Optimization Model For Multiple Project Management

Course Title: Operations Research
Course Number: ETM 540/640
Instructor: Dr. Tim Anderson
Term: Winter 2017
Date: March 26th, 2017

Team Members:
Aruna Bhatia
Gaya Haciane
Juchun Cheng
Sivagamasundari Sathappan
Tejas Deshpande

Table of Contents

List of Tables

List of Figures

1 Executive Summary

In any company, there are plenty of factors which can move a project past its deadline. It's very common for some project tasks (Activities) to be complex than anticipated or to have turnover on the project that requires to bring new resources up to speed. Sometimes, many project activities are simply underestimated and they end up taking more time than predicted. Regardless of how it happens, most of the times managers discover that projects are trending beyond committed deadlines. If it’s still the beginning of the project, some corrective actions could be taken to bring project back on track. But, closer to project deadline, choices dwindle.

In spite of such situations, project deadlines can’t be easily delayed or postponed as most of the projects in a company are interdependent in nature while sharing resources to maintain economies of scale. Changing customer demands or pressure of stiff competition in the market also force managers to get the project done before deadline.In such cases, companies prefer the option of ‘Project Crashing’ which simply means applying additional resources to the critical path or the sequence of activities that must be completed in required schedule. It's always possible to just throw more resources on the critical path, but crashing also means to get the biggest schedule gain for the least amount of incremental costs.

To study the issue of Project Crashing and high crashing cost related to it, we examine the case of Siemens Energy Inc. Siemens Energy Inc. is a division of a large European company Siemens AG which has operations in manufacturing and electronics worldwide. Similar to all companies, Siemens Inc. also deals with the issue of managing multiple projects in a given period of time. Management at Siemens Inc. would like to prefer developing a standardized method or model that can help managers to make informed and rational decision which can help them to accomplish all the project activities in time with the least extra cost incurred.

To begin with, we reviewed past literature available on Project Management. Most of them mainly focus on approaches used for schedule compression. In addition, we also reviewed past literature on optimization methods used for project crashing. Understading the past researches and the need of a robust and widely applicable optimization model for this classic but important problem, an attempt was made to address this Project Management. Hence, a model was developed using Linear Programming and Excel Solver which aims at meeting the project deadline with minimum cost.

2 Introduction

This project is conducted for Siemens Energy Inc.. Siemens Energy Inc. is a subsidiary of Siemens AG, the largest manufacturing and electronics company in Europe with branch offices abroad. Along with Energy, Industry, Healthcare, and Infrastructure & Cities are some principle division of the company. Siemens Energy is the world’s leading supplier for the generation, transmission and distribution of power. Known for innovation, excellence and responsibility, Siemens leadership in the increasingly complex energy business makes it a first-choice supplier for customers around the world. To achieve such brand reputation, company is continuously working towards efficiency and effective allocation of resources [1][2].

Considering the complex nature of business of the company, changing schedules according to customer demand and multiple activities getting managed with shared resources make it difficult to achieve target in time in case of most of the projects. Task dependencies included in almost all projects increase the complexity of this problem. Similar to many other companies, managers of Siemens Energy Inc. deal with the major challenge related to completion of projects before deadline. Due to complexity of tasks/activities related to project and least flexibility of time resource, delivering projects on time, or before time becomes difficult. For a company like Siemens, which has operations and business practices carried out worldwide, managing schedules according to different time zones and completing target activities in time have been the most challenging task.

Focusing on increasingly common situation in companies, where managers handle several projects simultaneously, some of these projects share resources in terms of time, employees, materials etc. The prospect of keeping multiple projects on track, as well as successfully managing several project teams at the same time definitely add to the potential for multiple project failures. Also, for most of the projects, several tasks and activities are inter-dependent. Hence, delays, overruns, or other unanticipated changes to a single project's schedule impact all related projects also. Shared resource pool makes it more difficult in nature [3][4].

In spite of all these complex situations in project management, target finishing date can't be postponed or delayed, company has to arrange additional resources to get it done in required time. In some situations, company has to prepone the deadline as well due to customer demand or anticipated changes in near future. In such circumstances, companies apply 'project crashing', that is spending more money to get project/tasks done more quickly than decided [5][6].

Here in case of Siemens Energy Inc. projects, a model if being proposed which could be used as standardized approach for meeting schedule using Project Crashing with minimum crashing cost.

