/;;
Route/1..3/;;
PortSeq/1..3/;;
Time/1..10/:Day;
VR(Ship,Route):CMV,yv,RCS;
IM(Port,ArrivalNo):Dv,Dq,Dt,yp,qp,tp,sp,Fsmin,FpTH;
IMT(Port,ArrivalNo,Time):Fp;
IMV(Port,ArrivalNo,Ship):qv,tv,zv,wv,FsTH,L;
IMVT(Port,ArrivalNo,Ship,Time):Fs;
IMVR(Port,ArrivalNo,Ship,Route):A,Qvcg,Tvcg;
IMS(Port,ArrivalNo,PortSeq):W,Qpcg,Tpcg;
IMJNV(Port,ArrivalNo,Port,ArrivalNo,Ship):x;
IMJN(Port,ArrivalNo,Port,ArrivalNo):;
IMJNVR(Port,ArrivalNo,Port,ArrivalNo,Ship,Route): Xcg;
IJV(Port,Port,Ship): TT, mu;
IS(Port,PortSeq):CMP,z,RCP;

!Pair of port and arrival number including initial position;
SA(Port,ArrivalNo):; !SA=ST;
!Pair of port and arrival number without initial position;
SN(Port,ArrivalNo):;
!Initial position of ship V for the M-th arrival to harbor I;
SO(Port,ArrivalNo,Ship):;
!CAP = vehicle capacity
QQ = vehicle initial capacity
Dy = dual value for each ship
Dz = dual value for each port
Je = 1 for supply port -1 for demand port
Ri = production/consumption rate
ST = setup time (setup port + setup loading/unloading)
TQ = loading/unloading time per unit product
IL = initial storage in port
SMX = max capacity of port
SC = setup cost (port setup + loading/unloading setup)
CMV = sailing cost for route r using ship v
yv = decision variable for route r using ship v
RCS = reduced cost for each ship-route combination
Dv = dual value for constraint 80
Dq = dual value for constraint 81
Dt = dual value for constraint 82
yp = decision variable for port being visited at certain
portsequence
qp = product quantity being loaded/unloaded at i,m
tp = day at i,m
sp = storage level at i,m
Fsmin = the least reduced cost of ship dynamic programming
FpTH = the least reduced cost of port dynamic programming
Fp = reduced cost of port dynamic programming
qv = product quantity being loaded/unloaded by ship at i,m
tv = day when ship depart at i,m
zv = decision variable for port i at portsequence s
```

```

wv = decision variable for port i at portsequence s
FsTH = reduced cost of ship dynamic programming
L = quantity onboard loaded ship v after departing from node i,m
Fs = reduced cost of ship dynamic programming
A = decision variable i,m,v,r
Qvcg = quantity i,m,v,r
Tvcg = day at i,m,v,r
W = decision variable at i,s 0 = visited
Qpcg = quantity i,m,s
Tpcg = day at i,m,s
x = decision variable i,m,j,n,v
Xcg = decision variable i,m,j,n,v,r
TT = travelling time
mu = transportation cost from i to j
CMP = total cost from usage of port i at portsequence s
z = ship v finishes its route at node i,m
RCP = reduced cost for each port-portsequence combination;
ENDSETS

SUBMODEL MASTER_PROB:
[MSTROBJ] min = @sum(VR(v,r):CMV(v,r)*yv(v,r)) +
@sum(IS(i,s):CMP(i,s)*z(i,s));
!Subject to;
!constraint 80;
@for(Port(i):
    @for(ArrivalNo(m):[R_Sche]
@sum(VR(v,r):A(i,m,v,r)*yv(v,r))+@sum(PortSeq(s):W(i,m,s)*z(i,s))=
1));
!constraint 81;
@for(Port(i):
    @for(ArrivalNo(m):[R_Quant]
@sum(VR(v,r):Qvcg(i,m,v,r)*yv(v,r))-
@sum(PortSeq(s):Qpcg(i,m,s)*z(i,s))=0));
!constraint 82;
@for(Port(i):
    @for(ArrivalNo(m):[R_Time]
@sum(VR(v,r):Tvcg(i,m,v,r)*yv(v,r))-
@sum(PortSeq(s):Tpcg(i,m,s)*z(i,s))=0));
!constraint 83;
@for(Ship(v):[R_Ship] @sum(Route(j):yv(v,j))=1);
!constraint 84;
@for(Port(i):[R_Port] @sum(PortSeq(j):z(i,j))=1);
!constraint 87;
@for(IMJNVR(i,m,j,n,v,r):@bin(@sum(Route(r):Xcg(i,m,j,n,v,r)*yv(v,
r))));
!integer requirement;
@for(IS(i,s):@bin(z));
@for(IMVR(i,m,v,r): @bin(A));
@for(IMS(i,m,s): @bin(W));

