2015

Short-Term Resource Allocation and Management

Denis da Cruz Pinha

Short-Term Resource Allocation and Management

Denis da Cruz Pinha

Dissertation submitted to the
Statler College of Engineering and Mineral Resources at
West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Industrial Engineering

Rashpal Ahluwalia, Ph.D., Chair
Fei Dai, Ph.D.
V. Jagannathan, Ph.D
Majid Jaridi, Ph.D.
Feng Yang, Ph.D.
.

Department of Industrial and Management Systems Engineering

Morgantown, West Virginia

2015

Keywords: Resource Management, Resource Constrained Project Scheduling, Project Management, Short-Term, Discrete Event Simulation, Decision Support System

Abstract

Short-Term Resource Allocation and Management

Denis da Cruz Pinha

Almost all sectors of the economy, such as, government, healthcare, education, ship repair, construction, and manufacturing require project management. A key component of project management deals with scheduling of tasks such that limited resources are utilized in an effective manner. Current research on resource constrained project-scheduling has been classified as: a) Single project with single mode for various tasks, b) Single project with multiple task modes, c) Multiple projects with single task mode, and d) Multiple projects with multiple task modes.

This work extends the current multi-project, multi-mode scheduling techniques. The resources can be renewable, and non-renewable. In addition, it focuses on short term scheduling, that is, scheduling on an hourly, daily, or weekly basis. Long term scheduling assumes a stable system, that is, resources, priorities, and other constraints do no change during the scheduling period. In this research, short term scheduling assumes a dynamic system, that is, resources, priorities, and other constraints change over time.

A hybrid approach is proposed to address the dynamic nature of the problem. It is based on discrete event simulation and a set of empirical rules provided by the project manager. The project manager is assumed to be highly knowledgeable about the project. He/she is regarded as an integral part of the system. Such an approach is better suited to deal with "real world" scheduling. The proposed approach does not seek to provide a single "optimum" solution, instead, it generates a series of feasible solutions, along with the impact of each solution on schedule and cost.

Two project case studies dealing with finding an optimum solution were selected from the literature. The proposed technique was applied to the data set in these studies. In both cases the proposed approach found the "optimum" solution. The model was then applied to two additional problems to test the features that could not be tested on the dataset from the literature.

As for practical implications, the proposed approach enhances the decision making process, by providing more resource allocation flexibility, and results in improved solutions in terms of total project duration and cost. From an academic viewpoint, this research enriches the existing literature, as it provides an extension of the resource constrained project scheduling problems, a discrete event simulation and four cases studies which highlights relevant issues to model properly the complexity of real-life projects.

Acknowledgements

I would like to express my deepest gratitude to my advisor, Dr. Rashpal Ahluwalia or Dr. A, as I have kindly called him for 4 years. Dr. A supported me through good and tough times during my research period. His experience, his positive attitude for life, his financial support and his valuable advices helped me to overcome several research and life challenges. Dr.A's role was far beyond what I ever expected of an advisor. His guidance, his knowledge, and his kindness fulfilled a perfect combination for me. I finish this important step in my life to start a new one having Dr.A as an inspiration, as a good friend, and also knowing that several opportunities still there for us to work together. I feel so glad to have had Dr. A in my life.

Besides my advisor, I would like to thank my dissertation committee: Dr. Jaridi, Dr. Yang, Dr. Dai and Dr. Juggy for their insightful comments and encouragement, but also for the questions which improved my research from various perspectives. It is an honor for me to have you as my committee members. Special feeling of gratitude to Ms. Marie and Mr. Howard who have helped me infinite times during my time at West Virginia University.

I also would like to express my gratitude to Dr. Nimbarte, Dr. Currie, Dr. Creese, and Dr. Iskander for providing me a chance to work as Teaching Assistant with Dr. A and also for providing a friendly research environment in the Department of Industrial and Management System Engineering at West Virginia University.

Dedication

I dedicate my dissertation to my wife, Carolina, and my son, Tiago. I would never have been able to finish my dissertation without their help. So much love!

A special feeling of gratitude to my loving parents, Cesar and Vera. My brother Daniel, his wife Paula, Vinicius, and Bernardo who have helped me infinite times. I also would like to dedicate this dissertation to Tia Helena and Tio Jose Lourenco who have supported me with encouragement words and optimism in tough times. My sister Andreia, brother Cesar, nephew Victor, my cousin Renato and my father in law Oscar who have never left my side, in spite of the distance. They have given me so much support and encouragement to finish my dissertation. I also dedicate this dissertation to my good friends Junior, Ana Candida, Bernardo, and little one Eduardo who have supported me throughout the process. I will always appreciate all they have done.

Special gratitude to Dr. Carvalho (Andrea) and Dr. Saisse (Manoel). They provided me several insights which improved my dissertation. I would also like to express my special feelings to Dr. Queiroz (Max) and Dr. Cury (Jose) of the Department of Automation and Systems Engineering at Federal University of Santa Catarina. They sparked my interest to pursue a higher education level and they are good friends.

I would like finally to thank God for supporting me spiritually throughout writing this dissertation and my life.

Table of Contents

List of Tables

List of Figures

Glossary

Term	Description
Task	Lowest level of work element (it cannot be divided further), that influences project cost, duration, or resource utilization.
Mode	Alternate ways of accomplishing the same task
Non-renewable Resource	Nonrenewable resource is a resource that is consumed when a task is carried out, e.g. raw material, power, fuel, etc.
Renewable Resources	Renewable resource are not consumed, e.g. facilities, equipment, workers, etc.
Doubly Constrained Resource	Doubly-constrained resources are a combination of the two previous categories and are constrained per period.
Regular Measures of Performance	The minimization of the project duration is certainly the most common regular measure of performance. Also, the minimization of lateness.
Non-regular Measures of Performance	Non-regular measures of performance are those in which the regular measure definition does not apply. This measure introduces financial aspects such as cost, penalty costs, and quality cost.

Acronyms

Acronyms	Description
RCPSP	Resource-Constrained Project Scheduling Problem
MRCPSP	Multi-Mode Resource Constrained Project Scheduling Problem
FRCPSP	RCPSP with Flexible resource profiles
RCMPSP	Resource-Constrained Multi-Project Scheduling Problem
MRCMPSP	Multi-mode Resource-Constrained Multi-Project Scheduling Problem
CMRCMPSP	Combinatorial Multi-mode Resource-Constrained Multi-Project Scheduling Problem
SSGS	Serial schedule generation scheme
PSGS	Parallel schedule generation scheme
PMSGS	Parallel mode schedule generation scheme
ToN	Task-on-Node
PERT	Program Evaluation Review Technique
CPM	Critical Path Method
JIT	Just-in-Time
OPT	Optimized Production Technology

Notations

Scalar	Description
NP	Number of projects
NS	Number of skills
NA_i	Number of tasks of project i
$NM_{i,j}$	Number of modes for project i, task j
$NC_{i,j,m}$	Number of subsets of resources for project i, task j, mode m
$NR_{i,j,m,c}$	Number of resources for project i, task j, mode m, subset of resource c
NRP	Number of resource for all projects
NPR_r	Number of priority rules for resource r
NRS_s	Number of resources with skill s
NMQ_t	Number of current modes in the queue at time t
$NIDR_z$	Number of time interval partitions during a day for resource z
$NSuc_{i,j}$	Number of successors for project i, task j
$NPred_{i,j}$	Number of predecessors for project i, task j
$d_{i,j,m}$	Duration time for project i, task j, mode m
$E_{i,j}$	Eligible date to start for project i, task j
$M_{i,j}$	Mode chosen for project i, task j
q	Index for current modes in the queue 1< q < NMQ_t
$MQ_{q,t}$	Mode with index q in the queue at time t
$\mathbf{T}$	interval time, $0 \leq t \leq T$
$\boldsymbol{rn_{i,j,m,c}}$	Required quantity of renewable resources with skill/capability c operating on mode m for task j of project i
$\boldsymbol{w_{i,j,m,c}}$	Required quantity of nonrenewable resource with capability c operating on mode m for task j of project i
$\boldsymbol{d_{i,j,m}}$	Duration of task j of project v operating on mode m
$(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{m}, \boldsymbol{c}, \boldsymbol{r})$	The 5-tuple (i, j, m, c, r) provides an index re ∈ RE
$(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{m}, \boldsymbol{c}, \boldsymbol{y})$	The 5-tuple (i, j, m, c, y) provides an index nr ∈ NR
$\boldsymbol{z}_{\left(\boldsymbol{Re}_{(i,j,m,c,r)},\boldsymbol{t}\right)}$	= 1 , if the renewable resource r with skill c required, operating on mode m for task j of project i is available at time t based on its own calendar = 0, otherwise
$\boldsymbol{z}_{\left(\boldsymbol{Nr}_{(i,j,m,c,y)},\boldsymbol{t}\right)}$	= 1, if the nonrenewable resource y with capability c required, operating on mode for task j of project i is available at time t = 0, otherwise
$\boldsymbol{dd_{i,j}}$	Assigned due date for task j of project i
$\boldsymbol{dd_i}$	Assigned due date for project i
$\boldsymbol{c_{i,j}}$	Tardiness cost task j of project i per time unit
$\boldsymbol{c_i}$	Tardiness cost of project i per time unit
$\boldsymbol{cs_{(re_{sr})}}$	Cost associated by using a given skill out of multiples of renewable resource re
$\boldsymbol{cr_{re}}$	Cost/unit time of renewable resource re in regular time
$\boldsymbol{cro_{re}}$	Cost/unit time of renewable resource re in over time
$\boldsymbol{crm_{re}}$	Cost/unit time of renewable resource re in maintenance
$\boldsymbol{cw_{nr}}$	Cost/unit of nonrenewable resource nr
$\boldsymbol{car_s}$	Cost for adding a renewable resource with skill s
$\boldsymbol{caw_{nr}}$	Cost for purchasing one unit of nonrenewable resource nr
$\boldsymbol{AR_{s,t}}$	Quantity of renewable resources available with skill/capability s at time t
$\boldsymbol{AW_{s,t}}$	Quantity of nonrenewable resource with capability s available (on hand) at time t
$(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{m}, \boldsymbol{c})$	The 4-tuple (i, j, m, c) provides an index s ∈ S
$\boldsymbol{ARD_{s,t}}$	Quantity of renewable resources added with skill/capability s at time t
$\boldsymbol{AWD_{nr,t}}$	Total amount of nonrenewable resource nr delivered at time t
$\boldsymbol{MRD_{s,t}}$	Quantity of renewable resources with skill/capability s in maintenance at time t

$\boldsymbol{TAWD_{nr}}$	Total amount of nonrenewable resource nr ordered
$\boldsymbol{TARD_s}$	Total quantity of renewable resources added with skill/capability s
$\boldsymbol{TR_{re}}$	Total amount of time used of renewable resource re in regular time
$\boldsymbol{TW_{nr}}$	Total quantity used of nonrenewable resource nr
$\boldsymbol{TRO_{re}}$	Total amount of time used of renewable resource re in over time
$\boldsymbol{TRM_{re}}$	Total amount of time in maintenance of renewable resource re
$\boldsymbol{TC_{i,j}}$	Tardiness cost for task j of project i
$\boldsymbol{TC_i}$	Tardiness cost of project i
$\boldsymbol{TCS_{(i,j,m,c,r)}}$	Cost calculated for choosing a given skill out of multiples of renewable resource re
$\boldsymbol{TCS}$	Total cost associated of choosing a skill out of multiples for all renewable resources, in all modes, tasks, projects and time.
$\boldsymbol{TCR}$	Total cost associated of resources, in all modes, tasks, projects and time.

Index	Description
i	Project index, $1 \leq i \leq NP$
j	Task index, $1 \leq j \leq NA_i$
m	Mode index, $1 \leq m \leq NM_{i,j}$
c	Subset index, $1 \leq c \leq NC_{i,j,m}$
r	Resource index, $1 \leq r \leq NR_{i,j,m,c}$
pr	Rule index, $1 \leq pr \leq NPR_r$
s	Skill index, $1 \leq s \leq NS$
pre	Predecessor index, $1 \leq pre \leq NPre_{ij}$
suc	Successor index, $1 \leq suc \leq NSuc_{ij}$
iN	Last task (indexed with N) of project i

Set	Description
RE	set of renewable resources, $re \in RE$
NR	set of nonrenewable resources, $nr \in NR$
RN	set of resources, $rn \in (RE \cup NR)$
I	set of projects, $i \in I$
$\mathbf{J_i}$	set of tasks of project i, $j \in J_i$
P	set of all precedence relationships
$\mathbf{M_{i,j}}$	set of modes for task j of project i, $m \in M_{i,j}$
$\mathbf{C_{i,j,m}}$	set of skills/capabilities operating on mode m of task j and project i, $c \in C_{i,j,m}$
$\mathbf{R_{i,j,m,c}}$	Set of renewable resources with skill/capability c operating on mode m of task j, and project i, $r \in R_{i,j,m,c} \subseteq RE$
$\mathbf{Y_{i,j,m,c}}$	set of nonrenewable resources with capability c operating on mode m, for task j of project i $y \in Y_{i,j,m,c}$ $Y_{i,j,m,c} \subseteq NR$
S	Set of subsets of renewable resources with same skills or capabilities, $s \in S$
$\mathbf{SRe_{re}}$	Set of skills for renewable resource re, $sr \in SRE_{re}$

Variable	Description
$\boldsymbol{st_{i,j}}$	Start time of task j, project i
$\boldsymbol{ft_{i,j}}$	Finish time of task j, project i
$\boldsymbol{x_{i,j,m,t}}$	= 1, if task j of project i, operating on mode m, is started at time t = 0. otherwise
$\boldsymbol{h_{i,j,m}}$	= 1, if task j of project i is operating on mode m = 0, otherwise
$\boldsymbol{u_{\left(Re_{(i,j,m,c,r)},t\right)}}$	= 1, if the renewable resource re with skill c required, operating on mode for task j of project i is available at time t because it is not being used for any other mode among all the projects = 0, otherwise

Chapter 1: Introduction and Literature Review

1.1 Project Scheduling Methods

The Project Management Institute defines project as a "temporary endeavor designed to produce a unique product, service, or result" [PMI, 2013]. Temporary implies that projects have a specific start and end dates. Project management is "the process of planning, organizing, motivating, controlling resources, procedures and protocols to achieve specific goals of a project" [PMI, 2013]. Successful conclusion of a project can result in a product, service, or an improved process to deliver a product or a service. Projects exist in every sector of the economy, including manufacturing, healthcare, education, and government. Project Scheduling is a complex task, largely due to multiple tasks, that compete for limited resources. Resources can be skilled labor, capital, equipment, facilities, etc. Significant amount of research has been conducted on allocating limited resources among several competing tasks. Bulk of this research falls under the general area of Operations Research (OR) and Simulation, and the models are either analytical or heuristic.

Project scheduling research has its origins in job shop and flow shop scheduling methods. In order to deal with the complexity of the scheduling problems, authors have suggested a wide variety of techniques over the years. Among these are mathematical programming by Cheng and Hall [Cheng and Hall, 2008], queue rules by Lu et al. [Lu et al., 2010], expert systems by Pesch and Tetzlaff [Pesch and Tetzlaff, 1996], neural networks by Chaudhuri and Kaja [Chaudhuri and Kaja, 2010], multi-agent system by Zhou et al [Zhou et al, 2010], optimized production scheduling by Goldratt [Goldratt, 1988] and supervisory control theory by Pinha et al. [Pinha et al. , 2011].

The scheduling policies and the concept of robustness were addressed by [1- Feng et al., 2012], and [2- Feng et al, 2012]. Briefly, their approach covered buffer capacity, arrival rate based on a Poisson process, operation time and setup time based on the exponential distribution. Their approach provided good scheduling performance for the developed criterion. They utilized seven heuristic dispatching rules to determine the overall “best” performance. However, basic production issues, such as, operation precedence and multiple resources required to complete a task were not covered. They demonstrated their approach on a single machine producing multiple products.

Recently, the concept of Transient State (TS) and Steady State (SS), from systems theory point of view, has been applied to production and project planning. Machine failures and repairs cause system perturbations, resulting in transition from SS to TS [Yang and Jingang, 2012]. The TS affects output dynamics, therefore studies have been performed to measure or estimate the time to reach SS from TS. Statistical approaches have been applied to model jobs arrival rates, services rates, the time to machine failure, time to machine repair (gamma distribution) and work in process. Yang [Yang and Jingang, 2012] compared the true dynamic response to the predicted response. Their approach provides average values of outputs with respect to the TS.

Numerous challenges exist when modelling “real world” situations. Several researcher are actively engaged in addressing these challenges [Naber and Kolisch, 2014], [Xu and Feng, 2014]. The research on project scheduling is well ahead of practice. The most common tool that is utilized in industry is Microsoft Project [MS Project, 2015]. This tool has several limitations. It is unable to efficiently handle resource constraints and multiple projects, resulting cost over runs and schedule slippage.

This chapter provides a review of scheduling methods as follows:

1. Program Evaluation and Review Technique - Critical Path Method
2. Resource Constrained Project Scheduling Problem (RCPSP)
3. Multi-mode Resource Constrained Project Scheduling Problem (MRCPSP)
4. Resource Constrained Multi-Project Scheduling Problem (RCMPSP)
5. Resource Constrained Project Scheduling Problem with flexible resource profile (FRCPSP)
6. Multi-mode Resource Constrained Multi-Project Scheduling Problem (MRCMPSP)
7. Project scheduling in the ship repair industry
8. Project scheduling in the construction industry

1.1.1 Critical Path Method

Critical Path Method (CPM) was introduced in the late 1950's [Moder and Phillips, 1964]. This technique represents project as a network using graph theory [Hazewinkel, 2001]. CPM is based on deterministic task time. The Task-on-Node (ToN) diagram is used to describe project tasks. The nodes represent tasks and arrows define the precedence relationship. Figure 1 shows six tasks, with node ST and node EN being dummy start and end tasks, respectively. The number above each node represents task duration. Task 6 in the Figure 1 has 2 incoming arrows while it has only one outgoing arrow. Task 6 has two predecessors (tasks 4 and 5) and only one successor (task EN) [PMI, 2013]. The network is a directed graph and no loop cycles are allowed. CPM assumes no constraints on resources. In spite of the severe assumption this method is widely used in practice, mainly due to its simplicity.

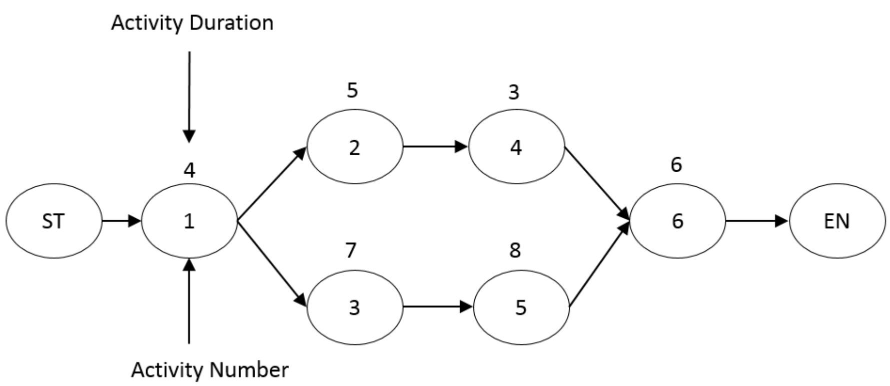


Figure 1: Task on Node Diagram for the Critical Path Method

1.1.2 Program Evaluation and Review Technique

Program Evaluation and Review Technique (PERT) was also introduced in the late 1950's [Fazar, 1959]. The Navy's Special Projects Office was charged with the Polaris-Submarine weapon system and the Fleet Ballistic Missile capability. PERT was then developed by a partnership between Booz Allen Hamilton and the U.S. Navy. This technique represents project as a network using graph theory

[Hazewinkel, 2001]. PERT also utilizes the Task-on-Node (ToN) diagram to describe project tasks. However, PERT allows probabilistic values for task duration time and it assumes that each task duration is a random variable between two extreme values: an optimistic time (a), pessimistic time (b). Also, a realistic time (m) is taken into account. It usually follows a beta probability distribution [Vanhoucke, 2013]. The expected time t of a beta distribution can be approximated by the weighted average shown in equation 1:

$$t = (a + 4m + b)/6 \tag{1}$$

Nodes and arrows follow precedence relationship as in CPM. PERT also assume no constraints on resources. In Figure 2, task 3's parameters are a = 4, m=7 and b=9.

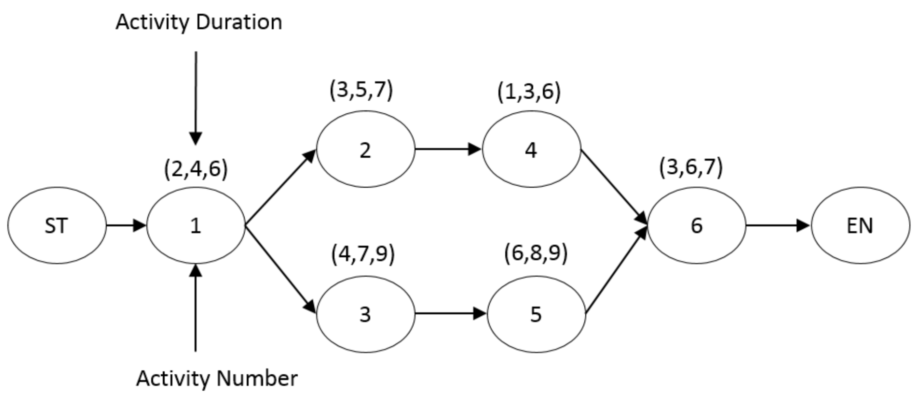


Figure 2: Task on Node Diagram for the PERT

1.1.3 Resource Constrained Project Scheduling Problem (RCPSP)

Resource Constrained Project Scheduling Problem (RCPSP) was introduced by Pritsker, Watters, and Wolfe [Pritsker et. al, 1969]. They assumed a single task duration and resource requirement pair. Several variations of the RCPSP method have been proposed. Brucker [Brucker et al, 1999] introduced the notion of limited renewable and non-renewable resources and conflicts between multiple resources. Hartmann and Briskorn [Hartmann and Briskorn, 2010] provided an extensive survey of variants and extensions of the RCPSP. They described two distinct classes of solutions, one for regular measures of performance, and the other for non-regular measures of performance. Bianco and Caramia [Bianco, L. and Caramia, 2013] developed an exact formulation for RCPSP.

Vanhoucke [Vanhoucke, 2013] described the above problem mathematically. Its objective function was to minimize the total duration of a project by minimizing the start time of the last task, subjected to precedence relationships and limited resources. Maenhout and Vanhoucke (2015) proposed an exact algorithm to solve the RCPSP integrated with project staffing problem considering different rules depending upon some labour regulations.

Figure 3 shows the Task-on-Node (ToN) network for RCPSP. The numbers and letters above each node represent task duration and resource requirement pair. Table 1 shows hypothetical task duration and resource requirements for three tasks. Each node has only one pair of task duration and resource requirement. This pair is predetermined and fixed, e.g. in Table 1, duration of task 2 is 5 time units and it requires 12% of resource 1 and 50% of resource 2. Details of this model are provided in [Vanhoucke, 2013].

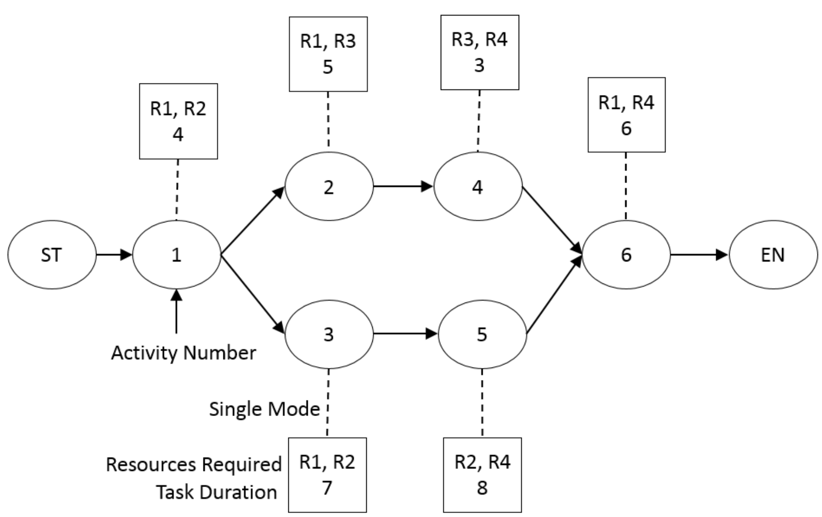


Figure 3: Task on Node Network for RCPSP

Table 1: Resource requirements			
Task	**Duration**	**Resource 1**	**Resource 2**
ST	0	0	0
2	5	12	50
3	7	35	54

1.1.4 Multi-Mode Resource Constrained Project Scheduling Problem (MRCPSP)

Multi-mode Resource Constrained Project Scheduling Problem (MRCPSP) is an extension of the RCPSP method**.** It was first introduced in 1977 by Elmaghraby [Elmaghraby, 1977]. Difference between RCPSP and MRCPSP is that RCPSP had only one pair of task duration-resource requirements to perform a task, whereas in MRCPSP each task can be performed by selecting one out of several different combinations of task duration-resource requirements. The various combinations are called modes. Naber and Kolisch [Naber and Kolisch, 2014] described mode as "a non-preemptive, constant resource usage of a task over its entire predetermined fixed duration." Figure 4 shows the ToN network for the MRCPSP. E.g. in Figure 4, task 3 has three modes with durations 7, 8, and 10 unit of time, respectively. Table 2 shows task duration-resource requirements and modes for task 3. Each node has task duration-resource requirements pair. The pairs are predetermined and fixed, e.g. task 3 has one mode of seven units of time and it requires resource R1 and R2. Table 3 describes additional details.

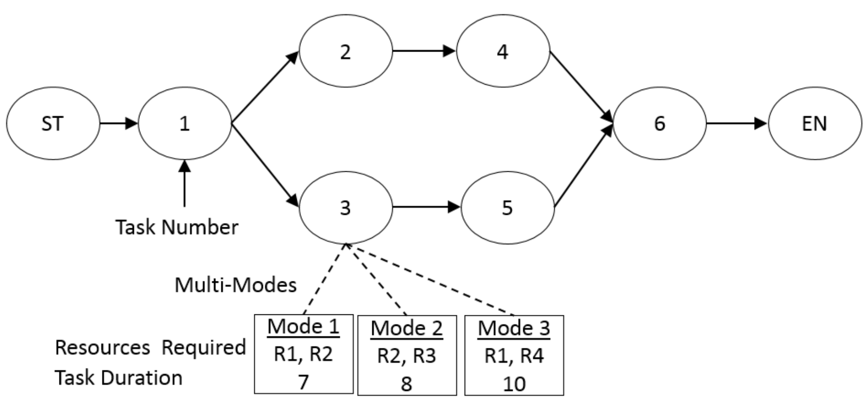


Figure 4: Task on Node for MRCPSP

Table 2: Modes for task 3	
Task	**Modes**
3	(7, R1, R2), (8, R2, R3), (10, R1, R4)

Table 3: Mode Description		
Modes	**Resource 1**	**Resource 2**
(7, R1, R2)	12	50
(8, R2, R3)	35	54
(10, R1, R4)	10	40

Various authors [Alcaraz and Maroto, 2003], [Bouleimen and Lecocq, 2003], [Hartmann, 2001], [Jarboui et. al, 2008], [Jozefowska et. al, 2001], [Ozdamar, 1999], [Varma et. al, 2007], [Brucker and Knust, 2001], [Calhoun et al, 2002], [De Reyck and Herroelen, 1999], [Drexl et al, 2000], [Heilmann, 2001], [Heilmann, 2003], [Zhu et. al, 2006], [Sabzehparvar and Seyed-Hosseini, 2008], [Peteghem and Vanhoucke, 2010], [Bouleimen and Lecocq, 2003], [Jarboui et al, 2008], [Kolisch and Drexl, 1997] have proposed both heuristic and exact solutions for the above problem.

1.1.5 Resource Constrained Multi-Project Scheduling Problem (RCMPSP)

Further extension of RCPSP is Resource Constrained Multi-Project Scheduling Problem (RCMPSP). RCMPSP can handle situations where multiple projects compete for the same resource. In RCMPSP, a task has only one possible combination of duration-resource requirement (one mode), as in RCPSP. However, RCMPSP can handle several projects, and project tasks need to be scheduled simultaneously, under precedence and resources constrains. Browning and Yassine handled RCMPSP by revising priority rules [Browning and Yassine, 2010]. Xue [Xue et al, 2010] applied neural network approach to solve RCMPSP. Zhang and Sun [Zhang and Sun, 2011] used priority-rule based heuristics. Laslo [Laslo and Goldberg, 2008] identified uncertainty in the multi-project environment. Chen and Shahandashti [Chen and Shahandashti, 2009] used the genetic algorithm. Bouleimen and Lecocq [Bouleimen and Lecocq, 2003] utilized the simulated annealing approach. Araúzo [Araúzo et al, 2010] applied Multi-Agent System to the RCMPSP problem.

1.1.6 Resource Constrained Project Scheduling Problem with Flexible Resource Profile (FRCPSP)

The MRCPSP uses a multi-mode approach to perform each task, compared to RCPSP, which has only one mode. However, both approaches use a fixed pair of task duration-resource requirements. Kolisch [Kolisch et al, 2003] proposed that task duration-resource requirement pair must be determined by the method and should not be fixed. Naber and Kolisch [Naber and Kolisch, 2014] proposed using Mixed Integer Programming (MIP) model to solve what they termed as Resource Constrained Project Scheduling Problem with flexible resource profile (FRCPSP).

Formulation of FRCPSP was also described by Naber and Kolisch [Naber and Kolisch, 2014]. They explained flexible resource profile by an example. That is, if a given task requires 10 person-days, it may be allocated as a constant profile of 2 persons for 5 days (as in the RCPSP), or as a flexible profile of 3 persons for 2 days and 2 persons for 2 days. The authors stated that one can derive FRCPSP as being MRCPSP by building several modes for each task. The issue with this approach is that it only works for discrete resources. For continuous resources, such translation is extremely cumbersome. The authors compared their solution with other exact methods and concluded that efficient solution methods, both exact and heuristic, for real-world applications need to be studied further. The authors also cited an interesting project scheduling problem in the healthcare industry, where FRCPSP can be applied. They regarded fellowship training program as a project, that consists of a number of different surgical type tasks, requiring a certain number of surgical sessions (resources) to be completed over the training period, subject to the pre-requisite requirements (precedence) and limited number of surgeries (resource availability) that the medical fellow can conduct in each period. Other work on FRCPSPs can be found in Fundeling (2006), Fundeling and Trautmann (2010), Ranjbar and Kianfar (2010) and Baumann and Trautmann (2013).

1.1.7 Multi-mode Resource Constrained Multi-Project Scheduling Problem (MRCMPSP)

Many researchers have worked on several variations of the RCPSP, RCMPSP and MRCPSP. However, the case of Multi-mode Resource Constrained Multi-Project Scheduling Problem (MRCMPSP) has not been addressed adequately. The MRCMPSP approach allows several project tasks to be handled simultaneously, under precedence and resources constraints, and each task can have several modes. Thus far, MRCMPSP is the most complex scenario cited in the literature. Only few papers addressed the MRCMPSP. It referred to the studies by Speranza and Vercellis (1993) and by Xu and Feng (2014). Xu and Feng applied a modified particle swarm optimization algorithm as a heuristic method to manage construction of a large-scale hydropower plant. They affirmed that exact methods are not able to solve complex real world problems.

The RCPSP and its more complex extensions are regarded as a standard approach to project scheduling. They belong to the class of strongly NP-hard problems [Kolisch, 1996]. Majority of resource-constrained scheduling methods can broadly be classified as exact or heuristic. Goal of the exact procedures is to find the optimal solution. These methods however, are only suitable for small projects with several assumptions to simplify the problem. Many authors have attempted to incorporate additional capabilities by simulation [Araúzo et al, 2010]. The heuristic methods aim to find a good, and not necessarily the best schedule, using more realistic assumptions [Vanhoucke, 2013].

1.2 Project Management in Industry

1.2.1 Construction Industry

According to AlSehaimi et al. (2013), the construction industry has lower productivity compared to other industries. Low productivity is largely due to poor resource management, availability of skilled labor, and reliability of the material supply chain. A study by the U.S. National Research Council (NRC 2009) also came to a similar conclusion. They indicated that in order to remain competitive the construction industry needs to manage project schedules, labor, material, and energy costs more effectively.

The construction industry, like other industries, has lacked a valid resource-constrained approach to plan their tasks [Lu et. al, 2008]. Popular project scheduling software systems such as Primavera [Primavera, 2015] and Microsoft Project [MS Project, 2015] have been applied in this sector, largely due to absence of a more suitable tool. When resources are scarce and shared along different tasks in complex projects, resource levelling is cumbersome under traditional approaches. Lu [Lu, 2003], Lu et al. [Lu et. al, 2008] utilized on simulation based methods to schedule tasks in the construction industry.

The goal of simulation based modelling is to provide an understanding of problems by building a computational logic of complex real-life problem [Law and Kelton, 2000]. There are two major categories with respect how the system changes overtime, discrete event simulation and continuous simulation [Pritsker and O'Reilly, 1999]. In general, a discrete event simulation is concerned with stimuli named events. The occurrence of an event at discrete points in time modifies the system, thus altering its current state at that time [Pinha et al, 2011]. Majority of simulation application in the construction engineering utilize on the discrete event simulation, a review of discrete event simulation methods in the constructions can be found in [Lau et al, 2014]. Further details of discrete event simulation approach are described in chapter 4.

Other work dealing with real-life projects in construction is described by Lu [Lu, 2003]. He developed a simplified discrete-event simulation approach (SDESA) by extracting features from existing simulation methods to plan the dynamic and interactive construction systems. [Lu and Chan Lu, 2004] present an extension of SDESA to deal with resource availability and multiple calendars having only one shift; [Lu et. al, 2008] describe an approach called Simplified Simulation-based Scheduling system (S3) based on SDESA, a new float determination method for CPM and Particle Swarm Optimizer (PSO), and [Lau et al, 2014] presented an approach to discretize continuous resources. Recent work of Siu et al [Siu et al, 2014] propose a bi-level project simulation methodology to first determine the optimal resource quantities and the shortest time duration as needed for accomplishing each work package; and second to estimate total project duration and budget through Monte Carlo simulation.

1.2.2 Ship Repair Industry

According to [Van Dijk, 2002], the traditional time-driven approaches such as the Critical Path Method (CPM), has several shortfalls for the ship repair industry. Roelof van Dijk [Van Dijk, 2002] worked with The Royal Netherlands Navy Dockyard (RNND) on a ship repair scheduling. RNND is responsible for maintenance, repair, and modifications of the Dutch naval vessels. They claim that production planning and scheduling in shipyards is unreliable. They further state that traditional time-driven approaches such as CPM have several shortcomings. For instance, resource capacity requirements are consequence of plans, but they should have been taken into consideration simultaneously as input with time and costs. Therefore, according to the author, plans performed with CPM are not reliable even for a single project. This situation is worse for the multi-project environment where resources must be shared among multiple projects. The integration of different stakeholders with different levels of responsibility was also mentioned as a critical need.

[Van Dijk, 2002] approach is an extension of the approach proposed by [De Boer,1998], [De Boer et al., 1997]. Wullink's work [Wullink, 2005], [Wullink et al., 2004]. It deals with resource loading under uncertainties. He utilized a scenario based approach and the concept of robustness to deal with demand and capacity uncertainties. He did not consider precedence relationships, release dates, and rush orders. Dlugokecki [Dlugokecki et al., 2010] proposed a project management approach inspired in Ballard [Ballard, 2000]. His work showed improvement in cost savings and higher level of productivity only for building new ships, not for ship repair tasks. Ballard and Choo [Ballard, 2000], [Choo, 2003] presented a resource model to manage construction projects. Their model lacked complexities of the ship repair industry.

Works of [Mourtzis, 2005], [Chyrssolouris et al., 2004], and [Chyrssolouris, 1999] have integrated different stakeholders in their planning process. They took a systems approach to planning and utilized state of the art information technology tools, such as heuristics and event-driven simulation, to allocate resources. They identified major differences between production planning and scheduling for the ship building industry vs. the ship repair industry. Some of the differences are types of facilities, types of equipment, worker skill levels, work flow patterns, shifting priorities, cost and delivery schedule [Chabane, 2004]. Authors [Charris and Arboleda, 2013], [Mello and Strandhage, 2011] worked on supply chain management for shipyards. [Pinha et al, 2011] presented an application of the supervisory control theory to schedule ship repair tasks. Preliminary work was described in [Pinha and Ahluwalia, 2014] and [Pinha and Ahluwalia, 2013]. The authors designed a database schema for the ship repair industry and proposed Discrete-Event-Simulation model. Zhou [Zhou et al., 2013] proposed solutions for repairing war ships; however, their work lacked several real constraints. Papakosta's approach [Papakostas et al., 2010] was based on [Chryssolouris and Dicke, 1992], [Chryssolouris et al., 1992, 1991], [Chryssolouri, 2005] for scheduling maintenance of airplanes whereas [Framinan and Ruiz, 2012, 2010], [Moghaddam and Usher, 2011], and [Yamashita et al, 2007] proposed alternate approaches.

Numerous gaps exist with respect to modeling of ship repair facilities. Currently, majority of production planning and scheduling task are considered static in nature and utilize Microsoft Excel or Microsoft Project software to scheduling, often resulting in cost over runs, schedule slippage, and low throughput. Traditional project management approach uses a pre-determined approach to resource allocate. The flexibility in resource allocation based on worker skills can reduce costs and increase throughput. This is an urgent need in the ship repair industry [DoN, 2013], [NSRP, 2013], [Leadership, 2013], [MARAD, 2013]. Also, there is a need for an integrated tool for resource allocation dealing with multi-modes, multi-projects, and resource constraints.

Application of project scheduling techniques in this industry has been very limited. US and European governments have identified project scheduling in shipyards as a major area of concern [NSRP, 2013], [Leadership, 2013], [MARAD, 2013]. Scheduling in the ship repair industry is further complicated due to unexpected demand, changing priorities, lack of skilled workers, high capital investment in specialized equipment, and environmental issues [Dlugokecki et. al., 2010], [Wullink, 2005], [Wullink et. al., 2004].

1.2.3 Limitations of the Commonly Used Tools

Table 4 summarizes pros and cons of some the scheduling approaches described in literature.

Table 4: Pros and Cos

Class	Pros	Cos	Authors
Mathematical Formulations	1-Exact solutions	1. Applied to small problems constraints 2. Many assumptions	[Vanhoucke, 2013] [Naber and Kolisch, 2014]
Optimization with Heuristics	1-Fuzzy operation times	1. Lack of daily calendar 2. Only Finish-to-Start, no lag-times 3. Limited few tasks and resources 4. Limited to single skill or operation	[Xu and Feng, 2014]
Discrete Event Simulation	1. Determine resource requirement 2. Estimate total project duration 3. Probability of scenarios to estimate durations	1. No modes nor multi-projects 2. No dynamic queues (first in - first out) 3. No multiple priority rules 4. No multiple-skills for resources	[Lu, 2003] [Lu and Chan Lu, 2004] [Lu, Lam, and Dai, 2008] [Lau et al, 2014] [Siu et al, 2014] [Mourtzis, 2005] [Chyrssolouris et al., 2004] [Chyrssolouris, 1999]
Microsoft Project and Primavera	4. Friendly user interface 5. Levelling Resources based on rules 6. Central database for managing multiple projects	5. No modes 6. No multiple priority rules for resources 7. No multiple-skills for resources 8. Faulty interpretation in handling multiple calendars for resources 9. Limited quantity of time intervals per day (5) 10. Does not allow task duration with less than one minute. 11. Only one rule applied to all resources.	[MS Project, 2015] [Primavera, 2015]

A recent work of Coughlan et al. (2015) utilized the Branch-Price-and-Cut algorithm for Multi-mode resource levelling to solve instance of problems with 50 jobs. The authors stated that even today instances with only 30 jobs are hard to solve to optimality. Also, a survey performed by Rehm and Thiede (2012) at Technische Universität Dresden about project scheduling methods presents interesting results. Rehm and Thiede state that since 1981, several solutions methods have been proposed. Since 1993, an average of 3 new solutions a year have been published. A relevant finding of their survey is that majority of solutions methods proposed are able to handle up to 51 tasks, and they utilize heuristics and use genetic algorithm. Regards to their objectives, most of the solutions are time-oriented and do not consider several resource constraints. No solution was found that covered all real-life production concerns. Rehm and Thiede state that due to innumerous existing gaps, further research is necessary. The authors conclude that due to the importance of this field, study will persist and new methods are likely to continue in the coming years.

Microsoft Project [MS Project, 2015] is certainly one of the most common software tool used in the industry. It is relative easy to use. However, resource allocation for competing tasks (levelling resources) is limited with only three general rules. Primavera [Primavera, 2015] is another software tool widely used in the industry. It can handle large data sets. It utilizes Oracle Database [Oracle, 2015], resource allocation for competing tasks is limited and cumbersome in Primavera.

Within the context of project scheduling, modelling of real world situations is a challenging task. Araúzo et al (2010) stated that classical methods, based on mathematical programming, could help decision making when problem complexity is low and the system is somewhat static. On the other hand, these characteristics are seldom true in real world projects. Microsoft and Primavera Project Planner (P3 and P6 versions), which are the most common tools used for project management, present shortcomings on relevant issues in some contexts. For instance, they do not guarantee accuracy in results when tasks require multiple resources that have different calendars associated. Faulty interpretations in calculating finish time for tasks exist. They do not support multiple task modes (i.e., only one set of resources for each task). They lack an efficient and automatic resource allocation procedure; they simply adopt fixed standard rules for levelling all resources in a context where rules may differ among the organization departments. These shortcomings reinforce the need for techniques to properly assist project management.

ProChain [ProChain, 2015] and ProTrack [ProTrack, 2015] are also used tools. ProChain has been applied more often in US projects whereas ProTrack is used in European countries. ProTrack simulates time and cost trade-offs, however neither multi-mode nor multi-project is offered. ProChain has been developed by a consulting company in US and technical support and/or customizations to attend customer needs are offered. ProChain allows lag times, but no resource allocation is done for competing tasks.

In a recent action research, Abrantes and Figueiredo (2015) worked closely with industrial partners and brought strong empirical evidence that resource management is indeed a concern in organizations driven by projects. They presented the challenges in resource management for real-life multiple-projects contexts, characterized by high dynamism and dependency complexity among projects. Resource management methods have received large attention in the academic literature, however resource management approaches to support organizations are still not established [Abrantes and Figueiredo , 2015].

Maenhout and Vanhoucke (2015) stated as future research in their work: “We aim to develop solution techniques that are able to tackle real-life problems. This means that on the one hand, we should increase the problem size (in terms of number of activities) scheduling one or multiple projects. On the other hand, we should expand the problem definition and incorporate additional personnel and task characteristics such as personnel skills, multiple modes of operation for a task, task pre-emption, etc. Exact optimization techniques will not be able to cope with the further extension of the problem definition and size. For this reason, we will focus on (hybrid) heuristic optimization techniques that combine the advantages of both mathematical programming techniques and meta-heuristics optimization”.

Chapter 2: Research Objectives

2.1 Introduction

All sectors of the economy require project management. In spite of differences in industries such as constructions, ship repair, manufacturing, healthcare, and education, there are many similarities with respect to project management, e.g. how to manage and how to allocate scarce resources is an issue found in all sector. Maintenance/Repair of ships, civil constructions, production of specialized equipment for industries such as coal, petroleum, power plants, and wind plants are examples of production environment driven by projects. Typically, project delays result in huge financial losses for the customer.

Project lead times depend not only on its own task times, but also due to waiting time in queues due to lack of resource. Thus, managers need to account for the current status of projects, and the current availability of resources to come up with project duration. In multi-project environment, this assessment is difficult because the number of tasks is very high and resource availability is unstable. Current resource availability requires a detailed assessment which is difficult to perform with traditional project management approaches. Tables 5 and 6 lists typical project planning attributes [Costa and Jardim, 2010].

Table 5: Project planning attributes

Attributes	Values
Demand Pattern	Customer orders
Production volume	Low. Usually as a unit
Production frequency	None or low repetitive
Services	Opened/including innumerous services
Production mix	High unstable
Lead times	High
Resources layout	Functional, Fixed-Position

Table 6: Typical issues for planning and controlling	
Typical	**Obstacles**
Scheduling	Search for an appropriate solution in a virtually infinite universe of possibilities
Short-term	Complex assessment of cost-benefit being made under intense pressure on a daily basis
Deadline	Lack of previous data and lead times depends on the task mix
Budget	Lack of previous data and profitability dependent on capacity adjustments
Re-planning	Instability are facts intrinsic in the project environment
Production	Need for controlling specific task dealing with large volume of data

An approach which supports multiple combinatorial modes and multiple projects is needed. In several projects, tasks can be performed by selecting one out of several different set of resources. Usually, a project manager is in charge of one project, when multi-projects must be run simultaneously, different plans (projects) exist. These sub-plans frequently results in sub-optimization of overall plan, resulting in lower productivity. Required resources such as workers, tools, materials are not adequately shared among projects, resulting in conflicts along managers and schedule slippage.

There is a need to support large scale, real-life projects, where people have multiple skills and equipment can perform a variety of tasks with varying efficiency levels. Methodologies suitable for dealing with multiple projects on a daily basis (differing in terms of size, resource constrained, and in considering real status of each task) remains the main challenge for scheduling. Also, one of the gaps identified from the literature is how to model production capacity of projects. If the capacity is not well described, the results might not represent accurately the production reality.

Assuming tasks require multiple skills, the model must be able to find from the pool of resource those which are able to perform tasks having the skills required. By not allowing multi-skills for resources, increases cost and reduces productivity. In addition, if no relationship exists between a resource and its skills, it is not possible to have resource allocation based on skills. This gap exists mainly due to the traditional project management approach to individually deal with time, cost and capacity. It results in complicating the planning process and ultimately it has a negative impact on productivity.

Why project scheduling is hard? Wall (1996) identified four different project scheduling issues that make it difficult.

1) The Size of the Problem: The size of the problem scheduling can be approximated by what-where-when matrix. Tasks (What) require resources (Where) for certain period of times (When).

2) Modelling issues: An optimum solution under incorrect assumptions is not very useful. A feasible solution under realistic constraints is what practitioners seek.