3 Project Objective

In spite of any unanticipated change, it is impossible for the company to extend the deadline of any project in most of the cases. It not only affects the performance of particular project but several other projects can also get affected due to inter-dependencies between them. Hence, to finish projects on time or sometimes before time, company has to allocate additional resources bearing extra cost.
Siemens Energy Inc. managers come across such situations multiple times, when they have to spend more money to get the project done more quickly. In other words, they have to apply 'Project crashing'. The key challenge to project crashing is to attain maximum reduction in schedule time with minimum cost

Hence, our objective is to deal with this challenge and propose standardized approach to optimize 'crashing cost'. The project aims to develop an optimization model which could be used to optimize multiple project management processes, with the focus on schedule optimization.

The model aims to minimize the crashing cost and ultimately the 'Total cost' with maximum reduction in schedule time estimated by management.

4 Literature Review

A project is a collection of activities/tasks to accomplish a specific objective within a defined deadline and budget. Project management comprises of scope, time, cost, risk, resource, quality, communication and integration management.[10] Projects managers have to deal with and also, the fierce competition in the markets today, make it important to complete a project within a deadline and sometimes even earlier. The constant competition amongst different project activities for available resources makes it even necessary to finish each and every activity in time. Inter-dependancies of the projects, activities also require all activities to meet individual deadlines alongwith the final project deadline. Failure to do so can cost a business in many ways.

Inspite of all such situations, expediting a project becomes inevitable which is when the concept of project crashing comes into play [7]. Project crashing requires completing some of the activities earlier than the normal completion time. Critical path method determines the minimum time required to complete the project and time required to complete each activity. Crash time is the minimum time required to accomplish the activity with additional resources or extra effort which results in overtime charges. Hence, crashing an activity involves increase in cost to attain decrease in time. But, it's not easy to just add resources and finish the activity before time. It should be done at minimum possible cost. When manager needs to expedite a project, he needs to determine which activity should be crashed to meet the new deadline at minimum cost.

Schedule compression is an important technique for schedule control in project management plan. Kathy Schwalbe states that its very common to have unreasonable schedules for projects in the company. Hence, it is the responsibility of managers and team members to evaluate and accept onlyrealistic schedule which can meet deadlines. Kathy Schwalbe also states that, though there are multiple techniques which help managers to develop a realistic schedule, resolving human related issues becomes equally important while meeting project deadlines [12].

The most common software used by managers for project management is MS Project. Siemens Inc. also uses the same software to manage projects and deadlines. MS Project is useful to provide all information related to project management such as list of activities, cost for each activity, resources corresponding to each activity, start and end date of an activity, critical path, total project completion cost. But, when it comes to project crashing, MS Project is not that helpful as it doesn't deal with time and cost management. Not only MS project but existing management softwares like SAS/OR Project Management, Primavera Cost Manager, Quick Gantt, Project Simulation Games, etc also are in capable of solving the cost- time optimization problems [10].

Helena Gaspers suggests the way of using sensitivity analysis generated by MS Excel solver to make time cost simulations[10]. In addition to that, Nhat-Duc Hoang statesthat linearity assumptions can be relaxed in case of time-cost function of an activity to take different forms such as concave, convex, quadratic and hybrid of concave and convex [13].

Traditionally, trial and error approach was used to demonstrate the logic of crashing. While doing it manually, the activities to be crashed are selected and then they are crashed one at a time. This can sometimes change the critical path altogether. Thus, when the project size increases and the number of activities increase making it cumbersome, inefficient and unmanageable as well. Linear programming can be used in this area of project management to come up with an optimal solution to this problem [7]. AnaghaKatti and Milind Daradealso used linear solver to crash project to deal with time cost trade off. The main objective of this paper was to minimize the cost and determine the number of days each activity can be crashed for. Their model determined the number of days each activity can be crashed for but could not come up with the early start time after crashing [11]. K Li, B Shao and P Zelbst used AON (acitivity – on- node) network approach to find an efficient way to address this problem. In their model they basically tried to compress the path which takes maximum time to complete in order to meet the schedule withing a given time. This approach however has a limitation that the manager would have to know and identify all the paths in te network and will have to calculate each path's duration in order to find a solution [8].

5 Data collection

Each project of Siemens Energy product is comprised of a different combination of subcomponents which includes A, B, C, …to H. Thus, the schedule and cost of each project is based on the configuration and quantity of its subcomponents. Because the required resources used in each subcomponents are shared, the dependency between each project is related to when the subcomponents are completed for each project. Table 1 shows the examples of projects in a quarter timeframe.