!Dantzig-Wolfe Decomposition Constraints;
!ensure ship v to have only one predecessor port before port
arrival (i,m);
@for(IMVR(i,m,v,r):A(i,m,v,r)=
@sum(IMJNVR(j,n,i,m,v,r):Xcg(j,n,i,m,v,r)));
!constraint 34;
@for(IM(i,m):tp(i,m)=@sum(Ship(v):tv(i,m,v)));
!constraint 35;
@for(IM(i,m):qp(i,m)=@sum(Ship(v):qv(i,m,v)));

```

```

!constraint 67;
@for (IMJNVR(i,m,j,n,v,r):x(i,m,j,n,v) =
@sum(Route(r):Xcg(i,m,j,n,v,r)*yv(v,r)));
!constraint 69;
@for (IMV(i,m,v):qv(i,m,v) = @sum(Route(r):Qvcg(i,m,v,r)*yv(v,r)));
!constraint 70;
@for (IMV(i,m,v):tv(i,m,v) = @sum(Route(r):Tvcg(i,m,v,r)*yv(v,r)));
!constraint 71;
@for (Ship(v):@sum(Route(r):yv(v,r))=1);
!constraint 72;
@FOR (VR(v,r):@GIN(yv(v,r)));
!constraint 73;
@for (IMS(i,m,s):yp(i,m) = @sum(PortSeq(s):W(i,m,s)*z(i,s)));
!constraint 75;
@for (IM(i,m):qp(i,m) = @sum(PortSeq(s):Qpcg(i,m,s)*z(i,s)));
!constraint 76;
@for (IM(i,m):tp(i,m) = @sum(PortSeq(s):Tpcg(i,m,s)*z(i,s)));
!constraint 77;
@for (Port(i):@sum(PortSeq(s):z(i,s))=1);
!constraint 78;
@FOR (IS(i,s):@GIN(z(i,s)));

ENDSUBMODEL

SUBMODEL RESTRICTED_MASTER_PROBLEM:
[R_MSTROBJ] min = @sum(VR(v,r):CMV(v,r)*yv(v,r)) +
@sum(IS(i,s):CMP(i,s)*z(i,s));
!Subject to;
!constraint 80;
@for (Port(i):
    @for (ArrivalNo(m):[R_Sche]
@sum(VR(v,r):A(i,m,v,r)*yv(v,r))+@sum(PortSeq(s):W(i,m,s)*z(i,s))=
1));
!constraint 81;
@for (Port(i):
    @for (ArrivalNo(m):[R_Quant]
@sum(VR(v,r):Qvcg(i,m,v,r)*yv(v,r))-
@sum(PortSeq(s):Qpcg(i,m,s)*z(i,s))=0));
!constraint 82;
@for (Port(i):
    @for (ArrivalNo(m):[R_Time]
@sum(VR(v,r):Tvcg(i,m,v,r)*yv(v,r))-
@sum(PortSeq(s):Tpcg(i,m,s)*z(i,s))=0));
!constraint 83;
@for (Ship(v):[R_Ship] @sum(Route(j):yv(v,j))=1);
!constraint 84;
@for (Port(i):[R_Port] @sum(PortSeq(j):z(i,j))=1);
!constraint 87;
@for (IMJNVR(i,m,j,n,v,r):@bin(@sum(Route(r):Xcg(i,m,j,n,v,r)*yv(v,
r))));
!integer requirement;
@for (IS(i,s):@bin(z));
@for (IMVR(i,m,v,r): @bin(A));
@for (IMS(i,m,s): @bin(W));

```

ENDSUBMODEL

SUBMODEL SHIP_SUBPROBLEM:

```

[SUBOBJ1] MIN = @sum(IMJNV(i,m,j,n,v) | i#NE#j : (mu(i,j,v) -
Dv(j,n)) * x(i,m,j,n,v) - qv(i,m,v) * Dq(i,m) - tv(j,n,v) * Dt(j,n));

!reduced cost for port arrival i,m with ship v in route r = 0 if
it's located in Initial position of ship V for the M-th arrival to
harbor I;
@for(SO(i,m,v) : @for(Time(t) : Fs(i,m,v,t)=0));