3) The dynamic nature of problems: Real world situations are unpredictable (machine breakdowns, human absence, inclement weather, etc.). In such cases the pre-planned project schedule are of little value.

4) Infeasible Solutions: If required renewable resources are available during the entire project duration, some delays still occur, but feasible solutions are certainly found. However, if nonrenewable resources exist, and they are scarce, feasible solution cannot be guaranteed.

2.2 Research objectives

The overall research objective of this study was to reduce total project duration and cost by providing various alternatives to resource allocation. This approach is based on the notion that project constraints are not constant. The project manager is provided several feasible solutions, which can include the "optimal" solution. Given the dynamic nature of projects, project managers are better equipped to choose one of the several solutions to attend their current needs. The details of specific research objectives are provided below:

Objective 1: Extension of the RCPSP model: The existing RCPSP approach to resource allocation in project scheduling was extended to deal with practical aspects of project scheduling. The concept of mode was extended to include multiple skills of varying levels of competence. A mode is no longer only a set of required resources capable to execute a task. Mode is also a set of combinatorial subsets of required resources capable to execute a task. The revised extension model is called Combinatorial Multi-Mode Resource Constrained Multi-Project Scheduling Problem (CMRCMPSP**).**

Objective 2: Focus on short-term project management: Most existing project management systems take a long term view of the problem. They assume a static system. This work focused on the short-term scheduling problem. Short-term is significantly more complex than long term scheduling, primarily due to need to respond to perturbation in near real time. It allows project manager decide on timely manner which solution among several is the good one to attend his/her current needs.

Objective 3: Provide a capability to support multi-modes, multiple skills, and multiple calendars: Short term needs such as multiple calendars, and constraints on resources are addressed. Project scheduling in industry is mainly carried out by applying traditional approaches, using tools such as Microsoft Project or Primavera. In construction industry, project managers commonly add resources constraints such as resource limitations and resource calendars. When projects are subjected to this kind of resource constraints, the correct project duration value cannot be determined by using mainstream software such as Primavera and MS Project.

The above objectives will assist project managers in dealing with unexpected events, queue lengths, priority rules, and user inputs, in near real time. Project scheduling strategies on a daily basis are volatile and unpredictable. They depend upon changing customer requirements and resource availability. Unexpected events always exist, and therefore, handling these events is critical to effectively running day-to-day operations. Decisions made yesterday may not be feasible today. The difficulty of exact solutions is in part due in articulating objective function or functions that represent the overall objectives and are dynamic in nature. Classical methods, based on mathematical programming, can help to make decisions when problem complexity is low and the system is static. These characteristics seldom represent project reality.

Chapter 3: Method

This chapter describes the solution method in four parts: (1) the Combinatorial Multi-Mode Resource Constrained Project Scheduling Problem (CMRCMPSP) as an extension of the RCPSP, (2) the problem model in the form of assumptions about tasks and resources with their associated constraints, (3) the simulation method, and (4) a description of the overall system architecture.

3.1 The Combinatorial Multi-Mode Resource Constrained Multi-Project Scheduling Problem

The Combinatorial Multi-mode Resource Constrained Multi-project Scheduling Problem (CMRCMPSP) is general and its covers the RCPSP, and its main extensions such as MRCPSP, RCMPSP, and MRCMPS. The choice of representation of a problem impacts its complexity and the search space (schedule options). A very specific representation significantly reduces the size of the search (options), and works on only a single problem instance. Nonetheless, a general representation, allows for more types of problems may be solved at the expense of searching a larger space.

The MRCPSP and MRCMPS can be performed by selecting one out of several different combinations of resource requirements (modes). The proposed method allows for unlimited modes. It first attempts to assign the required resources in the first mode (mode 1) for tasks, in case a required resource of mode 1 is not available, then it attempts to assign the required resources of mode 2, this logic is applied to all modes. Ultimately, if no resource requirements are met for at least one mode out of the possible set of modes of a given task, then it must wait until resource requirements are met for at least one mode. The sequential mode order allows project manager to assign ranks for different modes. The multiple modes flexibility covers the concept of resource-driven task duration defined in Wongwai and Malaikrisanachalee (2011), where tasks can start with partial resource requirements fulfilment. In the proposed method, a given task can have several modes, one of them may represent the least resource requirements to start a task.

Depending upon the application, the mode order may represent quality, speed, cost, etc. Project manager defines number of modes for each task according to current needs. Mode durations are independent of resource requirements by other modes. For instance, if mode 1 requires more resources in comparison of other modes, does not imply that duration of mode 1 will be shorter, it may be longer or shorter depending upon the process. Less equipment or fewer labours may work more efficiently than several other equipment or labours. Multiple modes provide additional flexibility in developing project schedules.

Many researchers have worked on several variations of the RCPSP such as RCMPSP and MRCPSP. The case of Multi-mode Resource Constrained Multi-Project Scheduling Problem (MRCMPSP) has not been addressed adequately due to its size and complexity. This research extends the MRCMPSP by proposing a new and general extension of all the aforementioned extensions of the RCPSP. The Combinatorial Multi-Mode Resource Constrained Multi Project Scheduling Problem (CMRCMPSP) is created. The proposed CMRCMPSP is capable to solve RCPSP, RCMPSP, MRCPSP, and MRCMPSP. Recall that MRCMPSP is the general extension of RCPSP found in literature and it allows several project tasks to be handled simultaneously, under precedence and resources constraints, and each task can have several modes. The figure 5 shows only one project with 6 tasks with start node (ST) and end node (EN) for illustrative purposes. Each task can have multiple modes, e.g., task 3 has 3 modes. Task 3, mode 1 requires resources R1 and R2 and the task 3's duration operating under this mode is 7 units of time. Task 3, mode 2 requires resources R2 and R3 and the task 3's duration operating under this mode is 8 units of time. Task 3, mode 3 requires resources R1 and R4 and the task 3's duration operating under this mode is 10 units of time.

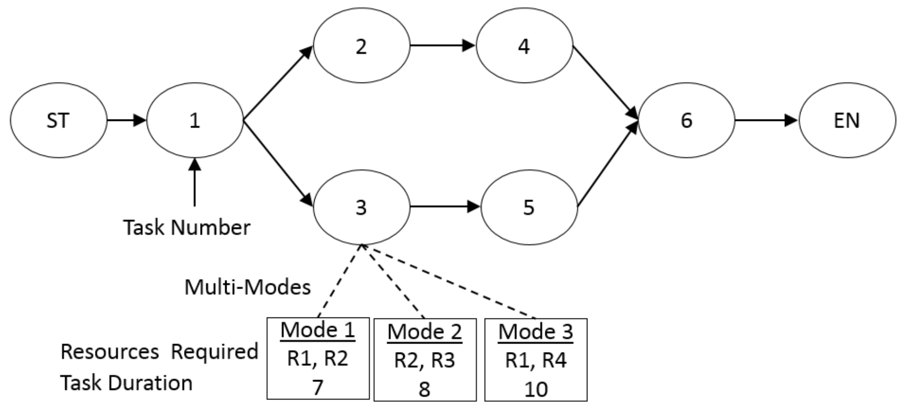


Figure 5: Task on Node of MRCMPSP model

The Combinatorial Multi-Mode Resource Constrained Multi Project Scheduling Problem (CMRCMPSP) allows several project tasks to be handled simultaneously, under precedence and resources constraints, and each task can have several modes. Naber and Kolisch [Naber and Kolisch, 2014] describe mode as a set of resources for executing a task over its entire predetermined fixed duration. In MRCMPSP and CMRCMPSP each task can be performed by selecting one out of several different modes. However, CMRCMPSP differs from MRCMPSP, because a mode in CMRCMPSP is no longer a set of resources only, but it is a set of combinatorial subsets of unknown required resources capable of executing a given task. In order to illustrate how CMRCMPSP differs from MRCMPSP a small example is shown in the Figure 6. It shows one project with 6 tasks along a start node (ST) and an end node (EN). Each task can have multiple modes, e.g., task 3 has 2 modes, mode 1 requires 2 welders, 3 cutters, R12, and nonrenewable resource R15. Task 3’s duration under mode 1 is 10 units of time. Task 3, mode 2, requires 3 welders, 1 cutter, R13 and R15 and the task 3’s duration operating under mode 2 is 12 units of time.

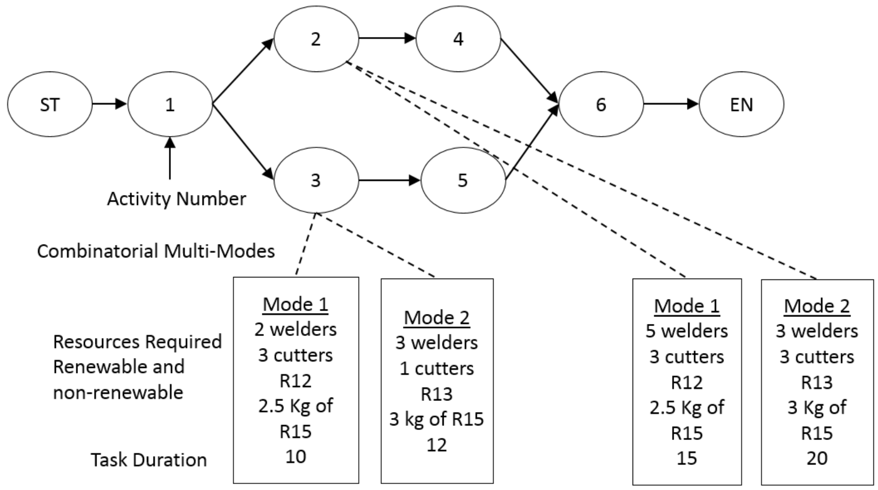


Figure 6: Task on Node of CMRCMPSP model

Table 7 describes the resources, their associated skills levels and their types (renewable or non-renewable). Notice that some resources can have more than one skill associated with them. The set of combinatorial subsets of unknown required resources can be formed with single-skilled resources in CMRCMPSP. The multi-skilled resources are used in this example to adhere to project environment reality. Very few approaches are capable of handling multi-skilled workers.

The various values in the tables show skill levels (1-Excelent, 2- Good, 3-Reasonable, 4-Poor). For instance, R1 has 3 skills, welding at skill level 1, cutting at skill level 2 and painting at skill level 4. The same logic is applied to all other resources. If no skill value is assigned to a resource, it is assumed to have no skills. Last row of Table 7 shows that there exist 8 welders, 6 cutters and 6 painters in the resource pool.

Table 7: Resource and Skill				
Resources	**Welding**	**Cutting**	**Painting**	**Type**
R1 (worker 1)	1	2	4	Renewable
R2 (worker 2)	1			Renewable
R3 (worker 3)	1			Renewable
R4 (worker 4)			1	Renewable
R5 (worker 5)		1	3	Renewable
R6 (worker 6)		1		Renewable
R7 (worker 7)	1	2	3	Renewable
R8 (worker 8)	1	2	3	Renewable
R9 (worker 9)	1			Renewable
R10 (worker 10)	1		2	Renewable
R11 (worker 11)	1	2		Renewable
R12 (Machine 1)				Renewable
R13 (Machine 2)				Renewable
R14 (Machine 3)				Renewable
R15 (Electrode)				Non-renewable
Quantity	8	6	6	

CMRCMPSP employs the term combinatorial due to the fact it allows for a finite combinatorial number of options to execute tasks. This differs widely from a standard multi-mode approach because number of modes does not define number of different options to perform a task. In the previous extension of multi-mode RCPSP, a given task with two modes has only two options for being executed. However, in the proposed approach a task with two modes, as it will be shown, can provide more than two options. In Figure 6, Task 3, Mode 1 requires 2 welders out of 8, 3 cutters out of 6, and resources R12, R15. Task 3, Mode 2 requires 3 welders out of 8, 1 cutter out of 6, and resources R13, R15. In this research, the combinatorial mode is defined as a set of combinatorial subsets of required resources. Task 3, Mode 1, the 2 welders required is the first subset, the 3 cutters is the second subset, and so on so forth. For each required subset, a search among a combinatorial number of resources is described by equation 1:

$$\binom{n}{p} = \frac{n!}{(n-p)!\,p!} \tag{1}$$

The number of combinations p of n elements where p is the required quantity of resources with a given skill and n is the available quantity of resources in the resource pool capable of meeting the required skill. In the above example, Task 3, Mode 1 requires 2 welders out of 8, 3 cutters out of 6, R12, and R15. Task 3, Mode 2 requires 3 welders out of 8, 1 cutter out of 6, R13, and R15. There are a total of 896 different ways to perform Task 3, not just 2. Equation (2) shows the number of alternative combinations for welders in mode 1:

$$\binom{8}{2} = \frac{8!}{(8-2)!\,2!} = 28 \tag{2}$$

Equation (3) shows the number of different combinations for cutters in mode 1.

$$\binom{6}{3} = \frac{6!}{(6-3)!\,3!} = 20 \tag{3}$$

$\binom{8}{2}\binom{6}{3} = 560$ different sets of resources to perform Task 3 by selection Mode 1

Equation (4) shows the number of alternative combinations for welders in mode 2.

$$\binom{8}{3} = \frac{8!}{(8-3)!\,3!} = 56 \tag{4}$$

Equation (5) shows the number of different combinations for cutters in mode 2.

$$\binom{6}{1} = \frac{6!}{(6-1)!\,1!} = 6 \tag{5}$$

$\binom{8}{3}\binom{6}{1} = 336$ different set of resources to perform Task 3 by selection Mode 2

Therefore, there are (560 + 336) 896 alternate ways to perform task 3. A general equation for calculating the number of options to perform a task in the CMRCMPSP is given by equation 6.

$$\sum_{m \in M_{i,j}} \left(\prod_{c \in C_{i,j,m}} \frac{n!}{(n-p)!\,p!} \right) \tag{6}$$

For the above example, for Mode 1, the number of combination of welders and cutters is given by $\binom{8}{2}\binom{6}{3} = 560$, whereas for Mode 2 the number of combination of welders and cutters is given by $\binom{8}{3}\binom{6}{1} = 336$.

$\mathrm{m} \in \mathrm{M}_{1,3} = \{\text{Mode 1}, \text{Mode 2}\}$

$\mathrm{c} \in \mathrm{C}_{1,3,1} = \{2 \text{ welders out of } 8, 3 \text{ cutters out of } 6, \mathrm{R12}, \mathrm{R15}\}$

$\mathrm{c} \in \mathrm{C}_{1,3,2} = \{3 \text{ welders out of } 8, 1 \text{ cutter out of } 6, \mathrm{R13}, \mathrm{R15}\}$

$\mathrm{rn}_{(\mathrm{v,j,m,c})} = \mathrm{rn}_{(1,3,1,1)} = 2;\ \mathrm{rn}_{(1,3,1,2)} = 3$

$$\sum_{m \in M_{i,j}} \left(\prod_{c \in C_{i,j,m}} \frac{n!}{(n-p)!\,p!} \right) = \frac{8!}{(8-2)!\,2!} * \frac{6!}{(6-3)!\,3!} * 1 * 1 + \frac{8!}{(8-3)!\,3!} * \frac{6!}{(6-1)!\,1!} * 1 * 1 = 896$$

Current literature does not address the above combinatorial problem.

3.2 The problem model

Figure 7 provides an overview of the problem model. It shows three simultaneous projects (separated by a dotted line). Each task, represented by a node, can have multiple modes, e.g., Task 3 of Project 1 has two modes. Each mode can require subsets of resources as described in the previous section. Resource can be renewable or non-renewable. In Figure 7, double circle for Tasks 1 of Project 1 indicates that some work has already been performed on this task and the model takes that into account. Tables 8 and 9 describe the proposed solution method in terms of task and resource.

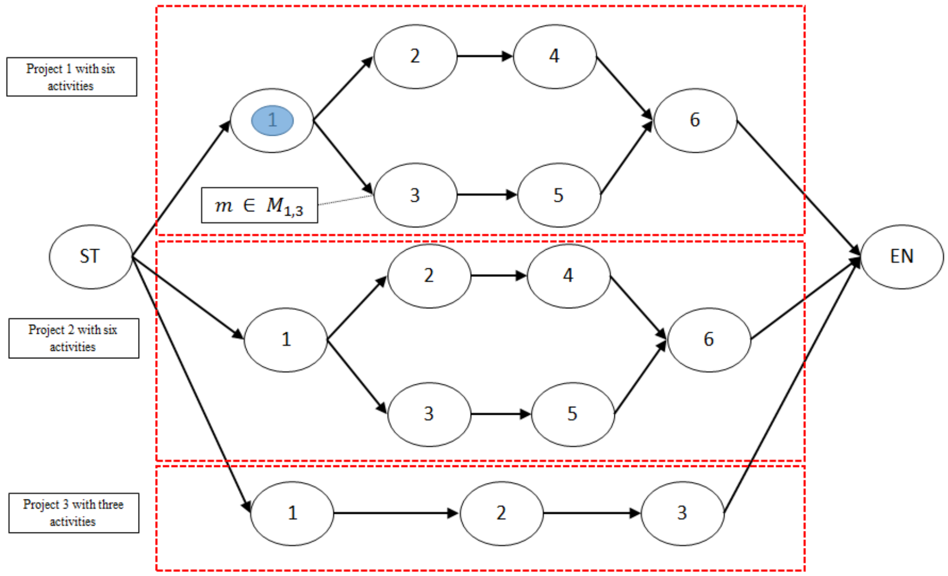


Figure 7: Task on node representation for multiple projects

Table 8: Task Modelling

Modelling	Description
Multi-projects	Tasks can belong to several different projects
Combinatorial	Tasks can have multiple modes. Each mode has its own subsets of resources.
Precedence	Tasks follow a precedence relationship.
Temporal Interruptions.	Tasks may have temporal constraints such that they can only be done at a certain time.

Table 9: Resource Modelling

Modelling	Description
Multi-skilled	Resources can have more than one skill
Multiple calendars	Each resource can have its own calendar.
Renewable, non-renewable resources, doubly constrained resources	There are three types of resources. Renewable resource are not consumed, e.g. facilities, equipment, workers, etc. Nonrenewable resource is a resource that is consumed when a task is carried out, e.g. raw material, power, fuel, etc. A plan of nonrenewable resources is considered. Delivery dates of non-renewable resources are considered in this model (doubly constrained resource)
Multiple priority rules	Each resource can have its own list of priority rules, e.g., rule for resource 1 (welding resource for example) do not apply to resource 2 (painting worker) and vice-versa.
Maintenance and Over times are allowed	Each resource can have maintenance and idle times. Human resources can have overtime.

3.3 Mathematical formulation of the CMRCMPSP model

The focus of most current approaches is to formulate the problem as an optimization problem with a goal of finding the optimal solution. Several assumptions are made in order to reduce mathematical complexity. Various models use heuristic or a genetic algorithm to make the problem tractable. To obtain an optimal solution for the proposed CMRCMPSP using mathematical formulation is intractable and computational prohibitive. After over simplification, many authors have been able to handle up to 51 tasks (Rehm and Thiede , 2012). Formulation of the RCPSP model is as described below (Vanhoucke, 2013).

$$f: minimize\ s_n \tag{7}$$

Subject to:

$$s_i + d_i \leq s_j\ \forall (i,j) \in A \tag{8}$$

$$\sum_{i \in S(t)} r_{i,k} \leq a_k\ \forall k = 1, \dots, K; t = 1, \dots, T \tag{9}$$

$$s_1 = 0 \tag{10}$$

$$s_i \in\ Z^+\ i = 1, \dots, n \tag{11}$$

The objective function in equation (7) minimizes the entire project duration by minimizing the start time of the dummy end task. The constraint set given in equation (8) maintains the finish-start precedence relations with a time-lag of zero among the tasks. Equation (9) is used to model the availability of the renewable resources. It stipulates that the sum of the resource requirements of all activities in progress at time period [t-1, t] is not allowed to exceed the availability of the renewable resource over the complete time horizon of the project T. Equation (10) forces the start task, and hence the project, to start at time zero. Equation (11) ensures that the task start times assume nonnegative integer values. The formulation of the MRCPSP model described by (Vanhoucke, 2013) is as follows:

$$Minimize \sum_{t=0}^{T} t * x_{n,1,t} \tag{12}$$

Subject to:

$$\sum_{m \in M_i} \sum_{t=0}^{T} x_{i,m,t} = 1 \; i \in I \tag{13}$$

$$\sum_{m \in M_i} \sum_{t=0}^{T} t * x_{i,m,t} - \sum_{m \in M_i} \sum_{t=0}^{T} (t - d_{j,m}) \, x_{j,m,t} \leq 0 \tag{14}$$

$$\sum_{i \in I} \sum_{m \in M_i} \sum_{t=0}^{T} r_{i,m,k} * x_{i,m,t} \leq AR_t \tag{15}$$

$$x_{i,m,t} \in \{0,1\} \; \forall i, m, t \tag{16}$$

The objective in equation (12) is to minimize the entire project duration by minimizing the start time of the end task. The constraint set given in equation (13) enforces that tasks can be executed only once during entire time horizon by selecting one mode out of the set of task modes. The constraint set given in equation (14) maintains the finish-start precedence relations with a time-lag of zero among the tasks. Equation (15) is used to model the limited availability of the renewable resources. It stipulates that the sum of the resource requirements of all tasks in progress during time period [0, T] is not allowed to exceed the availability of the renewable resource over the complete time horizon of the project T. Equation (16) ensures that $x_{i,m,t}$ variables assume integer values.

The mathematical formulations of RCPSP and MRCPSP fall in the class of Mixed-Integer Linear Programming models (MILP). Definition of MILP can be found in (Benichou et. at., 1971). The proposed CMRCMPSP model falls in the class of the Mixed-Integer Nonlinear Programming models. The problem is nonlinear if the objective function is nonlinear and/or the feasible region is determined by nonlinear constraints (MIT – Nonlinear, 2015). The CMRCMPSP model can be mathematically described as follows:

$$f = \min((\sum_{i \in I}(TC_i + \sum_{j \in J_i} TC_{i,j}) + TCR + TCS) \tag{17}$$

Subject to:

$$\sum_{m \in M_{i,j}} x_{i,j,m,t} = 1 , \forall j \in J_i , \forall i \in I, \text{for all } t \in [0, T] \tag{18}$$

$$x_{i,j,m,t} * t \geq st_{i,j} \ \forall j \in J_i , \forall i \in I \tag{19}$$

$$ft_{i,j} \geq st_{i,j} + d_{i,j,m} \ \forall j \in J_i , \forall i \in I, \forall m \in M_{i,j} \tag{20}$$

$$rn_{(i,j,m,c)} * x_{i,j,m,t} \leq AR_{((i,j,m,c),t)} , \forall i \in I, \forall j \in J_i , \forall m \in M_{i,j} , \forall c \in C_{i,j,m} , \forall t \in [0, T] \tag{21}$$

$$AR_{((i,j,m,c),t)} \leq \sum_{r \in R_{i,j,m,c}} z_{(Re_{(i,j,m,c,r)},t)} * u_{(Re_{(i,j,m,c,r)},t)} , \forall i \in I, \forall j \in J_i , \forall m \in M_{i,j} , \forall t \in [0, T] \tag{22}$$

$$\sum_{i \in I} \sum_{j \in J_i} \sum_{m \in M_{i,j}} \sum_{c \in C_{i,j,m}} \sum_{r \in R_{i,j,m,c}} u_{(Re_{(i,j,m,c,r)},t)} \leq 1 , , \forall t \in [0, T] \tag{23}$$

$$\sum_{m \in M_{i,b}} t * x_{i_1,b,m,t} \geq \sum_{m \in M_{i,a}} \left(t + d_{i_1,a,m}\right) * x_{i_2,a,m,t} , \forall (a, b) \in P, \forall i_1, i_2 \in I, \quad for\ all\ t \in [0, T] \tag{24}$$

$$TCS_{(i,j,m,c,r)} \geq cs_{(sr)} * x_{i,j,m,t} , \forall i \in I, \forall j \in J_i , \forall m \in M_{i,j} , \forall c \in C_{i,j,m} , \forall r \in R_{i,j,m,c} , \forall \text{sr} \in \text{SRE}_{(i,j,m,c,r)}, \forall t \in [0, T] \tag{25}$$

$$TCS \geq \sum_{i \in I} \sum_{j \in J_i} \sum_{m \in M_{i,j}} \sum_{c \in C_{i,j,m}} \sum_{r \in R_{i,j,m,c}} TCS_{(i,j,m,c,r)} \ for\ all\ t \in [0, T] \tag{26}$$

$$AW_{((i,j,m,c),t)} \leq \sum_{y \in Y_{i,j,m,c}} z_{(Nr_{(i,j,m,c,y)},t)} \;,\forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, ,\forall c\, \in C_{i,j,m}\, ,\forall t\, \in [0,T] \quad (27)$$

$$w_{i,j,m,c} * x_{i,j,m,t} \leq AW_{((i,j,m,c),t)}, \forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, ,\forall c\, \in C_{i,j,m}\, ,\forall t\, \in [0,T] \quad (28)$$

$$AR_{((i,j,m,c),t+1)} \leq AR_{((i,j,m,c),t)} + ARD_{((i,j,m,c),t)} - MRD_{((i,j,m,c),t)}, \forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, , \forall c\, \in C_{i,j,m}\, ,\forall t\, \in [0,T] \quad (29)$$

$$AW_{nr,t+1} \leq\; AW_{nr,t} + AWD_{nr,t}, \forall t\, \in [0,T] \quad (30)$$

$$TCR \geq \sum_{nr \in NR} cw_{nr} TW_{nr} + \sum_{nr \in NR} caw_{nr} TAWD_{nr} + \sum_{re \in RE} cr_{re} TRe_{re} + \sum_{re \in RE} car_{re} TARD_{re} + \sum_{re \in RE} cro_{re} TRO_{re} + \sum_{re \in RE} crm_{re} TRM_{re}\, , for\ all\ \ t\, \in [0,T] \quad (31)$$

$$TC_{i,j} \geq c_{i,j} * \left(\sum_{m \in M_{i,j}} \left(t + d_{i,j,m}\right) * x_{i,j,m,t} - dd_{i,j} \right), \forall i\, \in I, \forall j\, \in J_i\, , for\ all\ \ t\, \in [0,T] \quad (32)$$

$$TC_i \geq c_i * \left(\sum_{m \in M_{i,N}} \left(t + d_{i,N,m}\right) * x_{i,N,m,t} - dd_i \right), \forall i\, \in I, for\ all\ \ t\, \in [0,T] \quad (33)$$

$$x_{i,j,m,t}\; \in \{0,1\}\, \forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, ,\forall t\, \in [0,T] \quad (34)$$

$$u_{(Re_{(i,j,m,c,r)},t)}\; \in \{0,1\}\, \forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, ,\forall c\, \in C_{i,j,m}\, ,\forall r\, \in R_{i,j,m,c}\, \forall t\, \in [0,T] \quad (35)$$

$$z_{(Re_{(i,j,m,c,r)},t)}\; \in \{0,1\}\, \forall\, i\, \in I, \forall j\, \in J_i\, ,\forall m\, \in M_{i,j}\, ,\forall c\, \in C_{i,j,m}\, ,\forall r\, \in R_{i,j,m,c}\, ,\forall t\, \in [0,T], AR_{re,t} \in Z^{+}\, , AW_{nr,t}\; Q\;\; ,\forall re\, \in RE, \forall nr\, \in NR \quad (36)$$

Objective function given by equation (17) is to minimize the total tardiness over all projects. Constraint set defined by equation (18) ensures that all tasks are scheduled only once by selecting only one mode. Constraint set defined by equation (19) calculates start time for all tasks. Constraint set defined by equation (20) calculates finish time for all tasks. Constraint set defined by equation (21) imposes the maximum quantity of available renewable resource with capability c at time t for any project, task, and mode. Constraints set (22) calculate the maximum level of available renewable resources with capability c at time t according to the parameter z and binary variable u. It is a nonlinear constraints. Constraint set (23) ensures that resources are not assigned for more than one project, task, mode, capability at the same time t. Constraint set (24) reflects the precedence relationships among the tasks of all projects. Constraint set (25) calculates the total cost associated of choosing a capability s of a renewable resources at time t. Resource have multi-skills/capabilities, the first skill of the set (index 1) is the less costly, two is more costly, and so on so forth. This is a non-linear constraint, both are variables. Constraint set (26) calculates the total cost associated of choosing the capabilities of renewable resources for all time t. Constraint set (27) imposes the maximum level for the nonrenewable resource with capability c usage at time t over all projects. Constraint set (28) imposes the maximum quantity of available nonrenewable resources with capability c at time t for any project, task, and mode. Constraints set (29) calculate the maximum level of renewable resources with skill/capability c at time t according to the sum of the current quantity of renewable resources with skill/capability c at time t and renewable resources with skill/capability c added on time t minus the quantity of renewable resources with skill/capability c on maintenance. Constraints set (30) calculate the maximum level of nonrenewable resources nr at time t according to the sum of the current quantity of nonrenewable resources nr on hand and nonrenewable nr on order delivered on time t. Constraint set (31) calculates the total cost of the general renewable and nonrenewable resources. Constraint set (32) calculates the penalty cost due to tardiness values for each task. Constraint set (33) calculates the penalty cost due to tardiness values for each project. Constraint set (34) specifies the feasible ranges for the decision variables. Constraint set (35) specifies the feasible ranges for the decision variables. Constraint set (36) specifies the feasible ranges for the decision variables.

3.4 The Simulation Method

None of the existing methods, exact or heuristic have developed an approach to solve the CMRCMPSP. Mathematical modelling of the CMRCMPSP is cumbersome. This research creates a discrete event based simulation to model and solve the CMRCMPSP. The solution method is called STREAM (Short-Term Resource Allocation and Management). In addition to simulation, STREAM has several other algorithms. Section 3.3.1 describes the Discrete Event Simulation (DES) algorithm. Section 3.3.2 describes the Multi-mode Scheduling and Queue Management (MSQM) algorithm. Section 3.3.3 describes the Priority Rules Management (PRM) algorithm. Section 3.3.4 describes the Worker Skills Management (WSM) algorithm. Section 3.3.5 describes the Task Duration Management (TDM) algorithm. Section 3.3.6 describes the Task Precedence Management (TPM) algorithm. The six algorithms were formulated and coded specifically to handle the CMRCMPSP model.

The solution method was primarily created with the goal to start tasks as early as they become eligible to be performed. If a given task is eligible to start and all required resources are available, the task will be started immediately. In order to solve resource conflicts among tasks during the simulation, STREAM uses multiple priority rules. STREAM attempts to require the least possible amount of resources to avoid waiting times for tasks. E.g., if 20 resources exist in the resource pool, STREAM may not assign all of them. STREAM assigns resources limited to 20 as long as they avoid resource conflicts among tasks. If more than 20 resources are required for tasks, some tasks will have to wait until required resources become available. Delays are automatically calculated, no over allocation in resources are permitted.

3.4.1 Discrete Event Simulation

Simulation utilizes to a wide array of methods to mimic the behavior of complex systems [Kelton et al., 2002]. Simulation can be used to model a static or a dynamic environments. The focus of this work is on the dynamic systems. Figure 8 is an example of a continuous dynamic system. A weather system in which temperatures change overtime is an example of continuous dynamic system. A warehouse material arrival, material delivered, purchase orders, inventory levels are examples of a discrete dynamic system. Such systems follow a step function. Figure 9 shows a discrete dynamic system. Matloff [Matloff, 2008] provides an example where events decrease and increase the inventory levels, which are discrete variables. Chryssolouris [Chryssolouris , 2005] defines discrete-event simulation as following:

"Most simulation software programs model a manufacturing system as it evolves overtime by a representation in which the variables that track the systems state (the state variables) change instantaneously at separate points in time. These points in time, are the ones at which an event occurs, where an event is defined as an instantaneous occurrence that may change the state of the system. A model of this type is called a discrete event simulation model".

In shorts, for continuous systems, the state of the system changes continuously overtime, whereas for discrete dynamic systems, the state change occurs at discrete points in time. Program a discrete dynamic system is difficult because several tasks may have to be performed in parallel [Matloff, 2008].

"*Simulation programming can often be difficult—difficult to write the code, and difficult to debug. The reason for this is that it really is a form of parallel programming, with many different tasks in progress simultaneously, and parallel programming can be challenging. For this reason, many people have tried to develop separate simulation languages, or at least simulation paradigms (i.e. programming styles) which enable to programmer to achieve clarity in simulation code. Special simulation languages have been invented in the past, notably SIMULA, which was invented in the 1960s and has significance today in that it was the language which invented the concept of object-oriented programming that is so popular today. However, the trend today is to develop simulation libraries which can be called from ordinary languages such as C++, instead of inventing entire new languages".*

Figure 8 shows a system in which its behavior changes continuously overtime. It can be described by a function with a dependent variable x(t) and independent variable t [Cury, 2001]. In Figure 9 events are represented by letters α, β and λ. Each event affects the system differently depending upon its current state.

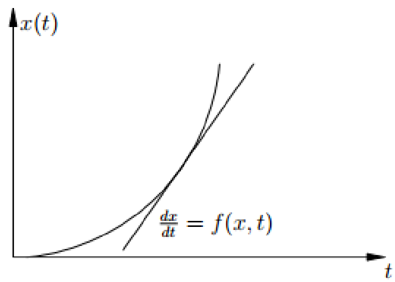


Figure 8:Typical behavior of Continuous Dynamic System

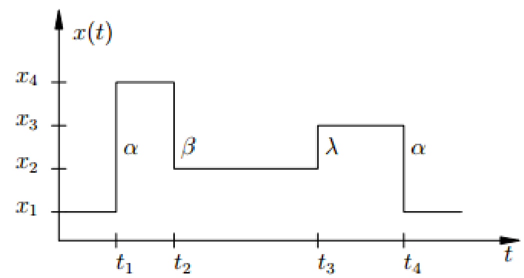


Figure 9: Typical behavior of Discrete Dynamic System

The simulation algorithm is based on discrete dynamic events and events can be inserted or deleted from the list of events during the simulation period. Events such as: start task, finish task, start resource maintenance, finish resource maintenance, can be used. Simulation period is the time horizon ranging from a date defined by the manager and date of the last task executed by simulation. Detailed description of general discrete dynamic event systems can be found in [Chryssolouris, 2005] and [Matloff, 2008].

The STREAM described in this work is based on a discrete dynamic event simulation using deterministic task duration. However, probabilistic task duration time (t) can be used by utilizing the approximated beta distribution where optimistic time is given by (a), pessimistic time is given by (b), and is the most likely time is given by (m). Expected task duration time (t) is approximated by the weighted average as (a+4m+b)/6 [Vanhoucke 2013]. STREAM prefers to use deterministic time values due to lack of reliable historical data, particularly when using multiple modes and multiple skills. Tasks can be rescheduled based on the current status of the system. The actual remaining time to finish tasks is also taken into account [1-Pinha et al. 2015, 2-Pinha et al. 2015]. Based on the nature of short-term project management, this research is concerned with discrete dynamic systems using deterministic task durations. Figure 10 provides a hierarchical representation of different classes of systems. Shaded boxes show scope of STREAM.

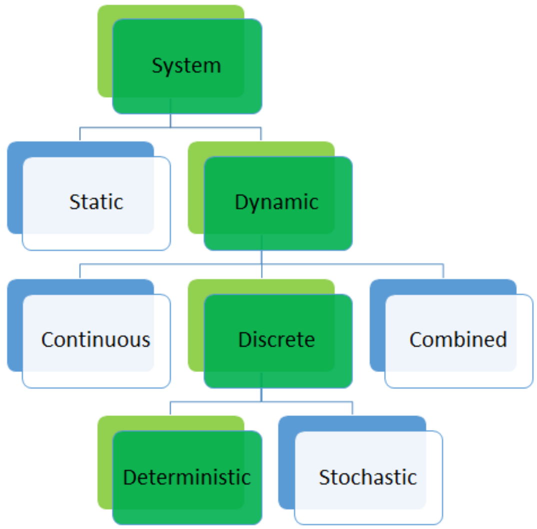


Figure 10: Overview Static and Dynamic System

The core algorithm of the Discrete Event Simulation (DES) is shown in Table 10. It generates outputs by scheduling tasks where the start date is specified by the manager. The *simulation clock*, a global variable, is initialized in the line 2. This variable controls how the clock moves in the simulated period. The project manager provides its initial value. Modes of eligible tasks join the queue (line 3). The “Run Queue” procedure (line 4) is activated to create events of finish tasks based on eligible modes. Also, priority rules are triggered by the “Run Queue” procedure in order to identify the winning mode in the queue when resource conflict exists. The “Run Queue” algorithm is described in Table 12. While events exist, the simulation keeps running until there are no more events (lines 5- 25). Line 6 selects the most imminent event to occur. Line 7 verifies if the most imminent event is to finish a task, if so, the *simulation clock* is updated in line 8 and the *finish time* is registered in line 9. In line 10, resources are released. The task’s successors may also be released if the number of predecessors executed by the simulator flagged as done is equal to the number of predecessors (line 14-15). In order to release a successor node, all predecessor nodes must be finished. Lines 16 to 18 register dates and modes are added to the queue. The executed event is removed from the list of events (line 22). Line 23 calls the procedure "Run Queue."

Table 10: Discrete Event Simulation (DES) Algorithm

```
1:  procedure DES
2:      Initialize simulation clock
3:      Modes of eligible tasks are sent to Queue
4:      Run Queue
5:      while exist events to be executed do
6:          Select the most imminent event to occur
7:          if Event type = Finish task then
8:              SimulationClock ← ExecutionDate
9:              Finish ← SimulationClock
10:             Release Resources
11:             Release Successors
12:             if Lagtime = 0 then
13:                 for Suc ← 1, NSuc_ij do
14:                     N. Pred. done ← N. Pred. done+1
15:                     if Number of Predecessors = N. Pred. done then
16:                         Eligible ← True
17:                         EligibleDate ← SimulationClock
18:                         Add modes to the queue
19:                     end if
20:                 end for
21:             end if
22:             Remove the executed event
23:             Run Queue
24:         end if
25:     end while
26: end procedure
```

3.4.2 Multi-mode Scheduling and Queue Management

This section presents the Multi-mode Scheduling and Queue Management (MSQM) algorithm. It establishes a concept of eligible modes, instead of eligible tasks as found in [Vanhoucke, 2013]. The Parallel Schedule Generation Scheme (PSGS) presented in [Vanhoucke, 2013] was expanded and named as Parallel Mode Schedule Generation Scheme (PMSGS). The PMSGS was presented in [Pinha et al, 2015]. Figure 11 shows a standard queue scheme commonly found in literature where tasks must select a queue to join and the resource must select a queue to server. Figure 12 also shows a standard representation of a queue found in the literature where tasks must select one resource out many [Khoshnevis, 1994].

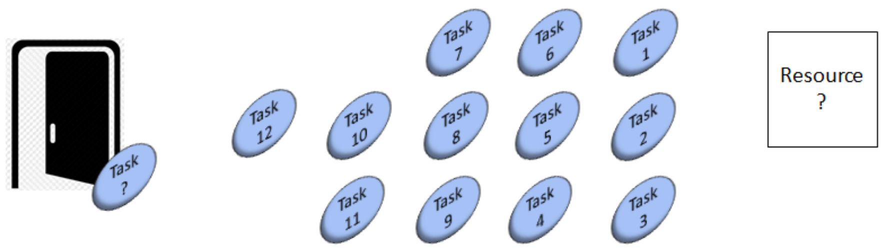


Figure 11: Selection of a queue to join and selection of a queue to serve

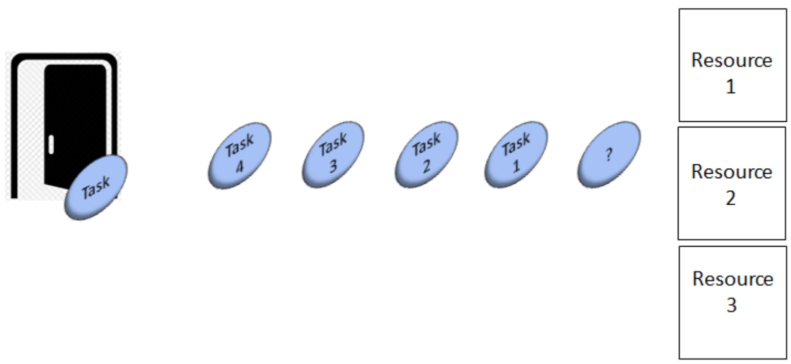


Figure 12: Selection from one out of several servers

It is difficult to organize queues when a single task has 896 different options and each resource has its own rules. The traditional queue management schemes do not support and cannot be applied to queue problem in CMRCMPSP. Table 11 shows resource requirements for Tasks 2 and 3 of Project 1. Figure 13 recalls the example 1 described in Section 3.1.

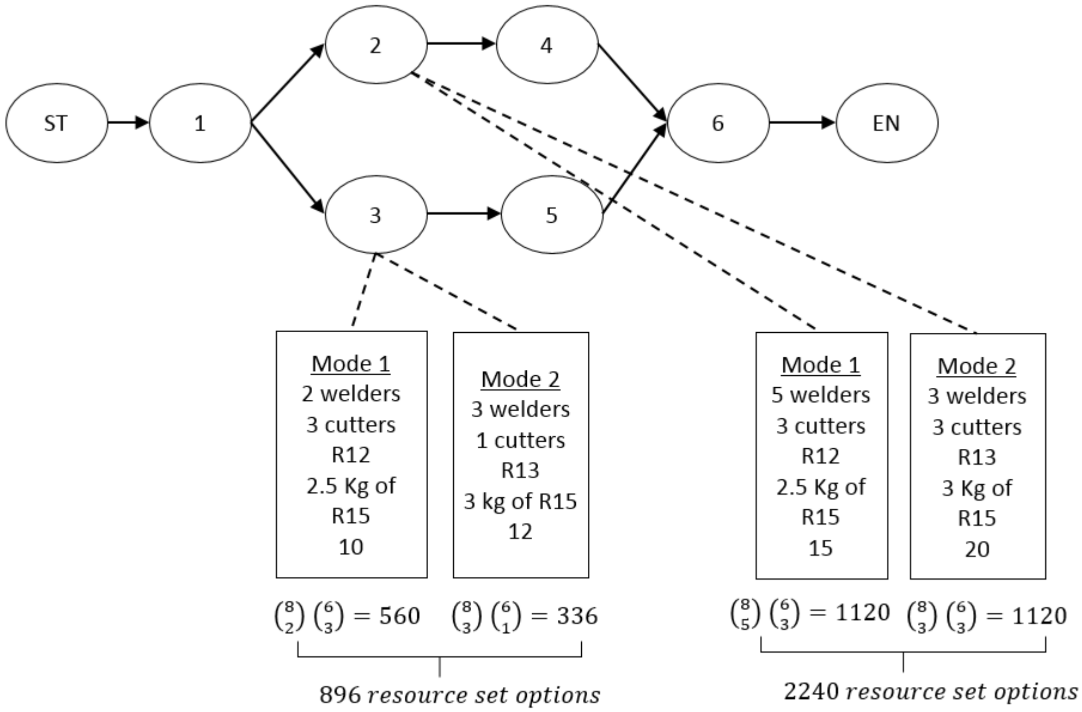


Figure 13: Task on node for the CMRCMPSP

Table 11: Mode description for task 2 and task 3 of project 1.			
Task	**Mode**	**Resource Requirements**	**Duration**
2	1	2 welders out of 8 , 3 cutters out of 6, R12 and 2.5 Kg of R15	10
	2	3 welders out of 8, 1 cutter out of 6,, R13 and 3 Kg of R15	12
3	1	5 welders out of 8, 3 cutters out of 6,, R12 and 2.5 Kg of R15	15
	2	3 welders out of 8, 3 cutters out of 6,, R13 and 3 Kg of R15	20

Figure 14 shows the complexity of the queue management for the CMRCMPSP. It only shows the 4 eligible modes associated with the eligible Tasks 2 and 3, of Project 1. The four oval shapes represent modes. The white oval shapes are modes related to Task 3, of Project 1 and the black ones are associated with Task 2, of Project 1. For simplification purposes, only Modes 1 of both tasks have association with their resource combinations.

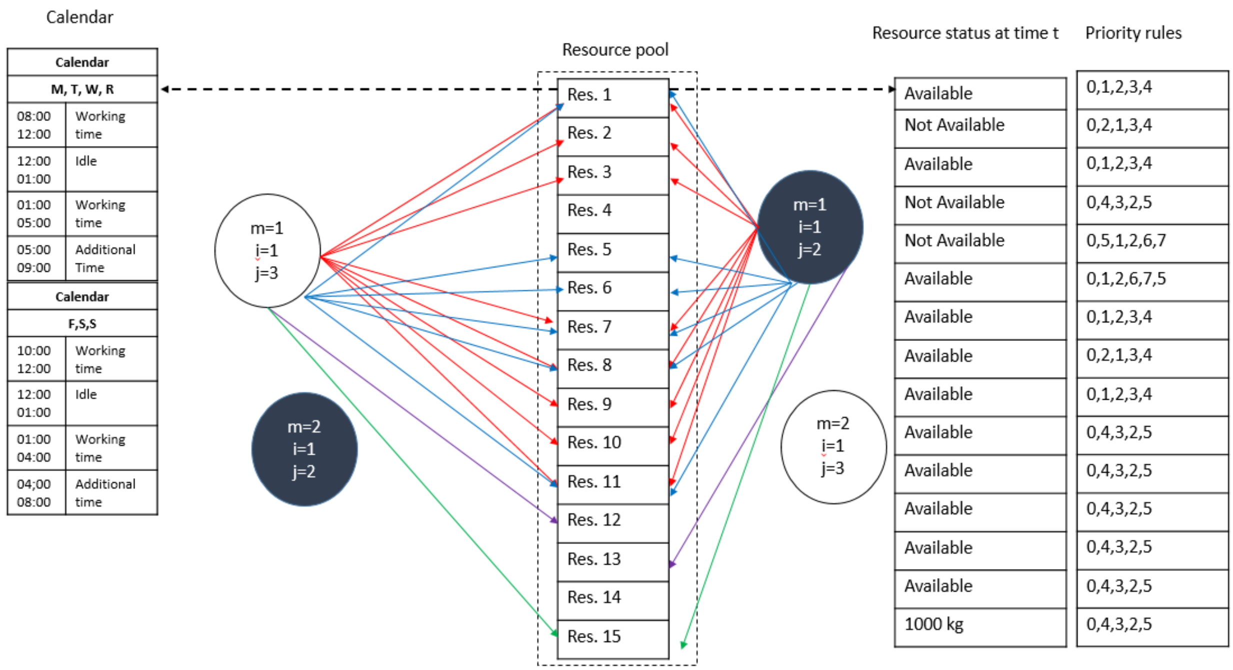


Figure 14: CMRCMPSP Queue Schema

As can be seen from the Figure 14 several links exist between modes and corresponding resources. These links are the resource requirements for each mode. Unavailability of a resource out of several resources can hold a task for being started. Resources can be renewable or nonrenewable. The queue management approach utilized by STREAM to solve the CMRCMPSP is as follows:

1. Break individual task into their respective modes.
2. There is no predefined and organized queue.
   a. There is no benefit to organizing a queue ahead of time because each resource has its own priority rules. It is difficult to organize a queue if the same mode requires different resources, and their rules are in conflict to each other.
   b. Eligible modes from eligible tasks are listed as possible candidate for execution.
   c. The decision of which mode wins among eligible modes is performed at each decision point in time.