Table 1 Precedence Relationship within each Project for Network 1

Project 1	
Activity	Immediate Predecessor(s)
A	-
B	A
C	B
H	C

Project 2	
Activity	Immediate Predecessor(s)
A	-
B	A
C	B
D	C
E	D
F	E
G	F
H	G

Project 3	
Activity	Immediate Predecessor(s)
A	-
B	A
C	B
G	C
H	G

Furthermore, we collected the information of the normal work and crashing work of each subcomponent activity as below:

Table 2 Cost and Duration Data for Projects in Network 1

Activity	Normal Duration (day)	Crash Duration (day)	Normal Cost ($/day)	Crash Cost ($/day)
A	5	4	400	650
B	7	5	400	650
C	6	6	400	400
D	8	5	400	650
E	9	7	400	650
F	10	8	400	650
G	7	5	300	500
H	3	3	300	300

Based on the dependency of shared resource between the activities, an AON network to see the relationship among these 3 projects was made. Activities in project 1 as AP1, BP1, CP1 and HP1 to show the resources used. Same rule was applied to the activities in project 2 and project 3. Finally the AON network was made to include all three projects. Figure 1 shows the AON network of project 1,2, and 3 together and which has been named as Network 1. This was done keeping in mind that the deadline of these projects together is 55 days and the activities amongst the projects are sharing resources, so they all have to be completed together in 55 days as these have to feed in to a larger project in the future. Two such networks have been used in the study to see the working of the model with more than 1 networks and also so that the model can be generalized.

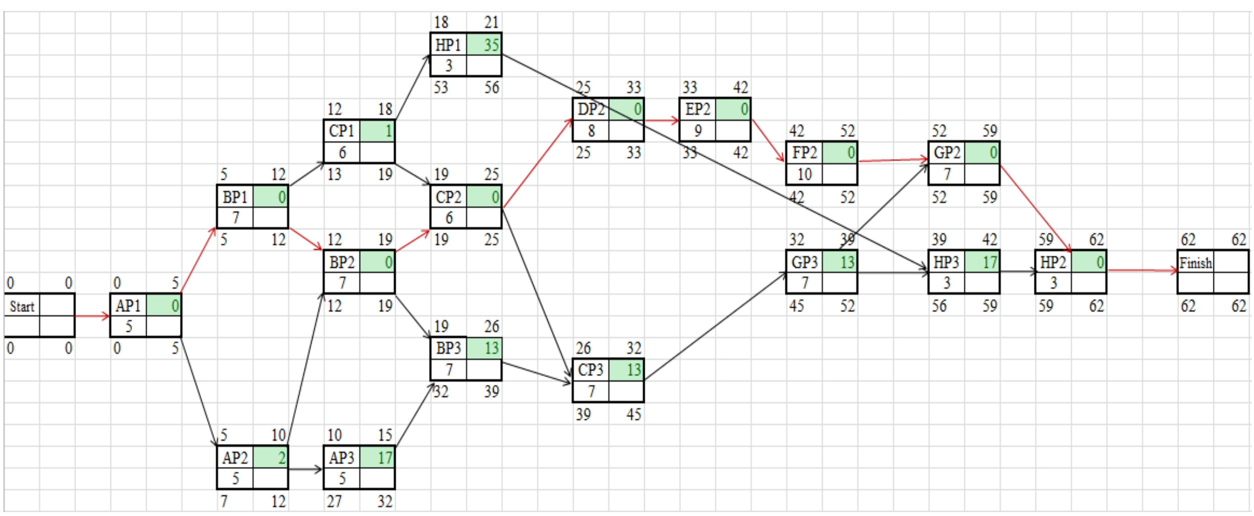

Figure 1 AON (Activity on node) Network 1

Similarily, Project 4, 5, and 6 formed Network 2, which is also used as an input data to the model. The precendence relationship for these projects have been detailed in Table 3 and the normal work and crash work related data is given in Table 4. Again, the deadline for these projects is 77 days and the activities amongst the projects are sharing resources, to show this an AON network has been developed as shown in Figure 2.

Table 3 Precedence Relationship within each Project for Network 2

Project 4

Activity	Immediate Predecessor(s)
A1	-
B1	A1
C1	B1
E1	C1
H1	E1

Project 5

Activity	Immediate Predecessor(s)
A1	-
B1	A1
C1	B1
D1	C1
E1	D1
F1	E1
G1	F1
H1	G1

Project 6

Activity	Immediate Predecessor(s)
A1	-
B1	A1
C1	B1
G1	C1
H1	G1

Table 4 Cost and Duration Data for Projects in Network 2

Activity	Normal Duration (day)	Crash Duration (day)	Normal Cost ($/day)	Crash Cost ($/day)
A1	10	8	600	775
B1	9	7	600	775
C1	8	7	600	775
D1	5	5	500	500
E1	12	12	500	500
F1	7	7	500	500
G1	11	10	500	715
H1	9	9	450	450