!Recursive Function Ship Shortest Path DP, Equation 95;
@for(IM(j,n) :
    @for(IM(i,m) | j#NE#i :
        @for(Ship(v) :
            @for(SO(o,p,v) | j#NE#o #AND# m#NE#p :
                @for(Time(t) : Fs(j,n,v,t)=
@min(Time(k) | k#LE#t : mu(i,j,v) - Dv(j,n) - Dq(i,m) * @abs(qv(j,n,v) -
qv(i,m,v))
-
Dt(j,n) * Day(k) + Fs(i,m,v,k))))))));

!FsTH, Equation 96;
@for(IMV(i,m,v) : FsTH(i,m,v)=@min(IMVT(i,m,v,t) : Fs(i,m,v,t)));
!The least reduced cost for ship v in route r;
@for(IMV(i,m,v) : Fsmin(i,m)=@min(Ship(v) : FsTH(i,m,v)));

!decision var ximjnv;
@for(SO(o,p,u) : @for(IMV(i,m,v) : x(o,p,i,m,u)=
@IF(FsTH(i,m,v) #EQ# Fsmin(i,m) #AND# Fsmin(i,m) #LT#0,1,0));
!decision var wv;
@for(IMV(i,m,v) : wv(i,m,v) = @IF(FsTH(i,m,v) #EQ# Fsmin(i,m)
#AND# Fsmin(i,m) #LT#0,1,0));
!decision var tv;
@for(IMVT(i,m,v,t) : tv(i,m,v,t) = @IF(FsTH(i,m,v) #EQ# Fsmin(i,m)
#AND# FsTH(i,m,v) #EQ# Fs(i,m,v,t), t, 0));

!Ship Constraints;
!constraint 36;
@for(IMV(i,m,v) : @gin(tv(i,m,v)));
!constraint 39;
@FOR(SO(i,m,v) : @SUM(SN(j,n) | i #NE# j : x(i,m,j,n,v) + zv(i,m,v)) = 1);
!constraint 40;
@FOR(SN(i,m) :
    @FOR(Ship(v) : @SUM(SA(j,n) | i #NE# j : x(j,n,i,m,v)) -
@SUM(SN(j,n) | i #NE# j : x(i,m,j,n,v)) - zv(i,m,v) = 0);
!constraint 41;
@FOR(Ship(v) : @SUM(SN(i,m) : zv(i,m,v)) = 1);
!constraint 43;
@FOR(IMJNV(i,m,j,n,v) : @BIN(x(i,m,j,n,v)));
!constraint 44;
@FOR(IMV(i,m,v) : @BIN(zv(i,m,v)));
!constraint 47;
@FOR(SO(i,m,v) : Je(i) * qv(i,m,v) - L(i,m,v) = - QQ(v));

@FOR(IMV(i,m,v) | i #EQ# 1 #AND# m #EQ# 1 #AND# v #EQ#
2 : @BND(0, qv(i,m,v), 0));
@FOR(IMV(i,m,v) | i #EQ# 3 #AND# m #EQ# 1 #AND# v #EQ# 1
: @BND(0, qv(i,m,v), 0));
!constraint 45a;
@FOR(SO(i,m,v) :

```