3. Verify if all the required resources for the first mode of a given task are available.

    a. If so, verify if all the required resources for the first mode of given different task are available.

    b. If no common resources exist between these two modes, mode 1 of the first task is a tentative winner and further comparison is performed with another mode.

    c. If a common resource exists, a priority rule defines which mode wins for the first pair comparison

    d. If the modes are still tied up, a second priority rule is used, and so forth. The comparison is performed while modes exist. In the end, only one mode wins

In order to deal with the queue management complexity in CMRCMPSP a new approach was utilized. The PMSGS selects and ranks eligible modes instead of eligible tasks at each decision point time through the simulation period. Eligible modes are those that are connected to eligible tasks. Different tasks can have modes that require a same set of resources, at the same time. Priority rules, as described in section 3.4.3, based on heuristics corresponding to a given application were developed to handle such conflicts. A separate procedure named "Run Queue" was implemented to control priorities, dynamically. If one priority rule is not enough to break a tie among modes, the second priority rule is used. Once a mode has been selected, other modes related to that task are dropped from the mode queue. This process is performed for all eligible modes simultaneously, at a given decision point time in PMSGS.

Figure 15 shows a hypothetical PMSGS scenario for Tasks 2 and 3. Due to the precedence order, Task 1 must be finished before Tasks 2 and 3. Assuming that Task 1 has been completed, Tasks 2, and 3 become eligible to start and therefore need to be scheduled at the given simulation clock time. Their modes, therefore, are automatically eligible and if resource conflict exists, the priority rules may have to be triggered. Notations i, j, and m are index for projects, tasks and modes, respectively.

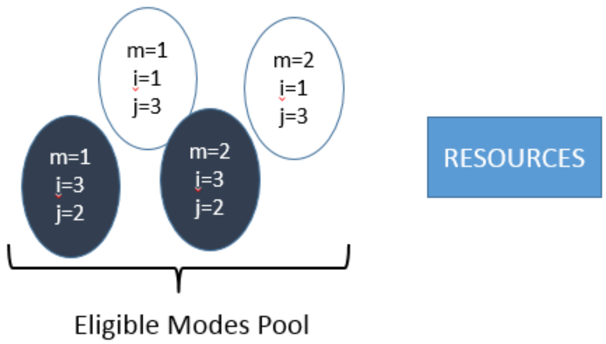


Figure 15: Hypothetical Scenario

Table 12 summarizes the Run Queue Algorithm. This procedure verifies if required resources are available for current modes in the queue and which mode currently has the highest priority, if resource conflict exists. The term current priority is used because modes are not sorted into the queue to be scheduled. When this procedure is called at a given simulation clock, the last priorities of modes may have nothing do to with the current situation. One mode which could have been in the second priority at the last decision time, may have the last priority at the current decision time. Modes which might have been added to the queue at to the current simulation clock may completely change priorities in the queue. Therefore, modes in queue change dynamically, according to their priorities at each decision point. In other words, whenever the procedure "Run Queue" is called, priorities are set dynamically for a given mode.

Line 2 of the Run Queue algorithm verifies if number of current modes in the queue is equal to one, if so, there is no competition among modes and then resources are required if they are available (Line 3). If resources are available, the algorithm registers the start time, requires resources and remove mode of queue (lines 4 to 6). If there is more than one mode in the queue, the algorithm goes over all the current modes (line 8). Each mode in the queue has an index *q1*. Line 10 verifies if resources required for the mode with index *q1* are available, if so, the algorithm verifies for all other modes with index different than *q1*, in this case *q2*. From lines 12 to 24, the algorithm searches for the highest priority mode. If modes require common resources, the search is performed by selecting two modes at time, and one is selected based on the priority rule. The final winner is the one which has the highest priority over all the modes at a given decision time. Lines 25 to 33, once a winner has been found, the start time is registered, resources are required, the event of finish task is added to the list of events and modes are removed from the queue. Line 25 checks if resource is not available for the competing mode with index *q2*, at this decision time or if no common resource exists between mode with index *q1* and *q2*. If one of this condition is false, it means no competition exists among the two modes and mode with index *q1* wins. While there are modes in queue which require available resources, event of finish task is added to the list of events. Events are added through "*Run Queue*" procedure in lines 4 and 26. The variable q is the index for current modes in the queue ranging from $1< q < NMQ_t$. NMQ_t is the number of current modes in the queue at time t; and MQ_{qt} is the mode with index q in the queue at time t.

Table 13 describes the start time recording procedure used by the Run Queue algorithm. It aims to find when a given task is able to start based on the current *simulation clock* (line 2). Also projected $Finish_{event}$ is calculated (line 4 or 6). A variable *Key* is assigned to be the concatenated code between the date of the projected $Finish_{event}$ and the *Task Index* (line 8). Line 9 the event of *Finish Task* is added to the list of events.

Table 12: Multi-mode Scheduling and Queue Management (MSQM)

```
procedure RUN QUEUE
    if NMQ_t = 1 then
        if (All resources are available for the mode) then
            Register start time(1)
            Require resources(1)
            Remove mode of queue(1)
        end if
    else
        for q1 ← 1, NMQ_t do
            if (All resources are available for MQ_{q_1 t}) then
                winning ← q_1
                AvailableRes ← False
                for q2 ← 1, NMQ_t do
                    if (winning ≠ q2) then
                        if (All resources are available for MQ_{q_2 t}) then
                            AvailableRes ← True
                            if CR(MQ_{(winning)t} and MQ_{q_2 t}) then
                                ComResource ← True
                                MQ_{(winning)t} ← Priority Rules
                            else
                                ComResource ← False
                            end if
                        end if
                    end if
                end for
                Register start time(Winning)
                Require resources(Winning)
                Remove mode of queue(Winning)
            end if
        end for
    end if
end procedure
```

Table 13: Start Time Recording Algorithm (DES/MSQM)

procedure Register Start Time(Task Index)
    $Start \leftarrow (TaskIndex, SimulationClock)$
    if $Duration \neq 0$ **then**
        $Finish_{event} \leftarrow Start + TaskDurationManagement$
    else
        $Finish_{Event} \leftarrow Start$
    end if
    $Key \leftarrow (Finish_{Event}, TaskIndex)$
    Add Event$(Key, "FinishTask", Finish_{Event}, TaskIndex)$
    APM$(TaskIndex)$
end procedure

3.4.3 Priority Rules Management

Vanhoucke [Vanhoucke, 2013] specified four types of priority rules: a) Task based priority rules, b) Network based priority rules, c) Critical path based priority rules, and d) Resource based priority rules. Multi-modes with multi-constrained resources for each task require complex management of queues. Resources availability ultimately decides which task will have priority.

A mode, in this research, has information regarding correlated tasks. There is traceability along projects, tasks, modes, and resource requirements. Therefore, this method will work, regardless of the rule, for all four of types of priorities. This research uses composite priority rules of the four rule classes mentioned by Vanhoucke [Vanhoucke, 2013]. However, instead of using a weighted combination, we propose a class called “Independent Composite Priority Rules”. The rules are specified for each resource, or set of resources, in priority order, without any weight. Each resource has the flexibility to have its own rules, e.g., rule for Resource 1 (welding resource for example) do not apply to Resource 2 (painting worker) and vice-versa. Indeed, it is common practice that each unit runs its own operations.

If the priority rules for Resource 1 are: 1) Customer deadline, 2) Shortest processing time, 3) Immediate successor need, and 4) Earliest completion time, and if two modes of different tasks are in the queue requiring Resource 1, then the priority rule 1 is used to resolve the issue. If the tasks have the same customer deadline, then the second rule is applied and so on so forth.

Figure 16 shows a schema of the four modes in the queue requiring a different set of resources. This schema based on example 1 shows only one combination out of several for Tasks 2 and 3 with their respective Mode 1 (Table 8 and Figure 6). The resource request process is made every time a resource is released. The status of resources is taken into consideration and once a resource conflict exists, priority rules are called to determine which mode wins. Notice that each resource may have its own priority rule list. A single combination for performing Task 2, Mode1, and Task 3, Mode 1 can be described as follows:

Task 2, Mode 1 = {R1, R2, (welders), R5, R6, R11 (cutters), R12, R15}.

Task 3, Mode 1 = {R3, R7, R8, R9, R10 (welders), R5, R6, R11 (cutters), R12, R15}.

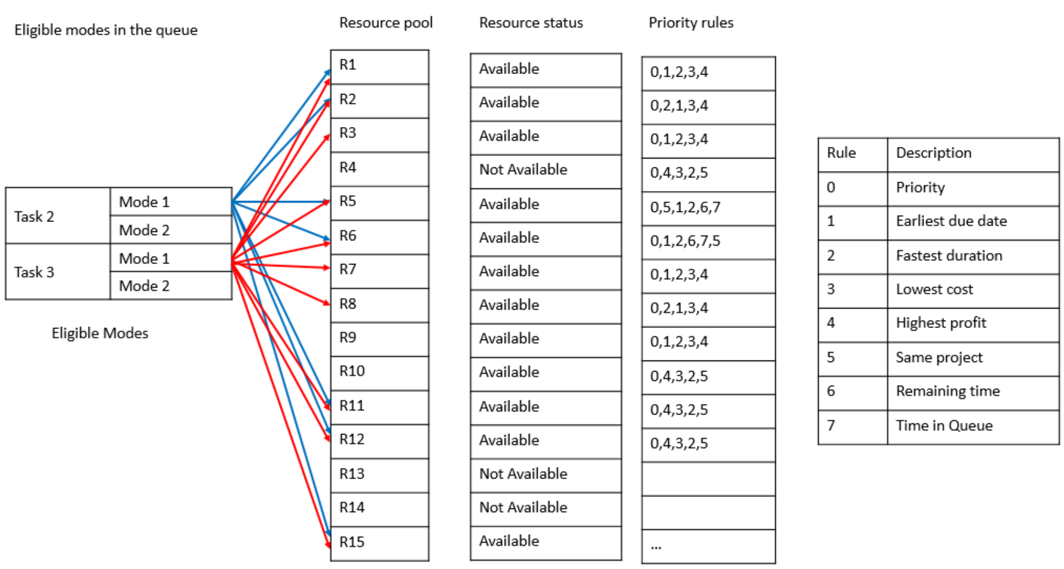


Figure 16: Modes x Resource pool

Assuming that only Modes 1 and 2 for Tasks 2 and 3 are in the queue at given decision point in time. A Venn diagram depicted in Figure 17 shows which resources are in conflict. Resource R5, R6 and R11 are required by both modes. Therefore, there is a resource conflict. Notice, all of these resources are available. They have been previously identified as available, the question at this point is: Which one of the two modes should be selected? Priority rules are called and applied only for resources R5, R6, and R11 to solve conflict between Modes 1 of Tasks 2 and 3.

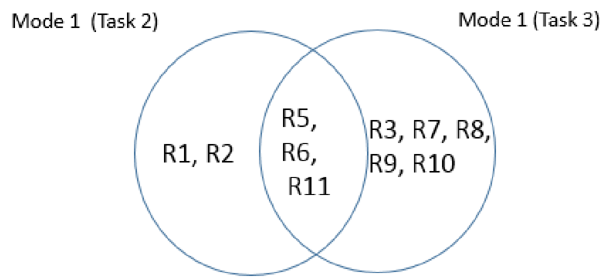


Figure 17: Venn diagram for resource conflicts

The Priority Rules Management (PRM) algorithm is shown in Table 14. This algorithm has 3 parameters: *Task 1 index*, *Task 2 index* and *Resource Index*. At each iteration, two modes of different tasks are utilized by the algorithm. This information comes from *Task 1 index* and *Task 2 Index*. The selected mode for each task was previously found by the Run Queue Algorithm. The third parameter "Resource Index" provides the index of a common resource along the modes, in other words, the conflicting resources. This procedure returns one temporary winning mode named "*PreliminaryWinner*". The tentative winning mode becomes a definitive winning mode once all modes in the queue have been verified. Based on the *Resource Index,* its priority rules number is known. Line 2 to line 19 perform a loop is performed in order to run all possible priority rules (from rule 1 to NPR_{nr}) to break ties among two modes, if necessary. Rule with index 1 is the highest priority rule, rule 2 is more important than 3, and so forth. In line 3 assuming that a given resource has the rule "Earliest Deadline", there will be a comparison between the deadline of *Task 1 index* and *Task 2 index*, the task which presents the earliest deadline will be the "*PreliminaryWinner*" (Line 4). The same logic follows from lines 5 to 13. Line 14 checks if the two tasks are still tied (*Preliminary Winner* = -1) and also checks if all possible priority rules have been used. If so, the *PreliminaryWinner* is equal to *Task 1 Index* because all effort has been done and they are still tied. If such condition occurs, first index is chosen. Line 16, on the other hand, checks if any Priority rule was enough to break a tie (*PreliminaryWinner* ≠ -1). If so, the loop ends because the resource conflict has been solved for at least one priority rule. NPR_z is the number of priority rules for resource *z*, and *pr* is the rule index with range $1<pr< NPR_z$.

Table 14: Priority Rules Management (PRM) Algorithm

```
procedure PRM(Task 1 index, Task 2 Index, Resource Index=nr)
    for pr ← 1, NPR_nr do
        if ("Earliest Deadline" = Rule_pr) then
            PreliminaryWinner ← winner from Deadline
        else if ("Fastest Duration" = Rule_pr) then
            PreliminaryWinnerr ← winner from Fastest Duration
        else if (Rule 3 = Rule_pr) then
            PreliminaryWinner ← winner from Rule 3
        else if (...) then
            PreliminaryWinner ← winner from ...
        else if (Rule n = Rule_pr) then
            PreliminaryWinner ← winner from Rule n
        end if
        if (PreliminaryWinner = -1) and pr=NPR_z then
            PreliminaryWinner ← Task 1 Index
        else if (PreliminaryWinner ≠ −1) then
            Exit For
        end if
    end for
end procedure
```

3.4.4 Worker Skills Management

STREAM does not use a pre-determined approach to project management. Instead, it is based on the notion of dynamically allocating resources. The flexibility in resource allocation is extended to worker skills to reduce project cost and duration. Figure 18 shows the Venn diagram for skills for example 1, as described in the Section 3.1.

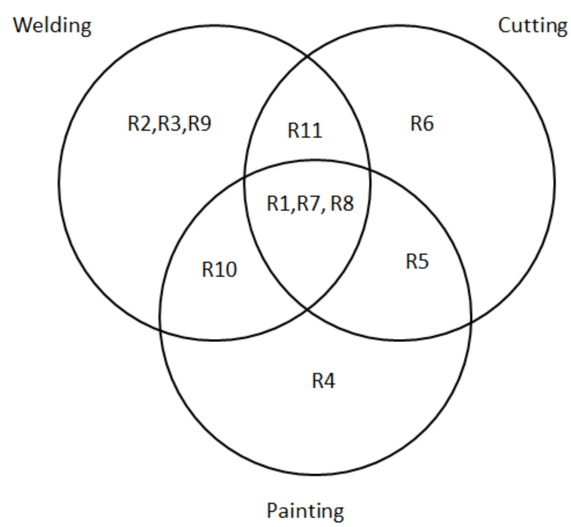


Figure 18: Venn diagram for skilled resources

One of the gaps identified in the literature is modelling of production capacity. If the capacity is not well defined, the results will not be accurate. Several authors have raised this issue, but very little information is available on how to address this problem. Traditional project scheduling is limited to single-skilled resource assumption. This does not represent the real-world practice where workers have multiple skills and are assigned to perform tasks where they are not primarily specialized [Wongwai and Malaikrisanachalee 2011]. However, some limitations regarding multi-skilled workers still exist: different work hours for workers, cost assessment, and overtimes, etc. STREAM's approach differs from Wongwai and Malaikrisanachalee (2011) by providing qualitative rank for different skills, such as excellent, good, satisfactory, and not applicable for each resource as introduced in section 3.1.

In STREAM, if a task requires a certain skill, say welding, it checks the available resource that have welding as their primary skill. If none is found, it checks for resources that have welding as their secondary skill, if found, it assigns that resource to the task. The resource could have painting as the primary skill, cutiing and welding as secondary skill. When a worker is assigned to perform the task using one of his many skills, then his/her other skills are not available to other task. Typical project management tools assume resources to have a single skill, resulting in schedule delay and higher cost.

In STREAM a method called Worker Skill Management (WSM) was created. WSM has two parts, i) Available Resource and Skills Management (ARSM) and ii) Resource and Skills Requirement Management (RSRM). Table 15 describes the ARSM algorithm. Table 16 describes the RSRM algorithm. The ARSM verifies if resources are available with required skills at a given decision point in time. The RSRM actually requires and assigns the resources based on the skill required.

The ARSM has two parameters, *SkillRequiredSelectedMode* and *NumberOfResources* for each required skill. The algorithm starts with a loop which runs with a variable *s* equal to 1, to the number of skills (*NS*). Line 3 checks if a skill required for a selected mode is equal to a general $skill_s$. If yes, it runs another loop from 1 to NRS_s from (4 to line 8). Line 5 checks if the $Resource_{rs}$, which has the required skill, is available. If so, line 6 the variable *Counter* is added of one. Line 11 verifies if the variable *Counter* is greater or equal to the number of resources required by a given task, if so, that means resources are available (line 12), if not, resource are not available (line 14). *NS* is the number of skills; *s* is the skill index with range $1 \leq s \leq NS$; and NRS_s is the number of resources with $skill_s$; *rs* is the resource index with range

$1 \leq rs \leq NRS_s$.

Table 15: Available Resource and Skills Management (ARSM) algorithm/WSM

```
1: procedure ARSM(TaskIndex)
2:     for m ← 1, NM_ij do
3:         Available ← True
4:         for c ← 1, NC_ijm do
5:             if (scope of the resource = "Resource") then
6:                 if Resource_c is not available then
7:                     Available ← False
8:                 end if
9:             else
10:                for s ← 1, NS do
11:                    if (SkillRequired = Skill_s) then
12:                        for rs ← 1, NRS_s do
13:                            if Resource_rs is available then
14:                                Counter ← Counter + 1
15:                            end if
16:                        end for
17:                        if Counter >= NumberOfResources then
18:                            Available ← True
19:                        else
20:                            Available ← False
21:                        end if
22:                    end if
23:                end for
24:            end if
25:        end for
26:        if Available = True then
27:            Exit For
28:        end if
29:    end for
30: end procedure
```

The RSRM has a single parameter, *Task Index*. The algorithm initializes a variable *k* which goes from 1 to the number of resource for the mode selected (NC_{ijm}). It handles resources and skills associated with a given resource. Line 3 checks if the scope of the select resource is "Resource". It can be either a resource or a skill. If it is "Resource", the resource for the select mode becomes occupied (line 4), else, the scope is "Skill" and then from lines 6 to 19, a loop of skills is performed in order to find the skill required for the mode selected (Line 7 – If a skill required for a selected mode is equal to a general $skill_s$).

If yes, run another loop from 1 to NRS_s from line 8 to line 17. Line 9 checks if the variable *Counter* is less or equal to NRS_s. If so, line 10 verifies if $Resource_{rs}$ is available, if it is available then $Resource_{rs}$ becomes unavailable in line 11 and the variable *Counter* goes up by one in line 12. If the variable *Counter* is greater than the number of resources required by a given task, the loop is finished (line 15). Variable *Counter* being greater than the number of resources required means that the number of required resources for a given skills to execute a task has been satisfied. *NS* is the number of skills; NR_{ijm} is the number of resources for project *i*, task *j*, mode *m*; NRS_s is the number of resources with skill *s; rs* is the resource index with range $1 \leq rs \leq NRS_{s.}$ *;s* is the skill index with range $1 \leq s \leq NS$; and *k* is the resource index with $1 \leq k \leq NR_{ijm.}$

Table 16: Resource and Skills Requirement Management (RSRM) algorithm/WSM

```
 1: procedure RSRM(SelectedMode, NumberOfResources)
 2:     for c ← 1, NC_ijm do
 3:         if (scope of the resource in the selected mode = "Resource") then
 4:             Available ← False
 5:         else
 6:             for s ← 1, NS do
 7:                 if SkillRequired = Skill_s then
 8:                     for rs ← 1, NRS_s do
 9:                         if Counter <= NumberOfResources then
10:                             if Resource_rs is available then
11:                                 Available ← False
12:                                 Counter ← Counter + 1
13:                             end if
14:                         else
15:                             Exit For
16:                         end if
17:                     end for
18:                 end if
19:             end for
20:         end if
21:     end for
22: end procedure
```

3.4.5 Task Duration Management

STREAM allows each resource to have several time intervals during a given day. That means a single resource can have different daily calendar, that is, a given skilled worker can work from 8:00 to 17:00 on Mondays, from 9:00 to 16:00 on Tuesdays, and so on. A "shift" is defined as eight hour work day and the time interval is further classified as "Regular", "Idle," and "Overtime." Time intervals classification can vary for each work day, that is, Monday could have 8:00 to 12:00 (Regular), 12:00 to 13:00 (Idle), and 13:00 to 17:00 (Regular). Tuesday's time classification could be 9:00 to 9:30 (idle), 9:30 to 12:30 (regular), 12:30 to 13:00 (idle), and 13:00 to 16:00 (regular), etc. Time intervals would vary according to project needs. Such an approach allows unlimited flexibility for time classification. The longest calculated duration along all resources will be the actual time taken to execute a given task using a given mode. Since each resource has its own calendar, the slowest resource or the most constrained one will result in the actual task duration. Real world projects have several calendar constraints, adding further complexity to the model. Table 17 shows the calendar for the fourteen renewable resources of example 1 presented in section 3.1. Table 18 shows a work day for resource R1 as an example.

Table 17: Resource Calendar

Resource	Number of days/week	Days
R1	5	M, W, T, Th., F
R2	5	M, W, T, Th., F
R3	5	M, W, T, Th., F
R4	6	M, W, T, Th., F, S
R5	4	M, W, T, Th.
R6	7	M, W, T, Th., F, S, Su.
R7	6	M, W, T, Th., F, S
R8	5	M, W, T, Th., F
R9	5	M, W, T, Th., F
R10	5	M, W, T, Th., F
R11	5	M, W, T, Th., F
R12	5	M, W, T, Th., F
R13	5	M, W, T, Th., F
R14	5	M, W, T, Th., F

Table 18: Work Day Classification for Resource R1 on Monday			
TIP	**From**	**To**	**Description**
1	00:00	8:00	Idle (non-working time)
2	08:00	12:00	Regular (working time)
3	12:00	13:00	Idle (lunch time)
4	13:00	18:00	Regular (working time)
5	18:00	24:00	Overtime (working time with additional pay)

Project management tools primarily focus on production times. Non-production times are extremely complicated and cumbersome to handle ahead of time, because it requires an exhaustive analysis of resource queues. It is difficult to estimate how long a task will wait for its resources once it is eligible to start. Traditional approaches do not account for idle time. Calendar constraints have significant impact on project duration time. For instance, if the duration of a given task requires 24 hours using a set of resources, it does not mean that this task can be carried out in three days of 8 hours shift. Depending upon resource availability, the task could take 48 hours, 96 hours, and so forth. The duration will depend upon break constraints and also the multiple resource calendars involved with the task

Table 19 summarizes the Task Duration Management (TDM) algorithm. It calculates the actual duration of tasks, taking into account constraints such as, maintenance times, lunch time, and other idle times. These types of calendar constraints exist for all projects and they add complexity to the simulation model. Additional times such as overtimes are considered and it can also affect tasks duration. Therefore, the simulation, through TDM algorithm, will calculate the actual finish time considering all the complexities mentioned above. NR_{ijm} is the number of resources for project i, task j, and mode m; $NIDR_z$ is the number of time interval partitions during a day for resource z; and k Resource index, $1 \leq k \leq NR_{ijm}$.

The TDM algorithm has three parameters, *Task Index*, *Duration, and Current Clock*. Duration is the production time provided by managers and *Current Clock* takes the information of the current value of the global variable *Simulation Clock*. Line 2 assigns the week day of the current clock to variable *Day*, for instance if the current clock is (2/26/2016 1:30 PM), variable *Day* becomes equal to 5 because 2/26/2016 12:00 is Thursday (Sunday = 1, Monday = 2, Tuesday = 3, Wednesday = 4, Thursday = 5, Friday = 6 and

Saturday = 7). From line 3 to 26 there is a loop that runs from 1 to number of required resources of the selected mode (NR_{ijm}). Line 4 *Remaining* time is assigned to be equal to *Duration.* When *Remaining* reaches 0, it means the task could have been executed in its total time across any calendar constraint. Line 5 calls the Availability Verification Management (*AVM*) procedure.

The AVM algorithm is shown in Table 20. Its goal is to figure out what is the time interval during the day for the *Current Clock* and it also adds possible idle times accordingly to variable *Duration*. Since each time interval has its own start time and end time, this procedure finds the time interval partition (TIP) and if it is a “Regular”, “Idle”, or “Additional”. Line 29 in the AVM algorithm checks if the current TIP is “Regular”, if so, verification if any maintenance or other idle times will exist is needed. In line 30, the variable *Add Idle Time* controls how much idle time exists through the TIP. In line 31, the variable *Remaining* is decreased by the time TIP value. If the TIP is “Idle” (line 32) no changes is made for the variable *Remaining*, only for *Add Idle Time* (line 33). If TIP is “Additional” (Line 34) then variable *Remaining* is decreased by the amount of time the TIP allows in additional time (line 35). “*Authorized Additional Time”* is a procedure which finds if additional time is authorized. Line 36, *Add Idle Time* is added by the current TIP end time minus any authorized additional time. Line 38 checks if *Remaining* is equal to 0, if so, exit procedure.

Table 19: Task Duration Management (TDM) algorithm

```
procedure TDM(TaskIndex, Duration, CurrentClock)
    Day ← Weekday(CurrentClock)
    for c ← 1, NC_ijm do
        for r ← 1, NR_ijmc do
            Remaining ← Duration
            AVM(CurrentClock)
            if tip + 1 <= NIDR_z then
                for tip ← tip + 1, NIDR_r do
                    AVM(CurrentClock)
                end for
            end if
            while Remaining > 0 do
                if Day ≠ 7 and Day ≠ 1 then
                    Day ← Day + 1
                else if Day = 7 then
                    Day ← 1
                else if Day = 1 then
                    Day ← 7
                end if
                for tip ← 1, NIDR_r do
                    AVM(CurrentClock)
                end for
            end while
            DurationCalculated ← Duration + AddIdleTime
            if DurationCalculated >= LongestDuration then
                LongestDuration ← DurationCalculated
            end if
        end for
    end for
end procedure
```

Table 20: Availability Verification Management (AVM) algorithm

```
procedure AVM(CurrentClock)
    if Time Interval Partition is "Regular" then
        AddIdleTime ← AddIdleTime +Verify Idle Times and Maintenance
        Remaining ← Remaining - (TIPEndTime - CurrentClock)
    else if Time Interval Partition is "Idle" then
        AddIdleTime ← AddIdleTime + (TIPEndTime-CurrentClock)
    else if Time Interval Partition is "Additional" then
        Remaining ← Remaining - (AuthorizedAdditionalTime)
        AddIdleTime ← AddIdleTime + (TIPEndTime-CurrentClock)-
(AuthorizedAdditionalTime)+(Verify Idle Times and Maintenance)
    end if
    if Remaining <= 0 then
        Exit Procedure
    end if
end procedure
```

Line 6 of the TDM algorithm, if the *Current Clock* is located within last TIP of the current day, then no more searching is required for that day. The variable TIP says which TIP the *Current Clock* is in. For instance, if the time of *Current Clock* is 1:30 PM, by looking at Table 19, the TIP uses "Regular" and variable TIP is set to 4. If TIP is 4, then one more TIP exists during the day. Therefore, line 6 verifies if the variable TIP +1 is less or equal to the number of TIPs during the day *($NIDR_z$)* for the resource with index r. If so, that means there will be more TIPs during the same day, and then *AVM* procedure is called for each TIP (lines 7 - 9). A loop is called in line 11 while *Remaining* is greater than 0. Notice the variable *Day* is assigned to be one unit of day more at each cycle (line 12). Lines 13 to 17 are to adjust the correct value of variable *Day* when it is either 7 or 1. Lines 18 to 20, once the variable *Day* has been assigned, all the TIPs must be run again. At each TIP, for each cycle, the *AVM* procedure is called. When the algorithm exits loop at the line 22, the variable *Duration Calculated* becomes equal to original *Duration* plus any additional idle times preserved into the variable *Add Idle Time*. There is only one more check to be performed, Lines 23 to 25 guarantees that longest duration along all resources is the actual time taken to execute the task using the selected mode. Recall that one mode has several resources and each resource might have its own calendar. Therefore, the slowest resource or the most constrained one will provide the longest duration.

3.4.6 Task Precedence Management

Table 21 presents the Task Precedence Management (TPM) algorithm. It shows how the simulation system handles precedence with lag-times. For tasks with no lag-time are found, DES algorithm determines how to release successors. It is triggered with a Finish Event occurrence. In the DES algorithm, release a successor may be possible when a Finish event occurs. In the TPM algorithm, on the other hand, events of “Release a Successor” are added to the list of future events, not at the current *Simulation Clock.*

Line 2 the variable *Start* is assigned to be equal to *Current Clock.* Line *3,* the variable *Finish* is equal to *Start* plus *Calculate Duration.* An event of Finish Task is added into the list of events (line 4). A loop between lines 5 and 15 is performed. It runs from 1 to the number of successors for a task. For each precedence relationship having lag-time different than 0 (line 6), a *Release Successor Date* variable is calculated in line 7. Lines 8 to 12 verifies if any predecessor has a finish date greater than the release date calculated, if so, *Release Successor Date* is assigned to be equal *to Pred. Finish* (predecessor finish date) . Line 13 adds an event of Release Successor. $NSuc_{ij}$ is the number of successors for project *i*, task *j*; $NPred_{ij}$ is the number of predecessor for project *i*, task *j*; *pre* is the predecessor index,

$1 \leq pre \leq NPre_{ij};$ and *suc* is the successor index, $1 \leq suc \leq NSuc_{ij}.$

Table 21: Task Precedence Management algorithm (TPM)

```
procedure TPM
    for Suc ← 1, NSuc_ij do
        if Lagtime ≠ 0 then
            ReleaseSucessorDate ← Finish+TaskDurationManagement_Lagtime
            N. Pred. done ← N. Pred. done+1
            if Number of Predecessors = N. Pred. done then
                for Pre ← 1, NPre_ij do
                    if Pred.Finish > ReleaseSucessorDate then
                        ReleaseSucessorDate ← Pred.Finish
                    end if
                end for
                Key ← (ReleaseSucessorDate, TaskIndex)
                Add Event(Key, "ReleaseSuccessor", ReleaseSucessorDate, TaskIndex)
            end if
        end if
    end for
end procedure
```

3.5 System Input and Output

The input-output schema of STREAM is shown in Figure 19. All of the outputs are based on resource allocation method of STREAM. Each resource has several associated information such as, available times, fixed cost, and variable costs. Tasks information includes, due dates, bonus, and penalty cost. Projects and tasks due dates are inputs and project and task finish dates are the computed outputs.

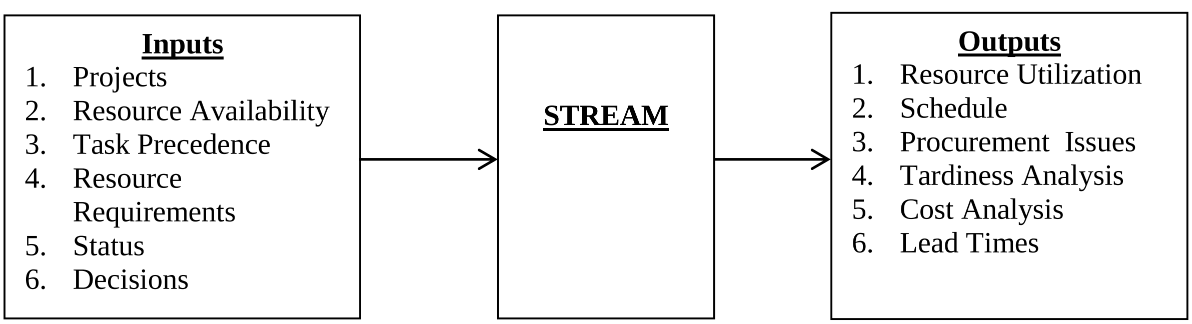


Figure 19: Input-Output Schema

Figure 20 illustrates the simulation cycle in which decisions can be made at every simulation cycle and system performance can be assessed. Outputs are discussed in Chapter 4 through the case studies.

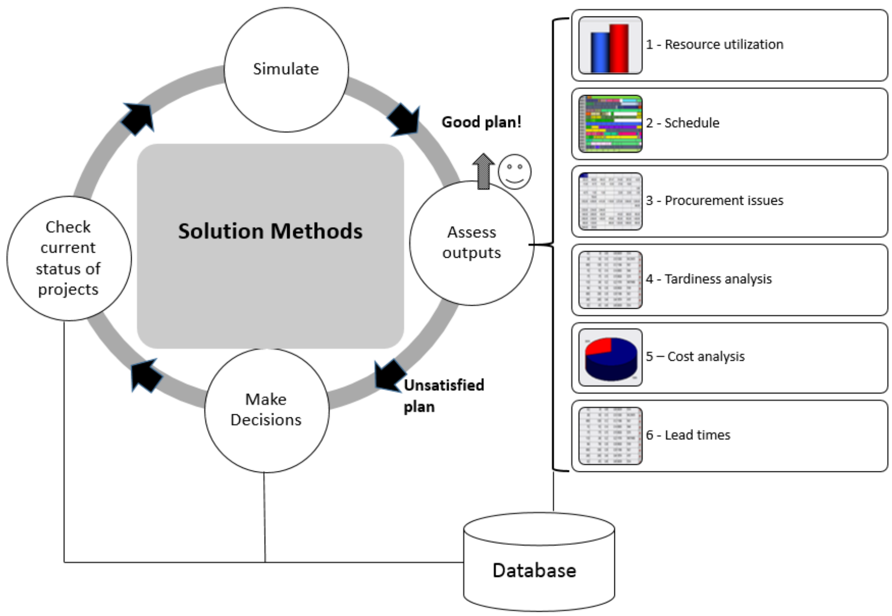


Figure 20: Simulation Cycle Schema

Two databases were implemented, one for the server side and another for the client side. The server side database maintains the simulator level data and its structure remains fixed. It has a fixed number of tables that provide all the necessary data to the simulator. By reading the data set, the proposed solution method presented in Chapter 3 is able to generate the project schedule as output.

The client side database is customized to meet the needs of particular clients or applications. Each client's database has unique data to the application. E.g., the ship repair industry requires dada that is not needed by the construction industry and vice-versa. The server side database was implemented to address all general project scheduling requirements. Customizations are carried out in the client or application database.

The server side database utilizes 10 tables. Tables dealing with reports are located in the client side database. Structured Query Language (SQL) [SQL, 2014] was used to map data from the client database to the server database. The sever side database tables and associated fields are described in the Appendix I. Table AI.1 describes the resources. Table AI.2 describes the resource idle times. The nonrenewable resources table which carries information such as quantity on hand and on order is described in Table AI.3. Table AI.4 describes the additional times for resources. Available time for resources and multiple skilled resources are presented in Tables AI.5 and AI.6, respectively. Table AI.7 describes the tasks. Details for Task/modes, Task/Modes/Resources and task precedence are described in Tables AI.8, AI.9, and AI.10, respectively.

Chapter 4: Method Application and Validation

Two cases studies reported in literature were carried out to test and validate STREAM. The results produce by STREAM were validated with published results. The data of the third and fourth case studies were collected on-site.

4.1 Case study 1: Construction of a Single Cell Box Culvert

This case study is based on the data provided by Lu et al. (2008). It deals with a construction of a single cell box culvert in a drainage project in Hong Kong. Lu et al. (2008) models the problem as optimization problem aiming to minimize the total project duration. Multi-modes, multi-skilled resources and non-renewable resources are not considered. The optimized project duration with limited resources found in Lu et al. (2008) is 275 days.

The case study is used to illustrate some of STREAM's capabilities (i.e., multi-modes, multi-skilled resources, the combination of multi-modes and multi-skilled resources, and non-renewable resources), which may impact on project's cost and duration. As the problem size is not too small and not large, these capabilities can be demonstrated through numerical examples. Additionally, as this case study provides a real-life project, it represents an invaluable context to validate STREAM. In this research, the work of Lu et al. (2008) is strictly used for obtaining data from their case study. A comparison between the proposed approach and Lu et al. (2008)'s approach is not made as the goals and capabilities of both approaches are by nature different.

This case study is organized in four subsections; 1) Input data, 2) The set of scenarios used to validate STREAM are described, 3) Results applying STREAM, and 4) Compares the results of the assessed scenarios.

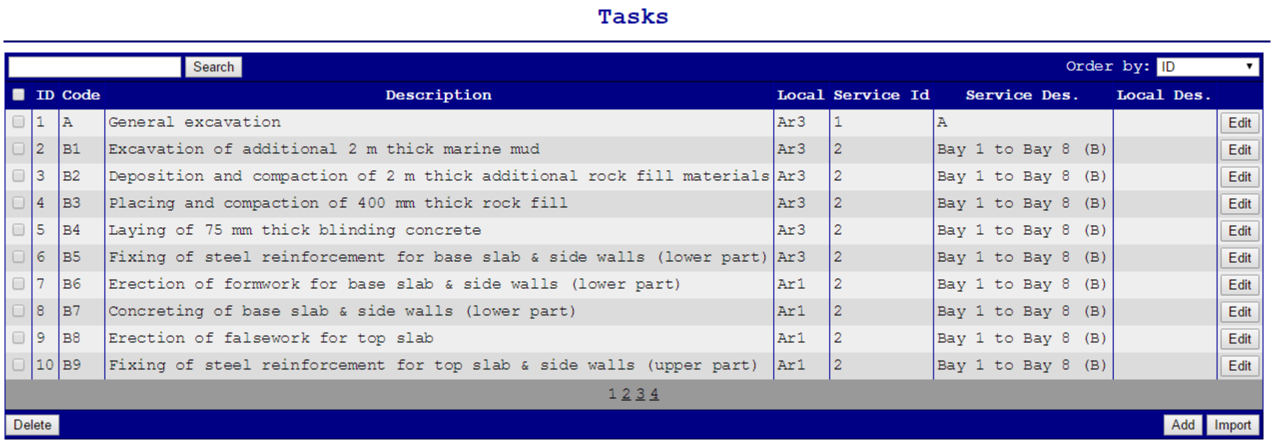
4.1.1 Inputs

Construction of a single cell box culvert project involves 8 different types of resources and 33 tasks. The project assumed eight hours per work day and distinct calendars were used for different resources. Table 22 shows the tasks, their durations, and their predecessors. The resources are shown in Table 23. The resources are: Bar bender and fixer (BBF), Backhoe with excavator (BE), Crawler mounted crane (CMC), Carpenter (CF), Concreting Labour (CLB), Drain Layer (DL), Skilled labour (LB) and Roller (RR). Figure 21 shows the Task screen and Figure 22 shows the Resource screen in STREAM. The resource calendars are displayed in Table 24. Appendix II describes the data files associated with this case study. Table AII.1 shows task resource requirements for mode 1 and resource capacity limits. The original case study assumes only one mode for each task. To demonstrate STREAM's capabilities, information regarding multi-modes, multi-skilled workers, non-renewable resources and costs are used as additional inputs in this case study.

Regarding multi-mode tasks (as tasks may be performed by selecting one out of several combinations of resource requirements), an additional mode (mode 2) was included for each task. For instance, according to mode 1, task A lasts 16 days and requires four BEs and eight LBs, and according to mode 2, it lasts 23 days and requires two BEs and five LBs. Table AII.2 in the Appendix II shows task resource requirements for mode 2. Resource skills are categorized as excellent, good, and satisfactory (Table 25). As example pf skills levels of each resource is shown in Table 26. E.g., resources CF perform CF tasks with excellence, CLB tasks in a good manner and LB tasks in a satisfactory way. Resources CLB perform CLB tasks with excellence, CF tasks in a good manner and LB tasks in a satisfactory way. To evaluate the financial impact of a scenario, STREAM requires data for fixed costs, variable costs, penalty costs, and bonus costs. Additionally, due dates for tasks are also relevant as they represent major milestones commonly found in project management. An example of fixed and variable costs for the resources is shown in Table AII.3 in the Appendix II. STREAM allows for penalty cost and bonus. Bonus is for early task completion and penalty is for delayed tasks. Bonus and penalty are calculated by comparing task finish time and task due date. These costs can be ignored by providing zero values for penalty costs and bonus. Penalty cost and bonus are in terms of loss and profit per day, respectively. Table AII.4 in the Appendix II shows task costs. A cost/time trade-off analysis is relevant and can be assessed when multi-skilled workers exist (if a task requires a CLB skill and a LB is assigned to the task, the cost associated refers to the LB worker).

Table 22: Task Description, Duration, and Predecessors

	Task	Description	Duration	Pred.
1	A	General excavation	16	
2	B1	Excavation of additional 2 m thick marine mud	8	A
3	B2	Deposition and compaction of 2 m thick additional rock fill materials	16	B1
4	B3	Placing and compaction of 400 mm thick rock fill	12	B2
5	B4	Laying of 75 mm thick blinding concrete	3	B3
6	B5	Fixing of steel reinforcement for base slab & side walls (lower part)	32	B4
7	B6	Erection of formwork for base slab & side walls (lower part)	16	B4
8	B7	Concreting of base slab & side walls (lower part)	4	B5,B6
9	B8	Erection of false work for top slab	32	B7
10	B9	Fixing of steel reinforcement for top slab & side walls (upper part)	20	B8
11	B10	Erection of formwork for top slab & wide walls (upper part)	20	B8
12	B11	Concreting of top slab & side walls (upper part)	5	B9,B10
13	C1	Placing and compaction of 400 mm thick rock fill	12	A
14	C2	Laying of 75 mm thick blinding concrete	3	C1
15	C3	Fixing of steel reinforcement for base slab & side walls (lower part)	32	C2
16	C4	Erection of formwork for base slab & side walls (lower part)	16	C2
17	C5	Concreting of base slab & side walls (lower part)	3	C3,C4
18	C6	Erection of false work for top slab	32	C5
19	C7	Fixing of steel reinforcement for top slab & side walls (upper part)	20	C6
20	C8	Erection of formwork for top slab & wide walls (upper part)	20	C6
21	C9	Concreting of top slab & side walls (upper part)	5	C7,C8
22	D1	Placing and compaction of 400 mm thick rock fill	2	C9
23	D2	Laying of 75 mm thick blinding concrete	1	D1
24	D3	Fixing of steel reinforcement for base slab & side walls (lower part)	5	D2
25	D4	Erection of formwork for base slab & side walls (lower part)	2	D3
26	D5	Concreting of base slab & side walls (lower part)	1	D4
27	D6	Erection of false work for top slab	4	D5,E2
28	D7	Fixing of steel reinforcement for top slab & side walls (upper part)	6	D6
29	D8	Erection of formwork for top slab & wide walls (upper part)	3	D7
30	D9	Concreting of top slab & side walls (upper part)	1	D6,D7
31	E1	Approval on concrete pipe manufacturer & delivery of concrete pipe	90	
31	E2	Positioning of pre-cast concrete pipes at end wall	1	E1
33	F	Backfilling & compaction B11, D9 7 4 8 4 Available resources limit	7	B11,D9



Tasks

Search | Order by: ID

ID	Code	Description	Local	Service Id	Service Des.	Local Des.	
1	A	General excavation	Ar3	1	A		Edit
2	B1	Excavation of additional 2 m thick marine mud	Ar3	2	Bay 1 to Bay 8 (B)		Edit
3	B2	Deposition and compaction of 2 m thick additional rock fill materials	Ar3	2	Bay 1 to Bay 8 (B)		Edit
4	B3	Placing and compaction of 400 mm thick rock fill	Ar3	2	Bay 1 to Bay 8 (B)		Edit
5	B4	Laying of 75 mm thick blinding concrete	Ar3	2	Bay 1 to Bay 8 (B)		Edit
6	B5	Fixing of steel reinforcement for base slab & side walls (lower part)	Ar3	2	Bay 1 to Bay 8 (B)		Edit
7	B6	Erection of formwork for base slab & side walls (lower part)	Ar1	2	Bay 1 to Bay 8 (B)		Edit
8	B7	Concreting of base slab & side walls (lower part)	Ar1	2	Bay 1 to Bay 8 (B)		Edit
9	B8	Erection of falsework for top slab	Ar1	2	Bay 1 to Bay 8 (B)		Edit
10	B9	Fixing of steel reinforcement for top slab & side walls (upper part)	Ar1	2	Bay 1 to Bay 8 (B)		Edit

1 2 3 4

Delete | Add | Import

Figure 21: Task screen in STREAM

Table 23: Resource Description		
Resource	**Description**	**Limits**
BBF	Bar bender and fixer	4
BE	Backhoe with excavator	4
CMC	Crawler mounted crane	3
CF	Carpenter	2
CLB	Concreting labour	5
DL	Drain layer	1
LB	Skilled labour	8
RR	Roller	4

Resources

Search | Order by: ID

ID	Code	Description	Group ID	Availability	Efficiency	Direct Cost	Variable Cost	Over Cost	Energy Cost	Status	Queue Type	Quantity	
1	BBF1	Bar bender and fixer 1	1	4	100%	$1,000.00	$13.00	$10.00	$0.00	Available	Finite	1	Edit
2	BBF2	Bar bender and fixer 2	1	4	100%	$1,000.00	$13.00	$10.00	$0.00	Available	Finite	1	Edit
3	BBF3	Bar bender and fixer 3	1	4	100%	$1,000.00	$13.00	$10.00	$0.00	Available	Finite	1	Edit
4	BBF4	Bar bender and fixer 4	1	4	100%	$1,000.00	$13.00	$10.00	$0.00	Available	Finite	1	Edit
5	BE1	Backhoes Excavator 1	2	4	100%	$1,000.00	$38.00	$10.00	$0.00	Available	Finite	1	Edit
6	BE2	Backhoes Excavator 2	2	4	100%	$1,000.00	$38.00	$10.00	$0.00	Available	Finite	1	Edit
7	BE3	Backhoes Excavator 3	2	4	100%	$1,000.00	$38.00	$10.00	$0.00	Available	Finite	1	Edit
8	BE4	Backhoes Excavator 4	2	4	100%	$1,000.00	$38.00	$10.00	$0.00	Available	Finite	1	Edit
9	CMC1	Crawler mounted crane 1	3	4	100%	$1,000.00	$150.00	$10.00	$0.00	Available	Finite	1	Edit
10	CMC2	Crawler mounted crane 2	3	4	100%	$1,000.00	$150.00	$10.00	$0.00	Available	Finite	1	Edit
11	CMC3	Crawler mounted crane 3	3	4	100%	$1,000.00	$150.00	$10.00	$0.00	Available	Finite	1	Edit
12	CF1	Carpenter 1	4	2	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
13	CF2	Carpenter 2	4	2	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
14	CLB1	Concreting labor 1	5	3	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
15	CLB2	Concreting labor 2	5	3	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
16	CLB3	Concreting labor 3	5	3	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
17	CLB4	Concreting labor 4	5	3	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
18	CLB5	Concreting labor 5	5	3	100%	$1,000.00	$19.67	$10.00	$0.00	Available	Finite	1	Edit
19	DL1	Drainlayer 1	6	4	100%	$1,000.00	$85.00	$10.00	$0.00	Available	Finite	1	Edit
20	LB1	Skilled labor 1	7	2	100%	$1,000.00	$25.00	$10.00	$0.00	Available	Finite	1	Edit

1 2

Delete | Add | Import

Figure 22: Resource screen in STREAM

Table 24: Resource Calendar		
Resource	**Number of days/week**	**Days**
BBF	7	M, W, T, Th., F, S, Su.
BE	7	M, W, T, Th., F, S, Su.
CMC	7	M, W, T, Th., F, S, Su.
CF	6	M, W, T, Th., F, S
CLB	4	M, W, T, Th.
DL	7	M, W, T, Th., F, S, Su.
LB	6	M, W, T, Th., F, S
RR	7	M, W, T, Th., F, S, Su.