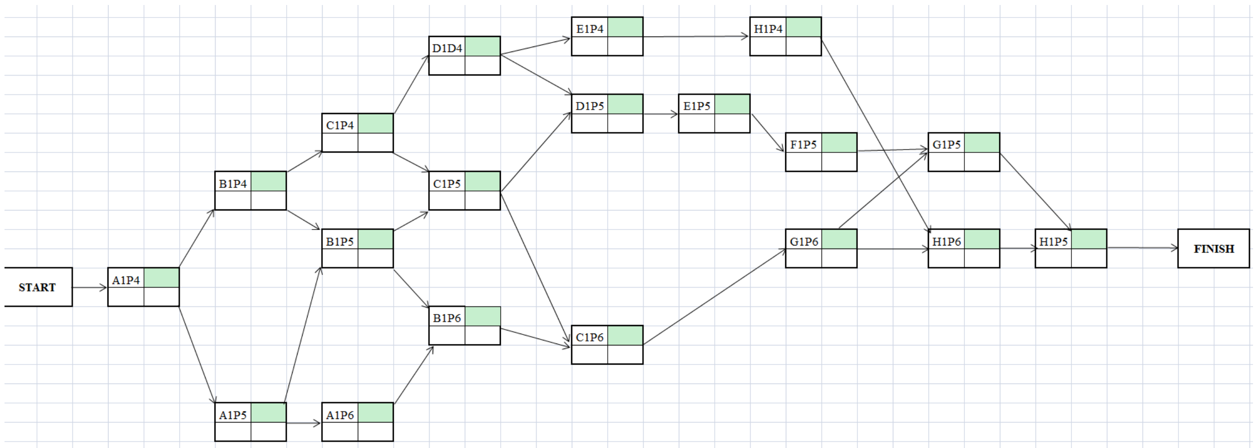

Figure 2 AON (Activity on node) Network 2

6 Methodology

For Siemens Energy Inc, we built a model to determine the minimum cost required to complete multiple projects within the new defined deadline. New deadline is earlier than the original deadline. We built a model for crashing two networks. Network 1 has 17 activities and Network 2 has 19 activities. Model determines the number of days each activity can be crashed at a minimum cost. It also determines the early start time of each activity after crashing. This optimization reduces the effort and manual work required by the managers to build a schedule as linear formulations are easy for solver to solve the problem. Global optimal solution can also be identified using linear solver.

6.1 Mathematical Formulation

Assumptions:

1. Few activities are using same resources within a Project.
2. Resources used in Network1 are different than the resources being used in Network2.
3. Activity can be crashed only for a full day not partial day.

Parameter:

C_i = Cost of crashing activity i per day
M_i = Maximum number of days activity i can be crashed
E_s = Early Start Time of the successor of activity i
N_i = Normal duration of activity i
S_n = Max Early Start Time of last activity in the network

Decision Variable:

Network 1 has 17 activities and Network 2 has 19 activities. It determines the number of days activity i is crashed (X_i) in each project and the early start time of activity i (E_i) after crashing. $X_i$and $E_i$are integer variables. It should be greater than or equal to 0.

In total, we have 72 decision variables as each activity has number of days crashed and its early start time as the decision variable.

Objective:

The goal of the project is to meet the project deadlines at the minimum cost.

$$Min \sum_{i=1}^{36} CiXi$$

Total minimum cost required to crash Network 1 and Network 2 is a result of multiplying the number of days activity i is crashed (X_i) and cost of crashing an activity i per day (C_i). Total number of activities in Project A and Project B is 36.

Constraints:

Maximum Days an Activity i can be crashed

$$Xi \leq Mi \qquad \forall\, i, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (1)$$

Number of days activity i can be crashed (X_i) should be less than or equal to maximum number of days activity i can be crashed(M_i). Maximum number of days activity i can be crashed is the normal duration of activity i (N_i) minus crash duration of activity i (D_i). This constraints assures that no activity can be crashed beyond its maximum limit.

Early Start Time Constraint for activity i

$$Xi - Ei + Es \geq Ni \qquad \forall\, i, s, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Early start time of successor of activity i (E_s) should be greater than or equal to start time for activity i(E_i) plus the normal time for activity i (N_i) minus the amount by which activity i is crashed (X_i). Early start constraints corresponds to precedence relationship in each project network. This constraints guarantees that each successor activity is only started after the activity before it is completed.

Early Start Time Constraint for last activity i in the AON

$$Ei - Xi \leq Sn \qquad \forall\, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots .. (3)$$

Early start time for the last activity i (E_i) in each project minus the number of days this activity is crashed (X_i) should be less than or equal to the max early start time of the last activity(S_n). This constraint is to minimize the completion time of the project.