```

        @FOR(SN(j,n)|i #NE# j: L(i,m,v) + Je(j)*qv(j,n,v) -
L(j,n,v) + CAP(v)*x(i,m,j,n,v) <= CAP(v));
@FOR(Ship(v):
    @FOR(SN(i,m):
        @FOR(SN(j,n)|i #NE# j: L(i,m,v) + Je(j)*qv(j,n,v) - L(j,n,v)
+ CAP(v)*x(i,m,j,n,v) <= CAP(v)));
!Constrain 45b;
@FOR(SO(i,m,v):
    @FOR(SN(j,n)|i #NE# j: L(i,m,v) + Je(j)*qv(j,n,v) -
L(j,n,v) - CAP(v)*x(i,m,j,n,v) >= - CAP(v));
@FOR(Ship(v):
    @FOR(SN(i,m):
        @FOR(SN(j,n)|i #NE# j: L(i,m,v) + Je(j)*qv(j,n,v) -
L(j,n,v) - CAP(v)*x(i,m,j,n,v) >= - CAP(v)));
!constraint 46a;
@FOR(SO(i,m,v):
    @FOR(SN(j,n)|j #EQ# 1: L(i,m,v) + CAP(v)*x(i,m,j,n,v) <=
CAP(v));
@FOR(Ship(v):
    @FOR(SN(i,m):
        @FOR(SN(j,n)|j #EQ# 1: L(i,m,v) + CAP(v)*x(i,m,j,n,v) <=
CAP(v)));
!Constrain 46b;
@FOR(SO(i,m,v):
    @FOR(SN(j,n)|j #EQ# 1: L(i,m,v) - CAP(v)*x(i,m,j,n,v) >= -
CAP(v));
@FOR(Ship(v):
    @FOR(SN(i,m):
        @FOR(SN(j,n)|j #EQ# 1: L(i,m,v) - CAP(v)*x(i,m,j,n,v) >= -
CAP(v)));
!constraint 48;
@FOR(Ship(v):
    @FOR(SN(i,m): qv(i,m,v) <= @SUM(SA(j,n)|i #NE# j: CAP(v) *
x(j,n,i,m,v)));
!constraint 49;
@FOR(Ship(v):
    @FOR(SN(i,m): L(i,m,v) <= @SUM(SA(j,n)|i #NE# j: CAP(v) *
x(j,n,i,m,v)));
!constraint 50;
@FOR(Ship(v):
    @FOR(SA(i,m): qv(i,m,v) <= CAP(v) * wv(i,m,v));
!constraint 51;
@FOR(SA(i,m): @SUM(Ship(v):wv(i,m,v)) <= 1);
!constraint 52;
@FOR(IMV(i,m,v):@BIN(wv(i,m,v)));
!constraint 53;
@FOR(IMV(i,m,v):@GIN(qv(i,m,v)));
@FOR(IMV(i,m,v):@GIN(L(i,m,v)));
!constraint 54;
@FOR(SO(i,m,v):
    @FOR(SN(j,n)|i #NE# j: tv(i,m,v) + ST(i)*wv(i,m,v) +
TQ(i)*qv(i,m,v) + TT(i,j,v) - tv(j,n,v) + 2*THorizon*x(i,m,j,n,v)
<= 2*THorizon));
@FOR(Ship(v):
    @FOR(SN(i,m):
        @FOR(SN(j,n)|i #NE# j: tv(i,m,v) + ST(i)*wv(i,m,v) +
TQ(i)*qv(i,m,v) + TT(i,j,v) - tv(j,n,v) + 2*THorizon*x(i,m,j,n,v)
<= 2*THorizon));
ENDSUBMODEL

```

```

SUBMODEL PORT_SUBPROBLEM:
[SUBOBJ2] MIN = @sum(IM(i,m):yp(i,m)*-Dv(i,m)) +
@sum(IM(i,m):qp(i,m)*Dq(i,m)) + @sum(IM(i,m):tp(i,m)*Dt(i,m));

!FpTH di initial position =0;
@for(SO(i,m,v):@for(Time(t): Fp(i,m,t)= 0));
!Recursive Function Port, Equation 92;
@for(IM(i,n):
    @for(IM(i,m):
        @for(Time(t):Fp(i,n,t)=
@min(Time(k)|k#LE#t:@sum(IM(i,m)|m#LE#n-1:SC(i)-
Dv(i,m+1))+Dq(i,m)*@abs(qp(i,n)-qp(i,m))

        +Dt(i,m)*Day(t)+Fp(i,m,k)))));
!FpTH, Equation 93;
@for(IM(i,m):FpTH(i,m)=@min(IMT(i,m,t):Fp(i,m,t)));

    !decision var yp;
    @for(IM(i,m):yp(i,m)= @IF(FpTH(i,m) #LT# 0,0,1));
    !decision var tp;
    @for(IMT(i,m,t):tp(i,m)= @IF(FpTH(i,m)#LT#0 #AND#
FpTH(i,m)#EQ#Fp(i,m,t),t,0));