Table 25: Resource Skill Levels	
Level	**Description**
0	None
1	Excellent
2	Good
3	Satisfactory

Table 26: Multiple Skill Example							
	Skill level						
Resource	**BBF**	**CMC**	**CF**	**CLB**	**DL**	**LB**	**RR**
BBF	1	0	0	0	0	0	0
CMC	0	1	0	0	0	0	0
CF	0	0	1	2	0	3	0
CLB	0	0	2	1	0	3	0
DL	0	0	0	0	1	0	0
LB	0	0	3	2	0	1	0
RR	0	0	0	0	0	0	1

4.1.2 Scenarios

Seven different scenarios were considered for this case study. Scenarios 1, 2, and 3 are the same as those considered by Lu et al. (2008). Scenarios 4, 5, 6, and 7 utilize multi-skills, multi-modes, multi-skills and modes, and non-renewable resources respectively. Brief description of each scenario is as follows:

Scenario 1: Determine schedule based on task precedence, resource calendars and resource availability.

Scenario 2: Minimize Total Project Duration (TPD), while maintaining task precedence, resource calendars, and resource availability.

Scenario 3: Minimize TPD by finding the optimum quantity of resources, while maintaining task precedence, and resource calendars.

Scenario 4: Determine schedule based on task precedence, resource calendars and resource availability constraints, where human resources can have multiple skills.

Scenario 5: Determine schedule based on task precedence, resource calendars and resource availability constraints, where tasks can be carried out in multiple ways (multi-modes).

Scenario 6: Determine schedule based on task precedence, resource calendars and resource availability constraints, where human resources can have multiple skills and tasks can be carried out in multiple ways (multi-modes).

Scenario 7: Determine schedule based on task precedence, resource calendars and resource availability constraints, where non-renewable resources and replenishment plan are considered.

4.1.3 Results applying STREAM

The seven scenarios were tested using STREAM. Each one reveals a particular capability of the proposed solution approach in providing resource allocation flexibilities.

Case study 1: Scenarios 1 and 2

For scenarios 1 and 2 STREAM found the same results. STREAM found TPD to be 275 days as in Lu et al. (2008) and cost of $14,347,972. STREAM provided addition insight to the project. Skilled labour (LB) had the highest utilization (93.35%), whereas the drain layers (DL) had the lowest utilization (0.14%). The project manager may consider that LB as a cause of resource bottleneck and he/she may choose to add another LB. STREAM does not permit over allocation of resources and delays are automatically computed. The resource utilization computed and the required quantity for each type of resource is described in Table 27. The detailed schedule is shown in Table 28. Figure 23 shows the Resource Utilization screen in STREAM. Each bar shows the idle and usage percentage time for each resource assigned for tasks. Red presents idle time percentage whereas blue is use percentage time.

Table 27: Total Project Duration and Resource Utilization by STREAM for Scenario 1 and 2

Resource	**STREAM**	**Resource utilization (%)**
BBF	4	43.29
BE	4	19.68
CMC	3	34.53
CF	2	31.51
CLB	5	11.83
DL	1	0.14
LB	8	**93.35**
RR	4	13.38
Total Project Duration (TPD)	275	

Table 28: Detailed Schedule by STREAM for Scenario 1 and 2				
Task	**Start Date**	**End Date**	**Due Date**	**Deviation (d)**
A	5/28/2015	6/15/2015	6/16/2015	-0.29
B1	6/16/2015	6/24/2015	6/26/2015	-1.29
B2	6/25/2015	7/13/2015	7/12/2015	1.71
B3	8/1/2015	8/14/2015	7/28/2015	17.71
B4	9/22/2015	9/24/2015	8/1/2015	54.71
B5	10/20/2015	11/25/2015	9/21/2015	65.71
B6	10/20/2015	11/6/2015	8/15/2015	83.71
B7	11/26/2015	12/2/2015	9/14/2015	79.71
B8	12/3/2015	1/8/2016	10/21/2015	79.71
B9	1/14/2016	2/5/2016	11/20/2015	77.71
B10	1/9/2016	2/1/2016	11/20/2015	73.71
B11	2/9/2016	2/16/2016	11/26/2015	82.71
C1	6/16/2015	6/29/2015	6/30/2015	-0.29
C2	6/30/2015	7/2/2015	7/2/2015	0.71
C3	7/3/2015	8/8/2015	8/15/2015	-6.29
C4	7/14/2015	7/31/2015	8/15/2015	-14.29
C5	8/10/2015	8/13/2015	8/18/2015	-4.29
C6	8/15/2015	9/21/2015	9/20/2015	1.71
C7	9/26/2015	10/19/2015	10/21/2015	-1.29
C8	9/26/2015	10/19/2015	10/21/2015	-1.29
C9	11/9/2015	11/16/2015	10/25/2015	22.71
D1	11/17/2015	11/18/2015	10/28/2015	21.71
D2	11/19/2015	11/19/2015	12/2/2015	-12.29
D3	11/20/2015	11/25/2015	12/5/2015	-9.29
D4	11/26/2015	11/27/2015	12/8/2015	-10.29
D5	12/3/2015	12/3/2015	1/10/2016	-37.29
D6	1/9/2016	1/13/2016	1/10/2016	3.71
D7	2/2/2016	2/8/2016	1/13/2016	26.71
D8	2/6/2016	2/9/2016	1/13/2016	27.71
D9	2/17/2016	2/17/2016	1/25/2016	23.71
E1	5/28/2015	8/25/2015	8/30/2015	-4.29
E2	9/25/2015	9/25/2015	9/2/2015	23.71
F	2/18/2016	2/25/2016	2/20/2016	5.71

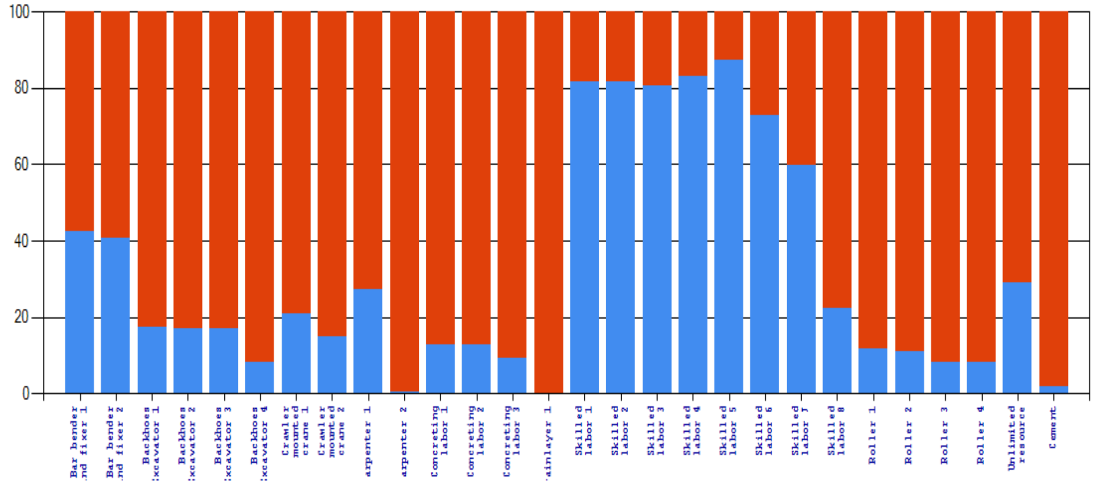


Figure 23: Resource Utilization Screen in STREAM

Case study 1 scenario 3: Finding the optimal quantity of resources

In this scenario 3, the objective is to find the optimal number of resource units for the aforementioned bottleneck resource LB. STREAM utilizes a simulation approach to determine the optimal number. Starting from the original 8 units of LB, one additional unit was considered at each simulation run. Table 29 shows ten simulation runs. After the eighth run STREAM does not find any improvement in total project duration and cost as shown in Table 29. Resource utilization with additional LBs for scenario 3 is shown in Table 30. The detailed schedule for scenario 3 is presented in Table AII.5 in the Appendix II. Figures 24 and 25 show project duration and cost as a function of skilled labors, respectively. Figure 26 shows a linear increase in project cost when more than fifteen skilled labor is used.

Table 29: Cost Analysis by STREAM for Scenario 3

	LB	Project	Start date	Finish date	% Time	Total Project	% Cost
1	**8**	**275**	**5/28/2015**	**2/25/2016**		**$14,347,972**	
2	9	268	5/28/2015	2/18/2016	2.55%	$13,344,318	6.83%
3	10	261	5/28/2015	2/11/2016	5.09%	$11,623,640	18.99%
4	11	225	5/28/2015	1/6/2016	18.18%	$6,256,752	56.39%
5	12	218	5/28/2015	12/30/2015	20.73%	$5,187,256	63.85%
6	13	190	5/28/2015	12/2/2015	30.91%	$2,253,504	84.29%
7	14	190	5/28/2015	12/2/2015	30.91%	$2,254,504	84.29%
8	**15**	**185**	**5/28/2015**	**11/27/2015**	**32.73%**	**$1,898,007**	**86.77%**
9	16	185	5/28/2015	11/27/2015	32.73%	$1,899,007	86.76%
10	17	185	5/28/2015	11/27/2015	32.73%	$1,900,007	86.76%

Table 30: Total Resource Duration and Resource Utilization by STREAM for Scenario 3		
Resource	**STREAM**	**Resource utilization (%)**
BBF	4	34.31
BE	4	25.13
CMC	3	26.47
CF	2	23.20
CLB	5	13.97
DL	1	0.20
LB	15	61.16
RR	4	16.64
Total Project Duration (TPD)	185	

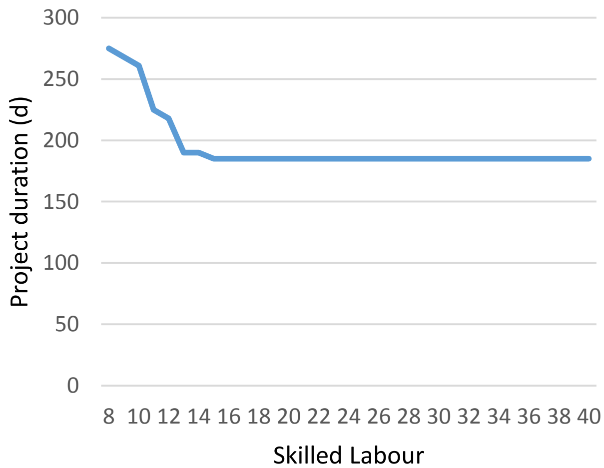


Figure 24: Project Duration (days)

Figure 25: Project Cost (million USD)

Figure 26: Project Cost (million USD) for Skilled Labour between 15 and 38

Case study 1: Scenario 4: Human resources with multiple skills

This scenario deals with the case of human resources having multi-skills. The assumptions are that: resource calendars, task precedence and task durations are the same as in scenario 1, the quantity of resources used is either equal to or less than scenario 1, but labours have multiple skills.

Four different simulation runs were carried out for this scenario. In run 1, the LBs have CF and CLB skills, and all CFs and CLBs were removed. The limit on LBs was kept the same as in Table 23. In run 2, the CLBs have CF skills, and all CFs were removed. The number of CLBs was kept the same as in Table 23. In run 3, LBs have additional CF skills and the two CFs were removed. The number of LBs was kept the same as in Table 23. In run 4, LBs, CFs and CLBs have three skills each. The number of LBs, CFs and CLBs was kept the same as in Table 23.

Table 31 shows the total project duration and total project cost for the four simulation runs. The total project duration and total project cost for the four simulation runs are compared with the baseline scenario with TPD of 275 days. The negative value of duration (days) refers to additional days required, compared to the baseline value. Negative values on cost difference refer to the increase in cost considering the baseline value. Figures 27 and 28 show project durations and costs based on the four simulation runs

Table 31: Total Project Duration and Cost Analysis by STREAM for Scenario 4							
	TBD	**Start date**	**Finish date**	**% Time**	**Total Project**	**Cost**	**% Cost**
1	370	5/28/2015	5/30/2016	-34.55%	$39,801,807	-$25,453,834	-177.40%
2	304	5/28/2015	3/25/2016	-10.55%	$22,289,949	-$7,941,976	-55.35%
3	296	5/28/2015	3/18/2016	-7.64%	$17,143,265	-$2,795,292	-19.48%
4	224	5/28/2015	1/5/2016	18.55%	$7,923,455	$6,424,517	44.78%

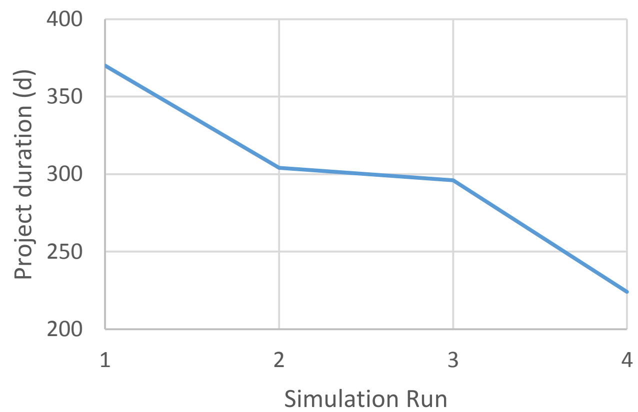


Figure 27: Project Duration (days) for Four Simulation Runs

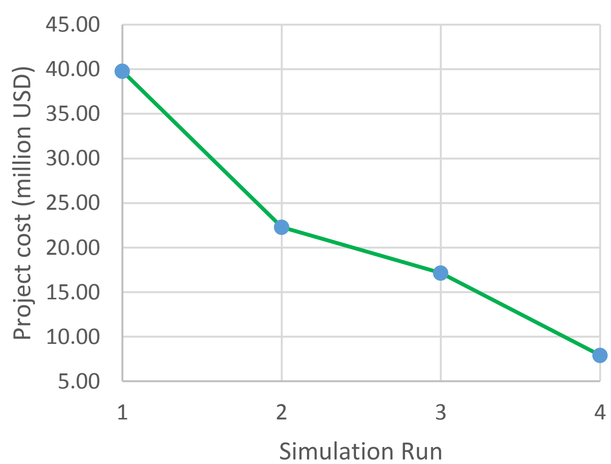


Figure 28: Project Cost (million USD) for Four Simulation Runs

Simulation run number four of scenario 4 provides the “best” solution, where TPD is reduced from 275 days to 224 days (51 fewer days), and total project cost is reduced from $14,347,972 to $7,923,455, a 44.78% cost reduction amounting to $6,424,517.

Case study 1: Scenario 5: Tasks with multiple modes

This scenario deals with the case of multi-modes. The assumptions for this scenario are similar to scenario 4, except for that labour does not have multiple skills, but multi-modes are allowed now. Figure 29 shows the Task/Modes screen of STREAM. It shows two mode for Task 2.

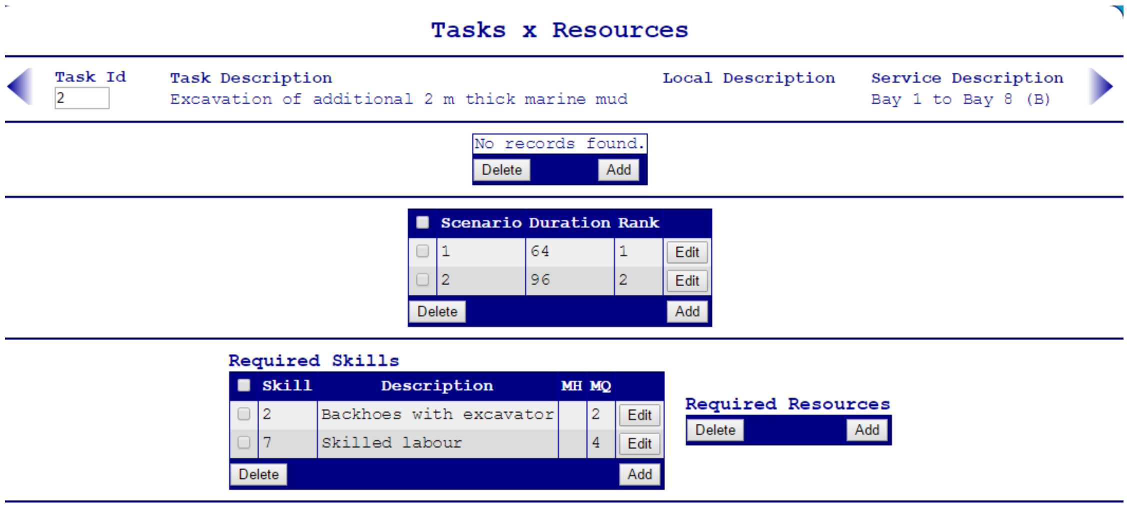


Figure 29: Task/Mode/Resources screen in STREAM

Under this scenario, TPD is reduced from 275 days to 255 days (20 fewer days) and total project cost is reduced from $14,347,972 to $10,855,420, a 24.3% cost reduction amounting to $3,492,551, as shown in Table 32.

Table 32: Project Duration and Cost Analysis by STREAM for Scenario 5

TPD	Start date	Finish	Total Project	Total Cost for 275	Dif. in Cost	Dif. in days
255	5/28/2015	2/5/2016	$10,855,420	$14,347,972	$3,492,551	20

Case study 1: Scenario 6: Scheduling considering that humans have multiple skills and that tasks have multiple modes

This scenario deals with the case of multi-skills and multi-modes. The assumptions for this scenario are the same as in scenario 4, but project tasks can also have multi-modes. Under this scenario, project duration is reduced from 275 days to 218 days (57 fewer days) and total project cost is reduced from $14,347,972 to $6,460,206, a cost reduction of nearly 50%, amounting to $7,887,765, as shown in Table 33. Table AII.6 in the Appendix II shows the detailed schedule, along with schedule deviations, as generated by STREAM for this scenario.

Table 33: Project Duration and Cost Analysis by STREAM for Scenario 6

TPD	Start date	Finish date	Total Project	Total Cost for 275	Dif. in	Dif. in days
218	5/28/2015	12/30/2015	$6,460,206	$14,347,972	$7,887,765	57

Case study 1: Scenario 7: Resources can be non-renewable or renewable

This scenario deals with non-renewable resources with replenishment plan (doubly constrained resource). This scenario is proposed to demonstrate how STREAM can warn project managers on non-renewable resources shortages with a replenishment plan. It does not consider multi-modes and multi-skills capabilities for simplicity purposes, although all the presented capabilities can be used simultaneously. In other words, the assumptions for this scenario are the same as in scenario 1, but a non-renewable resource is also required for a given task.

In this scenario, task C2, in addition to its original resource requirements shown in Table AII.1 described in the Appendix II, also requires 100 Kg of cement. This scenario assumes that 50 Kg of cement is available on hand, and 60 Kg is on order to be delivered on 07/15/2015. Table 34 presents one of the STREAM's outputs. It shows the tasks that are waiting for resources along the planning horizon, their eligible dates (the earliest date that all task predecessors are finished), the start date (the date the task is actually started), the waiting time (the difference in days between Start Date and Eligible Date), the needed resource, and the quantity that is lacking. For instance, task C2 was eligible to start at 6/29/2015, but it could not be started, even though all the required resources were available, except for the 50 kg of cement. Therefore, the task waited until 7/15/2015 to start. Whether or not there are non-renewable resources, STREAM can always warn about the required but unavailable resources and inform resulting task waiting times. This output is especially important in projects (constructions, shipyards, government, etc) where several resources (renewable or non-renewable) are required by individual tasks. Matching multiple resources for task execution is essential. Additionally, resources must be presented at commonly available periods in order to perform the tasks.

Table 34: Task waiting for resources in scenario 7 (by STREAM)

Task	Eligible Date	Start Date	Waiting time (d)	Resource Needed	Qty.
C2	**6/29/2015**	**7/15/2015**	**15.62**	**Cement**	**50**
C4	7/20/2015	7/28/2015	7.62	Skilled Labor	3
B4	7/27/2015	8/27/2015	30.62	Skilled Labor	1
E2	8/25/2015	9/2/2015	7.62	Skilled Labor	1
C5	8/26/2015	9/3/2015	7.62	Skilled Labor	2
B6	9/1/2015	9/10/2015	8.62	Skilled Labor	2
B5	9/1/2015	9/3/2015	1.62	Skilled Labor	2
C6	9/9/2015	10/10/2015	30.62	Skilled Labor	1
B7	10/9/2015	11/17/2015	38.62	Skilled Labor	1
C8	11/16/2015	11/24/2015	7.62	Skilled Labor	3
B8	11/23/2015	12/17/2015	23.62	Skilled Labor	1
C9	12/16/2015	1/25/2016	39.62	Skilled Labor	1
B10	1/22/2016	2/2/2016	10.62	Skilled Labor	2
D1	2/1/2016	2/16/2016	14.62	Skilled Labor	1
D5	2/26/2016	3/7/2016	9.62	Concreting Labor	1

For scenario 7, TPD was 307 days and cost was $24,625,122. If a project manager renegotiates with the supplier anticipation on the cement order delivery date, project duration and cost reductions can be obtained. For example, if the 60 Kg of cement is delivered on 6/29/2015, TPD and cost would be 275 days and $14,347,972, respectively.

Comparison of scenarios

This case study demonstrate the advantages of simulating multiple scenarios, as opposed to providing a single solution. In practice, these solutions exist, but project managers do not have the luxury of evaluating such alternatives on a day-to-day basis.

The first two scenarios do no provide resource allocation flexibility, TPD and cost were 275 days and $14,347,972, respectively. TPD and cost obtained by STREAM for Scenario 3 (when additional resources are allowed) were 185 days and $1,898,007, respectively. If no additional resources can be added, scenarios 4, 5 and 6, provide a practical short-term solutions. Scenarios 4 and 5 provide better results when compared with the base schedule. Under scenario 6 (when no resource could be added but multi-skills and multi-modes are considered), STREAM found the "best" solution. TPD is reduced from 275 days to 218 days (57 fewer days) and total project cost is reduced from $14,347,972 to $6,460,206, representing a cost reduction of nearly 50%. Finally, the issue of non-renewable resource shortage which exists in real-life projects, is described in scenario 7. In this case, STREAM provided information that can contribute to avoid critical non-renewable resource shortages, e.g., the renegotiation of the critical resource resulted in $10,000,000 savings in the overall project cost. STREAM generates outputs for each scenario in about 2 seconds on a standard laptop computer.

Table 35 summarizes total project duration and total project costs for the seven scenarios, whereas Figure 30 and 31 provide graphical representations of the data displayed in Table 35. Start date is a hypothetical project starting date. Project managers define it. STREAM calculated the finish date for each individual task under the given scenario. Table 35 summarizes TPD, start date, finish date, and cost for all scenarios of case study 1.

Table 35: Total Project Duration and Total Cost for the Seven Scenarios				
	STREAM			
Scenario	**TPD**	**Start date**	**Finish date**	**Total Project Cost (USD)**
1	275	5/28/2015	02/25/2016	$14,347,972
2	275	5/28/2015	02/25/2016	$14,347,972
3	185	5/28/2015	11/27/2015	$1,898,007
4	224	5/28/2015	1/5/2016	$7,923,455
5	255	5/28/2015	2/5/2016	$10,855,420
6	**218**	**5/28/2015**	**12/30/2015**	**$6,460,206**
7	307	5/28/2015	3/30/2016	$24,625,122

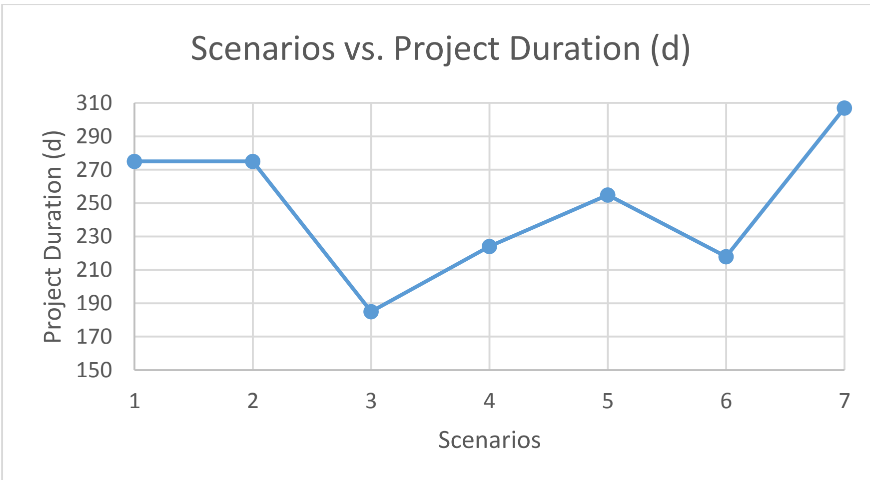


Figure 30: Project Duration (days) for Seven Scenarios

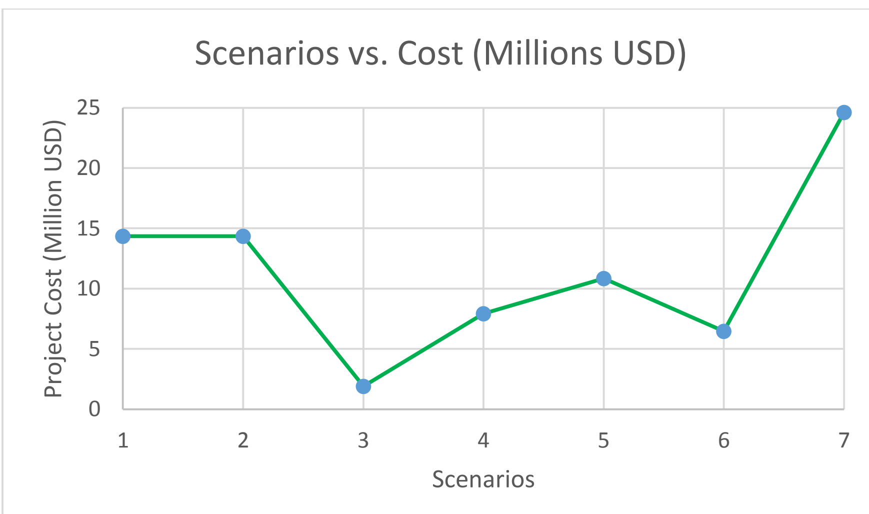


Figure 31: Project Cost (Millions USD) for Seven Scenarios

4.2 Case study 2: Oil Refinery Maintenance

Refineries like other industries need to perform preventive maintenance, renovations, or upgrades on a regular basis. Refinery maintenance tasks are planned in order to service and to upgrade equipment and the reactor in a refinery. These procedures are carried out to ensure safe operations, stay competitive, and meet government regulations. The refinery usually shuts down part of the production during planned maintenance. Planned shutdowns are also known as turnarounds. Turnarounds can last for a week or a few months. Turnaround duration mainly depends on the project complexity [Kendrickoil, 2015].

4.2.1 Inputs

The details of an oil refinery turnaround project was described in Siu et al. (2015). It requested 107 tasks and 19 different types of resources. Limits of resources required to accomplish the 107 tasks are shown in Table 36. Table AIII.1 in the Appendix III describes the tasks, their durations, predecessors and their resource requirements, e.g. Task 1 requires two Boilermakers and two Boilermaker welders.

Table 36: Resource Description (Siu et. al, 2015)

	Resource	Description	Limits
1	A	Boilermaker	0–20
2	B	Boilermaker welder	0–10
3	C	MSG80 3,600 t	0–2
4	D	Rigger	0–7
5	E	Scaffolder	0–10
6	F	Iron worker	0–10
7	G	Sterling 130 t crane	0–6
8	H	Technical	0–5
9	I	Inspection	0–2
10	J	Complex process operator	0–2
11	K	Refractory	5–15
12	L	Liquid penetrant inspection	0–1
13	M	X-Ray	0–1
14	N	PAUT inspection	0–1
15	O	Painter	0–4
16	P	Pipefitter	0–5
17	Q	Pipefitter welder	0–4
18	R	Inspector	0–2
19	S	Supervisor	0–2

4.2.2 Case study 2: Mathematical formulation by Siu et al. (2015)

Siu et al. (2015) formulated the turnaround project with the objective to find the shortest total project duration, the earliest finish time for each task and the minimal resource requirements. They formulated the problem as optimization problem and use strong mathematical formulation. The optimal resource requirements were identified between the lower and upper bounds of resource supply limits. The objective function defined by Siu et al. (2015) was:

$$\text{minimize } f = \underbrace{\sum_{\text{task}} \sum_{t=0}^{T} (\alpha^{t}{}_{\text{Task}} t x^{t}{}_{\text{task}})}_{\text{i) Task completion times}} + \underbrace{\sum_{t=0}^{T} (\beta^{t}{}_{\text{proj}} t x^{t}{}_{\text{proj}})}_{\text{ii) project completion time}} + \underbrace{\sum_{\text{Res}} \gamma_{\text{Res}} R_{\text{Res}}}_{\text{iii) Resource supply}} \tag{37}$$

The objective function consists of three parts: i) task completion times, ii) project completion time, and iii) resource supply. Parameter $\alpha^{t}{}_{\text{task}}$ is the relative importance of completion time of particular task at time t; parameter $\beta^{t}{}_{\text{Proj}}$ is the relative importance of completion time of the project at time t; and parameter γ_{Res} is the relative importance of supply quantity for particular resource. Variable $x^{t}{}_{\text{task}}$ is a binary variable, 1 represents a task completion at time t, or 0 otherwise; $x^{t}{}_{\text{Proj}}$is a binary variable, 1 represents a project completion at time t, or 0 otherwise; R_{Res} is a supply limit of particular resource. Siu et al. (2015) also defines the task precedence and resource limits constraints.

Siu et al. (2015) proposed a two-stage optimization solution. The first stage is to determine optimum project duration (shortest duration) and optimum resource supply (minimal quantity of resources). The second stage, given the analytical solutions resulting from the first stage, is designed to minimize the task finish times. The first stage is performed by setting the parameters $\{\alpha^{t}{}_{\text{task}}, \beta^{t}{}_{\text{Proj}}, \gamma_{\text{Res}}\}$ as $\{0, \beta^{t}{}_{\text{Proj}}, \gamma_{\text{Res}}\}$, whereas in the second stages the parameters are as $\{\alpha^{t}{}_{\text{task}}, 0, 0\}$.

The objective was to determine the schedule which provides the shortest total project duration, the earliest finish time for each task, and utilizes minimum resources. Parameters $\{\alpha^{t}{}_{\text{task}}, \beta^{t}{}_{\text{Proj}}, \gamma_{\text{Res}}\}$ are assumed known. No resource calendars were utilized and it was assumed that all resources work 7 days a week, 24 hours a day. Case of multi-modes, multiple calendars, and multiple skills were not modelled.

Siu et al. (2015) used Matlab to solve the turnaround project. The optimum resource requirements and total project duration determined by Siu et al. (2015) is shown in Table 37. Table AIII.2 in the Appendix III shows the earliest finish time for each task, 0 is used as the project starting time.

Table 37: Optimal total project duration and resource requirements (Siu et. al, 2015)

Resource	Optimal resource requirements
A	14
B	9
C	1
D	6
E	6
F	6
G	6
H	2
I	2
J	2
K	7
L	1
M	1
N	1
O	2
P	2
Q	1
R	1
S	1
Total Project Duration (TPD)	**173 (hours)**

4.2.3 Case study 2: STREAM formulation

The outputs generated by STREAM for this case study included: i) Task waiting times, ii) Lack of resources when needed and iii) Resource utilization. The outputs can support project managers to reduce project duration, to find the earliest finish time for tasks and to determine the minimal resource requirements. Task waiting times output assists project managers to identify how long a given task will be waiting for resources. Lack of resources output helps project managers determine which resources are not available when a given task requires them. Resource utilization output shows how individual resources are utilized over the entire project duration. These three STREAM's outputs empower project managers in obtaining good and sometimes optimal solutions. In additional the optimal solution, STREAM provides alternate solutions to the project managers. Project manager has the option to deploy non-optimal feasible solution during project duration.

STREAM does not utilize the three parameters, ${\alpha^{t}}_{task}$, ${\beta^{t}}_{Proj}$, and γ_{Res}. STREAM was able to find the optimal solution to the oil refinery turnaround project without mathematically formulating it as an optimization problem. In addition, STREAM provided critical information to the project manager.

For the turnaround project case study, a rule which prioritizes a given task which has the largest number of successors is applied dynamically, at different point in time. This rule enables STREAM to minimize delay to least number of tasks. By applying this rule, STREAM aims to deliver a solution targeting the first and third goals (shortest project duration and earliest task finish times). STREAM also meets the goal to utilize least possible amount of resources. E.g., if 20 boilermakers exist in the resource pool, it does not imply that STREAM will assign all of them. STREAM assigns resources such that resources conflicts among tasks are avoided. If more than 20 boilermakers are required for tasks, some tasks will have to wait until the boilermakers are available. The same logic is applied for other resources. This procedure is directly related to the second goal (optimum resource supply).

The case study 2 has a unique scenario with the objective to determine the schedule which provides the shortest total project duration, the earliest finish time for each task, and utilizes minimum resources. Different simulation runs were conducted to demonstrate STREAM's ability to find diverse solutions. The results are validated with the baseline results (Tables 37 and AIII.2 in the Appendix III) provide by Siu et al. (2015). Four different simulation runs were carried out for the turnaround project. Details of the four simulation runs are as follows:

Run 1: Determine the shortest total project duration, the earliest finish time for each task, and the minimal quantity of resources. Table 36 shows resource supply limits, task precedence and duration are shown in Table AIII.1 in the Appendix III. No resource calendars were used in order to compare results with Siu et al. (2015).

Run 2: Same as run 1 (data and goals), but priorities were changed for two tasks based on the waiting time for tasks output. STREAM finds values for the three unknown parameters $\{\alpha^{t}{}_{task}, \beta^{t}{}_{Proj}, \gamma_{Res}\}$.

Run 3: Determine the shortest duration project and the minimal quantity of resources (Definition of Stage 1 by Siu et al. 2015). In this run, resource limits were changed based on resource utilization output calculated by STREAM in run 2

Run 4: Determine the earliest finish time for each task (Definition of Stage 2 by Siu et al. 2015 but increasing in resources are allowed). STREAM determined the earliest finish time for each task by eliminating any waiting time for tasks to be started. Resource supply limits are increased based on waiting time for tasks output, and the three parameters $\{\alpha^{t}{}_{task}, \beta^{t}{}_{Proj}, \gamma_{Res}\}$ are known based on run 2 results.

4.2.4 Results applying STREAM

Total project duration time computed by STREAM was 173 hours. It is same as the one found by Siu et al. 2015.

Simulation run #1

In run 1, STREAM was able to finish 103 tasks exactly at the same finish time as in Siu et al. 2015. Only four tasks (93, 96,101, and 103) finished at different times. STREAM finished tasks 93 and 96 in 3, and 6 hours, respectively, whereas Siu et al. 2015's finished them in 5 and 8 hours, respectively. STREAM finished task 101 and 103 in 5 and 8 hours, whereas in Siu et al. 2015 they finished in 3 and 6 hours respectively. Table AIII.3 in the Appendix III shows in detail finish times for each task along with a comparison with baseline. It is clear from the results that STREAM found a different optimal solution.

Tasks 93 and 96 were finished by STREAM two hours earlier than in Siu et al. 2015, whereas 101 and 103 were finished two hours later. Both methods provided the optimal solution for the turnaround project based on the objective function defined by Siu et al. (2015). STREAM does not explicitly search for an optimal solution. However, the results obtained by STREAM were comparable to those by Siu et al. (2015), who formulated the turnaround problem an optimization problem.

For simulation run 1, the task waiting times for required resources are shown in Table 38. Notice, the four tasks described are the only ones without a perfect task flow. Perfect task flow is when the task becomes eligible to start, and all its required resources are available and assigned to the task, without any waiting time. The four tasks described are those which may offer opportunities to improve the solution. This output was not provided by Siu et al. (2015). The ability to show which task will wait for resource is very valuable for the project manager. Manager can then plan for such event.

Table 38: Task waiting for resources (STREAM Run1)

Task	Eligible Date	Start Date	Waiting time (h)	Resource Needed	Qty.
93	5/28/2015 0:00	5/28/2015 2:00	2	Inspection	1
94	5/28/2015 0:00	5/28/2015 3:00	3	Inspection	1
101	5/28/2015 0:00	5/28/2015 4:00	4	Inspection	1
102	5/28/2015 0:00	5/28/2015 1:00	1	Complex process operator	1

In Table 38, the date 05/28/2015 is the hypothetical turnaround project start date. Eligible date is the date on which a task becomes eligible to start. The difference between start and eligible dates is the waiting time due to a lack of resource when needed. The lack of resource and its quantity are also described in Table 38. Task 93 is eligible to start at the beginning of the simulation (5/28/2015 0:00), however it can only be started 2 hours later due to the lack of resource. One resource (Inspection) is lacking and it becomes available only at 5/28/2015 2:00. The same logic is used for the other three tasks described in Table 38. STREAM was able to find a different optimal solution in run 1 with the goal of finding the shortest project duration, optimum resource supply, and earliest task finish times.

The line curve in Figure 32 represents the finish time for each individual task calculated by the mathematical formulation in Siu et al. (2015) (baseline schedule). The line curve was constructed with the finish times displayed in Table AIII.2 in the Appendix III. The shaded region represents the finish times calculated by STREAM. Tasks 93 and 96 were finished 2 hours earlier, whereas Tasks 101 and 103 were finished 2 hours later than the baseline schedule. Table AIII.3 in the Appendix III displays the data.

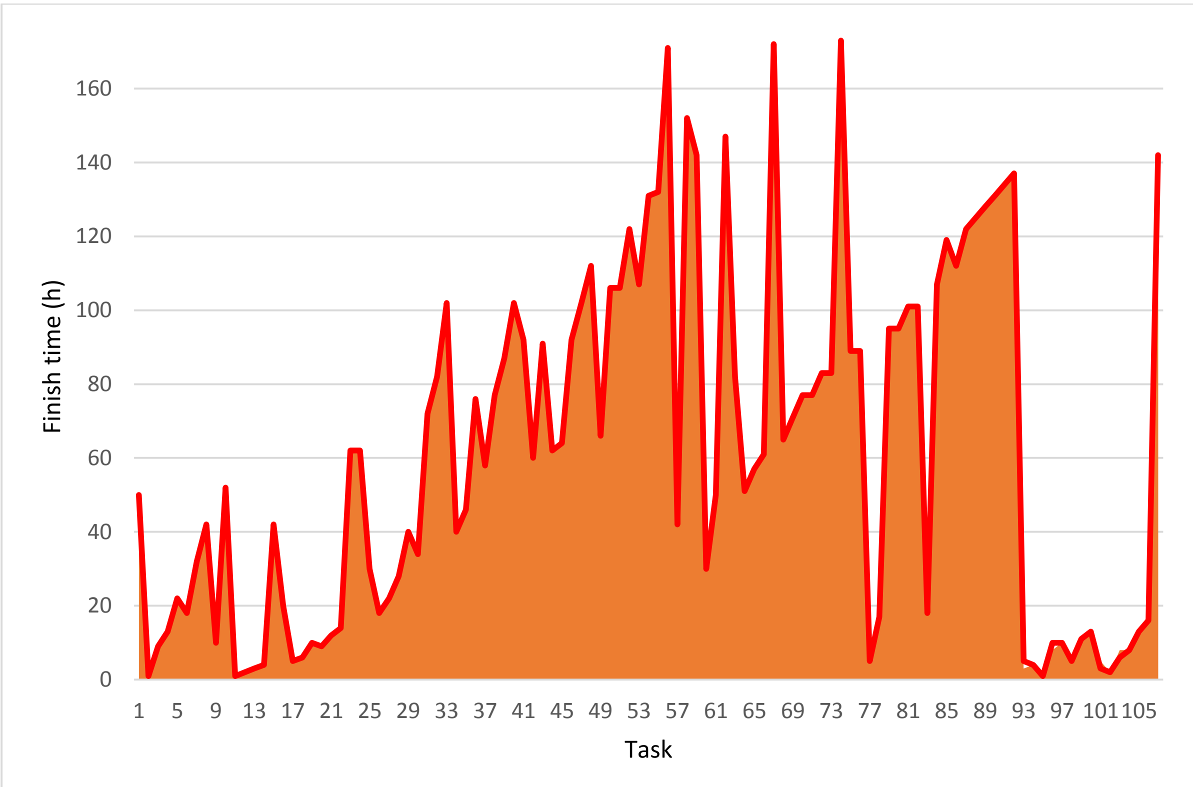


Figure 32: Task finish times for Siu et al., 2015 and STREAM Run 1

Simulation Run #2

Based on the outputs generated by run # 1, priorities for Tasks 101 and 93 were changed to 1000, and 400, respectively. STREAM utilizes a priority scale of 1-1000, with 1 being the lowest priority and 1000 the highest priority. By default all priorities are set to 500. Table 39 shows that the waiting times of Task 101 and 93 were changed based on the new priorities assigned. The required resources which are not available when required are also shown in Table 39.

Table 39: Task waiting for resources (STREAM Run 2)

Task	Eligible Date	Start Date	Waiting time (h)	Resource Needed	Qty.
93	5/28/2015 0:00	5/28/2015 4:00	4	Inspection (I)	1
94	5/28/2015 0:00	5/28/2015 3:00	3	Inspection	1
101	5/28/2015 0:00	5/28/2015 2:00	2	Inspection	1
102	5/28/2015 0:00	5/28/2015 1:00	1	Complex process operator	1

Figure 33 shows the resource utilization for the data described in Table AIII.4 in the Appendix III. Figure 33 shows the resource utilization graphs for each individual team. Resource group A (Boilermakers) and B (Boilermaker welders) have the highest utilization.

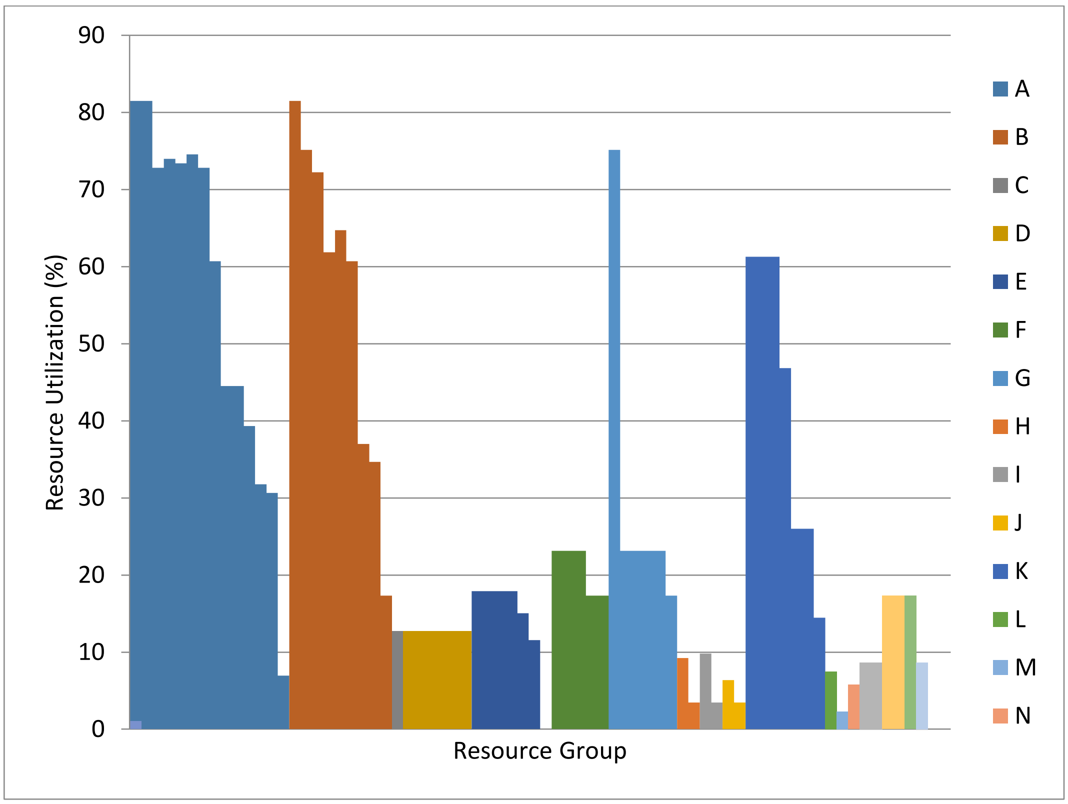


Figure 33: Resource utilization

The line curve in Figure 34 represents the finish time for each individual task calculated by the strong formulation in Siu et al. (2015). The line curve was constructed with the finish times displayed in Table AIII.2 in the Appendix III. The green shaded region represents the finish times calculated by

STREAM. Table AIII.3 in the Appendix III displays the data. None of the finish times calculated by STREAM were greater than the optimal solution. Thus, the green shaded region does not go beyond the line curve.