Integrality

$$Xi \in Integer \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

The number of days activity i is crashed (X_i) and early start time of activity i (E_i) should be an integer. Activity can be crashed for whole day and not partial day.

Non- negativity

$$Xi \geq 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots . (5)$$

The number of days activity i is crashed (X_i) and early start time of activity i (E_i) should be non negative (i.e. Greater than or equal to 0)

6.2 Linear Programming Model

Network 1 comprises of 17 activities and Network 2 comprises of 19 activties. Decision variables are the number of days each activity in Network 1 and Network 2 are crashed and Early Start Time for each activity in Network 1 and Network 2, there are 72 decision variable in total. The critical path for Network 1takes 62 days to complete but now it needs to be completed early in 55 days. Similarly, critical path of Network 2 takes 83 days to complete and it needs to be completed in 77 days now. The objective is to crash the project schedule at minimum cost. To meet the schedule for the new deadline we added the following constraints:

$$Xi \leq Mi \qquad \forall\, i, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots . (1)$$

$$Xi - Ei + Es \geq Ni \qquad \forall\, i, s, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

$$Ei - Xi \leq Sn \qquad \forall\, n \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots .. (3)$$

Excel solver was run after adding these constraints, the non- negativity and integer constraints to get to the following results.

Decision Variable

Table 5 shows the numbers of days an activity is crashed in Network 1. The activities those are crashed have a value greater than 0 and the activities which are not crashed have a value of 0. Similarly, Table 6 shows the numbers of days an activity is crashed in Network 2. The activities those are crashed have a value greater than 0 and the activities which are not crashed have a value of 0. In addition, Table 7 and Table 8 shows the Early Start Time of each activity after crashing in Network 1 and Network 2, respectively.

Table 5 Network 1 Days Crashed

DAYS CRASHED - NETWORK 1																
AP1	BP1	CP1	HP1	AP2	BP2	CP2	DP2	EP2	FP2	GP2	HP2	AP3	BP3	CP3	GP3	HP3
1	2	0	0	0	0	0	0	0	2	2	0	0	0	0	0	0

Table 6 Network 2 Days Crashed

DAYS CRASHED - NETWORK 2																		
A1P4	B1P4	C1P4	D1P4	E1P4	H1P4	A1P5	B1P5	C1P5	D1P5	E1P5	F1P5	G1P5	H1P5	A1P6	B1P6	C1P6	G1P6	H1P6
2	2	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0

Table 7 Network 1 Early start time

EARLY START TIME - NETWORK 1																
AP1	BP1	CP1	HP1	AP2	BP2	CP2	DP2	EP2	FP2	GP2	HP2	AP3	BP3	CP3	GP3	HP3
0	4	10	44	4	9	16	22	30	39	47	52	22	27	34	40	47

Table 8 Network 2 Early start time

EARLY START TIME - NETWORK 2																		
A1P4	B1P4	C1P4	D1P4	E1P4	H1P4	A1P5	B1P5	C1P5	D1P5	E1P5	F1P5	G1P5	H1P5	A1P6	B1P6	C1P6	G1P6	H1P6
0	8	15	22	27	39	8	17	26	34	39	51	58	68	17	27	39	47	59

Objective Function

Objective function is the total crash cost for crashing each activity inNetwork 1 and Network 2. Total crash cost is minimized.

Project 1

	DAYS CRASHED - NETWORK 1																
ACTIVITY	AP1	BP1	CP1	HP1	AP2	BP2	CP2	DP2	EP2	FP2	GP2	HP2	AP3	BP3	CP3	GP3	HP3
	1	2	0	0	0	0	0	0	0	2	2	0	0	0	0	0	0
CRASH COST/DAY	650	650	400	300	650	650	400	650	650	650	500	300	650	650	400	500	300

Project 2

DAYS CRASHED - NETWORK 2																		
A1P4	B1P4	C1P4	D1P4	E1P4	H1P4	A1P5	B1P5	C1P5	D1P5	E1P5	F1P5	G1P5	H1P5	A1P6	B1P6	C1P6	G1P6	H1P6
2	2	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
775	775	775	500	500	450	775	775	775	500	500	500	715	450	775	775	775	715	450

9615

Figure 3 Objective Function Value

Figure 3 shows the value of the objective function which was calculated as the SUMPRODUCT of the crash cost per day and the number of days each activity is crashed. The total cost of crashing both the networks is $9,615.