!Port Constraints;
!constraint 37;
@for(IM(i,m):@gin(qp(i,m)));
!constraint 42;
@FOR(SN(i,m):@SUM(Ship(v):@SUM(SA(j,n)|i #NE#
j:x(j,n,i,m,v))+yp(i,m)=1);
!constraint 55;
@FOR(SN(i,m)| m #NE# 1: yp(i,m) - yp(i,m-1)>=0);
!constraint 56;
@FOR(SA(i,m):@BIN(yp(i,m)));
@FOR(SA(i,m)|i #EQ# 1 #AND# m #EQ# 1:@BND(0,yp(i,m),0)); !y11=0,
because it's been visited (as starting point);
@FOR(SA(i,m)|i #EQ# 3 #AND# m #EQ# 1:@BND(0,yp(i,m),0));
!constraint 57;
@FOR(SN(i,m)| m #NE# 1:tp(i,m) - tp(i,m-1) >= 0);
!constraint 58;
@FOR(SA(i,m): tp(i,m) <= THorizon);
!constraint 59;
@FOR(SA(i,m):@GIN(tp(i,m)));
!constraint 61;
@FOR(IM(i,m)|m #EQ# 1: sp(i,m) = IL(i) + Je(i)*Ri(i)*tp(i,m));
!constraint 60;
@FOR(Ship(v):
    @FOR(SA(i,m)|i #EQ# 1: qv(i,m,v) <= (Je(i)*Ri(i) *(
ST(i)*wv(i,m,v) + TQ(i)*qv(i,m,v))) + sp(i,m));
!constraint 62;
@FOR(SA(i,m)|m #NE# 1: (sp(i,m-1) - Je(i)*qp(i,m-1) + Je(i)*Ri(i)
*(tp(i,m)-tp(i,m-1)) - sp(i,m)) = 0 );
!constraint 63;
@FOR(IM(i,m): 0 <= sp(i,m));
@FOR(IM(i,m): sp(i,m) <= SMX(i));
!constraint 64;
@FOR(SA(i,m): (sp(i,m) - Je(i)*qp(i,m) + @SUM(Ship(v):
Je(i)*Ri(i)*(ST(i)*wv(i,m,v) + TQ(i)*qp(i,m)))) >= 0 );

```

```

@FOR(SA(i,m): (sp(i,m) - Je(i)*qp(i,m) + @SUM(Ship(v):
Je(i)*Ri(i)*(ST(i)*wv(i,m,v) + TQ(i)*qp(i,m)))) <= SMX(i) );
!constraint 65;
!Constrain should be changed if there is additional visiting
ports;
@FOR(SA(i,m)|i #EQ# 1 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) >= 0 );
@FOR(SA(i,m)|i #EQ# 1 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) <= SMX(i) );
@FOR(SA(i,m)|i #EQ# 2 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) >= 0 );
@FOR(SA(i,m)|i #EQ# 2 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) <= SMX(i) );
@FOR(SA(i,m)|i #EQ# 3 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) >= 0 );
@FOR(SA(i,m)|i #EQ# 3 #AND# m #EQ# 3: (sp(i,m) - Je(i)*qp(i,m) +
Je(i)*Ri(i)*(THorizon-tp(i,m))) <= SMX(i) );
!constraint 66;
@FOR(IM(i,m):@GIN(sp(i,m)));

ENDSUBMODEL

CALC:
! Make first pattern an expensive super pattern;
!initial columnnya ceritanya rute 1 kapal 1 dari initial position
ke port 3, kapal 2 dari initial position ke port 1;
A(3,1,1,1)=1;
A(1,2,2,1)=1;
W(3,1,1)=0;
W(1,2,1)=0;
Qvcg(3,1,1,1)=1;
Qpcg(3,1,1)=1;
Qvcg(1,2,2,1)=1;
Qpcg(1,2,1)=1;
Tvcg(3,1,1,1)=2;
Tvcg(1,2,2,1)=2;
Tpcg(3,1,1)=2;
Tpcg(1,2,1)=2;
yv(1,1)=1;
yv(2,1)=1;
z(3,1)=1;
z(1,1)=1;
@for(Ship(v): CMV(v,1) = 999999);
CMP(3,1)= 999999;
CMP(1,1)= 999999;

! Loop as long as the reduced cost is attractive and there is
space;
LRCS = -2; !Clearly attractive initially;
LRCP = -2;
NR= 1;
NS= 1;
MXR= @size(Route);
MXS= @size(PortSeq);
@while(LRCS #LT# 0 #AND# LRCP #LT# 0 #AND# NR#NE#MXR #AND#
NS#NE#MXS:
!Solve for best route to run among ones generated so far;
@solve(RESTRICTED_MASTER_PROBLEM);
!Copy dual prices to SHIP and PORT submodel;

```