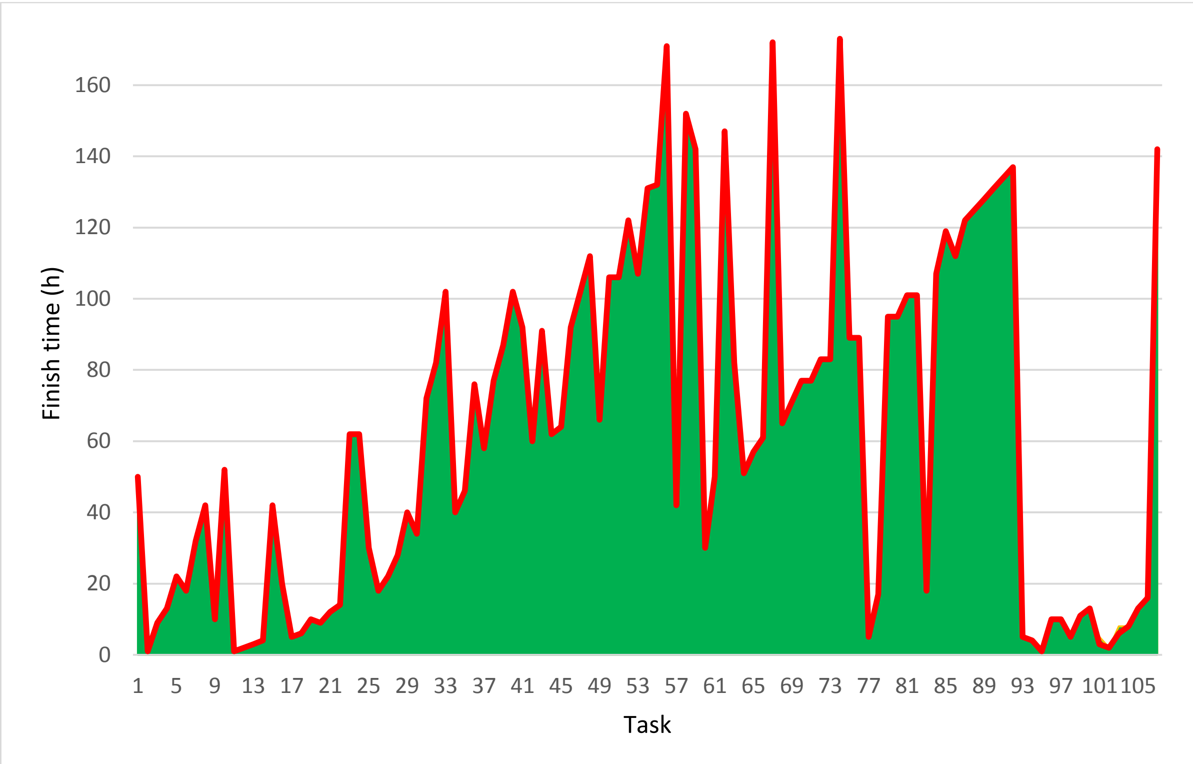


Figure 34: Task finish times for Siu et al., 2015 and STREAM Run 2

Simulation Run #3

In simulation run 3, STREAM attempts to minimize resource requirements while keeping project duration time constant to 173 hours. In order to do so, the resource utilization provided by STREAM in simulation run 2 is used. Resources were found to have low utilization (Table AIII.4 in the Appendix III), the 14th boilermaker (A) has only 6.94% of utilization duration the simulation. The other 13 boilermakers have higher utilization about 30%. This is an indication that 14 boilermakers may not be necessary to deliver the project in 173 hours, and one boilermaker can be reduced. The same logic is applied for other types of resources. The quantity of resource used for each type was defined by STREAM during the simulation run 2. The resource supply limits were the same as shown in Table 36, however STREAM has required the optimal resource supply. STREAM will required resources to finish tasks as earlier as possible. Resource types C, M, N, Q, R, and S were required only one unit for each type. Run 3, therefore, represents a case where resource supply limits are even more constrained while keeping the shortest project duration constant. Table 40 shows the resource requirements for run # 3. STREAM reduced the number of Boilermakers (A) from 14 to 13, number of Iron workers (F) and Sterling 130 t crane (G) was reduced from 6 to 5, number of technical (H), inspection (I), and complex process operator (J) were reduced from 2 to 1. Any reduction beyond of these limits will result in increasing the TPD. Table AIII.3 in the Appendix III shows the task finish times respectively for simulation run 3. It was expected that with fewer resources, some task finish times will increase. The task waiting times and required resources which are not available when needed are shown in Table 41 for simulation 3. It was also expected that with fewer resources, some waiting times will increase. The goal of STREAM is to provide several feasible options, it is left to the project manager to decide which option is appropriate for the current scenario.

Table 40: Case of reduced resource (STREAM Run 3)

	Resource	Description	Limits
1	A	Boilermaker	13
2	B	Boilermaker welder	9
3	C	MSG80 3,600 t	1
4	D	Rigger	6
5	E	Scaffolder	6
6	F	Iron worker	5
7	G	Sterling 130 t crane	5
8	H	Technical	1
9	I	Inspection	1
10	J	Complex process operator	1
11	K	Refractory	7
12	L	Liquid penetrant inspection	1
13	M	X-Ray	1
14	N	PAUT inspection	1
15	O	Painter	2
16	P	Pipefitter	2
17	Q	Pipefitter welder	1
18	R	Inspection	1
19	S	Supervisor	1

Table 41: Task waiting for resources (STREAM run 3)					
Task	**Eligible Date**	**Start Date**	**Waiting time (h)**	**Resource Needed**	**Qty.**
102	5/28/2015 0:00	5/28/2015 2:00	2	Technical	1
101	5/28/2015 0:00	5/28/2015 5:00	5	Technical	1
95	5/28/2015 0:00	5/28/2015 1:00	1	Technical	1
94	5/28/2015 0:00	5/28/2015 7:00	7	Technical	1
93	5/28/2015 0:00	5/28/2015 10:00	10	Technical	1
60	5/28/2015 0:00	5/29/2015 16:00	16	Iron worker	1
12	5/28/2015 1:00	5/28/2015 3:00	2	Technical	1
14	5/28/2015 5:00	5/28/2015 6:00	1	Technical	1
17	5/28/2015 7:00	5/28/2015 8:00	1	Technical	1
100	5/28/2015 12:00	5/28/2015 13:00	1	Boilermaker	1
21	5/28/2015 13:00	5/28/2015 14:00	1	Boilermaker	1
10	5/28/2015 22:00	5/29/2015 16:00	18	Sterling 130 t crane	1
61	5/29/2015 18:00	5/30/2015 2:00	8	Boilermaker	2

The line curve in Figure 35 represents the finish time for each individual task calculated by the mathematical formulation in Siu et al. (2015). The line curve was constructed with the finish times displayed in Table AIII.2 in the Appendix III. The shaded region represents the finish times calculated by STREAM. Table AIII.3 in the Appendix III displays the data. Some of the finish times calculated by STREAM were greater than the optimal solution. Tasks 70, 73, and 85 have finish times greater than the baseline schedule. Thus, the shaded region goes beyond the line curve. It is due to the intentional reduction of resources.

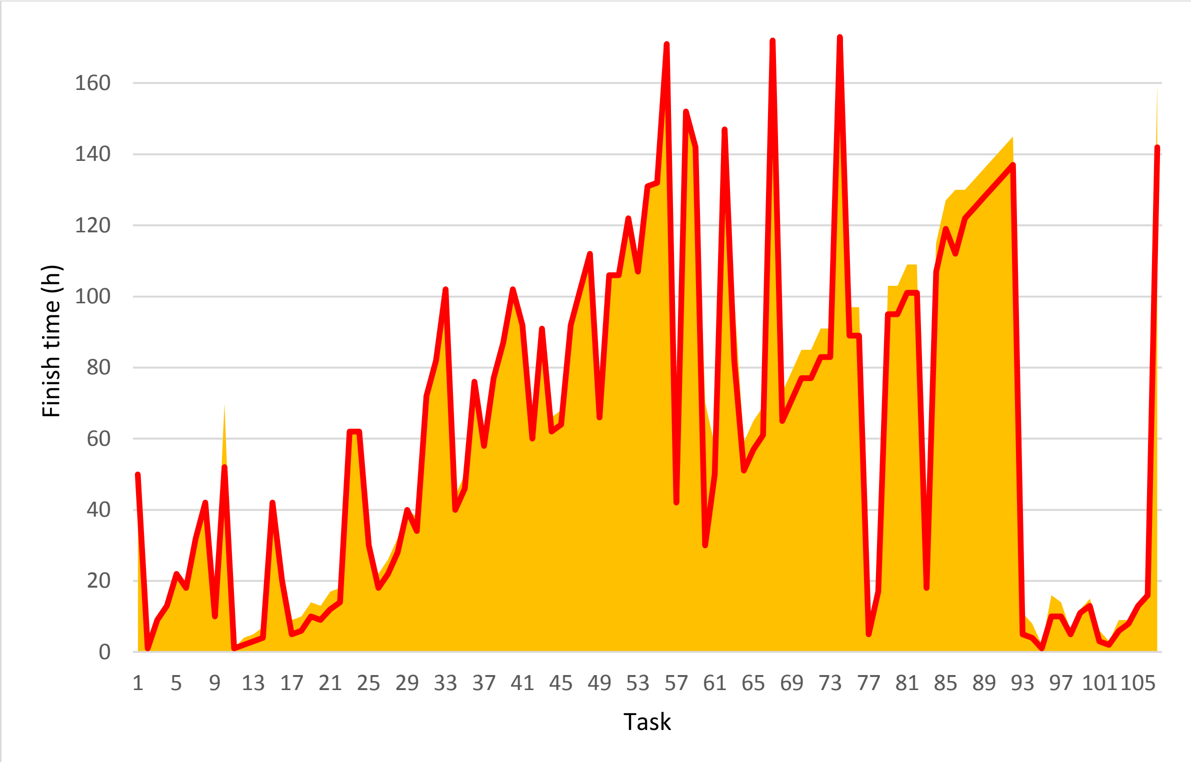


Figure 35: Task finish times for Siu et al., 2015 and STREAM Run 3

Simulation Run #4

The simulation run 4 aims to find the perfect task flow among all the tasks. That is, no task waits due to resource unavailability. Ultimately, the goal is to find the earliest finish time for each individual task while the least resource requirement is found. No queue is created because once a given task becomes eligible, its required resources are available. In order to achieve this goal, two STREAM outputs needed are utilized: i) task waiting times, and ii) required resources which are not available when required. Simulation run 2 outputs is used in run 4. Table 39 shows these outputs for simulation run 2. At first glance 3 resources of inspection and 1 complex process operator should be added. However, when 3 resources of inspections (I) are added, needs for more technical (H) and complex process operator (J) come up simultaneously. Tasks 93, 94, 101 and 102 have the same resource requirements (H[1]; I[1]; J[1]) as shown in Table 37. Table 42 shows the resource requirements for run 4 only. STREAM increased the number of technical (H), inspection (I), and complex process operator (J) from 2 to 6. Table

43 shows the difference between simulation run 4 and the baseline results in terms of resource requirements. Table AIII.3 in the Appendix III shows which tasks have been benefit from the additional resources, e.g., Task 93, 94, 96, 97, 101, 102, 103, 104, 105, and 106 are finishing earlier 4, 3, 4, 3, 2, 1, 2, 1, 1, and 1 hours respectively. Additional resources do not affect task finish times and project duration time. STREAM was able to schedule all tasks without waiting times.

Table 42: Case of increased resource (STREAM Run 4)

	Resource	Description	Limits
1	A	Boilermaker	14
2	B	Boilermaker welder	9
3	C	MSG80 3,600 t	1
4	D	Rigger	6
5	E	Scaffolder	6
6	F	Iron worker	6
7	G	Sterling 130 t crane	6
8	H	Technical	6
9	I	Inspection	6
10	J	Complex process operator	6
11	K	Refractory	7
12	L	Liquid penetrant inspection	1
13	M	X-Ray	1
14	N	PAUT inspection	1
15	O	Painter	2
16	P	Pipefitter	2
17	Q	Pipefitter welder	1
18	R	Inspection	1
19	S	Supervisor	1

The line curve in Figure 36 represents the finish time for each individual task calculated by the mathematical formulation in Siu et al. (2015). The line curve was constructed with the finish times displayed in Table AIII.2 in the Appendix III. None of the finish times calculated by STREAM were greater than the optimal solution provided by Siu et al. (2015). Thus, the shaded region does not go beyond the line curve. Instead, some finish times in STREAM were smaller than the optimal solution. It is due to the addition of resources.

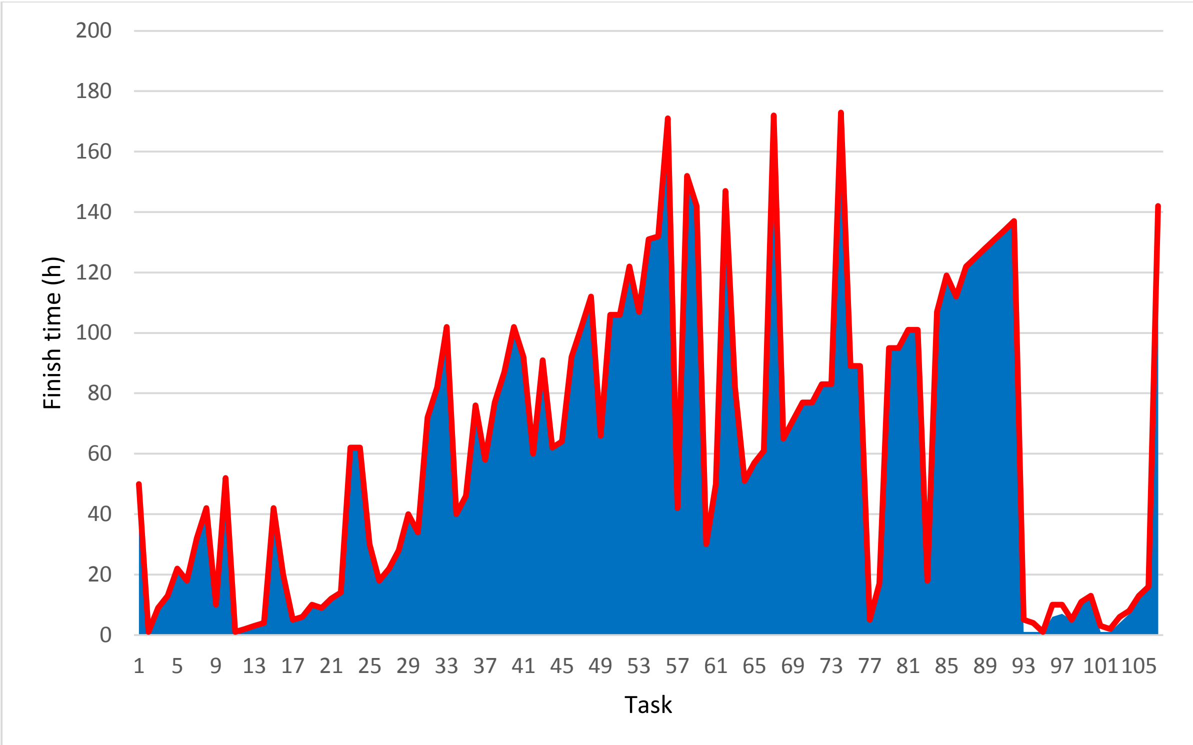


Figure 36: Task finish times for Siu et al., 2015 and STREAM Run 4

Simulations results and discussion

In runs 1 and 2, the optimal resource supply is also found as in Siu et al. (2015). Table 43 shows the baseline result (resource requirements) and validates it with the four simulation runs. The column Diff. in Table 43 represents the difference between STREAM and Siu et al. (2015) results. If it is 0 (zero) that means no difference was found, quantity of resource type A quantity in Siu et al. (2015) was 14 and STREAM also found 14 for runs 1 and 2.

STREAM was able to find the optimal solution in run 1 and run 2 by utilizing the two aforementioned STREAM outputs (Task waiting times and resource utilization). STREAM is not a "black box" for solving the optimization problems, where inputs come in and outputs are produced. STREAM is able to track in details what is happening during each simulation run. Notice that simulation runs 1 and 2 are optimal solutions, but knowing this information does not mean that tasks are not waiting for resources. Based on Tables 38, 39 and 41, there are tasks waiting for resources at different points in time.

Therefore, this additional information provided by STREAM empowers project managers to find different solution under the optimal solutions. Tables 43 and Table AIII.3 in the Appendix III show the resource requirements and finish times for the four simulation runs, respectively.

Table 43: Solution results

Resource	Run1		Run 2		Run 3		Run 4	
	Qty.	Diff.	Qty.	Diff.	Qty.	Diff.	Qty.	Diff.
A	14	0	14	0	13	-1	14	0
B	9	0	9	0	9	0	9	0
C	1	0	1	0	1	0	1	0
D	6	0	6	0	6	0	6	0
E	6	0	6	0	6	0	6	0
F	6	0	6	0	5	-1	6	0
G	6	0	6	0	5	-1	6	0
H	2	0	2	0	1	-1	6	4
I	2	0	2	0	1	-1	6	4
J	2	0	2	0	1	-1	6	4
K	7	0	7	0	7	0	7	0
L	1	0	1	0	1	0	1	0
M	1	0	1	0	1	0	1	0
N	1	0	1	0	1	0	1	0
O	2	0	2	0	2	0	2	0
P	2	0	2	0	2	0	2	0
Q	1	0	1	0	1	0	1	0
R	1	0	1	0	1	0	1	0
S	1	0	1	0	1	0	1	0

Figure 37 shows graphically the resource requirements for the four simulation runs and Siu et al. (2015) results. The line is the optimal resource requirement for resources obtained in Siu et al. (2015). Runs 1 and 2 obtained the optimal resource requirements, run 3 obtained fewer resources than the optimal solution, and run 4 more resources than the optimal solution.

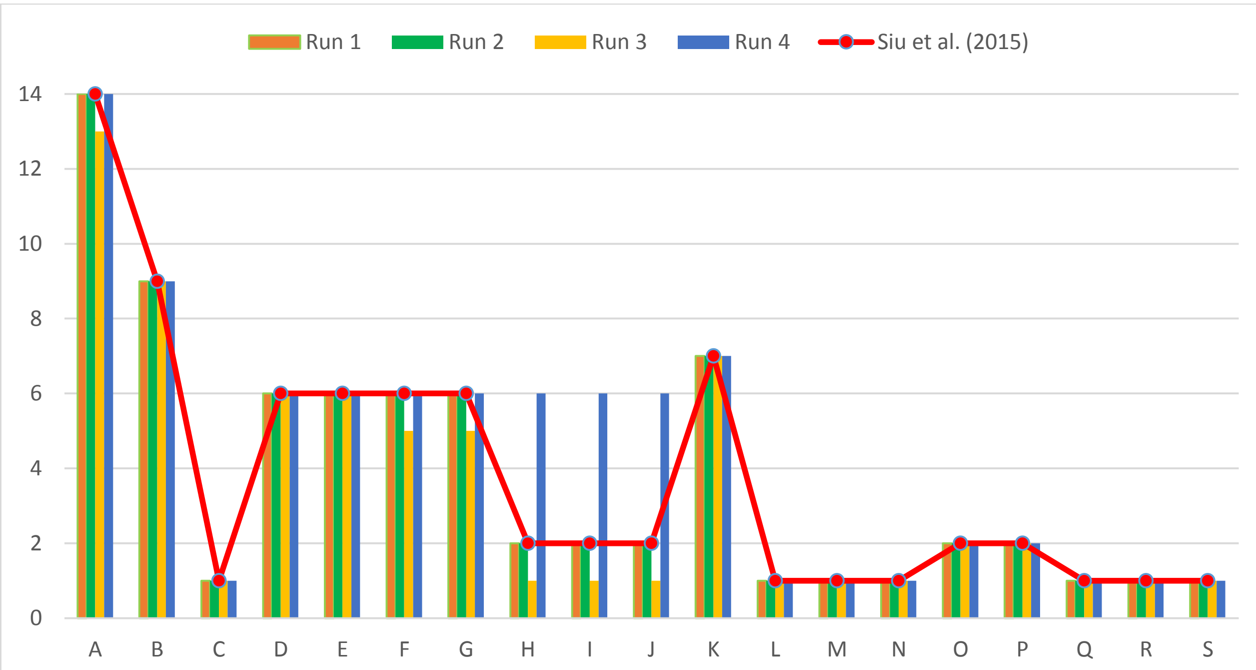


Figure 37: Resource requirements for Siu et al., 2015, STREAM Run 1, STREAM Run 2, and STREAM Run 3, STREAM Run 4

4.3 Case Study 3: Ship Repair

Under this case study real world data was collected from the largest shipyard in the southern hemisphere. It is located at Rio de Janeiro/Brazil. Figure 38 shows an aerial view of the repair shipyard.

Figure 38: Repair shipyard facility located in Brazil

In the shipyard three floating docks and two dry docks. Docks are the most valuable resources of a repair shipyard, they are expensive and limited. Each dock is equipped with its own cranes. The cranes directly support nearly all the work teams as they move around the shipyard. In addition to the docks, cranes and work teams, the shipyard has a series of other production resources distributed throughout the facility. Some of these resources are allocated and grouped in resource centers. The grouping depends on the process utilized to carry out tasks. The metalworking resources are managed by metalworking centers; the tube and valve resources are located close to one another; the plasma cutting machinery is in a specific center, and so forth.

Docks could be occupied by more than one ship at a time. It is a favorable business situation, but it complicates the operations management. A manager is assigned for each dock. In Brazilian repair shipyards, the manager is in charge of all tasks related to her/his dock. Dock managers are highly skilled, and influential. The dock managers are rewarded for efficient operation of their docks. They schedule tasks on their docks using a simple spreadsheet or Microsoft Project software. They compete for finite resources with other dock managers. Such practice results in optimizing operations at an individual dock, while sub-optimizing the overall projects. Such an approach leads to scheduling slippage and cost overruns [Pinha and Ahluwalia, 2013], [Pinha and Ahluwalia, 2014], [Van Dijk, 2002]. In addition, there is lack of communication among stakeholders (dock managers, customers, suppliers, etc.). The system becomes dependent on dock mangers resourcefulness and skill level of the workforce, but system knowledge is not preserved. Therefore, a unified and centralized data for projects is required.

When a task is carried out at a shipyard, the necessary resources are sent to the desired location. The ships are static and the resources are brought to them. The key resources at a repair shipyard are the work teams and their tools. Typical work teams are: welders, mechanics, painters, and quality control personnel. As resources, the work teams are the key for project managers in the shipyard. Resource centers (metalworking, mechanical, tubing, etc.) respond to tasks needs. The bottlenecks occur not when the teams are working in the resource centers, but when they are working on board the ships, this being the factor that ultimately determines how long a ship remains in dock.

This production environment based on projects provides all necessary features for testing STREAM. All ships and offshore platforms, however large or small, undergo scheduled or unscheduled repair and maintenance. The bidding process for ship repair jobs is highly competitive and global in scope [Pinha and Ahluwalia, 2013], [Pinha and Ahluwalia, 2014]. The ship repair industry is also prone to risks due to high level of capital investment in facilities (such as dry docks), equipment (such as heavy duty cranes), and skilled work force [Pinha et al, 2011]. This industry provides highly customized service and deals with unpredictable demand [Dlugokecki et al., 2010]. Some aspects of ship maintenance, such as cleaning and painting have been automated [Sjøbakk et al., 2013], [Navarro et al., 2013], [Navarro et al., 2011]. However, a vast majority of tasks are performed manually. Typical services include: a) Docking, b) Hand scraping, c) High pressure fresh water jet cleaning, d) Painting, e) Tank cleaning, f)

Steel work, g) Repair of ship's structure, h) Repair of propulsion system, i) Piping repair, j) Valve repair, k) Electrical system, l) Undocking, and m) Testing at sea. Production planning and scheduling is difficult due to finite resources, such as docks, cranes, and worker skills and uneven flow of repair orders [Pinha and Ahluwalia, 2013], [Pinha and Ahluwalia, 2014], [Dlugokecki et al., 2010], [Mourtzis, 2005], [Wullink et al., 2004], [Van Dijk, 2002], [Chyrssolouris, 1999], [Chyrssolouris et al., 2004], [De Boer, 1998], [De Boer et al, 1997]. Table 44 lists some of the resources, grouped by work teams, machines, tools, and material handling devices [Pinha et al., 2011].

Table 44: Types of Resources

Work Teams	Machines	Tools	Material Handling
Mechanical	Plasma Cutting	Hydro-jet pumps	Forklift
Structure	Pipe bending	Paint pumps	Trucks
Paint	Welding Machines	Hydraulic pumps	Cranes
Sand-blasting	Tube resources	Sand-blasting pumps	Pulley

4.3.1 Inputs

A project for repairing a complex vessel is described. It is a class leading ROV (remotely operated underwater vehicle) construction support vessel ideally suited to perform subsea operations across a wide range of water depths and environmental conditions. It is used to repair oil offshore platforms. Figure 39 shows the vessel. The project was planned with 111 tasks with 475 multiple skilled resources in the resource pool.

Figure 39: Vessel

Table 45 describes only partially the required tasks, and their precedence relationship data. The complete data is presented in Table AIV.1 in the Appendix IV. Table 46 shows the skills required for repairing this ship. The resource requirements for tasks/modes, and the multi-skilled resources are described in Tables AIV.2 and AIV.3 respectively in the Appendix IV.

	Table 45: Ship Repair Task Description	
Task	Description	Pred.
	Treatment and Painting	
1	install scuppers pipes (top side)	
2	high pressure water (top side)	1
3	treatment (top side)	2
4	painting (top side)	3
5	hull marks + name+ register port (top side)	4
.		
.		
.		
108	supply equipment for lifting 500 kg height min. 06 meters (life boats)	

Table 46: Skills	
Skill	Description
1	welder
2	blow torched
3	brazier
4	assembler
5	mechanic
6	electrician
7	fluid jet
8	painter
9	plumber
10	crane operator
11	machine operator
12	carrier
13	assembler scaffolding
14	carpenter
15	dock aid
16	support
17	safety
18	fireman
19	general services
20	quality control

Figure 40 shows a screen where project managers can associate skills for a given resource in STREAM. Resource “worker 20” can work as welder, assembler and fireman in the shipyard.

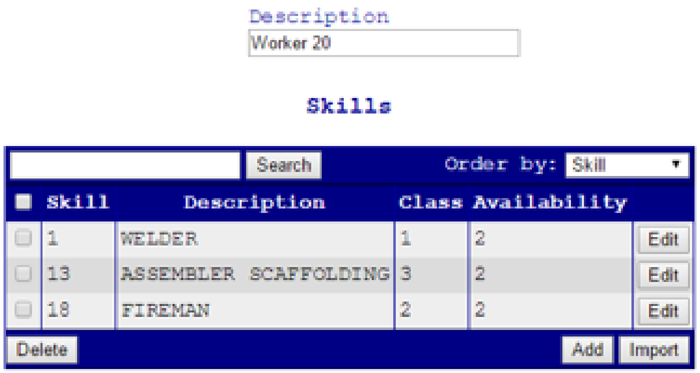


Figure 40: Multiple skills for a given resource in STREAM

4.3.2 Results applying STREAM

For this case study two simulations were run on a unique scenario. The first simulation was performed to assess which tasks are delayed due to lack of resources. The second run provides an improved result in light of outputs obtained for simulation run 1. The waiting times for some tasks in the first run is shown in Table 47. These tasks are waiting for resources which were not available when required. One example in this case study is proposed. Task 61 was eligible to start on 03-Jun-2015, but 36 units of the nonrenewable resource (electrode ok 4600.1 / 8) were not available on hand. Task 61 was started on 30-Jun-2015. This was the date when a new order quantity of the nonrenewable resource was delivered in the shipyard. Simulation run 1 generates the outputs considering the nonrenewable resource issue. What-if the delivery date for the nonrenewable resource could be anticipated for 03-Jun-2015. Simulation 2 is run to assess this, and a comparison between simulations 1 and 2 is made.

Table 47: Lack of resources when required (Run 1)					
Task	**Eligible Date**	**Start Date**	**Waiting Time**	**Qty.**	**Resources**
89	18-May-15	12-Jun-15	25.58	1	Fluid Jet
88	18-May-15	12-Jun-15	25.54	1	Fluid Jet
82	18-May-15	19-May-15	1.13	1	Support
84	19-May-15	22-May-15	3.04	5	Fluid Jet
35	19-May-15	01-Jun-15	13.04	14	Fluid Jet
50	22-May-15	04-Jun-15	13.08	10	Fluid Jet
43	22-May-15	03-Jun-15	11.83	12	Fluid Jet
68	28-May-15	08-Jun-15	10.62	1	Valves repairing
61	**03-Jun-15**	**30-Jun-15**	**26.92**	**36**	**electrode ok 4600.1 / 8**
51	12-Jun-15	24-Jun-15	11.75	7	Painter
21	18-Jun-15	19-Jun-15	0.92	1	Painter

Figure 41 shows the resource utilization bar chart for all required resource for repairing the vessel. X-axis represent resources, y-axis is the resource utilization which can vary from 0% to 100%. The colors for bars in the chart represent skills. E.g., red color bars are resources for painting, yellow are resources for plumbing, and so on. The chart provides a clear indication that painters are the resource most utilized.

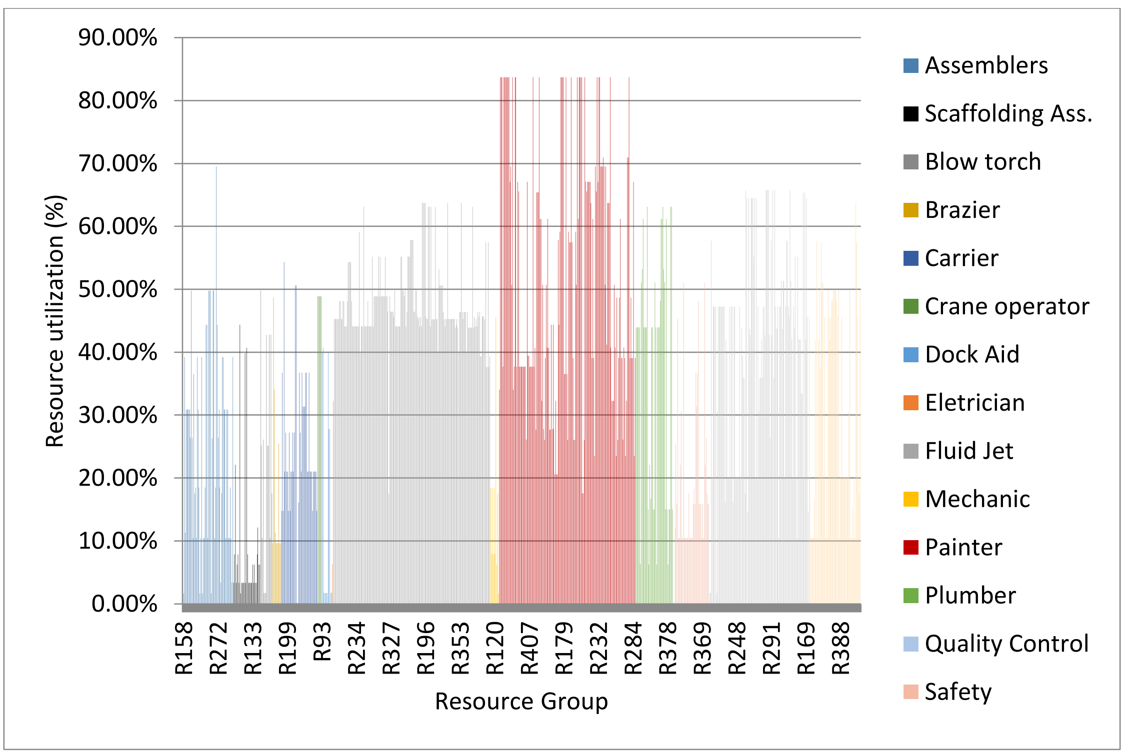


Figure 41: Resource Utilization (Run 1)

Table AIV.4 shows the individual cost and schedule for all the 111 tasks. Table 48 shows the total project cost and project duration.

Table 48: Project Duration and Cost Analysis (Run 1)			
Start date	**Finish date**	**Project Cost (USD)**	**Project Duration (days)**
5/18/2015	7/27/2015	$9,876,833	70

Figure 42 shows the accumulated cost among tasks. X-axis represent the tasks, which are numbered from 1 to 111. Y-axis represents the accumulated cost associated for performing tasks in millions (USD).

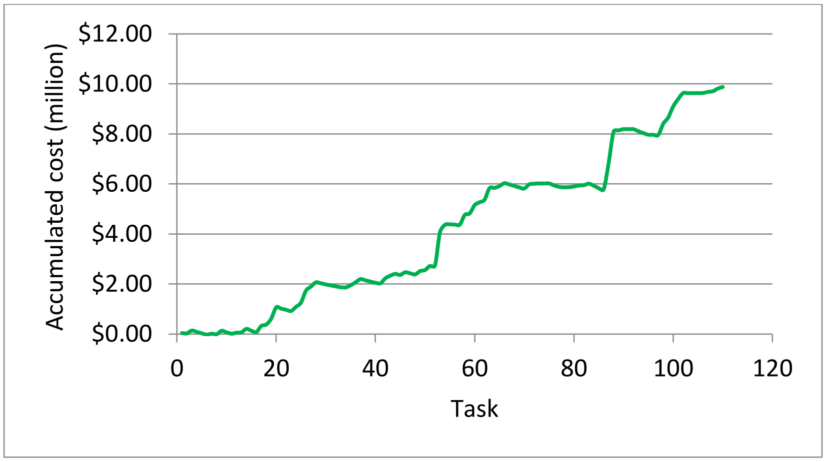


Figure 42: Accumulated Cost with nonrenewable issue in Task 61

Figure 43 shows the deviations against tasks due dates. Task 61 has a positive deviation of 12 days (late) as can be seen in the Figure 43 or by looking at Table AIV.4 in the Appendix IV. Table 47 shows that Task 61 has a waiting time due to a late delivery of the nonrenewable resource (electrode ok 4600.1 / 8). Thus, the deviation shown in the Figure 38 is caused by the nonrenewable resource. What-if the delivery date for the nonrenewable resource could be moved to 03-Jun-2015. How much cost could be saved? Simulation 2 is run to answer this question. Figure 44 shows the Tardiness analysis screen provided by STREAM.

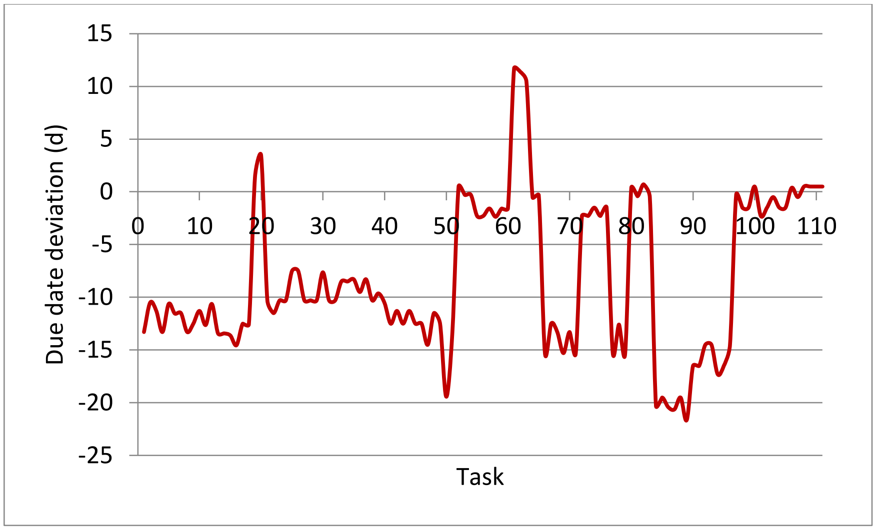
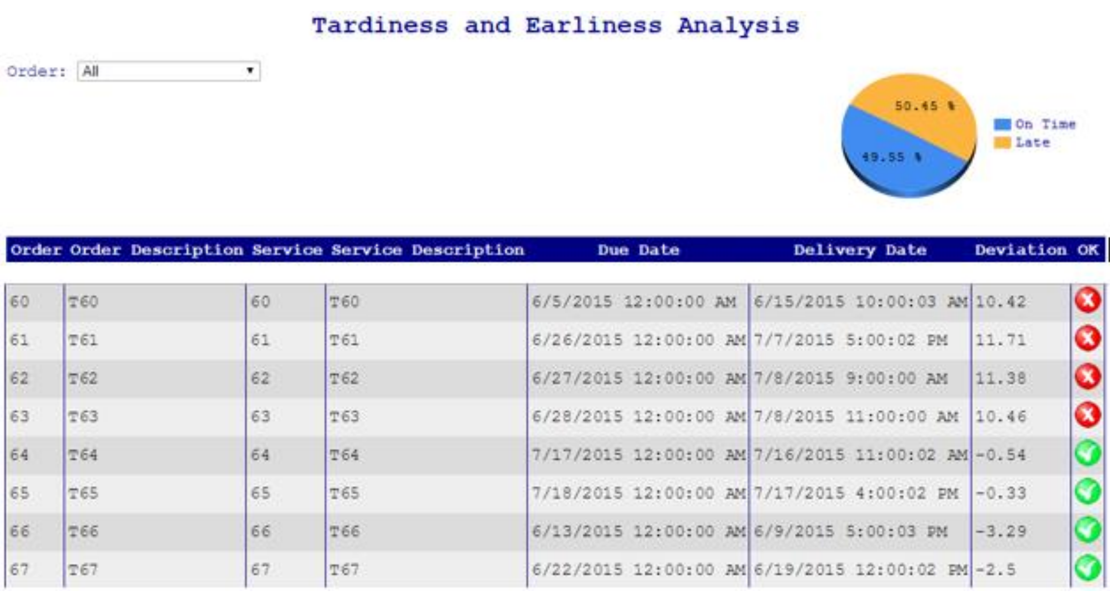


Figure 43: Deviation of Due Dates with nonrenewable resource issue



Order	Order Description	Service	Service Description	Due Date	Delivery Date	Deviation	OK
60	T60	60	T60	6/5/2015 12:00:00 AM	6/15/2015 10:00:03 AM	10.42	
61	T61	61	T61	6/26/2015 12:00:00 AM	7/7/2015 5:00:02 PM	11.71	
62	T62	62	T62	6/27/2015 12:00:00 AM	7/8/2015 9:00:00 AM	11.38	
63	T63	63	T63	6/28/2015 12:00:00 AM	7/8/2015 11:00:00 AM	10.46	
64	T64	64	T64	7/17/2015 12:00:00 AM	7/16/2015 11:00:02 AM	-0.54	
65	T65	65	T65	7/18/2015 12:00:00 AM	7/17/2015 4:00:02 PM	-0.33	
66	T66	66	T66	6/13/2015 12:00:00 AM	6/9/2015 5:00:03 PM	-3.29	
67	T67	67	T67	6/22/2015 12:00:00 AM	6/19/2015 12:00:02 PM	-2.5	

Figure 44: Tardiness Analysis screen in STREAM

Figure 45 shows the accumulated cost among tasks for run 2. A reduction in cost for Task 61 could be obtained due to reductions in penalty cost associated for this task.

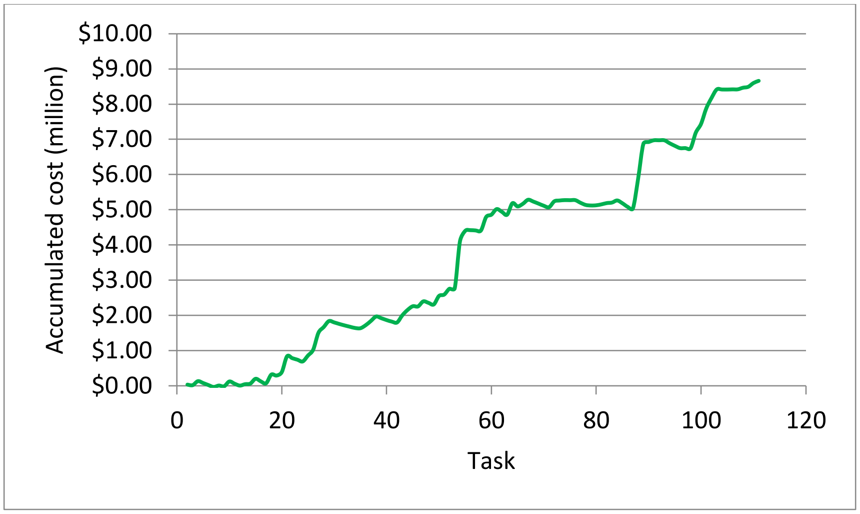


Figure 45: Accumulated cost with earlier deliver date for the nonrenewable resource

Figure 46 shows the deviations against tasks due dates after re-planning the delivery date for the nonrenewable resource (Run 2). Task 61 has no longer a positive deviation of 12 days, instead Task 61 has finished 14 days earlier than its due date. More than $1,000,000 in cost could be saved and 5 days in project reduction was found by re-planning the nonrenewable resource. Table 49 shows the project cost and project duration obtained by STREAM for the shipyard.

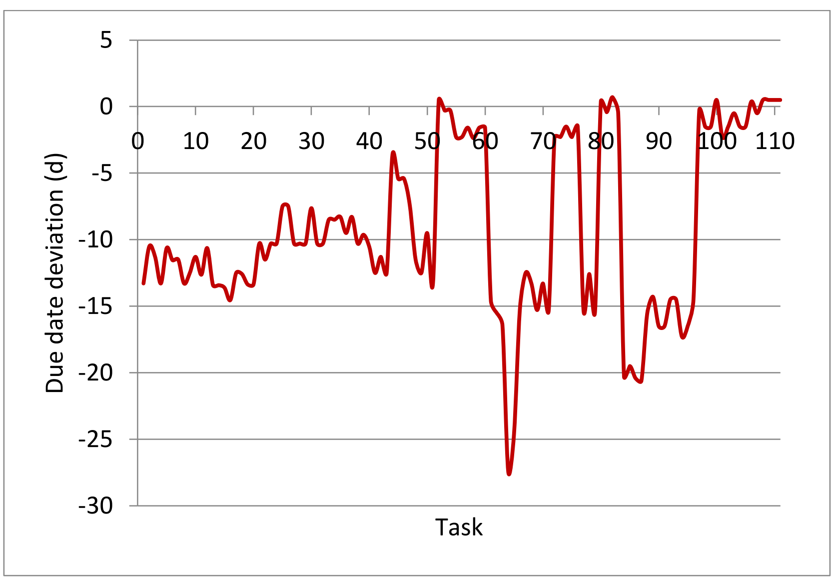


Figure 46: Deviation of Due Dates with Nonrenewable resource issue solved

Table 49: Project Duration and Cost Analysis			
Start date	**Finish date**	**Project Cost (USD)**	**Project Duration (days)**
5/18/2015	7/22/2015	$8,660,104	65

Figure 47 shows the Resource schedule screen in STREAM. This screen provides which tasks should be performed during the project for a given resource. The project manager has the option to select one resource and evaluate which tasks will be execute by it. Resource 1 will execute Tasks 75, 76, 77, 68, 69, and 70 on 6/3/2015, 6/10/2015, 6/16/2015, 6/18/2015, 7/1/2015 and 7/14/2015, respectively, as can be seen in Figure 47.

Resource Schedule

Resources: 1

Search Search by: Task Id

Task Id	Due Date	Duration	Eligible Time	Start Time	End Time
75	5/31/2015 12:00:00 AM	1.5	6/3/2015 12:00:00 PM	6/3/2015 12:00:00 PM	6/9/2015 5:00:02 PM
76	6/6/2015 12:00:00 AM	1.5	6/9/2015 5:00:02 PM	6/10/2015 7:59:59 AM	6/16/2015 12:00:01 PM
77	6/21/2015 12:00:00 AM	0.5	6/16/2015 12:00:01 PM	6/16/2015 1:00:01 PM	6/17/2015 5:00:01 PM
68	7/2/2015 12:00:00 AM	3	6/9/2015 5:00:03 PM	6/18/2015 8:00:00 AM	6/30/2015 5:00:03 PM
69	7/17/2015 12:00:00 AM	3	6/30/2015 5:00:03 PM	7/1/2015 7:59:59 AM	7/13/2015 5:00:03 PM
70	7/20/2015 12:00:00 AM	1	7/13/2015 5:00:03 PM	7/14/2015 7:59:59 AM	7/16/2015 5:00:01 PM

Figure 47: Resource schedule screen in STREAM

4.4 Case Study 4: Supply Chain Management of Motorcycle Assembly Line

Supply chain (SC) is as an added value stream to manufacturing goods. It includes various stages of manufacturing, supply of parts, production, supply of raw materials, intermediate parts, packaging, transportation, warehousing, and other logistics [Chen and Hall, 2007]. The Council of Supply Chain Management Professionals [CSCMP, 2014] define Supply Chain Management (SCM) as all tasks involved in sourcing and procurement, conversion, and all logistics management tasks. It also includes coordination and collaboration along stakeholders. SCM is a complex activity due to constant changes and uncertainties in the global market place [Ivanov and Sokolov, 2012]. Mulani and Lee (2002) state that supply chain managers spend more than half of their working time dealing with changes in the supply chain.

Reliable delivery dates are critical drivers to sustain a healthy supply chain. Partnership along a supply chain, in which suppliers and customers exchange information (purchase orders, invoices, shipping notices, etc), has worked efficiently to reduce supply issues. However, such healthy relationship is not always possible. Factors such as political instability, financial risks, inflation, tax rates and even cultural issues can affect the stability of a SC.

This case study deals with a highly unstable supply chain where suppliers fail to meet delivery schedule and Just-in-time techniques are not feasible. A framework of supply chain network is presented. The network model is described by the Task-on-Node (ToN) diagram.