7 Results and Analysis

Network 1 comprised of 17 interdependent activities. Normal completion time for this project was 62 days and its normal cost was $39,300. In the optimization of this problem solver simplex engine was run to minimize the additional cost incurred by crashing whilst embedding crashing days' limits into the constraints. The solver returned an improvement (decrease of completion time) by a week (7 days) and crashing cost of $4,250. The number of crashed activities was four. Activities AP1, BP1, FP2 and GP2 were crashed. AP1 was crashed by 1 day and BP1, FP2 and GP2 were crashed by 2 days each.

Network 2comprises of 19 interdependent activities. Normal completion time for this project was 83 days and its normal cost was $92,250. In the optimization of this problem solver simplex engine was run to minimize the additional cost incurred by crashing whilst embedding crashing days' limits into the constraints. The new completion time for this project is 77 days (6 days less) and crashing cost is$5,365. The number of crashed activities was five. Activities A1P4, B1P4, C1P4, A1P5 and G1P5 were crashed. A1P4 and B1P4 were crashed by 2 days each and C1P4, A1P5 and G1P5 were crashed by 1 day each. Table 9 summarizes the results:

Table 9 Summary of Results

Completion Time Network 1 (After crashing/Before Crashing)	55 /62
Crash Cost Network 1	$4,250
Normal Cost Network 1	$39,300
Total Cost of Network 1	$43,550
Completion Time Network 2 (After Crashing/Before Crashing)	77/ 83
Crash Cost Network 2	$5,365
Normal Cost of Network 2	$92,250
Total Cost of Network 2	$97,615
TOTAL COST	$141,165

Total number of days to complete Network 1 and Network 2 is 146. This was compressed to 133 days as a result of crashing which costs $9615. The cost of crashing increases with the increase in the number of crash days. Further, cost-time tradeoff is evaluated by plotting a graph using the minimum crash cost corresponding to different completion time for each Network.The corresponding graphs for Networks 1 and 2 are shown in figures 4 and 5 respectively. This kind of information can be valuable for managers as it can help them understand and evaluate the time- cost trade off and eventually enable them in making well informed, timely and rational decisions when needed.