```

@for (IM(i,m) : Dv(i,m) = -@dual(R_Sche(i,m)));
@for (IM(i,m) : Dq(i,m) = -@dual(R_Quant(i,m)));
@for (IM(i,m) : Dt(i,m) = -@dual(R_Time(i,m)));
@for (Ship(v) : Dy(v) = -@dual(R_Ship(v)));
@for (Port(i) : Dz(i) = -@dual(R_Port(i)));
!Generate the current most attractive route;
@solve(SHIP_SUBPROBLEM);
@solve(PORT_SUBPROBLEM);
@for (IMVR(i,m,v,r) | r#EQ#NR : RCS(v,r) = CMV(v,r) -
@sum(IM(i,m) : A(i,m,v,r) * Dv(i,m)) -
@sum(IM(i,m) : Qvcg(i,m,v,r) * Dq(i,m))
-
@sum(IM(i,m) : Tvcg(i,m,v,r) * Dt(i,m)) - Dy(v));
@for (IMS(i,m,s) | s#EQ#NS : RCP(i,s) = CMP(i,s) -
@sum(ArrivalNo(m) : W(i,m,s) * Dv(i,m)) +
@sum(ArrivalNo(m) : Qpcg(i,m,s) * Dq(i,m))
+ @
sum(ArrivalNo(m) : Tpcg(i,m,s) * Dt(i,m)) - Dz(i));
!marginal value of current best route/ calculate Reduce
Cost;
@for (Ship(v) : RCS(v,1)=-1);
@for (Port(i) : RCP(i,1)=-1);
LRCS = @min(VR(v,r) | r#EQ#NR : RCS(v,r));
LRCP = @min(IS(i,s) | s#EQ#NS : RCP(i,s));

!Add the route to the Master if it is attractive;
@IFC(LRCS #LT# 0 #AND# NR#EQ#MXR-1:NR= NR+1;

@for (IMJNVR(i,m,j,n,v,r) | r#EQ#NR : Xcg(i,m,j,n,v,r)=x(i,m,j,n,
v));
@for (IMJNVR(i,m,j,n,v,r) | r#EQ#NR : A(j,n,v,r)=x(i,m,j,n,v));
@for (IMVR(i,m,v,r) | r#EQ#NR : Qvcg(i,m,v,r)=qv(i,m,v));
@for (IMVR(i,m,v,r) | r#EQ#NR : Tvcg(i,m,v,r)=tv(i,m,v));
@for (VR(v,r) | r#EQ#NR : CMV(v,r)=@sum(IMJNVR(i,m,j,n,v,r) | r#EQ#
NR #AND# i#NE#j : mu(i,j,v) * Xcg(i,m,j,n,v,r)));
!Add the Port Sequence to the Master if it is attractive;
@IFC(LRCP #LT# 0 #AND# NS#EQ#MXS-1:NS=NS+1;
@for (IMS(i,m,s) | s#EQ#NS : W(i,m,s)=yp(i,m));
@for (IMS(i,m,s) | s#EQ#NS : Qpcg(i,m,s)=qp(i,m));
@for (IMS(i,m,s) | s#EQ#NS : Tpcg(i,m,s)=tp(i,m));
@for (IS(i,s) | s#EQ#NS : CMP(i,s) = SC(i) * (MXS-
@sum(PortSeq(s) : W(i,m,s)))));
!Finally solve Master as an IP;
@SOLVE( MASTER_PROB);
ENDCALC

```


WRITER BIOGRAPHY



The author's full name is Rafly Dwi Priyono, born in Jakarta on May 21st, 1998. The author is the second of three children. The author took formal education at Batu Ampar 05 Public Elementary School, 20 Public Junior High School, Jakarta, 39 Public Senior High School, Jakarta and Institute of Technology Sepuluh Nopember, Surabaya.

During college, the author joined the Industrial Engineering Student Association Organization, as a Staff of External Relations Department. The author is also an awardees of Bank Indonesia scholarship and is a member of organization fostered by Bank Indonesia called Generasi Baru Indonesia Institute of Technology Sepuluh Nopemer Commissariat from 2018 until 2020. The author takes role as an Staff of Public Relation Department (2018-2019) and as Chairman of Generasi Baru Indonesia Institute of Technology Sepuluh Nopemer Commissariat (2019-2020). The author also had the opportunity to become a laboratory assistant for Quantitative Modeling And Industrial Policy Analysis Laboratory from 2018 until 2020. This laboratory with core competences in the fields of optimization, simulation, statistics and data mining provides a lot of experience to the author, both as an administrator and as a course assistant in the laboratory.

In addition to the organizational activities above, the author has also received a scholarship for student exchange to Anadolu University, Eskisehir, Turkey in 2018. Then the author has also had practical work in the Pastry and Bakery Department as a Production Manager Assistant and was entrusted with re-designing the kitchen layout and doing stopwatches time study to measure the production capacity of the said departments.

The author can be contacted via email at jkt.rafly.dp@gmail.com or through the linkedIn profile www.linkedin.com/in/raflydwipriyono.