Significant amount of research has been conducted in the general area of Supply Chain Scheduling (SCS). Bulk of this research has focused on mathematical formulations, based on job shop and flow shop scheduling methods. In order to deal with the complexity of the scheduling problems, a wide variety of techniques were described in the Chapter 1.

Ivanov and Sokolov (2012) and Ivanov et al (2014) proposed an approach based on optimal control theory, specifically, on Optimal Program Control (OPC), combined with optimization methods for scheduling a supply chain. Figure 48 shows a general supply chain structure based on Ivanov et al (2014).

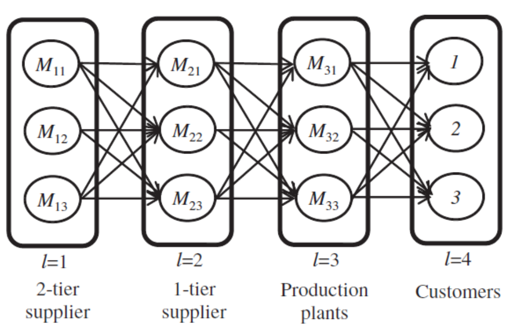


Figure 48: Supply chain structure from Ivanov et al (2014)

The above supply chain structure consists of 4 processing and 3 transportation stages. M_{ij} are the non-identical alternative production plants defined at each stage. Note that stages l=1, l=2, l=3 have 3 non-identical alternative resources each and l=4 is composed of 3 different customers.

Scheduling of tasks in a project and in a supply chain address several of the same problems. This work applied the algorithms described in Chapter 3. The concept of multi-stage presented earlier, such as 1st -tier, nth-tier suppliers, can be applied to supply chain network. However, in many cases, due to precedence among suppliers and customers along a chain, the notion of stages can become cumbersome. In this study nodes are used to describe precedence relationships. Figure 49 shows a task on node diagram for supply chain network. The supply chain network diagram shows tasks arranged in a logical precedence order along with suppliers and customers. The proposed approach makes the following assumptions:

1. The nodes represent tasks and arrows define the precedence relationship along the chain.
2. Each node represents a supplier or a customer, but a given node can either be a supplier or a customer. For instance in Figure 49, node 2 in the first chain is a supplier of node 4, but it is a customer of node 1. A predecessor node is a supplier to the successor node (customer). For modelling purposes, a supplier is any node with outgoing arrow, which delivers services or goods. A customer is any node with incoming arrows, which is connected with a supplier.
3. Release dates for tasks are unknown in advance. During the simulation cycle, tasks become eligible, subject to precedence. For processing tasks, resource requirements need to be satisfied.
4. A mode is an option to perform a given task, e.g., each mode can be functioning as a facility of a given supplier, a different supplier (backup supplier) or simply a list of different resources to deliver services or goods. Modes might have their resources occupied, delaying the supply chain flow.

Each node can have several modes or several alternative executors as found in Ivanov et al (2014). Instead of having explicitly alternative executors being part of the network (supply chain structure), this work proposes they work as attribute of the node. Only one mode is selected at a particular point in time. The customer might have to choose from three different suppliers, however, only one supplier is selected.

In Figure 49, the three rectangular boxes show three simultaneous chains. The number of modes for each node depends upon how many suppliers a customer has. Double circles for Task 1 in chain 1 indicates that some work has already been performed by this supplier. Node ST is the dummy start node for all chains; it is used to guarantee that all eligible suppliers can start producing goods at time 0. Node EN is the dummy end node; it registers completion time of last node of multiple chains. The meanings of nodes and modes can be different for different situations, however main features of the ToN network are preserved.

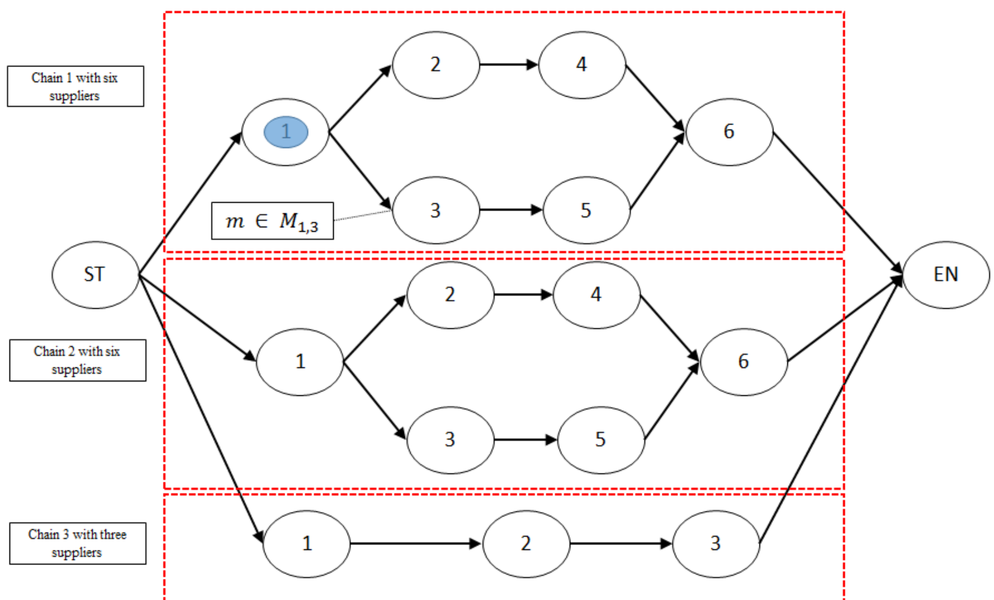


Figure 49: Supply chain schema

4.4.1 Inputs

The data for the motorcycle assembly case study comes from a real world company based in Manaus, Brazil. It is the third largest annual motorcycle production facility in the world, only behind facilities located in China and India. It is also the second largest foreign investment of a Japanese company. More than two millions motorcycles are produced each year. It produces more than 20 different motorcycle types. Once the motorcycles have been assembled, they are shipped to more than 600 dealers in 5 different regions of Brazil.

Several certified suppliers provide different parts (plastic parts, seats, rims, electronic components, tail-pipes, frames, fuel tanks, forks, wheels, covers, engine housings, cylinders, cylinder heads, etc.) to the motorcycle assembly facility and ultimately to assembly lines. The suppliers are under pressure to deliver, at the agreed price, high quality parts, in a timely manner. In order to ensure these requirements, the motorcycle company keeps its own staff at the supplier facilities. However, the issue of late deliveries remains a major cause of concern. There is lack of information on the current status of the supply chain (remaining time to accomplish tasks). Such information if available, could result in rescheduling of tasks. If one supplier fails, it affects the entire supply chain. Shortcomings of the current scenario are: i) lack of information when needed along the chain, ii) lack of coordination in delivering parts to the assembly lines, generating unnecessary inventory, iii) trucks waiting for parts, and iv) dealers not receiving what they were promised.

Figure 50 illustrates the SC network for motorcycle facility. This case study identified 55 tasks, which require 66 resources. Deterministic times were used. The computational time to find a feasible solution was about 2 seconds. The proposed approach schedules the supply chain based on current status of tasks; it enables managers to better handle day-to-day unexpected events.

Tables 50, 51 and 52 show nodes, modes and resources, respectively. In Figure 49, nodes 2 to 14 have 3 modes each (i.e., they have 3 different suppliers to deliver plastic parts, seats, rims, etc). These nodes (suppliers) have no shared resources. For modelling purposes, they are regarded as non-constrained resource. It is assumed the lead times (duration times) are provided by each supplier or calculated by other techniques such as Carvalho et al. (2013).

Nodes 15 to 27 describe shared resources such as box trucks. These are constrained resources. Eight types of trucks support the shipping process to the motorcycle facility. Nodes 28 to 40 describe the six workforce teams that unload trucks from different suppliers. These teams are also constrained resources.

Nodes 41, 42 and 43 correspond to preassemble operations, but 2 preassemble teams are allocated. Node 44 represents the assembly operation. Since assembly lines are setup for a particular motorcycle models, this case study considers only one duration time for this node. Nodes 45 and 46 have a duration time and they are not resource constrained.

Nodes 47 to 51 describe three shared workforce teams working on a wharf to load the trucks with the demanded motorcycles. These workforce teams are constrained. Nodes 52 to 56 represent travel time to deliver the motorcycle to different regions in Brazil. Each region has its own fleet of truck which are not resource constrained. Node ST is a start dummy node; it allows suppliers to start their productions simultaneously at given date. Node EN is an end dummy node. It shows that all requested motorcycles have been delivered to the dealers.

This case study assumes a demand of 10,000 units of a given motorcycle type. Based on its Bill of Materials (BOM), different parts are required along the supply chain. Duration times represent the total time to produce individual components, to ship components, to preassemble, and to do final assemble of the 10,000 units. For modelling purposes, the lot size for each supplier refers to assembly of 10,000 units (e.g. for a demand of 10,000 motorcycles, 20,000 wheels are required from the wheels suppliers, if no inventory exist). The lead time (duration) provided by supplier 1 (node 2 or Task 2) is 80 hours as shown in Table 50. Each supplier can have its internal lot size, however, this case study assumes each supplier provides the lead time to the motorcycle assembly facility. Due to confidentiality reasons, some of the data values were modified. Table 50 shows some node-mode durations. Tables 51 and 52 describe the nodes (tasks), and resources required for the case study, respectively.

Table 50: Nodes, modes and duration times from Figure 49		
N-M Pair	Hours	Description
(2,1)	80	Production Plastic Parts
(2,2)	96	Production Plastic Parts
(3,1)	160	Production Seats
(4,1)	112	Production Rims
(5,1)	320	Production Wheels
(41,1)	160	Preassemble Rims and Wheels
(44,1)	48	Assembling motorcycles

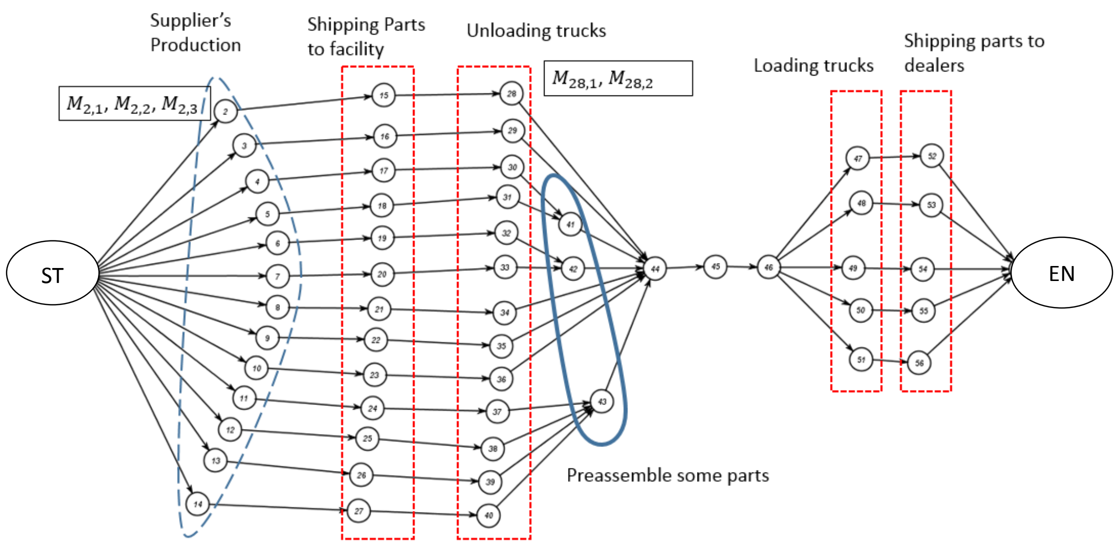


Figure 50: Motorcycle supply chain network

Table 51: Node description			
Node	**Description**	**Node**	**Description**
1	Start dummy node	30	Rims
Production in suppliers		31	Wheels
2	Plastic Parts	32	Electronic components
3	Seats	33	Frames
4	Rims	34	Fuel tanks
5	Wheels	35	Tail pipes
6	Electronic components	36	Forks
7	Frames	37	Covers
8	Fuel tanks	38	Cylinders
9	Tail pipes	39	Cylinders Heads
10	Forks	40	Engine housings
11	Covers	**Preassembling parts**	
12	Cylinders	41	Rims and Wheels
13	Cylinders Heads	42	Electronic Component and Frame
14	Engine housings	43	Covers, Cylinders, Cylinders Head and Engine Housing
Shipping parts to facility		**Assembling, shipping, loading trucks**	
15	Plastic Parts	44	Assembling motorcycles
16	Seats	45	Shipping by box trucks
17	Rims	46	Shipping by ships
18	Wheels	47	Load trucks to North
19	Electronic components	48	Load trucks to Northeast
20	Frames	49	Load trucks to Central West
21	Fuel tanks	50	Load trucks to South
22	Tail pipes	51	Load trucks to Southeast
23	Forks	52	Shipping to North
24	Covers	53	Shipping to Northeast
25	Cylinders	54	Shipping to Central West
26	Cylinders Heads	55	Shipping to South
27	Engine housings	56	Shipping to Southeast
27	Engine housings	57	End dummy node
Unloading trucks into the facility			
28	Plastic Parts		
29	Seats		

Table 52: Resource description			
Resource	**Description**	**Resource**	**Description**
R1	Supplier 1 of Plastic Parts	R34	Supplier 1 of Cylinders Heads
R2	Supplier 2 of Plastic Parts	R35	Supplier 2 of Cylinders Heads
R3	Supplier 3 of Plastic Parts	R36	Supplier 3 of Cylinders Heads
R4	Supplier 1 of Seats	R37	Supplier 1 of Engine housings
R5	Supplier 2 of Seats	R38	Supplier 2 of Engine housings
R6	Supplier 3 of Seats	R39	Supplier 3 of Engine housings
R7	Supplier 1 of Rims	R40	Box Truck 1
R8	Supplier 2 of Rims	R41	Box Truck 2
R9	Supplier 3 of Rims	R42	Box Truck 3
R10	Supplier 1 of Wheels	R43	Box Truck 4
R11	Supplier 2 of Wheels	R44	Box Truck 5
R12	Supplier 3 of Wheels	R45	Box Truck 6
R13	Supplier 1 of Electronic components	R46	Box Truck 7
R14	Supplier 2 of Electronic components	R47	Box Truck 8
R15	Supplier 3 of Electronic components	R48	Unloading trucks team 1
R16	Supplier 1 of Frames	R49	Unloading trucks team 2
R17	Supplier 2 of Frames	R50	Unloading trucks team 3
R18	Supplier 3 of Frames	R51	Unloading trucks team 4
R19	Supplier 1 of Fuel tanks	R52	Unloading trucks team 5
R20	Supplier 2 of Fuel tanks	R53	Unloading trucks team 6
R21	Supplier 3 of Fuel tanks	R54	Preassemble team 1
R22	Supplier 1 of Tail pipes	R55	Preassemble team 2
R23	Supplier 2 of Tail pipes	R56	Assemble team
R24	Supplier 3 of Tail pipes	R57	Trucks with motorcycle
R25	Supplier 1 of Forks	R58	Ships with motorcycle
R26	Supplier 2 of Forks	R59	Load trucks with Motorcycle team 1
R27	Supplier 3 of Forks	R60	Load trucks with Motorcycle team 2
R28	Supplier 1 of Covers	R61	Load trucks with Motorcycle team 3
R29	Supplier 2 of Covers	R62	Trucks region North
R30	Supplier 3 of Covers	R63	Trucks region Northeast
R31	Supplier 1 of Cylinders	R64	Trucks region Central west
R32	Supplier 2 of Cylinders	R65	Trucks region South
R33	Supplier 3 of Cylinders	R66	Trucks region Southeast

Priority rules such as best quality, lowest cost, fastest production, and fastest shipping were used to break a tie when modes required the same resources at the same time. Starting date of 10/9/2014 was selected. All of the motorcycles were delivered on 11/22/2014. Figures 51 shows the Gantt chart provided by STREAM with the scheduling results. Y-axis shows list of tasks which have to be performed while x-axis represents the time. Rectangles shows when tasks are being planned to be performed.

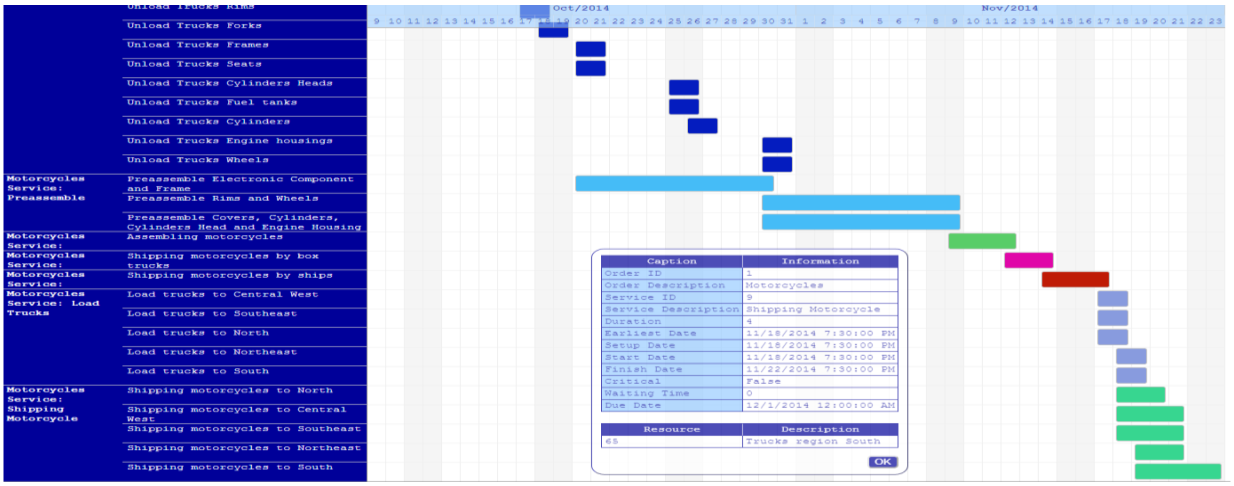


Figure 51: Gantt chart in STREAM for case study 4

The method allows supply chain managers to visualize the current status of the supply chain in a dynamic manner. If the system changes, the supply chain manager can generate the new schedule in near real time. Such approach enhances manager's ability to respond to unexpected events.

Chapter 5: Conclusions and Contributions

5.1 Conclusions

There are major gaps between project scheduling theory and practice. Many authors [Maenhout and Vanhoucke, 2015], [Abrantes and Figueiredo, 2015], [Coughlan et al., 2015], [Xu and Feng, 2014], [Naber and Kolisch, 2014], [Rehm and Thiede, 2012] have indicated a need to expanded project scheduling methods to meet the needs of “real-world” projects. This research provided one such extension. It expanded the RCPSP methods to deal with multiple task modes and multiple skills and calendars for resources. The revised extension of the RCPSP is called CMRCMPSP. The solution method was implemented into a software tool called STREAM. The object of the method remains the same, that is, minimizing total project duration and cost, subject to resource constraints.

Resource allocation problems are typically formulated mathematically, as an optimization problem, with an objective to minimize total project duration, subject to a set of resource constraints. Such methods attempt to provide a single “optimal” solution. There are many benefits to formulating the program mathematically. However, such formulations do not support large scale project scheduling problems and make several assumption in order to keep the problem mathematically tractable. They assume that resources will remain somewhat stable during project lifecycle. In “real world” projects resources and priorities change of a daily basis. The methodology presented in this research is based on a flexible discrete event simulation, where project manager is an integral part of the resource allocation process. Project manager is provided a method to dynamically adapt to changes in resources and priorities.

The proposed approach was applied to four “real-world” case studies: 1) Construction of a single cell box culvert, 2) Oil refinery turnaround, 3) Ship maintenance, and 4) Motorcycle assembly. First two case studies were based on data provided in the literature. The purpose of these case studies was to validate the results obtained by STREAM with those published in the literature. The ship maintenance and motorcycle assembly case studies were based on experience of the author in dealing with complex scheduling issues. These cases demonstrate advanced features (such as, multi-mode tasks, multi-skilled resources, combination of multi-modes and multi-skills, renewable and non-renewable resources) of STREAM that are not addressed in literature.

Regarding the modelling of input data, there is commonly a trade-off between a model's accuracy in capturing the relevant features of the project-planning environment and the resulting model complexity. In this research, different options to perform tasks (multi-modes), multi-skilled workers, and non-renewable resources, which often exist in real-life projects, were considered in the assessed scenarios to validate STREAM. However, it is also often that during the project scheduling and resource allocation decisions, these realistic options are not taken into account, mainly due to the lack of suitable tools for project planning. The resource allocation decision is left purely to the execution level, where the impact on project duration and cost is difficult to be computed. Ultimately the poor resource management results on cost overruns and schedule slippage.

Another issue discussed herein refers to STREAM´s flexibility to represent the decision-maker´s perspective towards the planning problem. For instance, project managers are highly knowledgeable professionals and are in the centre of the decision making process. Techniques which oversimplify the project problem and neglect the importance of project managers often do not support the day-to-day reality in projects. STREAM allows the project manager to intervene in the construction of the project schedule with managerial decisions (i.e., changing specific input data) in order to generate a concurrent new analysis. Through a sensitivity analysis, it is possible to test different plans evaluating 'what if?' scenarios and policies in an iterative and interactive planning process comparing the overall cost and project duration. These managerial decisions refer to actions such as varying material availability (changing release dates and due dates), adjusting capacity levels (altering the maximum number of working hours for specific resources, maintenance of equipment), and authorizing overtime for specific workers.

Besides assisting the project manager in choosing a suitable project schedule, STREAM also enhances the decision-making process by providing more detailed and precise information about the planning problem that is usually not available in the most common used tools for project scheduling. For instance, STREAM automatically assigns the resources capable to execute the tasks among the resource pool and shows which resources are utilized to perform each individual task. To provide this information, task/modes in STREAM can require skills instead of specified resources. Resource skills and their skill levels are also considered in this process. The information relative to the task waiting times contributes in environments where matching multiple resources for tasks is needed. This output highlights the tasks that were delayed due to insufficient number of resources. This information permits a timely assessment of possible shortcomings (e.g., the project manager may need to renegotiate a supply delivery date to avoid resource shortage.).

This research addressed the existing theory-practice gap in the area of project management. The gap with respect to the development of decision support tools for short term resource allocation and management within the project scheduling process was addressed. It aimed to reduce project duration and costs in dynamic environments by empowering project managers to assess different resource allocation flexibilities and to avoid non-renewable resource shortages. According to the results obtained, reductions in cost and project duration were found to be relevant when resource allocation flexibilities are considered.

From an academic perspective, the proposed STREAM approach contributes to literature in the project scheduling research area, as there are few studies that explore resource allocation flexibilities in dynamic environments. This research discusses relevant issues, found in real-life project planning contexts that are not fully considered by other researchers, such as the use of multi-mode tasks, multi-skilled resources, the combination of multi-mode tasks and multi-skilled resources, parallel scheduling of multiple tasks, priority rules for dynamic queue, multiple calendars with interruptions and non-renewable resources.

From the practitioner's standpoint, this research provides a pragmatic view of what can be obtained when project managers are empowered to assess different resource allocation flexibilities within the project scheduling process. Four case studies were used to validate and demonstrate STREAM capabilities in reducing project cost and duration. Several different scenarios, based on a real-life project were considered to validate the proposed STREAM approach, which proved to be useful to assess trade-offs between the use of these resource allocation flexibilities and the total cost and project duration. Moreover, STREAM enhances the decision making process by providing flexibility to represent the decision-maker´s perspective towards the planning problem. It also allows the project manager to intervene in the construction of the project schedule with managerial decisions in order to generate concurrent analysis. Besides assisting the project manager in choosing a suitable project schedule, STREAM also provides more detailed and precise information about the planning problem, usually not available in common used tools for project scheduling, thus permitting a timely assessment of possible shortcomings. The results obtained by STREAM for the case studies are summarized as follows:

Case study 1: A single cell box culvert construction in a drainage project described in the literature. Project details are described in Lu et al. (2008). They identified 33 tasks for this project. STREAM outperformed Particle Swarm Optimization (PSO) for two scenarios described in the literature. It was also able to find the optimal solution in about 1 second using the same data. It is worth noting that these results were obtained without mathematical formulation or attempting to find the optimal. STREAM run other scenarios where human resources have multiple skills, where tasks have multiple modes, and where human resources have multiple skills and tasks have multiple modes. For all the different scenarios, the analysis of cost and project duration is provided. Significant cost and time reductions were found through this method. A comparison between the different scenarios is made with the baseline (optimal solution) and the project duration is reduced from 275 days to 218 days (57 fewer days) and total project cost is reduced from $14,347,972 to $6,460,206, cost reduction of nearly 50%, amounting to $7,887,765.

Case study 2: An oil refinery turnaround project described in the literature. Project details are described in Siu et al. (2015). They identified 107 tasks for this project and 19 different types of resources. Siu et al. (2015) aims to determine the shortest duration project, the earliest finish time for each task and the minimal quantity of resources. They formulated the problem as optimization problem and use traditional mathematical formulation. STREAM was not only able to find a good solution, but it was able to find the same optimal solution found in Siu et al. (2015), a different optimal solution and two other good solutions. Also through this case study is shown how STREAM can support managers in finding unknown parameters such as priorities for tasks which depend upon current needs. Differently of the traditional mathematical formulation where modelers do not have control or can keep on track what happens during the optimization cycle (Input – (black box) – output), this approach provides output such as i) Task waiting times, ii) Lack of resources when needed, and iii) resource utilization. These outputs, showed through this case study, empower managers in reducing project duration, in finding earliest finish time for tasks and in determining the minimal resource requirements. STREAM run 4 simulations for this case study; the first simulation, a different optimal solution than presented in Siu et al. (2015) was found; the second run, the exactly same optimal solution presented in Siu et al. (2015) was obtained; the third simulation, the maximal reduction in number of resources which does not affect the shortest project duration is obtained; the four simulation provides the least number of resource to be added in order to avoid any waiting time for all tasks.

Case study 3: Real world data was collected from the largest shipyard in the southern hemisphere. The shipyard of this case study is located at Rio de Janeiro/Brazil. A project for repairing a complex vessel is described. It is a class leading ROV (remotely operated underwater vehicle) construction support vessel ideally suited to perform subsea operations across a wide range of water depths and environmental conditions. The project was planned with 111 tasks with 475 resources in the resource pool. Steel work, treatment and painting, valves, propeller, tank cleaning, support, and pipes are examples of group of tasks which were scheduled by STREAM. STREAM was able to finish the project in 70 days with a cost of

$9,542,465. A shortage of a nonrenewable resource (electrode ok 4600.1 / 8) delayed task T61 in 26 days. STREAM was able to identify it, and a renegotiation of delivery date was made. The project duration and cost were reduced to 65 days and $9,058,573 respectively. STREAM identified the resource shortage and more than $400,000 in savings could be obtained. In order to demonstrated the STREAM capability to run in multiple-projects and huge large project problems. STREAM runs a multi-project scenario with 100 projects in parallel similar as Skandi Victoria. The computational time to schedule the 100 projects was 13.5 minutes.

Case study 4: This case study deals with a supply chain scheduling of a motorcycle assembly facility located in Manaus, Brazil. The supply chain tasks were formulated as a network of 55 tasks competing for 66 resources. The method was able to provide a set of feasible solutions in about 1 second of computing time. Priority rules such as best quality, lowest cost, fastest production, and fastest shipping were used to break a tie when modes required the same resources at the same time.

The proposed approach is general purpose and applicable to a wide variety of situations. The proposed research used data from four case studies in different context. The correctness of the method is guaranteed because STREAM was able to find optimal solutions when it run in equal conditions to those case studies which attempted to optimize the problem. It is worth noting that STREAM does not have a mathematical formulation. However, this research showed that STREAM has the potential and delivered optimal solutions in about one second. Also a set of feasible solutions with cost and time trade-offs are always provided. While evolutionary methods and strong mathematical formulations start to have prohibitive computational times and not able to solve large problems, STREAM can solve problems with 3,000 tasks (5,000 modes) and 400 multi-skilled resources in 78 seconds, and with 10,434 tasks (15,000 modes) and 400 multi-skilled resources in 810 seconds. The problem with 10,434 tasks is based on a scenario with 100 projects running simultaneously. Based on the results and computational times obtained by STREAM for scheduling more than 10,000 tasks in realistic scenarios, this approach can be applied to real-life day-to-day projects without prohibitive computational times.

5.2 Contributions

This research enhances decision making capability of project managers by providing a flexible method for resource allocation, in order to reduce total project duration and cost. Specific contributions are described below:

The Resource Constrained Project Scheduling Problem (RCPSP) was expanded to include combinatorial number of task modes, multiple projects, resource skills, and calendars. The proposed RCPSP extension, Combinatorial Multi-mode Resource Constrained Multi-project Scheduling Problem (CMRCMPSP), includes capabilities provided by RCPSP, MRCPSP, RCMPSP, and MRCMPS methods. CMRCMPSP allows for real-life constraints and it makes fewer assumptions than the previous methods. It was shown that a given task with two modes instead only providing two options for executing a task, this research provided innumerous sets. The mode formed by a set of subsets of resources provides higher flexibility to resource allocation compared with previous approaches.

A discrete event simulation has been developed for the CMRCMPSP. It focused on manufacturing, constructions, maintenance of equipment and other systems driven by projects with resource constraints. It was implemented in Visual Basic.Net on a Windows platform and called as STREAM. It does not use the traditional queue concept, instead it selects modes from the eligible modes pool based on resource availability, skill level, and current priority. The Parallel Mode Schedule Generation Scheme is proposed. This is an extension of the Parallel Schedule Generation Scheme. It was shown that traditional queue management cannot support the CMRCMPSP. STREAM was tested and validated on two case studies published in the literature. The two case studies in the literature mathematically formulated the resource allocation problem as an optimization problem. STREAM produced the same results (under the same assumption) as those reported in the literature. After validating the software it was applied to two more case studies.

Focus of most resource allocation methods is on the mid (1 to 3 months) to long (3 months to 3 years) term time horizon. Focus of STREAM is on short term planning, that is, on a daily or hourly basis. Short term scheduling is significantly more complex than long term scheduling. Long term scheduling systems assume a stable system. STREAM considers the system to be dynamic. It can adapt to real world situations, e.g. a worker could get sick in the afternoon resulting in a changed resource profile. STREAM generates a series of feasible solutions, along with the impact of each solution on schedule and cost. Focus of STREAM is on providing a series of feasible solutions and let the project manager determine what is good under the current circumstances based on measure of performances.

STREAM is better equipped to meet real world needs because it is able to handle 1) Multiple combinatorial modes for tasks, 2) Multiple skills and capabilities for resources, 3) Different priority rules for different resources, and 4) Multiple calendars with interruptions. The two dominant software tools, Microsoft Project and Primavera do not provide such capabilities.

This research produced 3 peer reviewed conference papers [1-Pinha et al., 2015], [2-Pinha et al., 20115], [Pinha and Ahluwalia, 2013] and 3 papers in peer reviewed journals [Pinha and Ahluwalia, 2014], [Pinha et al., 2015], and [Pinha and Ahluwalia, 2015]. The last two are still under review.

5.3 Future Research

This research could be expanded to include:

1. **A capability to track and report on task status and resource utilization in near real-time:** To facilitate the integration of STREAM into the project day-to-day environment, the procedures, software tools and hardware to use STREAM properly must be identified. Questions to be addressed: i) How to collect the task and resource status in an efficient manner to re-plan a project? ii) What time during the day is the best time for project managers to schedule their projects? and iii) how resources will execute the tasks which STREAM has planned?
2. **Comparison of proposed schedule with the actual and use of Earned Value Management (EVM) to track project progress:** The Earned Value Management theory (EVM) widely used to control project progress could be developed into STREAM. STREAM is able to provide the baseline

schedule while EVM controls the project progress. The ultimately goal is to verify if STREAM reduces the discrepancy between of what was planned and what was actually performed.

3. **Comparison of forward and backward scheduling on project duration and cost**: Project duration and costs comparison between forward scheduling and backward scheduling methods could be made in order to identify whether or not reductions can be found.
4. **Development of industry based heuristics for priority rules:** if STREAM is to be integrated with a number of different project applications which have their own methodologies and rules. The process to facilitate the integration and to identify the rules is subject for further investigation. Standards for the exchange and sharing of the project rules based on the application could be defined.
5. **Implementation of STREAM on multiple platforms such as windows, web, mobile devices**: Further development is needed to use full capability of STREAM on multiple platform such as windows, web and mobile devices.
6. **Application of STREAM to other industries, e.g. energy, healthcare, aerospace, software development:** The results obtained so far encourage an investigation whether STREAM fits other industries such as energy, healthcare, aerospace, software development particularly in situations where resource allocation flexibilities exist..

References

Abrantes, R., Figueiredo, J., (2015), International Journal of Project Management, “Resource management process framework for dynamic NPD portfolios”, Volume 33, Issue 6, August 2015, Pages 1274-1288, ISSN 0263-7863

Alcaraz, J., Maroto, C., Ruiz, R., (2003), Journal of the Operational Research Society, “Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms”, 54, 614–626.

Araúzo, J., Pajares, J., Lopez-Paredes, A., (2010), Simulation Modelling Practice and Theory, “Simulating the dynamic scheduling of project portfolios”, Volume 18, Issue 10, November 2010, Pages 1428-1441, ISSN 1569-190X.

Ballard, G (2000), Ph.D. Thesis, “The Last Planner™ System of Production Control”, School of Civil Engineering, The University of Birmingham.

Baumann, P., Trautmann, N., (2013), In Proceedings of the IEEE international conference on industrial engineering and engineering management, “Optimal scheduling of work-content constrained projects”, 10–13 December. Bangkok, Thailand.

Benichou, J. M. Gauthier, P. Girodet, G. Hentges, G. Ribiere, O. Vincent, (1971), Mathematical Programming, “Experiments in mixed-integer linear programming”, 1971, Volume 1, Issue 1, pp 76-94,

Beşikci,U., Bilge,U., Ulusoy,G. (2015), European Journal of Operational Research, “Multi-mode resource constrained multi-project scheduling and resource portfolio problem”, Volume 240, Issue 1, 1 January 2015, Pages 22-31, ISSN 0377-2217

Bianco, L., Caramia, M., (2013), Flexible Services and Manufacturing Journal, “A new formulation for the project scheduling problem under limited resources”, 25, 6–24.

Bouleimen, K. , Lecocq, H., (2003), European Journal of Operational Research, “A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version”, 149 (2), 268–281.

Browning, T., Yassine, A., (2010), International Journal of Production Economics, “Resource-constrained multi-project scheduling: Priority rule performance revisited”, Volume 126, Issue 2, August.

Brucker, P., Knust, S., (2001), Lecture Notes in Computer Science 2079, “Resource-constrained project scheduling and timetabling”, 277–293.

Brucker, P., Drexl, A., Mohring, R., Neumann, K., Pesch, E., (1999), European Journal of Operational Research, “Resource-constrained project scheduling: notation, classification, models, and methods”, 112:3–41

Calhoun, K., Deckro, R., Moore, J., Chrissis, J., Hove, J., (2002), The international Journal of Management Science, “Planning and re-planning in project and production scheduling,” 30 (3) 155–170.

Chabane, H., (2004), xa.yimg.com/kq/groups/24634788/1393872093/name/Art111.pdf

Charris, E., Carlos D. Paternina–Arboleda, (2013), International Journal of Industrial and Systems Engineering, “Simulation model of the supply chain on a naval shipyard”, Volume 13, Issue - 3, 280-290, 2013-01-01, 10.1504/IJISE.2013.052277

Chen, P., Shahandashti, S., (2009), Automation in Construction, “Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints”, Volume 18, Issue 4, July 2009, Pages 434-443, ISSN 0926-5805.

Cheng, L.Z.; Hall, G.N. (2008), Manufacturing & Service Operations Management. "Maximum Profit Scheduling", vol.10, No. 1, winter 2008, pp.84-107.

Choo, H. (2003), Ph.D. Thesis, "Distributed Planning and Coordination to Support Lean Construction", School of Civil Engineering, The University of California, Berkeley.

Chryssolouris, G., J. Pierce, K. Dicke, (1992), International Journal of Flexible Manufacturing Systems, "A decision-making approach to the operation of flexible manufacturing systems", 4 (3–4) (1992) 309–330.

Chryssolouris, G., J. Pierce, K. Dicke, (1991), Journal of Manufacturing Systems, "An approach for allocating manufacturingresources to production tasks", 10 (5) (1991)368–382.

Chryssolouris, G., K. Dicke, M. Lee, (1992), International Journal of Production Research, "On the resources allocation problem", 30 (12) (1992) 2773–2795.

Chryssolouris, G., M. Lee, K. Dicke, (1991), International Journal of Computer Integrated Manufacturing, "An approach to short interval scheduling for discrete parts manufacturing", 4 (3) (1991) 157–168.

Chryssolouris, G., (2005), Manufacturing Systems: Theory and Practice, 2nd Edition New York, Springer-Verlag, 2005.

Chryssolouris, G., (1999), 10th International Conference on Computer Applications in Shipbuilding, "A Planning and Control Method for Shipyard Processes: A Ship-repair Yard Case Study", ICCAS 99.

Chryssolouris, G., Makris, S., Xanthakis, V., &Mourtzis, D., (2004). Journal of Computer Integrated Manufacturing, "Towards the Internet-based supply chain management for the ship repair industry", International 17(1), 45-57. doi:10.1080/0951192031000080885

Costa, R.S. e Jardim E.G.M. (2010), "Os cinco passos do pensamento enxuto", Rio de Janeiro, 2010. http://www.trilhaprojetos.com.br

Cury, 2001, http://www.das.ufsc.br/~cury/cursos/apostila.pdf

De Boer, R. (1998), Ph. D. thesis, "Resource-Constrained Multi-Project Management - A Hierarchical Decision Support System", University Of Twente, The Netherlands.

De Boer, R., J.M.J. Schutten, W.H.M. Zijm, (1997), CIRP Annals - Manufacturing Technology, "A Decision Support System for Ship Maintenance Capacity Planning", Vol. 46, Issue 1, pp. 391-396, ISSN 0007-8506, http://dx.doi.org/10.1016/S0007-8506(07)60850-6 (http://www.sciencedirect.com /science/article/ pii/S0007850607608506)

De Reyck, B., Herroelen, W., (1999), European Journal of Operational Research, "The multi-mode resource-constrained project scheduling problem with generalized precedence relations," 119 (2), 538–556.

Dlugokecki, V., Fanguy, D., Hepinstall, L., Tilstorm, D., (2010), Journal of Ship Production and Design, "Transforming the Ship Building and Ship Repair Project Environment", vol. 26, num. 4, pp 265-272, doi: 10.1109/IEEM.2011.6117887

Dlugokecki, V., Fanguy, D., Hepinstall, L., Tilstorm, D., (2010), Journal of Ship Production and Design, "Transforming the Ship Building and Ship Repair Project Environment", vol. 26, num. 4, pp 265-272

DoN, (2013), http://www.onr.navy.mil/~/media/Files/Funding-Announcements/BAA/2013/13-020.ashx

Drexl, A., Nissen, R., Patterson, J., Salewski, F., (2000), European Journal of Operational Research, "Progen/px – An instance generator for resource-constrained project scheduling problems with partially renewable resources and further extensions", 125 (1), 59–72.

Coughlan, E., Lübbecke, M., Schulz, J., European Journal of Operational Research, “A branch-price-and-cut algorithm for multi-mode resource leveling”, Volume 245, Issue 1, 16 August 2015, Pages 70-80, ISSN 0377-2217.

Elmaghraby, S., (1977), “Task networks: Project planning and control by network models”, Wiley, New York.

Fazar, W., (1959), The American Statistician, "Program Evaluation and Review Technique", Vol. 13, No. 2, p.10.

Feng, W. [1], Zheng, L., Jingshan, L., (2012). Robotics and Automation (ICRA), “The robustness of scheduling policies in multi-product manufacturing systems with sequence-dependent setup times and finite buffers”, 2012 IEEE International Conference on, pp.5074-5079.

Feng, W.[2], Zheng, L., Jingshan, L. (2012), International Journal of Production Research, “Scheduling policies in multi-product manufacturing systems with sequence-dependent setup times and finite buffers”, 50:24, 7479-7492.

Framinan, J., and R. Ruiz., (2012), International Journal of Production Research, “Guidelines for the Deployment and Implementation of Manufacturing Scheduling Systems”, 50 (7): 1799–1812.

Fundeling, C., (2006), Deutscher Universitat-Verlag, “Resourcen beschränkte Projektplanung bei vorgegebenen Arbeitsvolumina (Resource-constrained project scheduling with work contents).”

Fundeling, C., Trautmann, N., (2010), European Journal of Operational Research, “A priority-rule method for project scheduling with work-content constraints”, 203, 568–574.

Hartmann, S., (2001), Annals of Operations Research, “Project scheduling with multiple modes: A genetic algorithm”, 102, 111–135.

Hartmann, S., Briskorn, D., (2010), European Journal of Operational Research, “A survey of variants and extensions of the resource-constrained project scheduling problem”, Volume 207, Issue 1, 16 November 2010, Pages 1-14, ISSN 0377-2217.

Harvard Review Business, (2003). https://hbr.org/2003/09/why-good-projects-fail-anyway

Hazewinkel, M., (2001), "Graph theory", Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4

Heilmann, R., (2001), OR Spektrum, “Resource-constrained project scheduling: A heuristic for the multi-mode case”, 23, 335–357.

Heilmann, R., (2003), European Journal of Operational Research, “A branch-and-bound procedure for the multi-mode resource constrained project scheduling problem with minimum and maximum time lags”, 144, 348–365.

Jarboui, B., Damak, N., Siarry, P. , Rebai, A., (2008), Applied Mathematics and Computation, “A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems”, 195, 299–308.

Jozefowska, J., Mika, M., Rozycki, R., Waligora, G., Weglarz, J., (2001), Annals of Operations Research “Simulated annealing for multi-mode resource-constrained project scheduling,” 102, 137–155.

Kelton, W.D., Sadowski, R.P., Sadowski, D.A., 2002, Simulation with ARENA.

kendrickoil (2015, http://www.kendrickoil.com/

Khoshnevis, B., (1994). Discrete systems simulation. New York: McGraw-Hill.

Kolisch, R., (1996), European Journal of Operational Research, “Serial and parallel resource-constrained project scheduling methods revisited: theory and computation”, 90, 320–333.

Kolisch, R., Drexl, A., (1997), IIE Transactions, “Local for multi-mode resource-constrained project”, vol. 29, no. 11, pp. 987–999.

Kolisch,R., Meyer,K., Mohr, R., Schwindt, C., Urmann, M., (2003), Zeitschrift für Betriebswirtschaft, Ablaufplanung fur die Leitstrukturoptimierung in der Pharmaforschung (Scheduling of lead optimization in pharmaceutical research projects). 73, 825–848.

Laslo, Z., Goldberg, A., (2008), International Journal of Project Management, “Resource allocation under uncertainty in a multi-project matrix environment: Is organizational conflict inevitable?”, Volume 26, Issue 8, November 2008, Pages 773-788, ISSN 0263-7863

Leadership, (2013), http://ec.europa.eu/enterprise/sectors/maritime/files/shipbuilding/leadership2020-final-report_en.pdf

Law, A. M., and Kelton, W. D. (2000). Simulation modeling and analysis, 3rd Ed., McGraw-Hill, New York.

Lau, S., Lu, M., and Poon, C. (2014), J. Constr. Eng. Manage, “Formalized Approach to Discretize a Continuous Plant in Construction Simulations”, 140(8), 04014032.

Lu, M. Lam, H., Dai, F., (2008), Automation in Construction, Resource-constrained critical path analysis based on discrete event simulation and particle swarm optimization, Volume 17, Issue 6, August 2008, Pages 670-681

Lu, M., Chan, W.H., (2004), Proceedings of Winter Simulation Conference 2004, “Modeling concurrent operational delays in construction tasks with simplified discrete event simulation approach”, Washington D.C., 2004, pp. 1260–1267

Lu, M., (2003), Journal of Construction Engineering and Management, “Simplified Discrete-Event Simulation Approach for Construction Simulation”, 2003 129:5, 537-546

Maenhout, B., Vanhoucke, M., (2015), Journal of Scheduling, “An exact algorithm for an integrated project staffing problem with a homogeneous workforce”, p-1-27, August-2015.

Matloff, N., (2008), http://heather.cs.ucdavis.edu/~matloff/156/PLN/DESimIntro.pdf

MARAD,(2013), http://www.marad.dot.gov/documents/MARAD_Econ_Study_Final_Report_ 2013.pdf

Mello, M., J. O. Strandhagen, (2011), Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment, “Supply chain management in the shipbuilding industry: challenges and perspectives”, 225:261-270, June 24, 2011

MIT Nonlinear (2015). http://web.mit.edu/15.053/www/AMP-Chapter-13.pdf

Moder, J., and Phillips, C., (1964), Project Management with CPM and PERT"

Moghaddam, K., John S. Usher, (2011), Computers & Industrial Engineering, “Preventive maintenance and replacement scheduling for repairable and maintainable systems using dynamic programming”, Volume 60, Issue 4, May 2011, Pages 654-665, ISSN 0360-8352, http://dx.doi.org/10.1016/j.cie.2010.12.021.(http://www.sciencedirect.com/science/article/pii/S036083521100009X)

Mourtzis, D., (2005), International Journal of Computer Integrated Manufacturing, “An integrated system for managing ship repair operations”, 18:8, 721-733

MS Project (2015), http://office.microsoft.com/en-us/project/

Naber, A., Kolisch, R., (2014), European Journal of Operational Research, “MIP models for resource-constrained project scheduling with flexible resource profiles”, , Volume 239, Issue 2, 1 December 2014, Pages 335-348, ISSN 0377-2217.

Navarro, P., Iborra, Andrés; Fernández, Carlos; Sánchez, Pedro; Suardíaz, Juan, (2010), Sensors, “A Sensor System for Detection of Hull Surface Defects”, no. 8: 7067-7081.

Navarro, P., Muro, Juan S., Alcover, Pedro M., Fernández-Isla, Carlos, (2013), Sensors, “Sensors Systems for the Automation of Operations in the Ship Repair Industry”, no. 9: 12345-12374.

NSRP, (2013), http://www.nsrp.org/2-Solicitation_Documents/RA%2012-01_FINAL-v2.pdf

Oracle, (2015), https://www.oracle.com/database/index.html

Ozdamar, L., (1999), IEEE Transactions on Systems, Man, and Cybernetics, Applications and Reviews, “A genetic algorithm approach to a general category project scheduling problem”, 29 (1999) 44–59.