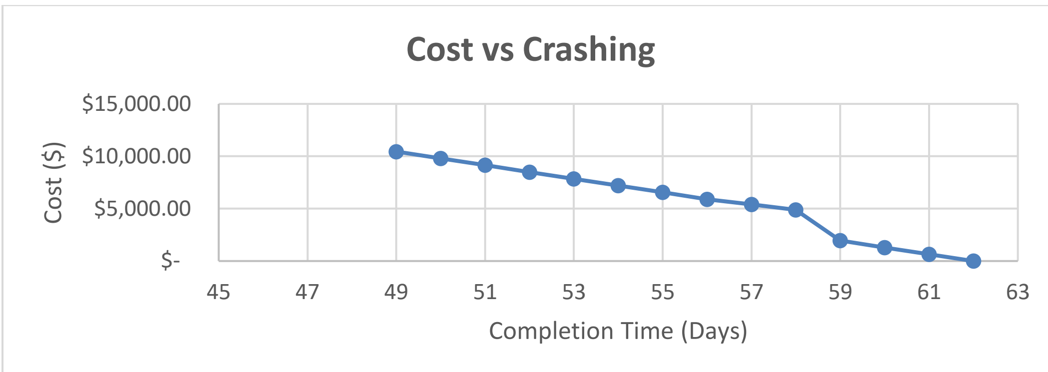

Figure 4 Network 1 – Relationship between Crash days and Crash cost

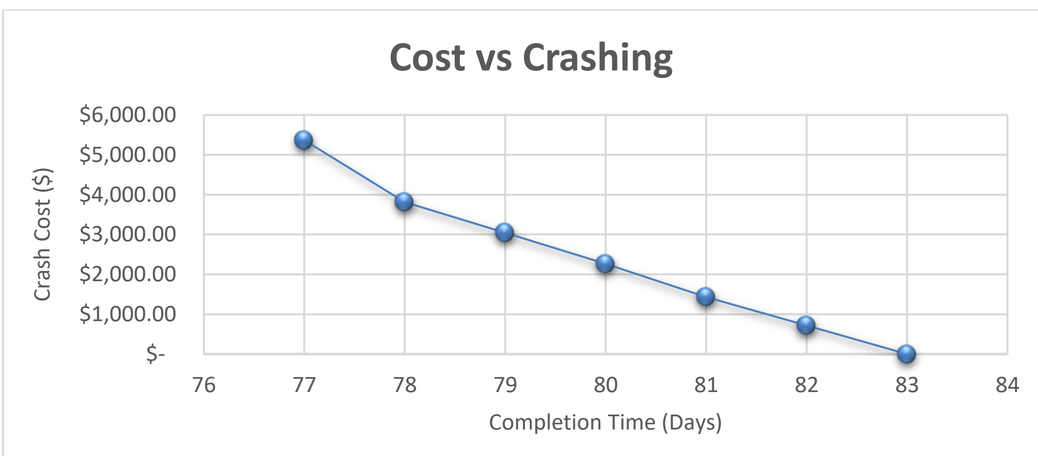

Figure 5 Network 2 – Relationship between Crash days and Crash cost

This paper proposes an efficient Linear programming Model which is robust in nature as it can be used for small as well as large projects in any company. Though, AnaghaKatti and Milind Darade used MS excel solver to crash schedule with minimum cost, they did not propose an LP model for the same. They mainly focused on extracting data from MS Project into MS Excel solver and solving for a solution. This type of approach can be difficult to understand as well as hard to generalize. It can also be difficult to extend this approach to a larger or different project without and LP Model.

The model being proposed in this paper can be used to large projects with multipe activities easily, as it does not requires manager to know and calculate the duration of each path in the network to meet the constraint requirement of the model. As it can be seen in the model proposed by Kunpeng Li, Bin Shao and Pamela Zelbst, it is required to have hard coding of all the paths in the AON. In the model being proposed in this paper, manager is only required to know the durations of the activity, crash duration, cost related to these activities and the precedence realationships amongst the activities. Thus, it qualifies to be a simpler and efficient LP model.

8 Conclusion

Efficient project management is essential to success and growth of any company. The growing competition, uncertainity and agility of the markets today often require company's to complete projects ahead of time, or meet the schedule requirements even after facing delays. This requires effective management of projects so as to minimize the overall cost incurred not only by the projects but cost by the business overall as these delays can cost your market share and eventually growth. Keeping the importance of this problem in mind, an optimization model is proposed which is simple, robust and efficient at the same time, achieving maximum gain in crashing schedule at minimum possible crashing cost.

Currently, Siemens Inc. uses the software of MS Project for managing different projects, but when it comes to Project Crashing, its still done with manual approach providing no guarantee that project crashing is done with the minimum possible cost. This model can help them to a great extent. The model presented by this paper could help project management teams of Siemens Inc. crash projects in an efficient way while giving them the opportunity to work on as many projects as they wish simultaneously. In addition, the early start time information in this model used by managers to track important milestones.

Furthermore, an attempt has been made to show the time- cost trade off using graphs which can help managers evaluate the impact of crashing on the cost of the project and always way in other related factors to make an informed and rational decision backed up with data and facts. Additionally, this model can be upgraded and extended for various sizes and types of projects. The optimization model developed in this paper can serve as a viable and efficient approach for compressing schedule with minimum cost.

9 Limitation & Scope for Future Research

The optimization model developed for this project is used to reduce crashing cost while attaining maximum schedule compression. Because of the limited time in hand to complete this project, the only focus of this project was 'crashing' which can be a costly affair. In future, this project can be extended to include fast tracking in the model. Fast tracking of the activities also serves the purpose of achieving early scedules with no added cost. In addition, detailed information about resources used for an activity can provide insights to develop an Optimization model and to identify which activities should be fast tracked and which should be crashed to meet schedule requirements. This model would be more effective in reducing cost as it will also consider fast tracking of the activities which doesn’t incur extra cost.

The linear programming model has been solved using an MS excel Solver in this paper, in future this model can also be solved using OMPR (Optimization Modeling with R). A programme can be written in R to implement the model, which will make it even more robust and give it a platform efficiency.

10 References

[1]“Siemens Energy at a Glance", 2017. https://w5.siemens.com/web/il/en/corporate/home/Siemens_Israel/Energy/Documents/1.pdf.

[2]"Siemens Company Official Website", 2017. [Online]. Available: http://www.siemens.com/us/en/home.html/.

[3] T. Mochal, "10 ways to get a slipping project back on track - TechRepublic", *TechRepublic*, 2008. [Online]. Available: http://www.techrepublic.com/blog/10-things/10-ways-to-get-a-slipping-project-back-on-track/.

[4] R. F. Deckro and J. E. Hebert, "Resource constrained project crashing," *Omega*, vol. 17, no. 1, pp. 69-79, 1989.

[5] "A Quick Guide to Crashing a Project Schedule", Brighthub Project Management, 2015. [Online]. Available: http://www.brighthubpm.com/resource-management/5055-project-crashing-what-options-do-you-have/.

[6] A. Babu and N. Suresh, "Project management with time, cost, and quality considerations," *European Journal of Operational Research*, vol. 88, no. 2, pp. 320-327, 1996.

[7] J. Meredith and S. Mantel Jr, *Project management*, 8th ed. Hoboken, N.J.: John Wiley & Sons, 2012.

[8] K. Li, B. Shao and P. Zelbst, "Project Crashing Using Excel Solver: A Simple AON Network Approach", *International Journal of Management & Information Systems (IJMIS)*, vol. 16, no. 2, pp. 177-182, 2012.

[9] C. Ragsdale, *Spreadsheet Modelling & Decision Analysis*, 1st ed. 1996.

[10] H.Gaspars-Wieloch, "Time- cost project management with solver", *Contemporary issues in Business, management and education*, 2012.

[11] A. Katti and M. Darade, "Project crashing to solve Time-Cost Trade-Off", *SSRG International Journal of Civil Engineering (SSRG-IJCE)*, vol. 3, no. 1, 2016.

[12] K. Schwalbe, Information technology project management, 1st ed. 1999.

[13] N. Hoang, "NIDE: A Novel Improved Differential Evolution for Construction Project Crashing Optimization", *Journal of Construction Engineering*, 2014.

11 Appendix

Table 10 Calculations for Network 1

Normal Time	Crash Time	Max crash Days	Activity	Immediate Predecessor(s)	Normal Duration (day)	Early start Times	
5	4	1	AP1	-	5	AP1->BP1	5
7	5	2	BP1	AP1	7	BP1->CP1	7
6	6	0	CP1	BP1	6	CP1->HP1	6
3	3	0	HP1	CP1	3	AP1->AP2	5
5	4	1	AP2	AP1	5	BP1->BP2	7
7	5	2	BP2	BP1, AP2	7	AP2->BP2	5
6	6	0	CP2	CP1, BP2	6	CP1->CP2	6
8	5	3	DP2	CP2	8	BP2->CP2	7
9	7	2	EP2	DP2	9	CP2->DP2	6
10	8	2	FP2	EP2	10	DP2->EP2	8
7	5	2	GP2	FP2,GP3	7	EP2->FP2	9
3	3	0	HP2	GP2,HP3	3	FP2->GP2	10
5	4	1	AP3	AP2	5	GP3->GP2	7
7	5	2	BP3	BP2, AP3	7	GP2->HP2	7
6	6	0	CP3	CP2, BP3	6	HP3->HP2	3
7	5	2	GP3	CP3	7	AP2->AP3	5
3	3	0	HP3	HP1	3	AP3->BP3	5
						BP2->BP3	7
						BP3->CP3	7
						CP2->CP3	6
						CP3->GP3	6
						GP3->HP3	7
						HP1->HP3	3

Table 11 Calculations for Network 2

Activity	Immediate Predecessor(s)	Normal Duration	Crash Duration	Max crash days	EST	
A1P4		10	8	2	A1P4-->B1P4	10
B1P4	A1P4	9	7	2	B1P4-->C1P4	9
C1P4	B1P4	8	7	1	C1P4-->D1P4	8
D1P4	C1P4	5	5	0	D1P4-->E1P4	5
E1P4	D1P4	12	12	0	E1P4-->H1P4	12
H1P4	E1P4	9	9	0	A1P4-->A1P5	10
A1P5	A1P4	10	8	2	A1P5-->B1P5	10
B1P5	B1P4, A1P5	9	7	2	B1P4-->B1P5	9
C1P5	C1P4, B1P5	8	7	1	B1P5-->C1P5	9
D1P5	D1P4, C1P5	5	5	0	C1P4-->C1P5	8
E1P5	E1P4, D1P5	12	12	0	C1P5-->D1P5	8
F1P5	E1P5	7	7	0	D1P4-->D1P5	5
G1P5	F1P5, G1P6	11	10	1	D1P5-->E1P5	5
H1P5	G1P5, H1P6	9	9	0	E1P4-->E1P5	12
A1P6	A1P5	10	8	2	E1P5-->F1P5	12
B1P6	B1P5, A1P6	9	7	2	F1P5-->G1P5	7
C1P6	C1P5, B1P6	8	7	1	G1P6-->G1P5	11
G1P6	C1P6	11	10	1	G1P5-->H1P5	11
H1P6	H1P4, G1P6	9	9	0	H1P6-->H1P5	9
					A1P5-->A1P6	10
					A1P6-->B1P6	10
					B1P5-->B1P6	9
					B1P6-->C1P6	9
					C1P5-->C1P6	8
					C1P6-->G1P6	8
					G1P6-->H1P6	11
					H1P4-->H1P6	9