Papakostas, N., Papachatzakis, P., Xanthakis, V., Mourtzis, D., Chryssolouris, G. (2010), Decision Support Systems. “An approach to operational aircraft maintenance planning”, Volume 48, Issue 4, March 2010, Pages 604-612, ISSN 0167-9236, http://dx.doi.org/10.1016/j.dss.2009.11.010.

Peteghem, V., Vanhoucke, M., (2010), European Journal of Operational Research, “A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem”, vol. 201, no. 2, pp. 409–418.

Pinha, D., Ahluwalia, R., Carvalho, A., (2015), Project Management Journal, “Flexible Resource Management and Its Effect on Project Cost and Duration”, (Under review).

Pinha, D., and Ahluwalia, R., (2015), European Journal of Operational Research, “Combinatorial Multi-Mode Resource Constrained Multi-Project Scheduling”, (Under review).

Pinha, D., Ahluwalia, R., Carvalho, A., (2015), 2015 IFAC Symposium on Information Control in Manufacturing, “Parallel Mode Schedule Generation Scheme”, Ottawa, Canada.

Pinha, D., Ahluwalia, R., Carvalho, A., Senna, P., (2015), 2015 IFAC Symposium on Information Control in Manufacturing”, “Supply Chain Scheduling: A Motorcycle Assembly Case Study”, Ottawa, Canada.

Pinha, D., Ahluwalia, R, (2013), Proceedings of the Industrial and System Engineering World Conference, “Decision Support System for Repair Shipyard Industry”, Las Vegas, USA.

Pinha, D, Ahluwalia, R, (2014) “Decision Support System for Production Planning in the Ship Repair Industry”. Industrial and Systems Engineering Review Journal, [S.l.], v. 2, n. 1, p. 52-61, jul. 2014. ISSN 2329-0188

Pinha, D., De Queiroz, M.H., Cury, J. E R, (2011), Proceedings of the IEEE Conference on Automation Science and Engineering (CASE), “Optimal scheduling of a repair shipyard based on Supervisory Control Theory”, 2011, vol., no., pp.39,44, August 24-27, 2011 doi: 10.1109/n CASE.2011.6042515

PMI, (2013), A Guide to the Project Management Body of Knowledge, Fifth Edition, Project Management Institute.

Primavera, (2015), https://www.oracle.com/applications/primavera/index.html

Pritsker, A. A. B., and O’Reilly, J. J. (1999). Simulation with visual SLAM and AweSim, 2nd Ed., Systems Publishing Corp., West Lafayette,IN

Pritsker, A., Watters, L., Wolfe, P., (1969), Manage. Sci., “Multi project scheduling with limited resources: A zero-one programming approach”, 16, 93–108

ProChain, (2015), https://www.prochain.com/

ProTrack, (2015), http://www.or-as.be/protrack

Ranjbar, M., Kianfar, F., (2010), Transaction E: Industrial Engineering, "Resource-constrained project scheduling problem with flexible work profiles: A genetic algorithm approach", 17, 25–35.

Rehm, M., & Thiede, J., (2012), Technische Universität Dresden, "A survey of recent methods for solving project scheduling problems", Fakultät Wirtschaftswissenschaften.

Sabzehparvar, M., Seyed-Hosseini, S., (2008), Journal of Supercomputing, "A mathematical model for the multimode resource-constrained project scheduling problem with mode dependent time lags", 44 (3), 257–273.

Siu, M., Lu, M. and AbouRizk, S., (2014), In Proceedings of the 2014 Winter Simulation Conference (WSC '14), "Bi-level project simulation methodology to integrate superintendent and project manager in decision making: shutdown/turnaround applications". IEEE Press, Piscataway, NJ, USA, 3363-3364.

Sjøbakk, B., Maria Kollberg Thomassen and Erlend Alfnes, (2013), International Workshop of Advanced Manufacturing and Automation, "Automation in the ETO Production Situation: The Case of a Norwegian Supplier of Ship Equipment", ISBN/ISBN2 978-82-321-0377-5/, ISSN/ISSN2 1892-8110.

Speranza, M. G, and C. Vercellis, (1992), European Journal of Operational Research, "Hierarchical models for multi-project planning and scheduling", 64, no. 2 (1993) 312–325.

SQL, (2014), http://www.microsoft.com/en-us/server-cloud/products/sql-server/

Beşikci,U., Bilge,U., Ulusoy,G., (2015), European Journal of Operational Research Multi-mode resource constrained multi-project scheduling and resource portfolio problem, , Volume 240, Issue 1, 1 January 2015, Pages 22-31, ISSN 0377-2217

Van Dijk, R. (2002), http://alexandria.tue.nl/repository/books/587366.pdf

Vanhoucke, M., (2013), "Project Management with Dynamic Scheduling, Baseline Scheduling, Risk Analysis and Project Control" 2nd ed. 2013, XVIII, 318 p. 123 illus.

Varma, V., Uzsoy, R., Pekny, J., Blau, G., (2007), Journal of Heuristics, "Lagrangian heuristics for scheduling new product development projects in the pharmaceutical industry", 13 (5), 403–433.

Wall, M., (1996), Ph.D. dissertation, "A Genetic Algorithm for Resource-Constrained Scheduling", Massachusetts Institute of Technology.

Wullink, G (2005), Ph.D. Thesis, "Resource Loading Under Uncertainty", Beta Research School for Operations Management and Logistics, The University of Twente, The Netherlands.

Wullink, G., Hans, E., Gademann, A., van Harten, A., (2004), International Journal of Production Research, "Scenario based approach for Flexible Resource Loading under Uncertainty", 42 (24), 5079-5098

Xu, J. and Feng. C., (2014), The Scientific World Journal, "Multimode Resource-Constrained Multiple Project Scheduling Problem under Fuzzy Random Environment and Its Application to a Large Scale Hydropower Construction Project," , vol. 2014, Article ID 463692, 20 pages, 2014.

Xue, H., Wei, S., Wang, Y. (2010), Apperceiving Computing and Intelligence Analysis (ICACIA) International Conference, "Resource-constrained multi-project scheduling based on ant colony neural network", pp.179,182.

Yamashita, D., Amaral Armentano Vinícius, and Manuel Laguna, (2007), Journal of Scheduling, "Robust Optimization Models for Project Scheduling with Resource Availability Cost",10.1 (2007): 67. ProQuest. Web. 24 Jan. 2014.

Yang, F., Jingang Liu (2012), European Journal of Operational Research, “Simulation-based transfer function modeling for transient analysis of general queueing systems”, Volume 223, Issue 1, 16 November 2012, Pages 150-166, ISSN 0377-2217.

Zhang, L., Sun, R., (2011), Service Systems and Service Management (ICSSSM), 2011 8th International Conference, "An improvement of resource-constrained multi-project scheduling model based on priority-rule based heuristics", pp.1,5, 25-27 June 2011

Zhou, S., Ying Feng Zheng, Zhang Jiang, Meng Qiang Duan, (2013), Applied Mechanics and Materials, “Research of Scheduling Management in Warship Scheduled Repair Project”, 409-410, 1564, DOI: 10.4028/www.scientific.net/AMM.409-410.1564

Zhu, G., Bard, J., Yu, G., (2006), INFORMS Journal on Computing “A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem”, 377–390.

Appendix I: Server Side Database Tables

Table AI.1: SIMU_RESOURCES			
Field	**Data Type**	**Size**	**Description**
ResourceID	Number	Integer	Resource ID
Description	Text	25	Resource Description
AvailabilityID	Number	Integer	Availability ID
DirectCost	Number	Double	Direct Cost
VariableCost	Number	Double	Variable cost/hour
AdditionalTimeCost	Number	Double	Additional cost/hour
MaintenanceCost	Number	Double	Maintenance cost/hour
EnergyCost	Number	Double	Energy cost/hour
Type	Text	25	Renewable and Nonrenewable

Table AI.2: SIMU_RESOURCES_IDLE_TIMES			
Field	**Data Type**	**Size**	**Description**
ResourceID	Number	Integer	Resource ID
StartTime	Date	10	Start time
EndTime	Date	10	End time

Table AI.3: SIMU_RESOURCES_NON_RENEWABLE			
Field	**Data Type**	**Size**	**Description**
InventoryId	Number	Integer	Resource ID
ResourceId	Number	Integer	Start time
Quantity	Number	Double	End time
InventoryType	Text	15	On hand/On order
DeliveryDate	Date	10	Delivery Date

Table AI.4: SIMU_RESOURCES_ADDITIONAL_TIMES			
Field	**Data Type**	**Size**	**Description**
ResourceID	Number	Integer	Resource ID
StartTime	Date	10	Start time
EndTime	Date	10	End time

Table AI.5: SIMU_RESOURCE_AVAILABILITY			
Field	**Data**	**Size**	**Description**
A_W_SId	Text	25	Concatenated code between AvailabilityId, WeekDay, and
AvailabilityId	Number	Integer	Availability ID
WeekDay	Number	Integer	Week day
StartTime	Date	10	Start Time Interval Partition
EndTime	Date	10	End Time Interval Partition
Type	Text	25	Idle, Regular or Additional

Table AI.6: SIMU_RESOURCES_SKILLS			
Field	**Data Type**	**Size**	**Description**
ResourceID	Number	Integer	Resource ID
SkillID	Number	Integer	Skill ID
Rank	Number	Integer	1-Excelent, 2-Good, 3-Reasonable, 4-

Table AI.7: SIMU_TASKS			
Field	**Data Type**	**Size**	**Description**
TaskID	Number	Integer	Task ID
Description	Text	25	Task description
NetDuration	Number	Double	Remaining time to finish a task
Revenue	Number	Double	Revenue
DueDate	Date	10	Due date
PenaltyCost	Number	Double	Penalty cost to no delivery on time
Bonus	Number	Double	Bonus to delivery on time

Table AI.8: SIMU_TASK_MODES			
Field	**Data Type**	**Size**	**Description**
TaskModeId	Text	25	Task Mode ID
TaskID	Number	Integer	Task ID
ModeID	Number	Integer	Mode ID
Duration	Number	Double	Duration (hours)

Table AI.9: SIMU_TASK_MODES_RESOURCES			
Field	**Data Type**	**Size**	**Description**
TaskModeResourceId	Text	25	Task Mode ID
TaskModeId	Text	25	Task ID
Code	Number	Integer	Skill ID or Resource ID
Consumption	Number	Double	Percentage required of a given resource
Qty	Number	Integer	Resource quantity
Type	Text	25	Skill or Resource

Table AI.10: SIMU_TASK_PRECEDENCE			
Field	**Data**	**Size**	**Description**
Task_TaskPredId	Text	25	Concatenated code between Task and its Predecessor Task ID
TaskID	Number	Integer	Task ID
TaskPredID	Number	Integer	Task Predecessor ID
TimeToRelease	Number	Double	Lag-time

Appendix II: Data Files for Case Study 1

	Table AII.1: Resources Required by Each Task for Mode 1								
	Task	**BBF**	**BE**	**CMC**	**CF**	**CLB**	**DL**	**LB**	**RR**
1	A		4					8	
2	B1		2					4	
3	B2		2	1				4	2
4	B3		1					4	2
5	B4			1		2		6	
6	B5	2						4	
7	B6				1			4	
8	B7			1		3		4	
9	B8							6	
10	B9	2		1				3	
11	B10				1			4	
12	B11			1		3		4	
13	C1		1					3	2
14	C2			1		2		4	
15	C3	2						3	
16	C4				1			4	
17	C5			1		3		4	
18	C6							5	
19	C7	2		1				3	
20	C8				1			4	
21	C9			1		3		3	
22	D1		1					2	1
23	D2			1		2		2	1
24	D3	1						2	
25	D4				1			4	
26	D5			1		3		2	1
27	D6							4	
28	D7	2		1				2	
29	D8				1			4	
30	D9			1		3		2	1
31	E1								
31	E2			1			1	6	
33	F		4					8	4
Resource Limits		**4**	**4**	**3**	**2**	**5**	**1**	**8**	**4**

Table AII.2: Duration and Resources Required by Tasks for Mode 2									
		Resource Requirements							
Task	**Duration (days)**	**BBF**	**BE**	**CMC**	**CF**	**CLB**	**DL**	**LB**	**RR**
A	23		2					5	
B1	12		2					2	
B2	23		2	1				2	1
B3	17		1					2	1
B4	5			1		1		2	
B5	45	1						2	
B6	23				1			2	
B7	6			1		2		2	
B8	45							3	
B9	28	1		1				2	
B10	28				1			2	
B11	7			1		2		2	
C1	17		1					2	1
C2	5			1		1		2	
C3	45	1						2	
C4	23				1			2	
C5	5			1		2		2	
C6	45							3	
C7	28	1		1				2	
C8	28				1			2	
C9	7			1		2		2	
D1	3		1					1	1
D2	2			1		1		1	1
D3	7	1						1	
D4	3				1			2	
D5	2			1		2		1	1
D6	6							2	
D7	9	1		1				1	
D8	5				1			2	
D9	2			1		2		1	1
E2	2			1				3	
F	10		2					4	2
Available resource limit		**4**	**4**	**3**	**2**	**5**	**1**	**8**	**4**

Table AII.3: Resource Costs Example		
Resource	**Fixed Cost ($)**	**Variable Cost (USD/h)**
BBF	1000	13
BE	1000	38
CMC	1000	150
CF	1000	19.67
CLB	1000	19.67
DL	1000	85
LB	1000	25
RR	1000	94

Table AII.4: Due Dates, Penalty Costs and Bonus Example			
Task	**Due Date**	**Penalty Cost (USD/d)**	**Bonus (USD/d)**
A	6/16/2015	12000	6000
B1	6/26/2015	12000	6000
B2	7/12/2015	12000	6000
B3	7/28/2015	12000	6000
B4	8/1/2015	12000	6000
B5	9/21/2015	12000	6000
B6	8/15/2015	12000	6000
B7	9/14/2015	12000	6000
B8	10/21/2015	12000	6000
B9	11/20/2015	12000	6000
B10	11/20/2015	12000	6000
B11	11/26/2015	12000	6000
C1	6/30/2015	12000	6000
C2	7/2/2015	12000	6000
C3	8/15/2015	12000	6000
C4	8/15/2015	12000	6000
C5	8/18/2015	12000	6000
C6	9/20/2015	12000	6000
C7	10/21/2015	12000	6000
C8	10/21/2015	12000	6000
C9	10/25/2015	12000	6000
D1	10/28/2015	12000	6000
D2	12/2/2015	12000	6000
D3	12/5/2015	12000	6000
D4	12/8/2015	12000	6000
D5	1/10/2016	12000	6000
D6	1/10/2016	12000	6000
D7	1/13/2016	12000	6000
D8	1/13/2016	12000	6000
D9	1/25/2016	12000	6000

E1	8/30/2015	12000	6000
E2	9/2/2015	12000	6000
F	2/20/2016	15000	6000

Table AII.5: Detailed Schedule by STREAM for Scenario 3

Task	Start Date	End Date	Due Date	Deviation (d)
A	5/28/2015	6/15/2015	6/16/2015	-0.29
B1	6/16/2015	6/24/2015	6/26/2015	-1.29
B2	6/25/2015	7/13/2015	7/12/2015	1.71
B3	7/14/2015	7/27/2015	7/28/2015	-0.29
B4	7/28/2015	7/30/2015	8/1/2015	-1.29
B5	7/31/2015	9/5/2015	9/21/2015	-15.29
B6	7/31/2015	8/18/2015	8/15/2015	3.71
B7	9/7/2015	9/10/2015	9/14/2015	-3.29
B8	9/11/2015	10/17/2015	10/21/2015	-3.29
B9	10/19/2015	11/10/2015	11/20/2015	-9.29
B10	10/19/2015	11/10/2015	11/20/2015	-9.29
B11	11/11/2015	11/18/2015	11/26/2015	-7.29
C1	6/16/2015	6/29/2015	6/30/2015	-0.29
C2	6/30/2015	7/2/2015	7/2/2015	0.71
C3	7/3/2015	8/8/2015	8/15/2015	-6.29
C4	7/3/2015	7/21/2015	8/15/2015	-24.29
C5	8/10/2015	8/13/2015	8/18/2015	-4.29
C6	8/14/2015	9/19/2015	9/20/2015	-0.29
C7	9/21/2015	10/13/2015	10/21/2015	-7.29
C8	9/21/2015	10/13/2015	10/21/2015	-7.29
C9	10/14/2015	10/21/2015	10/25/2015	-3.29
D1	10/22/2015	10/23/2015	10/28/2015	-4.29
D2	10/26/2015	10/26/2015	12/2/2015	-36.29
D3	10/27/2015	10/31/2015	12/5/2015	-34.29
D4	11/2/2015	11/3/2015	12/8/2015	-34.29
D5	11/4/2015	11/4/2015	1/10/2016	-66.29
D6	11/5/2015	11/9/2015	1/10/2016	-61.29
D7	11/10/2015	11/16/2015	1/13/2016	-57.29
D8	11/10/2015	11/12/2015	1/13/2016	-61.29
D9	11/19/2015	11/19/2015	1/25/2016	-66.29
E1	5/28/2015	8/25/2015	8/30/2015	-4.29
E2	8/26/2015	8/26/2015	9/2/2015	-6.29
F	11/20/2015	11/27/2015	2/20/2016	-84.29

Table AII.6: Detailed Scheduling by STREAM for Scenario 6				
Task	**Start Date**	**End Date**	**Due Date**	**Deviation (d)**
A	5/28/2015	6/15/2015	6/16/2015	-0.29
B1	6/16/2015	6/24/2015	6/26/2015	-1.29
B2	6/25/2015	7/13/2015	7/12/2015	1.71
B3	7/14/2015	7/27/2015	7/28/2015	-0.29
B4	7/28/2015	8/4/2015	8/1/2015	3.71
B5	8/5/2015	9/10/2015	9/21/2015	-10.29
B6	8/5/2015	9/1/2015	8/15/2015	17.71
B7	9/14/2015	9/17/2015	9/14/2015	3.71
B8	9/18/2015	10/24/2015	10/21/2015	3.71
B9	10/26/2015	11/17/2015	11/20/2015	-2.29
B10	10/26/2015	11/26/2015	11/20/2015	6.71
B11	11/30/2015	12/7/2015	11/26/2015	11.71
C1	6/16/2015	6/29/2015	6/30/2015	-0.29
C2	6/30/2015	7/2/2015	7/2/2015	0.71
C3	7/3/2015	8/8/2015	8/15/2015	-6.29
C4	7/6/2015	7/30/2015	8/15/2015	-15.29
C5	8/10/2015	8/17/2015	8/18/2015	-0.29
C6	8/18/2015	10/12/2015	9/20/2015	22.71
C7	10/13/2015	11/4/2015	10/21/2015	14.71
C8	10/13/2015	11/16/2015	10/21/2015	26.71
C9	11/17/2015	11/24/2015	10/25/2015	30.71
D1	11/25/2015	11/26/2015	10/28/2015	29.71
D2	11/30/2015	11/30/2015	12/2/2015	-1.29
D3	12/1/2015	12/5/2015	12/5/2015	0.71
D4	12/7/2015	12/8/2015	12/8/2015	0.71
D5	12/9/2015	12/9/2015	1/10/2016	-31.29
D6	12/10/2015	12/14/2015	1/10/2016	-26.29
D7	12/15/2015	12/21/2015	1/13/2016	-22.29
D8	12/15/2015	12/17/2015	1/13/2016	-26.29
D9	12/22/2015	12/22/2015	1/25/2016	-33.29
E1	5/28/2015	8/25/2015	8/30/2015	-4.29
E2	9/2/2015	9/2/2015	9/2/2015	0.71
F	12/23/2015	12/30/2015	2/20/2016	-51.29

Appendix III: Data Files for Case Study 2

Table AIII.1: Task Description, Duration, Predecessors, and Resource Requirements (Siu et. al, 2015)

T	Description	Duration	Pred	Resources
1	Install hex on external riser at cut line, approximately 30	50	-	A[2]; B[2]
2	Prejob meeting to install new reactor head	1	-	C[1]; D[6]
3	Position crane and install rigging on new head	8	2	C[1]; D[6]
4	Lift new head and swing amine unit	4	3	C[1]; D[6]
5	Remove rigging and boom clear of work area	4	6	C[1]; D[6]
6	Continue swing and lower new reactor head onto shell	5	4	C[1]; D[6]
7	Hoard in decking on lower dipleg bracing back to shell	10	5	E[6]
8	Fit and tack new head to existing reactor shell	20	5	A[4]; B[4]
9	Install landing from stairway to RX Platform 1	10	-	F[3]; G[5]
10	Install braces and structural section at platform	30	5	A[3];
11	Sign off to close regen manway MX-5 (plenum)	1	-	H[1]; I[1];
12	Sign off to close regen manway MX-3 (plenum)	1	11	H[1]; I[1];
13	Sign off to close regen OHL manway MX-4 (top OHL)	1	12	H[1]; I[1];
14	Sign off to close regen OHL manway MX-6 (vertical section)	1	13	H[1]; I[1];
15	Install bulkhead #2 in reactor at lower elevation, also acc	10	7	E[6]
16	Install landing from stairway to RX	10	9	F[3]; G[5]
17	Sign off to close regen OHL manway MX-7 (bottom section)	1	14	H[1]; I[1];
18	Sign off to close regen OHL manway (west of stack valve)	1	17	H[1]; I[1];
19	Install refractory plug—regen manway MX-5 (plenum)	4	18	A[1]; B[1]
20	Install manway cover—regen manway MX-3 (plenum)	3	18	A[2]
21	Install manway cover—regen OHL manway (west of stack valve)	3	20	A[2]
22	Install refractory plug—regen OHL manway MX-4 (Top OHL)	4	19	A[1]; B[1]
23	Weld out new reactor head to existing reactor shell (25%)	20	8	A[1]; B[3]
24	Lower riser into position, fit and tack	20	15	A[4]; B[1]
25	Install landing from stairway to RX Platform 2	10	16	F[3]; G[5]
26	Install refractory plug—regen OHL manway MX-6 (vertical sec	4	22	A[1]; B[1]
27	Install refractory plug—regen OHL manway MX-7 (bottom secti	4	26	A[1]; B[1]
28	Install refractory—regen manway MX-5 (Plenum)	6	27	K[3]
29	Install landing from stairway to RX Platform 3	10	25	F[3]; G[5]
30	Install refractory—regen OHL manway MX-4 (top OHL)	6	28	K[3]
31	Weld connect pressure tap piping from riser to shell, locat	10	23	P[2]; Q[1]
32	Weld out new reactor head to existing reactor shell (50%)	20	23	A[3]; B[3]
33	Weld out new riser duct to existing lower riser section	40	24	A[2]; B[2]
34	Install refractory—regen OHL manway MX-6 (vertical section)	6	30	K[3]
35	Install refractory—regen OHL manway MX-7 (bottom section)	6	34	K[3]
36	LPI weld connection pressure tap piping from riser to shell	4	31	L[1];
37	Refractory cure time—regen OHL manway MX—4, 5, 6, 7	12	35	-
38	Sign off install of pressure tap piping from riser to shell	1	36	H[1];

39	Weld connect pressure tap piping from riser to shell, locat	10	38	P[2];
40	Weld out new reactor head to existing reactor shell (75%)	20	32	A[3];
41	Backgouge reactor weld of new shell to existing shell	10	32	A[4];
42	Install cover plate—regen manway MX-5 (plenum)	2	37	A[2]
43	LPI weld connection pressure tap piping from riser to shell	4	39	L[1];
44	Install cover plate—regen OHL manway MX-4 (top OHL)	2	42	A[2]
45	Install cover plate—regen OHL manway MX-6 (vertical section	2	44	A[2]
46	Sign off install of pressure tap piping from riser to shell	1	43	H[1];
47	Weld connect TI piping from riser to shell, located above l	10	46	P[2];
48	Weld inside of new shell to existing shell	20	41	A[4];
49	Install cover plate—regen OHL manway MX-7 (bottom section)	2	45	A[2]
50	LPI weld connection TI piping from riser to shell, located	4	47	L[1];
51	Final NDE on riser weld	4	33	M[1]
52	Weld out new reactor head to existing reactor shell (100%)	20	40	A[3];
53	Sign off installation of TI piping from riser to shell, loc	1	50	H[1];
54	Install OD riser hex mesh at cut line, approximately 15 squ	25	51	A[2];
55	Phase array weld of new reactor head to existing reactor sh	10	52	N[1]
56	Install refractory in hex on external riser weld location,	40	54	K[6]
57	Ball test primary cyclones and sign off to install riser ma	2	29	H[1];
58	Layout and install refractory anchors on reactor head weldo	20	55	A[2];
59	Buff shell weld for painting	10	55	O[2]
60	Install bridge steel from new stairway to reactor head 30 —	30	-	F[3]
61	Install riser manway and seal weld	8	57	A[2];
62	Paint shell weld for painting	5	59	O[2]
63	Install platform 1, section 0–90 from RX to reg.	30	10	A[3];
64	NDE on riser manway cover	1	61	A[2];B[1]
65	Remove scaffold from ACB	6	64	E[5]
66	Final cleaning of ACB	4	65	A[2]
67	Install refractory on new reactor head to existing shell we	20	58	K[1]
68	Sign off to close reactor MW—MX—4, 5, 6, 7, 8, 9 (ACB)	4	66	H[1];
69	Close reactor MW—MX-4 (ACB)—install refractory plug	6	68	A[1];
70	Close reactor MW—MX-5 (ACB)—install refractory plug	6	69	A[1];
71	Close reactor MW—MX-4 (ACB)—install refractory in	6	69	K[4]
72	Close reactor MW—MX-6 (ACB)—install refractory plug	6	70	A[1];
73	Close reactor MW—MX-5 (ACB)—install refractory in	6	71	K[4]
74	Sign off refractory installation on riser OD 2	2	56	H[1];
75	Close reactor MW—MX-7 (ACB)—install refractory plug	6	72	A[1];
76	Close reactor MW—MX-6 (ACB)—install refractory in	6	73	K[4]
77	Riser—remove all internal scaffolding in riser	5	-	E[4]
78	Riser—weld on riser manway	12	77	A[1];
79	Close reactor MW—MX-8 (ACB)—install refractory plug	6	75	A[1];
80	Close reactor MW—MX-7 (ACB)—install refractory in	6	76	K[4]
81	Close reactor MW—MX-9 (ACB)—install refractory plug	6	79	A[1];
82	Close reactor MW—MX-8 (ACB)—install refractory in	6	80	K[4]

83	Riser—NDE on riser manway weld	1	78	A[1];
84	Close reactor MW—MX-9 (ACB)—install refractory in	6	82	K[4]
85	Close reactor MW—MW—MX-4, 5, 6, 7, 8, 9 (ACB)—cure	12	84	-
86	Install platform 2, section 0–90 from RX to reg.	30	63	A[3];
87	Close reactor MW—MX-4 (ACB)—close manway cover plate	3	85	A[2]
88	Close reactor MW—MX-5 (ACB)—close manway cover plate	3	87	A[2]
89	Close reactor MW—MX-6 (ACB)—close manway cover plate	3	88	A[2]
90	Close reactor MW—MX-7 (ACB)—close manway cover plate	3	89	A[2]
91	Close reactor MW—MX-8 (ACB)—close manway cover plate	3	90	A[2]
92	Close reactor MW—MX-9 (ACB)—close manway cover plate	3	91	A[2]
93	Sign off to close reactor MW—MX-1 (shell)	1	-	H[1];
94	Sign off to close reactor MW—big MW	1	-	H[1];
95	Sign off to close reactor MW—stripper cone	1	-	H[1];
96	Close reactor MW—MX-1 (shell)	5	93	A[2]
97	Close reactor MW—MX-big MW	6	94	A[4]
98	Install refractory plug—stripper cone manway	4	95	A[1];
99	Install refractory—stripper cone manway	6	98	K[3]
10	Install cover plate—stripper cone manway	2	99	A[2]
10	Sign off to close reactor MW—MX-10 (plenum)	1	-	H[1];
10	Sign off to close reactor MW—MX-11 (plenum)	1	-	H[1];
10	Close reactor MW—MX-10 (Plenum) —close manway cover	3	101	A[2]
10	Close reactor MW—MX-11 (Plenum) —install refractory plu0g	6	102	A[1];
10	Close reactor MW—MX-11 (Plenum) —install refractory in	5	104	K[4]
10	Close reactor MW—MX-11 (Plenum)—close manway cover	3	105	A[2]
10	Install platform 3, section 0–90 from RX to reg.	30	86	A[3];

Table AIII.2: Analytical solution – Task finish time (Siu et. al, 2015)		
Task	**Description**	**Finish Time**
1	Install hex on external riser at cut line, approximately 30	50
2	Prejob meeting to install new reactor head	1
3	Position crane and install rigging on new head	9
4	Lift new head and swing amine unit	13
5	Remove rigging and boom clear of work area	22
6	Continue swing and lower new reactor head onto shell	18
7	Hoard in decking on lower dipleg bracing back to shell	32
8	Fit and tack new head to existing reactor shell	42
9	Install landing from stairway to RX Platform 1	10
10	Install braces and structural section at platform	52
11	Sign off to close regen manway MX-5 (plenum)	1
12	Sign off to close regen manway MX-3 (plenum)	2
13	Sign off to close regen OHL manway MX-4 (top OHL)	3
14	Sign off to close regen OHL manway MX-6 (vertical section)	4
15	Install bulkhead #2 in reactor at lower elevation, also acc	42
16	Install landing from stairway to RX	20
17	Sign off to close regen OHL manway MX-7 (bottom section)	5
18	Sign off to close regen OHL manway (west of stack valve)	6
19	Install refractory plug—regen manway MX-5 (plenum)	10
20	Install manway cover—regen manway MX-3 (plenum)	9
21	Install manway cover—regen OHL manway (west of stack valve)	12
22	Install refractory plug—regen OHL manway MX-4 (Top OHL)	14
23	Weld out new reactor head to existing reactor shell (25%)	62
24	Lower riser into position, fit and tack	62
25	Install landing from stairway to RX Platform 2	30
26	Install refractory plug—regen OHL manway MX-6 (vertical sec	18
27	Install refractory plug—regen OHL manway MX-7 (bottom secti	22
28	Install refractory—regen manway MX-5 (Plenum)	28
29	Install landing from stairway to RX Platform 3	40
30	Install refractory—regen OHL manway MX-4 (top OHL)	34
31	Weld connect pressure tap piping from riser to shell, locat	72
32	Weld out new reactor head to existing reactor shell (50%)	82
33	Weld out new riser duct to existing lower riser section	102
34	Install refractory—regen OHL manway MX-6 (vertical section)	40
35	Install refractory—regen OHL manway MX-7 (bottom section)	46
36	LPI weld connection pressure tap piping from riser to shell	76
37	Refractory cure time—regen OHL manway MX—4, 5, 6, 7	58
38	Sign off install of pressure tap piping from riser to shell	77
39	Weld connect pressure tap piping from riser to shell, locat	87
40	Weld out new reactor head to existing reactor shell (75%)	102
41	Backgouge reactor weld of new shell to existing shell	92

42	Install cover plate—regen manway MX-5 (plenum)	60
43	LPI weld connection pressure tap piping from riser to shell	91
44	Install cover plate—regen OHL manway MX-4 (top OHL)	62
45	Install cover plate—regen OHL manway MX-6 (vertical section	64
46	Sign off install of pressure tap piping from riser to shell	92
47	Weld connect TI piping from riser to shell, located above l	102
48	Weld inside of new shell to existing shell	112
49	Install cover plate—regen OHL manway MX-7 (bottom section)	66
50	LPI weld connection TI piping from riser to shell, located	106
51	Final NDE on riser weld	106
52	Weld out new reactor head to existing reactor shell (100%)	122
53	Sign off installation of TI piping from riser to shell, loc	107
54	Install OD riser hex mesh at cut line, approximately 15 squ	131
55	Phase array weld of new reactor head to existing reactor sh	132
56	Install refractory in hex on external riser weld location,	171
57	Ball test primary cyclones and sign off to install riser ma	42
58	Layout and install refractory anchors on reactor head weldo	152
59	Buff shell weld for painting	142
60	Install bridge steel from new stairway to reactor head 30 —	30
61	Install riser manway and seal weld	50
62	Paint shell weld for painting	147
63	Install platform 1, section 0–90 from RX to reg.	82
64	NDE on riser manway cover	51
65	Remove scaffold from ACB	57
66	Final cleaning of ACB	61
67	Install refractory on new reactor head to existing shell we	172
68	Sign off to close reactor MW—MX—4, 5, 6, 7, 8, 9 (ACB)	65
69	Close reactor MW—MX-4 (ACB)—install refractory plug	71
70	Close reactor MW—MX-5 (ACB)—install refractory plug	77
71	Close reactor MW—MX-4 (ACB)—install refractory in manway ne	77
72	Close reactor MW—MX-6 (ACB)—install refractory plug	83
73	Close reactor MW—MX-5 (ACB)—install refractory in manway ne	83
74	Sign off refractory installation on riser OD 2	173
75	Close reactor MW—MX-7 (ACB)—install refractory plug	89
76	Close reactor MW—MX-6 (ACB)—install refractory in manway ne	89
77	Riser—remove all internal scaffolding in riser	5
78	Riser—weld on riser manway	17
79	Close reactor MW—MX-8 (ACB)—install refractory plug	95
80	Close reactor MW—MX-7 (ACB)—install refractory in manway ne	95
81	Close reactor MW—MX-9 (ACB)—install refractory plug	101
82	Close reactor MW—MX-8 (ACB)—install refractory in manway ne	101
83	Riser—NDE on riser manway weld	18
84	Close reactor MW—MX-9 (ACB)—install refractory in manway ne	107
85	Close reactor MW—MW—MX-4, 5, 6, 7, 8, 9 (ACB)—cure time	119

86	Install platform 2, section 0–90 from RX to reg.	112
87	Close reactor MW—MX-4 (ACB)—close manway cover plate	122
88	Close reactor MW—MX-5 (ACB)—close manway cover plate	125
89	Close reactor MW—MX-6 (ACB)—close manway cover plate	128
90	Close reactor MW—MX-7 (ACB)—close manway cover plate	131
91	Close reactor MW—MX-8 (ACB)—close manway cover plate	134
92	Close reactor MW—MX-9 (ACB)—close manway cover plate	137
93	Sign off to close reactor MW—MX-1 (shell)	5
94	Sign off to close reactor MW—big MW	4
95	Sign off to close reactor MW—stripper cone	1
96	Close reactor MW—MX-1 (shell)	10
97	Close reactor MW—MX-big MW	10
98	Install refractory plug—stripper cone manway	5
99	Install refractory—stripper cone manway	11
100	Install cover plate—stripper cone manway	13
101	Sign off to close reactor MW—MX-10 (plenum)	3
102	Sign off to close reactor MW—MX-11 (plenum)	2
103	Close reactor MW—MX-10 (Plenum) —close manway cover plate	6
104	Close reactor MW—MX-11 (Plenum) —install refractory plu0g	8
105	Close reactor MW—MX-11 (Plenum) —install refractory in man	13
106	Close reactor MW—MX-11 (Plenum)—close manway cover plate	16
107	Install platform 3, section 0–90 from RX to reg.	142

Table AIII.3: Solution results

Task	Run 1		Run 2		Run 3		Run 4	
	Time (h)	Diff.	Time	Diff.	Time	Diff.	Time	Diff.
1	50	0	50	0	50	0	50	0
2	1	0	1	0	1	0	1	0
3	9	0	9	0	9	0	9	0
4	13	0	13	0	13	0	13	0
5	22	0	22	0	22	0	22	0
6	18	0	18	0	18	0	18	0
7	32	0	32	0	32	0	32	0
8	42	0	42	0	42	0	42	0
9	10	0	10	0	10	0	10	0
10	52	0	52	0	70	18	52	0
11	1	0	1	0	1	0	1	0
12	2	0	2	0	4	2	2	0
13	3	0	3	0	5	2	3	0
14	4	0	4	0	7	3	4	0
15	42	0	42	0	42	0	42	0
16	20	0	20	0	20	0	20	0
17	5	0	5	0	9	4	5	0

18	6	0	6	0	10	4	6	0
19	10	0	10	0	14	4	10	0
20	9	0	9	0	13	4	9	0
21	12	0	12	0	17	5	12	0
22	14	0	14	0	18	4	14	0
23	62	0	62	0	62	0	62	0
24	62	0	62	0	62	0	62	0
25	30	0	30	0	30	0	30	0
26	18	0	18	0	22	4	18	0
27	22	0	22	0	26	4	22	0
28	28	0	28	0	32	4	28	0
29	40	0	40	0	40	0	40	0
30	34	0	34	0	38	4	34	0
31	72	0	72	0	72	0	72	0
32	82	0	82	0	82	0	82	0
33	102	0	102	0	102	0	102	0
34	40	0	40	0	44	4	40	0
35	46	0	46	0	50	4	46	0
36	76	0	76	0	76	0	76	0
37	58	0	58	0	62	4	58	0
38	77	0	77	0	77	0	77	0
39	87	0	87	0	87	0	87	0
40	102	0	102	0	102	0	102	0
41	92	0	92	0	92	0	92	0
42	60	0	60	0	64	4	60	0
43	91	0	91	0	91	0	91	0
44	62	0	62	0	66	4	62	0
45	64	0	64	0	68	4	64	0
46	92	0	92	0	92	0	92	0
47	102	0	102	0	102	0	102	0
48	112	0	112	0	112	0	112	0
49	66	0	66	0	70	4	66	0
50	106	0	106	0	106	0	106	0
51	106	0	106	0	106	0	106	0
52	122	0	122	0	122	0	122	0
53	107	0	107	0	107	0	107	0
54	131	0	131	0	131	0	131	0
55	132	0	132	0	132	0	132	0
56	171	0	171	0	171	0	171	0
57	42	0	42	0	42	0	42	0
58	152	0	152	0	152	0	152	0
59	142	0	142	0	142	0	142	0
60	30	0	30	0	70	40	30	0
61	50	0	50	0	58	8	50	0

62	147	0	147	0	147	0	147	0
63	82	0	82	0	100	18	82	0
64	51	0	51	0	59	8	51	0
65	57	0	57	0	65	8	57	0
66	61	0	61	0	69	8	61	0
67	172	0	172	0	172	0	172	0
68	65	0	65	0	73	8	65	0
69	71	0	71	0	79	8	71	0
70	77	0	77	0	85	8	77	0
71	77	0	77	0	85	8	77	0
72	83	0	83	0	91	8	83	0
73	83	0	83	0	91	8	83	0
74	173	0	173	0	173	0	173	0
75	89	0	89	0	97	8	89	0
76	89	0	89	0	97	8	89	0
77	5	0	5	0	5	0	5	0
78	17	0	17	0	17	0	17	0
79	95	0	95	0	103	8	95	0
80	95	0	95	0	103	8	95	0
81	101	0	101	0	109	8	101	0
82	101	0	101	0	109	8	101	0
83	18	0	18	0	18	0	18	0
84	107	0	107	0	115	8	107	0
85	119	0	119	0	127	8	119	0
86	112	0	112	0	130	18	112	0
87	122	0	122	0	130	8	122	0
88	125	0	125	0	133	8	125	0
89	128	0	128	0	136	8	128	0
90	131	0	131	0	139	8	131	0
91	134	0	134	0	142	8	134	0
92	137	0	137	0	145	8	137	0
93	3	-2	5	0	11	6	1	-4
94	4	0	4	0	8	4	1	-3
95	1	0	1	0	2	1	1	0
96	8	-2	10	0	16	6	6	-4
97	10	0	10	0	14	4	7	-3
98	5	0	5	0	6	1	5	0
99	11	0	11	0	12	1	11	0
100	13	0	13	0	15	2	13	0
101	5	2	3	0	6	3	1	-2
102	2	0	2	0	3	1	1	-1
103	8	2	6	0	9	3	4	-2
104	8	0	8	0	9	1	7	-1
105	13	0	13	0	14	1	12	-1

106	16	0	16	0	17	1	15	-1
107	142	0	142	0	160	18	142	0
TPD	**173**		**173**		**173**		**173**	

Table AIII.4: Resource Utilization (STREAM Run 2)

	Resource	Idle time %	Working time %
1	A1	18.50	81.50
2	A2	18.50	81.50
3	A3	27.17	72.83
4	A4	26.01	73.99
5	A5	26.59	73.41
6	A6	25.43	74.57
7	A7	27.17	72.83
8	A8	39.31	60.69
9	A9	55.49	44.51
10	A10	55.49	44.51
11	A11	60.69	39.31
12	A12	68.21	31.79
13	A13	69.36	30.64
14	A14	93.06	6.94
21	B1	18.50	81.50
22	B2	24.86	75.14
23	B3	27.75	72.25
24	B4	38.15	61.85
25	B5	35.26	64.74
26	B6	39.31	60.69
27	B7	63.01	36.99
28	B8	65.32	34.68
29	B9	82.66	17.34
31	C1	87.28	12.72
33	D1	87.28	12.72
34	D2	87.28	12.72
35	D3	87.28	12.72
36	D4	87.28	12.72
37	D5	87.28	12.72
38	D6	87.28	12.72
40	E1	82.08	17.92
41	E2	82.08	17.92
42	E3	82.08	17.92
43	E4	82.08	17.92
44	E5	84.97	15.03
45	E6	88.44	11.56
50	F1	76.88	23.12
51	F2	76.88	23.12

52	F3	76.88	23.12
53	F4	82.66	17.34
54	F5	82.66	17.34
55	F6	82.66	17.34
60	G1	24.86	75.14
61	G2	76.88	23.12
62	G3	76.88	23.12
63	G4	76.88	23.12
64	G5	76.88	23.12
65	G6	82.66	17.34
66	H1	90.75	9.25
67	H2	96.53	3.47
71	I1	90.17	9.83
72	I2	96.53	3.47
73	J1	93.64	6.36
74	J2	96.53	3.47
75	K1	38.73	61.27
76	K2	38.73	61.27
77	K3	38.73	61.27
78	K4	53.18	46.82
79	K5	73.99	26.01
80	K6	73.99	26.01
81	K7	85.55	14.45
90	L1	92.49	7.51
91	M1	97.69	2.31
92	N1	94.22	5.78
93	O1	91.33	8.67
94	O2	91.33	8.67
97	P1	82.66	17.34
98	P2	82.66	17.34
102	Q1	82.66	17.34
106	R1	91.33	8.67
108	S1	97.11	2.89

Appendix IV: Data Files for Case Study 3

	Table AIV.1: Ship Repair Task Description	
Task	Description	Pred.
	Treatment and Painting	
1	install scuppers pipes (top side)	
2	high pressure water (top side)	1
3	treatment (top side)	2
4	painting (top side)	3
5	hull marks+name+ register port (top side)	4
6	remove scuppers pipes (top side)	5
7	scraping (vertical bottom)	
8	high pressure water (vertical bottom)	7
9	treatment (vertical bottom)	8
10	painting (vertical bottom)	9
11	hull marks (vertical bottom)	10
12	protection speed - log/ echo sounder (flat bottom)	
13	high pressure water (flat bottom)	12
14	treatment (flat bottom)	13
15	painting (flat bottom)	14
16	hull marks (flat bottom)	15
17	mount stages (moon pool)	
18	high pressure water (moon pool)	17
19	treatment (moon pool)	18
20	painting (moon pool)	19
21	change zinc anodes (moon pool)	18
22	mount stages (sea chests and box coolers)	
23	open grids (sea chests and box coolers)	
24	install temporary lights 24 volts (sea chests and box coolers)	
25	clean (sea chests and box coolers)	
26	high pressure water (sea chests and box coolers)	
27	treatment and paint (sea chests and box coolers)	26
28	change zinc anodes (sea chests and box coolers)	27
29	paint (sea chests and box coolers)	27
30	remove temporary lights 24 volts (sea chests and box coolers)	29
31	close grids (sea chests and box coolers)	29
32	dismount stages (sea chests and box coolers)	29
33	mount stages (tunnel thruster / azimuth retractable)	
34	open grids (tunnel thruster / azimuth retractable)	
35	high pressure water (tunnel thruster / azimuth retractable)	33,34
36	treatment (tunnel thruster / azimuth retractable)	35

37	paint (tunnel thruster / azimuth retractable)	36
38	change zinc anodes (tunnel thruster / azimuth retractable)	36
39	close grids (tunnel thruster / azimuth retractable)	37,38
40	dismount stages (tunnel thruster / azimuth retractable)	39
41	mount stages (moon pool internal and external parts)	
42	scraping (moon pool internal and external parts)	41
43	hight pressure water (moon pool internal and external parts)	41
44	treat (moon pool internal and external parts)	43
45	paint (moon pool internal and external parts)	44
46	dismount stages (moon pool internal and external parts)	45
47	change zinc anodes (moon pool internal and external parts)	44
48	mount stages (hipap trucking)	
49	scraping (hipap trucking)	48
50	high pressure water (hipap trucking)	49
51	treat (hipap trucking)	50
52	paint (hipap trucking)	51
53	dismount stages (hipap trucking)	52
	Steel work	
54	remove (sacrifice anodes)	
55	mount / weld (sacrifice anodes)	54
56	mount stages (after body repair)	
57	mark (after body repair)	56
58	cut (after body repair)	57
59	prefabricate (after body repair)	58
60	mounting (after body repair)	59
61	welding (after body repair)	60
62	survey inspection (after body repair)	61
63	ultrasonic and vacuum test (after body repair)	62
64	treatment and paint (after body repair)	63
65	remove stages (after body repair)	64
	Valves	
66	remove from board (sea valves and overboard valves)	
67	transport to workshop (sea valves and overboard valves)	66
68	open valves on workshop (sea valves and overboard valves)	66
69	repair (sea valves and overboard valves)	68
70	test/ inspection (sea valves and overboard valves)	69
71	transport to board (sea valves and overboard valves)	70
72	mounting on board (sea valves and overboard valves)	71
73	remove from board (hipap valves and gate valves)	
74	transport to workshop (hipap valves and gate valves)	73
75	open valves (hipap valves and gate valves)	74
76	repair (hipap valves and gate valves)	75

77	test/ inspection (hipap valves and gate valves)	76
78	transport to board (hipap valves and gate valves)	77
79	mounting (hipap valves and gate valves)	78
	bottom plugs	
80	remove (bottom plugs)	
81	replaced (bottom plugs)	80
82	vacuum test (bottom plugs)	80
	Propeller	
83	mount stages (propeller)	
84	clean/ polish propeller (propeller)	83
85	install protection for painting (propeller)	84
86	remove protection (propeller)	85
87	dismount stages (propeller)	86
	Tank cleaning	
88	clean tank #1 (diesel tanks)	
89	clean tank #2 (fresh water tanks)	
	Support	
90	mount stages (n/a)	
91	weld eyes pad (n/a)	
92	remove eyes pad (n/a)	90.91
93	dismount stages (n/a)	90.91
	Pipes	
94	cut (bio guard antifouling system)	
95	fabricate (bio guard antifouling system)	94
96	mounting (bio guard antifouling system)	95
97	inspection (bio guard antifouling system)	96
98	test (bio guard antifouling system)	97
	treatment and painting	
99	mount stages (deck crane)	
100	degreasing (deck crane)	99
101	treatment st 3 (deck crane)	100
102	painting (deck crane)	101
103	dismount stages (deck crane)	102
	Support	
104	mount stages hydraulics hoses (deck crane)	
105	dismounting stages hydraulics hoses (deck crane)	104
	life boats	
106	prepare blocks for receive boat on shipyard-min.300 mm of height (life boats)	
107	mounting stages for access forward and after locations hooks (life boats)	
108	supply equipment for lifting 500 kg height min. 06 meters (life boats)	

Table AIV.2: Task-Modes-Resource Requirements			
Task	Mode	Qty.	Description
1	1	2	carrier
1	1	2	plumber
1	2	1	plumber
2	1	2	fluid jet
2	1	2	support
2	2	1	fluid jet
3	1	2	painter
3	2	1	painter
4	1	2	support
4	1	4	painter
4	2	4	painter
4	2	2	support
5	1	1	support
5	2	1	support
6	1	1	plumber
6	2	1	carrier
7	1	2	fluid jet
7	2	1	fluid jet
8	1	2	support
8	1	2	fluid jet
8	2	1	fluid jet
9	1	2	painter
9	2	1	painter
10	1	2	support
10	1	4	painter
10	2	4	painter
11	1	1	support
11	2	1	support
12	1	1	support
12	2	1	support
13	1	2	support
13	1	2	fluid jet
13	2	1	fluid jet
14	1	2	painter
14	2	1	painter
15	1	2	support
15	1	4	painter
15	2	4	painter
16	1	1	painter
16	2	1	support
17	1	1	assembler
17	2	2	assembler

18	1	2	support
18	1	2	fluid jet
18	2	1	fluid jet
19	1	2	painter
19	2	1	painter
20	1	5	painter
20	1	2	support
20	2	4	painter
21	1	1	assembler
21	1	2	welder
21	1	1	blow torched
21	1	1	painter
21	2	1	blow torched
21	2	1	assembler
21	2	1	welder
21	2	1	painter
22	1	3	assembler
22	2	3	assembler
23	1	2	mechanic
23	2	1	mechanic
24	1	1	electrician
24	2	1	electrician
25	1	10	fluid jet
26	1	10	fluid jet
27	1	10	painter
27	1	10	support
28	1	1	painter
28	1	2	welder
28	1	1	assembler
28	1	1	blow torched
28	2	1	assembler
28	2	1	welder
28	2	1	blow torched
28	2	1	painter
29	1	10	painter
29	2	10	painter
30	1	1	electrician
31	1	2	mechanic
32	1	3	assembler
33	1	3	assembler
34	1	4	mechanic
35	1	20	fluid jet
36	1	10	painter
37	1	10	painter

38	1	1	assembler
38	1	1	painter
38	1	1	blow torched
38	1	2	welder
38	2	1	welder
38	2	1	blow torched
38	2	1	assembler
38	2	1	painter
39	1	2	mechanic
40	1	3	assembler
41	1	3	assembler
42	1	2	fluid jet
43	1	20	fluid jet
44	1	10	painter
45	1	10	painter
46	1	3	assembler
47	1	2	welder
47	1	1	blow torched
47	1	1	assembler
47	1	1	painter
48	1	9	assembler
49	1	4	fluid jet
50	1	20	fluid jet
51	1	10	painter
52	1	10	painter
53	1	6	assembler
54	1	5	welder
54	1	5	support
54	1	5	blow torched
54	1	5	assembler
54	1	5	painter
55	1	5	welder
55	1	2	support
56	1	9	assembler scaffolding
57	1	1	support
58	1	5	blow torched
59	1	10	brazier
59	1	10	welder
60	1	10	assembler
61	1	20	welder
62	1	1	quality control
63	1	1	quality control
64	1	10	support
64	1	10	painter

65	1	6	assembler
66	1	10	plumber
67	1	1	plumber
67	1	10	carrier
67	1	1	Truck 1
68	1	1	welder
68	1	1	Valves Repairing
69	1	1	welder
69	1	1	Valves Repairing
70	1	1	welder
70	1	1	Valves Repairing
71	1	10	carrier
72	1	10	plumber
73	1	10	plumber
74	1	1	plumber
74	1	1	Truck 1
74	1	10	carrier
75	1	1	Valves Repairing
75	1	1	welder
76	1	1	Valves Repairing
76	1	1	welder
77	1	1	Valves Repairing
77	1	1	welder
78	1	10	carrier
79	1	10	plumber
80	1	4	support
80	1	4	dock aid
81	1	4	support
81	1	4	dock aid
82	1	4	support
82	1	4	dock aid
83	1	10	assembler scaffolding
84	1	10	fluid jet
85	1	10	painter
86	1	10	fluid jet
87	1	10	assembler scaffolding
88	1	20	fluid jet
88	1	20	safety
89	1	20	fluid jet
89	1	20	safety
90	1	10	support
91	1	10	support
92	1	10	support
93	1	10	support

94	1	1	Pipe Bending
94	1	1	fluid jet
95	1	1	Pipe Bending
95	1	1	fluid jet
96	1	1	Pipe Bending
96	1	1	fluid jet
97	1	1	Pipe Bending
97	1	1	fluid jet
98	1	1	Pipe Bending
98	1	1	fluid jet
99	1	1	crane operator
99	1	5	painter
99	1	5	fluid jet
100	1	5	fluid jet
100	1	1	crane operator
100	1	5	painter
101	1	5	fluid jet
101	1	1	crane operator
101	1	5	painter
102	1	1	crane operator
102	1	5	fluid jet
102	1	5	painter
103	1	5	painter
103	1	1	crane operator
103	1	5	fluid jet
104	1	1	support
105	1	1	support
106	1	1	support
107	1	2	assembler scaffolding
108	1	1	Pulley 1
108	1	1	support
108	1	1	machine operator
109	1	1	Power Generator 1
109	1	1	support
110	1	5	support
111	1	5	support

	Table AIV.3: Multi-skilled resources	
Resource	**Description**	**Rank**
R76	plumber	1
R76	quality control	2
R76	safety	3
R77	machine operator	1
R77	electrician	2
R77	welder	3
R78	fluid jet	1
R78	electrician	2
R78	plumber	3
R79	blow torched	1
R79	dock aid	2
R79	carrier	3
R80	assembler scaffolding	1
R80	safety	2
R81	general services	1
R81	machine operator	2
R81	quality control	3
R82	carrier	1
R82	mechanic	2
R82	assembler scaffolding	3
R83	support	1
R83	blow torched	3
R84	painter	1
R84	assembler	2
R84	electrician	3
R85	welder	1
R85	machine operator	2
R86	safety	1
R86	painter	2
R86	general services	3
R87	fluid jet	1
R87	crane operator	2
R87	fireman	3
R88	carpenter	1
R88	general services	2
R88	crane operator	3
R89	carpenter	1
R89	brazier	2
R89	fireman	3
R90	machine operator	1
R90	welder	2
R90	assembler scaffolding	3

R91	support	1
R91	assembler	3
R92	fireman	1
R92	carrier	2
R92	brazier	3
R93	crane operator	1
R93	general services	2
R93	fluid jet	3
R94	machine operator	1
R94	general services	2
R94	carrier	3
R95	welder	1
R95	fireman	2
R95	assembler scaffolding	3
R96	dock aid	1
R96	support	2
R96	electrician	3
R97	crane operator	1
R97	plumber	2
R97	carpenter	3
R98	electrician	1
R98	plumber	2
R98	brazier	3
R99	carrier	1
R99	fluid jet	2
R99	fireman	3
R100	brazier	1
R100	carrier	2
R100	fireman	3
R101	safety	1
R101	crane operator	2
R101	assembler	3
R102	assembler	1
R102	safety	2
R102	carrier	3
R103	blow torched	1
R103	brazier	2
R103	plumber	3
R104	brazier	1
R104	quality control	2
R104	safety	3
R105	painter	1
R105	mechanic	2
R105	machine operator	3

R106	quality control	1
R106	machine operator	2
R106	welder	3
R107	quality control	1
R107	welder	2
R108	dock aid	1
R108	fireman	2
R108	carpenter	3
R109	carrier	1
R109	dock aid	2
R109	safety	3
R110	carpenter	1
R110	general services	2
R110	plumber	3
R111	mechanic	1
R111	dock aid	2
R111	assembler	3
R112	general services	1
R112	fireman	2
R112	assembler scaffolding	3
R113	dock aid	1
R113	electrician	2
R113	painter	3
R114	general services	1
R114	fluid jet	2
R114	crane operator	3
R115	fireman	1
R115	brazier	2
R116	welder	1
R116	safety	3
R117	fireman	1
R117	blow torched	2
R117	plumber	3
R118	dock aid	1
R118	electrician	2
R118	general services	3
R119	brazier	1
R119	quality control	2
R119	carpenter	3
R120	mechanic	1
R120	safety	2
R120	assembler scaffolding	3
R121	carpenter	1
R121	brazier	2

R121	welder	3
R122	safety	1
R122	mechanic	2
R122	quality control	3
R123	assembler	1
R123	blow torched	3
R124	plumber	1
R124	brazier	2
R124	carpenter	3
R125	machine operator	1
R125	carpenter	2
R125	mechanic	3
R126	brazier	1
R126	blow torched	2
R126	dock aid	3
R127	machine operator	1
R127	carrier	2
R127	painter	3
R128	welder	1
R128	plumber	2
R128	safety	3
R129	brazier	1
R129	assembler	2
R129	quality control	3
R130	general services	1
R130	dock aid	2
R130	brazier	3
R131	crane operator	1
R131	plumber	2
R131	brazier	3
R132	mechanic	1
R132	dock aid	2
R132	plumber	3
R133	assembler scaffolding	1
R133	dock aid	2
R134	general services	1
R134	blow torched	2
R134	safety	3
R135	fireman	1
R135	machine operator	2
R135	carrier	3
R136	dock aid	1
R136	machine operator	2
R136	support	3

R137	crane operator	1
R137	painter	2
R137	electrician	3
R138	quality control	1
R138	plumber	2
R138	assembler scaffolding	3
R139	mechanic	1
R139	safety	3
R140	blow torched	1
R140	assembler	2
R140	welder	3
R141	carrier	1
R141	safety	2
R141	assembler scaffolding	3
R142	carpenter	1
R142	blow torched	2
R142	machine operator	3
R143	painter	1
R143	safety	2
R143	blow torched	3
R144	general services	1
R144	brazier	2
R144	machine operator	3
R145	quality control	1
R145	general services	3
R146	fireman	1
R146	carpenter	2
R146	assembler	3
R147	dock aid	1
R147	painter	2
R147	carpenter	3
R148	safety	1
R148	quality control	2
R148	support	3
R149	support	1
R149	carrier	2
R149	assembler	3
R150	mechanic	1
R150	fireman	2
R150	quality control	3
R151	carrier	1
R151	safety	2
R151	mechanic	3
R152	crane operator	1

R152	general services	2
R152	machine operator	3
R153	quality control	1
R153	machine operator	2
R153	general services	3
R154	assembler scaffolding	1
R154	machine operator	2
R154	quality control	3
R155	machine operator	1
R155	mechanic	2
R155	fireman	3
R156	crane operator	1
R156	fluid jet	2
R156	electrician	3
R157	mechanic	1
R157	welder	2
R157	fluid jet	3
R158	assembler	1
R158	general services	2
R158	crane operator	3
R159	assembler	1
R159	blow torched	2
R159	plumber	3
R160	assembler	1
R160	dock aid	2
R160	blow torched	3
R161	fluid jet	1
R161	painter	2
R161	machine operator	3
R162	welder	1
R162	general services	2
R162	blow torched	3
R163	assembler scaffolding	1
R163	brazier	2
R163	plumber	3
R164	crane operator	1
R164	quality control	2
R164	electrician	3
R165	assembler scaffolding	1
R165	quality control	2
R165	fireman	3
R166	blow torched	1
R166	electrician	2
R166	fluid jet	3

R167	crane operator	1
R167	blow torched	2
R167	painter	3
R168	safety	1
R168	assembler	2
R168	painter	3
R169	support	1
R169	dock aid	2
R169	assembler scaffolding	3
R170	blow torched	1
R170	dock aid	2
R170	general services	3
R171	plumber	1
R171	carpenter	2
R171	painter	3
R172	carrier	1
R172	quality control	2
R172	fireman	3
R173	painter	1
R173	crane operator	2
R173	support	3
R174	blow torched	1
R174	fluid jet	2
R174	dock aid	3
R175	machine operator	1
R175	assembler scaffolding	2
R175	general services	3
R176	fireman	1
R176	carpenter	2
R177	plumber	1
R177	safety	2
R177	painter	3
R178	fireman	1
R178	electrician	2
R179	painter	1
R179	assembler	2
R179	carpenter	3
R180	fluid jet	1
R180	crane operator	2
R180	general services	3
R181	carrier	1
R181	electrician	2
R181	general services	3
R182	mechanic	1

R182	machine operator	2
R182	assembler scaffolding	3
R183	painter	1
R183	welder	2
R183	crane operator	3
R184	safety	1
R184	electrician	2
R184	assembler	3
R185	assembler scaffolding	1
R185	fluid jet	2
R185	support	3
R186	mechanic	1
R186	welder	2
R186	crane operator	3
R187	mechanic	1
R187	assembler	2
R187	safety	3
R188	plumber	1
R188	blow torched	2
R188	electrician	3
R189	carrier	1
R189	assembler scaffolding	2
R189	brazier	3
R190	crane operator	1
R190	electrician	2
R191	electrician	1
R191	assembler	2
R191	crane operator	3
R192	crane operator	1
R192	brazier	2
R193	quality control	1
R193	carpenter	2
R193	dock aid	3
R194	quality control	1
R194	safety	2
R194	plumber	3
R195	dock aid	1
R195	plumber	2
R195	quality control	3
R196	painter	1
R196	fluid jet	2
R196	welder	3
R197	plumber	1
R197	brazier	2

R197	fluid jet	3
R198	mechanic	1
R198	support	2
R198	general services	3
R199	carrier	1
R199	assembler scaffolding	2
R199	machine operator	3
R200	fireman	1
R200	quality control	2
R200	brazier	3
R201	machine operator	1
R201	brazier	2
R201	carpenter	3
R202	electrician	1
R202	dock aid	2
R202	painter	3
R203	crane operator	1
R203	plumber	2
R203	carrier	3
R204	machine operator	1
R205	brazier	1
R205	carpenter	2
R205	carrier	3
R206	general services	1
R206	brazier	3
R207	electrician	1
R207	blow torched	2
R207	mechanic	3
R208	quality control	1
R208	general services	2
R209	assembler scaffolding	1
R209	general services	2
R209	carpenter	3
R210	blow torched	1
R210	brazier	2
R210	fireman	3
R211	crane operator	1
R211	brazier	2
R211	welder	3
R212	general services	1
R212	dock aid	2
R212	blow torched	3
R213	crane operator	1
R213	fireman	2

R213	quality control	3
R214	quality control	1
R214	carrier	2
R214	assembler scaffolding	3
R215	fireman	1
R215	assembler	2
R215	fluid jet	3
R216	carrier	1
R216	brazier	2
R216	fluid jet	3
R217	safety	1
R217	machine operator	2
R217	mechanic	3
R218	support	1
R218	blow torched	2
R218	assembler scaffolding	3
R219	plumber	1
R219	fireman	2
R220	painter	1
R220	dock aid	2
R220	plumber	3
R221	dock aid	1
R221	support	2
R221	fireman	3
R222	painter	1
R222	fluid jet	2
R222	electrician	3
R223	dock aid	1
R223	general services	2
R223	support	3
R224	welder	1
R224	blow torched	2
R225	dock aid	1
R225	assembler	2
R226	carpenter	1
R226	plumber	2
R226	brazier	3
R227	welder	1
R227	fluid jet	2
R227	carrier	3
R228	quality control	1
R228	assembler	2
R228	crane operator	3
R229	support	1

R229	assembler scaffolding	2
R229	blow torched	3
R230	quality control	1
R230	fireman	2
R230	assembler	3
R231	crane operator	1
R231	carrier	3
R232	painter	1
R232	assembler	2
R232	brazier	3
R233	mechanic	1
R233	assembler	2
R233	fireman	3
R234	fluid jet	1
R234	carrier	2
R234	mechanic	3
R235	fireman	1
R235	brazier	3
R236	brazier	1
R236	painter	2
R237	crane operator	1
R237	mechanic	2
R237	safety	3
R238	general services	1
R238	painter	2
R238	welder	3
R239	general services	1
R239	welder	2
R239	brazier	3
R240	machine operator	1
R240	carpenter	2
R241	brazier	1
R241	general services	2
R241	painter	3
R242	carrier	1
R242	carpenter	2
R242	assembler	3
R243	carpenter	1
R243	crane operator	2
R243	assembler	3
R244	plumber	1
R244	blow torched	2
R244	carrier	3
R245	machine operator	1

R245	support	2
R245	fluid jet	3
R246	assembler scaffolding	1
R246	quality control	2
R246	carpenter	3
R247	dock aid	1
R247	brazier	2
R247	fluid jet	3
R248	fireman	1
R248	support	2
R248	assembler scaffolding	3
R249	brazier	1
R249	support	2
R249	machine operator	3
R250	quality control	1
R250	electrician	2
R250	blow torched	3
R251	support	1
R251	fireman	2
R251	quality control	3
R252	dock aid	1
R252	crane operator	2
R252	plumber	3
R253	fireman	1
R253	blow torched	2
R254	blow torched	1
R254	brazier	2
R254	painter	3
R255	general services	1
R255	quality control	2
R256	quality control	1
R256	safety	2
R256	assembler scaffolding	3
R257	mechanic	1
R257	assembler	3
R258	mechanic	1
R258	welder	2
R259	carpenter	1
R259	carrier	2
R260	machine operator	1
R260	safety	3
R261	dock aid	1
R261	fireman	2
R261	mechanic	3

R262	brazier	1
R262	quality control	2
R262	electrician	3
R263	electrician	1
R263	brazier	2
R263	carrier	3
R264	plumber	1
R264	assembler scaffolding	2
R264	assembler	3
R265	dock aid	1
R265	carrier	2
R265	support	3
R266	blow torched	1
R266	carpenter	2
R266	assembler	3
R267	dock aid	1
R267	machine operator	2
R267	plumber	3
R268	welder	1
R268	assembler scaffolding	2
R268	safety	3
R269	mechanic	1
R269	blow torched	2
R269	carrier	3
R270	dock aid	1
R270	carpenter	2
R270	carrier	3
R271	support	1
R271	machine operator	2
R271	carrier	3
R272	assembler	1
R272	painter	2
R272	electrician	3
R273	blow torched	1
R273	general services	2
R273	safety	3
R274	carpenter	1
R274	blow torched	2
R274	machine operator	3
R275	assembler	1
R275	machine operator	2
R275	dock aid	3
R276	brazier	1
R276	general services	2

R276	crane operator	3
R277	fireman	1
R277	brazier	2
R277	dock aid	3
R278	crane operator	1
R278	painter	2
R278	brazier	3
R279	fireman	1
R279	electrician	2
R279	crane operator	3
R280	carrier	1
R280	crane operator	2
R280	electrician	3
R281	brazier	1
R281	support	2
R281	crane operator	3
R282	assembler scaffolding	1
R282	mechanic	2
R282	carrier	3
R283	carrier	1
R283	assembler scaffolding	2
R283	support	3
R284	safety	1
R284	painter	2
R284	carpenter	3
R285	machine operator	1
R285	general services	2
R285	painter	3
R286	dock aid	1
R286	carpenter	2
R286	welder	3
R287	brazier	1
R287	carpenter	2
R287	assembler	3
R288	painter	1
R288	assembler scaffolding	2
R288	dock aid	3
R289	carrier	1
R289	dock aid	2
R289	plumber	3
R290	general services	1
R290	dock aid	2
R290	assembler	3
R291	support	1

R291	welder	2
R291	assembler scaffolding	3
R292	crane operator	1
R292	mechanic	2
R292	blow torched	3
R293	electrician	1
R293	fireman	2
R293	carrier	3
R294	carrier	1
R294	fireman	2
R295	quality control	1
R295	machine operator	2
R296	assembler scaffolding	1
R296	crane operator	2
R296	dock aid	3
R297	assembler scaffolding	1
R297	dock aid	2
R297	plumber	3
R298	fluid jet	1
R298	general services	2
R298	dock aid	3
R299	support	1
R299	carrier	2
R299	assembler scaffolding	3
R300	carrier	1
R300	brazier	2
R300	mechanic	3
R301	fluid jet	1
R301	machine operator	2
R301	quality control	3
R302	blow torched	1
R302	plumber	2
R302	general services	3
R303	fireman	1
R303	electrician	3
R304	dock aid	1
R304	support	2
R304	fluid jet	3
R305	machine operator	1
R305	general services	2
R305	painter	3
R306	machine operator	1
R306	brazier	2
R306	mechanic	3

R307	electrician	1
R307	carrier	2
R307	fluid jet	3
R308	quality control	1
R308	carrier	2
R308	machine operator	3
R309	general services	1
R309	blow torched	2
R310	fireman	1
R310	blow torched	2
R310	carrier	3
R311	machine operator	1
R311	plumber	2
R312	electrician	1
R312	machine operator	2
R313	electrician	1
R313	blow torched	2
R313	plumber	3
R314	blow torched	1
R314	support	2
R315	safety	1
R315	quality control	2
R316	carrier	1
R316	brazier	2
R317	crane operator	1
R317	safety	2
R317	electrician	3
R318	electrician	1
R318	brazier	2
R318	welder	3
R319	carpenter	1
R319	painter	2
R319	general services	3
R320	fireman	1
R320	machine operator	2
R320	welder	3
R321	safety	1
R321	fireman	2
R321	machine operator	3
R322	carrier	1
R322	carpenter	2
R322	general services	3
R323	assembler scaffolding	1
R323	carpenter	2

R323	crane operator	3
R324	general services	1
R324	quality control	2
R325	assembler scaffolding	1
R325	quality control	2
R325	electrician	3
R326	assembler scaffolding	1
R326	safety	2
R326	general services	3
R327	machine operator	1
R327	fluid jet	3
R328	fluid jet	1
R328	assembler	2
R328	dock aid	3
R329	quality control	1
R329	assembler	2
R329	machine operator	3
R330	safety	1
R330	carrier	2
R330	plumber	3
R331	carpenter	1
R331	carrier	2
R331	painter	3
R332	brazier	1
R332	quality control	2
R332	assembler scaffolding	3
R333	safety	1
R333	blow torched	2
R333	quality control	3
R334	painter	1
R334	support	2
R334	assembler	3
R335	assembler	1
R335	support	2
R335	fireman	3
R336	crane operator	1
R336	electrician	2
R336	assembler scaffolding	3
R337	safety	1
R337	blow torched	2
R337	electrician	3
R338	fluid jet	1
R338	electrician	2
R338	carrier	3

R339	crane operator	1
R339	blow torched	2
R339	dock aid	3
R340	fireman	1
R340	assembler	2
R340	fluid jet	3
R341	blow torched	1
R341	brazier	2
R341	machine operator	3
R342	painter	1
R342	fireman	2
R342	crane operator	3
R343	machine operator	1
R343	blow torched	2
R343	safety	3
R344	plumber	1
R344	fluid jet	3
R345	painter	1
R345	blow torched	2
R345	general services	3
R346	assembler	1
R346	fireman	2
R346	support	3
R347	quality control	1
R347	dock aid	2
R348	carrier	1
R348	crane operator	2
R348	dock aid	3
R349	welder	1
R349	mechanic	2
R350	fluid jet	1
R350	plumber	2
R350	blow torched	3
R351	assembler scaffolding	1
R351	dock aid	2
R351	carrier	3
R352	blow torched	1
R352	fireman	2
R352	plumber	3
R353	fluid jet	1
R353	carrier	2
R353	assembler scaffolding	3
R354	support	1
R354	quality control	3

R355	safety	1
R355	assembler	2
R356	mechanic	1
R356	welder	2
R356	dock aid	3
R357	blow torched	1
R357	fireman	2
R358	assembler scaffolding	1
R358	brazier	2
R358	mechanic	3
R359	assembler scaffolding	1
R359	general services	2
R359	plumber	3
R360	carpenter	1
R360	electrician	2
R361	machine operator	1
R361	assembler scaffolding	2
R361	welder	3
R362	carpenter	1
R362	assembler scaffolding	2
R362	safety	3
R363	machine operator	1
R363	assembler scaffolding	2
R363	mechanic	3
R364	safety	1
R364	dock aid	2
R364	welder	3
R365	machine operator	1
R365	fluid jet	2
R365	assembler scaffolding	3
R366	assembler scaffolding	1
R366	general services	2
R367	painter	1
R367	plumber	2
R367	dock aid	3
R368	fluid jet	1
R368	dock aid	2
R368	fireman	3
R369	safety	1
R369	blow torched	2
R369	quality control	3
R370	safety	1
R370	fluid jet	2
R370	machine operator	3

R371	blow torched	1
R371	assembler	2
R371	support	3
R372	fluid jet	1
R372	general services	2
R373	safety	1
R373	crane operator	2
R373	machine operator	3
R374	carrier	1
R374	plumber	2
R374	brazier	3
R375	support	1
R375	machine operator	2
R375	plumber	3
R376	quality control	1
R376	mechanic	2
R376	blow torched	3
R377	mechanic	1
R377	brazier	2
R377	assembler	3
R378	plumber	1
R378	dock aid	2
R378	assembler scaffolding	3
R379	quality control	1
R379	safety	2
R379	plumber	3
R380	general services	1
R380	painter	2
R380	fluid jet	3
R381	welder	1
R381	painter	2
R381	fluid jet	3
R382	assembler	1
R382	general services	2
R382	machine operator	3
R383	assembler	1
R383	assembler scaffolding	2
R383	brazier	3
R384	fireman	1
R384	general services	2
R384	electrician	3
R385	support	1
R385	blow torched	2
R385	assembler	3

R386	support	1
R386	carpenter	2
R386	mechanic	3
R387	fireman	1
R387	blow torched	2
R388	welder	1
R388	blow torched	2
R388	mechanic	3
R389	brazier	1
R389	dock aid	2
R389	machine operator	3
R390	painter	1
R390	safety	2
R391	mechanic	1
R391	electrician	2
R392	brazier	1
R392	support	2
R392	machine operator	3
R393	general services	1
R393	dock aid	2
R393	plumber	3
R394	general services	1
R394	machine operator	2
R394	assembler	3
R395	plumber	1
R395	fluid jet	2
R395	blow torched	3
R396	electrician	1
R396	carpenter	2
R397	mechanic	1
R397	support	2
R397	crane operator	3
R398	safety	1
R398	fluid jet	2
R398	carrier	3
R399	support	1
R399	dock aid	2
R399	carrier	3
R400	general services	1
R400	quality control	2
R400	fluid jet	3
R401	assembler scaffolding	1
R401	fluid jet	2
R402	support	1

R402	crane operator	2
R402	fireman	3
R403	welder	1
R403	brazier	2
R403	quality control	3
R404	assembler scaffolding	1
R404	general services	2
R404	safety	3
R405	crane operator	1
R405	dock aid	2
R406	fluid jet	1
R406	brazier	2
R406	assembler scaffolding	3
R407	painter	1
R407	fireman	2
R407	support	3
R408	crane operator	1
R408	assembler	2
R408	support	3
R409	safety	1
R409	dock aid	2
R410	crane operator	1
R410	blow torched	2
R410	plumber	3
R411	assembler	1
R411	support	2
R411	quality control	3
R412	assembler scaffolding	1
R412	painter	2
R412	crane operator	3
R413	machine operator	1
R413	crane operator	2
R413	plumber	3
R414	electrician	1
R414	welder	2
R414	fireman	3
R415	fluid jet	1
R415	support	2
R415	quality control	3
R416	welder	1
R416	assembler scaffolding	2
R416	plumber	3
R417	blow torched	1
R417	fluid jet	2

R417	machine operator	3
R418	fluid jet	1
R418	assembler scaffolding	2
R418	blow torched	3
R419	painter	1
R419	crane operator	2
R419	safety	3
R420	safety	1
R420	crane operator	2
R420	carrier	3
R421	quality control	1
R421	blow torched	2
R421	assembler	3
R422	painter	1
R422	electrician	2
R422	brazier	3
R423	mechanic	1
R423	assembler scaffolding	2
R423	blow torched	3
R424	painter	1
R424	brazier	2
R424	mechanic	3
R425	dock aid	1
R425	carpenter	2
R425	brazier	3
R426	machine operator	1
R426	blow torched	2
R427	machine operator	1
R427	blow torched	2
R427	assembler	3
R428	dock aid	1
R428	blow torched	2
R428	support	3
R429	brazier	1
R429	safety	2
R429	assembler	3
R430	blow torched	1
R430	painter	2
R431	plumber	1
R431	blow torched	2
R431	general services	3
R432	support	1
R432	plumber	2
R432	blow torched	3

R433	mechanic	1
R433	quality control	2
R433	painter	3
R434	fireman	1
R434	crane operator	2
R434	assembler scaffolding	3
R435	general services	1
R435	brazier	2
R435	safety	3
R436	plumber	1
R436	safety	2
R436	brazier	3
R437	plumber	1
R437	machine operator	2
R437	dock aid	3
R438	electrician	1
R438	mechanic	2
R438	assembler scaffolding	3
R439	quality control	1
R439	machine operator	2
R439	carpenter	3
R440	fluid jet	1
R440	blow torched	2
R440	general services	3
R441	general services	1
R441	carrier	2
R441	machine operator	3
R442	painter	1
R442	brazier	2
R442	electrician	3
R443	brazier	1
R443	fireman	2
R443	machine operator	3
R444	carpenter	1
R444	blow torched	2
R444	dock aid	3
R445	general services	1
R445	blow torched	2
R446	blow torched	1
R446	quality control	2
R446	safety	3
R447	crane operator	1
R447	assembler	2
R447	welder	3

R448	support	1
R448	blow torched	2
R449	assembler	1
R449	dock aid	2
R449	blow torched	3
R450	mechanic	1
R450	welder	2
R450	dock aid	3
R451	general services	1
R451	assembler	2
R452	welder	1
R452	support	2
R452	fireman	3
R453	welder	1
R453	blow torched	2
R453	fluid jet	3
R454	assembler scaffolding	1
R454	brazier	2
R454	blow torched	3
R455	general services	1
R455	crane operator	2
R455	quality control	3
R456	support	1
R456	quality control	2
R456	fluid jet	3
R457	assembler scaffolding	1
R457	carrier	2
R458	safety	1
R458	painter	2
R459	carpenter	1
R459	mechanic	2
R459	blow torched	3
R460	blow torched	1
R460	crane operator	2
R460	assembler scaffolding	3
R461	crane operator	1
R461	welder	2
R461	mechanic	3
R462	assembler	1
R462	welder	2
R462	fluid jet	3
R463	support	1
R463	fluid jet	2
R463	plumber	3

R464	quality control	1
R464	welder	2
R464	general services	3
R465	plumber	1
R465	welder	2
R465	crane operator	3
R466	carrier	1
R466	carpenter	2
R466	general services	3
R467	carpenter	1
R467	assembler scaffolding	2
R467	painter	3
R468	plumber	1
R468	welder	3
R469	blow torched	1
R469	dock aid	2
R469	safety	3
R470	dock aid	1
R470	general services	2
R470	blow torched	3
R471	carpenter	1
R471	mechanic	2
R471	blow torched	3
R472	painter	1
R472	crane operator	2
R472	carrier	3
R473	crane operator	1
R473	quality control	2
R474	electrician	1
R474	carpenter	2
R474	support	3
R475	fluid jet	1
R475	carrier	2
R475	blow torched	3

Table AIV.4: Cost and Schedule								
T	**Due Date**	**Start Date**	**End Date**	**D.**	**Fixed Cost**	**Variable Cost**	**Penalty Cost**	**Bonus**
1	6/3/2015	5/18/2015	5/20/2015	-13.29	$446.0	$27,360.23	$0.00	$66,458.25
2	6/12/2015	5/21/2015	6/1/2015	-10.5	$410.00	$128,640.27	$0.00	$52,499.85
3	6/27/2015	6/1/2015	6/15/2015	-11.29	$111.00	$40,800.10	$0.00	$56,458.10
4	7/10/2015	6/16/2015	6/26/2015	-13.29	$658.00	$179,280.69	$0.00	$66,458.15
5	7/10/2015	6/29/2015	6/29/2015	-10.63	$54.00	$60.00	$0.00	$53,125.00
6	7/11/2015	6/29/2015	6/29/2015	-11.54	$55.00	$120.01	$0.00	$57,708.30
7	5/31/2015	5/18/2015	5/19/2015	-11.5	$117.00	$3,360.04	$0.00	$57,499.95
8	6/11/2015	5/19/2015	5/28/2015	-13.29	$472.00	$106,080.31	$0.00	$66,458.15
9	6/25/2015	5/29/2015	6/12/2015	-12.5	$111.00	$40,800.12	$0.00	$62,499.80
10	7/5/2015	6/12/2015	6/23/2015	-11.29	$642.00	$192,960.40	$0.00	$56,458.15
11	7/7/2015	6/24/2015	6/24/2015	-12.63	$52.00	$60.00	$0.00	$63,125.00
12	5/29/2015	5/18/2015	5/18/2015	-10.63	$59.00	$60.00	$0.00	$53,125.00
13	6/10/2015	5/18/2015	5/27/2015	-13.42	$448.00	$106,080.31	$0.00	$67,083.20
14	7/9/2015	5/27/2015	6/25/2015	-13.42	$104.00	$83,520.20	$0.00	$67,082.85
15	7/21/2015	6/25/2015	7/7/2015	-13.62	$616.00	$203,760.40	$0.00	$68,124.40
16	7/22/2015	7/7/2015	7/7/2015	-14.54	$103.00	$240.02	$0.00	$72,707.70
17	6/4/2015	5/18/2015	5/22/2015	-12.5	$54.00	$6,000.03	$0.00	$62,499.90
18	7/1/2015	5/22/2015	6/18/2015	-12.58	$462.00	$309,600.69	$0.00	$62,916.30
19	7/16/2015	6/18/2015	7/17/2015	1.42	$60.00	$41,760.10	$14,168.10	$0.00
20	7/24/2015	7/17/2015	7/27/2015	3.42	$736.00	$201,600.40	$34,168.30	$0.00
21	7/17/2015	6/19/2015	7/6/2015	-10.29	$1,112.00	$500,401.44	$0.00	$51,458.10
22	5/31/2015	5/18/2015	5/19/2015	-11.5	$171.00	$5,040.06	$0.00	$57,499.95
23	5/31/2015	5/18/2015	5/20/2015	-10.29	$105.00	$6,840.06	$0.00	$51,458.25
24	5/31/2015	5/18/2015	5/20/2015	-10.29	$59.00	$3,420.03	$0.00	$51,458.25
25	6/9/2015	5/18/2015	6/1/2015	-7.5	$548.00	$204,000.62	$0.00	$37,499.80
26	6/9/2015	5/18/2015	6/1/2015	-7.5	$566.00	$204,000.62	$0.00	$37,499.80
27	6/21/2015	6/1/2015	6/10/2015	-10.29	$2,232.00	$528,001.34	$0.00	$51,458.15
28	6/29/2015	6/11/2015	6/18/2015	-10.29	$1,080.00	$212,400.86	$0.00	$51,458.20
29	7/7/2015	6/11/2015	6/26/2015	-10.29	$571.00	$221,400.72	$0.00	$51,458.10
30	7/7/2015	6/29/2015	6/29/2015	-7.63	$59.00	$60.00	$0.00	$38,125.00
31	7/12/2015	6/29/2015	7/1/2015	-10.29	$110.00	$6,840.06	$0.00	$51,458.25
32	7/12/2015	6/29/2015	7/1/2015	-10.29	$167.00	$10,260.09	$0.00	$51,458.25
33	5/28/2015	5/18/2015	5/19/2015	-8.5	$172.00	$5,040.06	$0.00	$42,499.95
34	5/28/2015	5/18/2015	5/19/2015	-8.5	$221.00	$6,720.08	$0.00	$42,499.95
35	6/11/2015	6/1/2015	6/2/2015	-8.29	$1,109.00	$33,600.10	$0.00	$41,458.25
36	6/22/2015	6/3/2015	6/12/2015	-9.5	$536.00	$132,000.34	$0.00	$47,499.85

37	7/2/2015	6/12/2015	6/23/2015	-8.29	$534.00	$160,800.34	$0.00	$41,458.15
38	6/29/2015	6/12/2015	6/18/2015	-10.29	$1,080.00	$177,600.38	$0.00	$51,458.20
39	7/4/2015	6/24/2015	6/24/2015	-9.63	$110.00	$120.01	$0.00	$48,125.00
40	7/5/2015	6/24/2015	6/24/2015	-10.54	$167.00	$360.03	$0.00	$52,708.30
41	6/4/2015	5/18/2015	5/22/2015	-12.5	$168.00	$18,000.10	$0.00	$62,499.90
42	6/14/2015	5/22/2015	6/2/2015	-11.29	$110.00	$32,160.07	$0.00	$56,458.15
43	6/25/2015	6/3/2015	6/12/2015	-12.5	$1,107.00	$264,000.67	$0.00	$62,499.85
44	7/5/2015	6/12/2015	6/23/2015	-11.29	$553.00	$160,800.34	$0.00	$56,458.15
45	7/16/2015	6/24/2015	7/3/2015	-12.5	$533.00	$132,000.48	$0.00	$62,499.85
46	7/21/2015	7/3/2015	7/8/2015	-12.5	$172.00	$21,420.03	$0.00	$62,499.90
47	7/15/2015	6/24/2015	6/30/2015	-14.5	$1,064.00	$177,600.67	$0.00	$72,499.90
48	5/31/2015	5/18/2015	5/19/2015	-11.5	$515.00	$15,120.17	$0.00	$57,499.95
49	6/4/2015	5/19/2015	5/22/2015	-12.5	$224.00	$17,280.06	$0.00	$62,499.90
50	7/2/2015	6/4/2015	6/12/2015	-19.42	$1,114.00	$230,400.58	$0.00	$97,083.15
51	7/15/2015	6/24/2015	7/1/2015	-13.29	$539.00	$106,200.43	$0.00	$66,458.20
52	7/13/2015	7/2/2015	7/13/2015	0.5	$544.00	$160,800.48	$5,000.30	$0.00
53	7/15/2015	7/13/2015	7/14/2015	-0.29	$342.00	$10,080.03	$0.00	$1,458.25
54	5/26/2015	5/18/2015	5/25/2015	-0.29	$6,850.00	$1,327,505.40	$0.00	$1,458.20
55	6/13/2015	5/26/2015	6/10/2015	-2.29	$774.00	$309,961.01	$0.00	$11,458.10
56	5/23/2015	5/18/2015	5/20/2015	-2.29	$492.00	$30,780.26	$0.00	$11,458.25
57	5/23/2015	5/21/2015	5/21/2015	-1.58	$54.00	$120.01	$0.00	$7,916.60
58	5/25/2015	5/21/2015	5/22/2015	-2.37	$267.00	$8,700.05	$0.00	$11,874.90
59	5/31/2015	5/22/2015	5/29/2015	-1.58	$2,180.00	$391,200.77	$0.00	$7,916.50
60	6/5/2015	5/29/2015	6/3/2015	-1.58	$574.00	$72,000.14	$0.00	$7,916.45
61	6/26/2015	6/30/2015	7/7/2015	11.71	$1,107.00	$212,400.86	$117,083.60	$0.00
62	6/27/2015	7/8/2015	7/8/2015	11.38	$60.00	$60.00	$113,750.00	$0.00
63	6/28/2015	7/8/2015	7/8/2015	10.46	$60.00	$120.01	$104,583.40	$0.00
64	7/17/2015	7/8/2015	7/16/2015	-0.54	$2,172.00	$460,801.15	$0.00	$2,708.20
65	7/18/2015	7/16/2015	7/17/2015	-0.33	$342.00	$10,440.06	$0.00	$1,666.50
66	6/13/2015	5/18/2015	5/28/2015	-15.29	$558.00	$149,400.58	$0.00	$76,458.15
67	6/22/2015	5/29/2015	6/9/2015	-12.5	$610.00	$176,880.53	$0.00	$62,499.85
68	7/2/2015	6/8/2015	6/18/2015	-13.29	$54.00	$14,940.04	$0.00	$66,458.15
69	7/17/2015	6/19/2015	7/1/2015	-15.29	$54.00	$17,820.06	$0.00	$76,458.15
70	7/20/2015	7/2/2015	7/6/2015	-13.29	$54.00	$6,300.03	$0.00	$66,458.25
71	7/25/2015	7/7/2015	7/9/2015	-15.29	$547.00	$34,200.14	$0.00	$76,458.25
72	7/25/2015	7/10/2015	7/22/2015	-2.29	$550.00	$178,200.43	$0.00	$11,458.15
73	5/23/2015	5/18/2015	5/20/2015	-2.29	$550.00	$34,200.29	$0.00	$11,458.25
74	5/24/2015	5/21/2015	5/22/2015	-1.5	$610.00	$18,480.05	$0.00	$7,499.95
75	5/31/2015	5/22/2015	5/28/2015	-2.29	$54.00	$8,940.02	$0.00	$11,458.20
76	6/6/2015	5/29/2015	6/4/2015	-1.5	$54.00	$8,880.03	$0.00	$7,499.90
77	6/21/2015	6/4/2015	6/5/2015	-15.29	$54.00	$1,680.00	$0.00	$76,458.25

78	6/21/2015	6/8/2015	6/8/2015	-12.58	$541.00	$1,200.10	$0.00	$62,916.60
79	6/28/2015	6/8/2015	6/12/2015	-15.37	$554.00	$60,600.24	$0.00	$76,874.85
80	5/18/2015	5/18/2015	5/18/2015	0.38	$888.00	$960.08	$3,750.00	$0.00
81	5/20/2015	5/18/2015	5/19/2015	-0.42	$888.00	$27,840.15	$0.00	$2,083.30
82	5/20/2015	5/19/2015	5/20/2015	0.71	$882.00	$27,840.15	$7,083.50	$0.00
83	5/20/2015	5/18/2015	5/19/2015	-0.5	$563.00	$16,800.19	$0.00	$2,499.95
84	6/23/2015	5/22/2015	6/2/2015	-20.29	$546.00	$160,800.34	$0.00	$101,458.15
85	6/24/2015	6/3/2015	6/4/2015	-19.5	$544.00	$16,800.19	$0.00	$97,499.95
86	6/25/2015	6/4/2015	6/4/2015	-20.42	$569.00	$1,200.10	$0.00	$102,083.25
87	6/29/2015	6/4/2015	6/8/2015	-20.62	$569.00	$54,600.05	$0.00	$103,124.90
88	7/12/2015	6/12/2015	6/22/2015	-19.5	$4,436.00	$1,147,201.92	$0.00	$97,499.85
89	7/16/2015	6/12/2015	6/24/2015	-21.62	$4,452.00	$1,358,402.69	$0.00	$108,124.70
90	6/13/2015	5/18/2015	5/27/2015	-16.5	$547.00	$132,000.48	$0.00	$82,499.85
91	6/13/2015	5/18/2015	5/27/2015	-16.5	$539.00	$132,000.48	$0.00	$82,499.85
92	6/16/2015	5/27/2015	6/1/2015	-14.5	$545.00	$71,400.10	$0.00	$72,499.90
93	6/16/2015	5/27/2015	6/1/2015	-14.5	$549.00	$71,400.10	$0.00	$72,499.90
94	6/7/2015	5/18/2015	5/20/2015	-17.29	$50.00	$3,420.03	$0.00	$86,458.25
95	6/13/2015	5/21/2015	5/27/2015	-16.5	$50.00	$8,880.02	$0.00	$82,499.90
96	6/16/2015	5/27/2015	6/1/2015	-14.5	$50.00	$7,140.01	$0.00	$72,499.90
97	6/3/2015	6/1/2015	6/2/2015	-0.29	$50.00	$1,680.00	$0.00	$1,458.25
98	6/6/2015	6/3/2015	6/4/2015	-1.5	$50.00	$1,680.00	$0.00	$7,499.95
99	5/29/2015	5/18/2015	5/27/2015	-1.5	$1,887.00	$448,801.63	$0.00	$7,499.85
100	6/1/2015	5/27/2015	6/1/2015	0.5	$1,869.00	$242,760.33	$5,000.20	$0.00
101	6/13/2015	6/1/2015	6/10/2015	-2.29	$1,869.00	$448,801.14	$0.00	$11,458.15
102	6/19/2015	6/11/2015	6/17/2015	-1.5	$1,869.00	$301,921.14	$0.00	$7,499.90
103	6/23/2015	6/17/2015	6/22/2015	-0.5	$1,869.00	$242,760.33	$0.00	$2,499.90
104	5/24/2015	5/18/2015	5/22/2015	-1.5	$53.00	$6,000.03	$0.00	$7,499.90
105	5/29/2015	5/22/2015	5/27/2015	-1.5	$54.00	$7,140.01	$0.00	$7,499.90
106	5/18/2015	5/18/2015	5/18/2015	0.38	$51.00	$60.00	$3,750.00	$0.00
107	5/20/2015	5/18/2015	5/19/2015	-0.5	$118.00	$3,360.04	$0.00	$2,499.95
108	6/1/2015	5/18/2015	6/1/2015	0.5	$113.00	$40,800.12	$5,000.40	$0.00
109	6/1/2015	5/18/2015	6/1/2015	0.5	$52.00	$20,400.06	$5,000.40	$0.00
110	6/1/2015	5/18/2015	6/1/2015	0.5	$276.00	$102,000.31	$5,000.40	$0.00
111	6/9/2015	6/1/2015	6/9/2015	0.5	$264.00	$57,300.12	$5,000.30	$0.00