A System Dynamics Approach towards Design and Management for Software Development Projects

47186847 Mst. Taskia KHATUN
Human and Engineered Environmental Studies
September 2020
Supervisor: Assoc. Prof. Kazuo HIEKATA

The objective of this thesis work is to develop a method for supporting decisions for software development project management against uncertain rework and fluctuating productivity of engineers. This method would support for prediction and evaluation of process improvement of project planning, controlling based on overtime and hiring which would help to enhance our understanding of and make prediction about model-based decision making. For enhancing our understanding of project behavior, the proposed model describes causal loop diagram to represent a key dynamic, then reproduce detailed behavior and dynamics of a project by System Dynamics simulation model.

Keywords: Software Development, System Dynamics, Project Management, overtime, hiring

1 Introduction

Developing a software project has to follow a series of activities to decide the whole project scope and certain requirements including project schedule, utilization of the budget and the number of full-time equivalent workforces as well. But often problem occurs during development and project manager has to consider about taking control actions such as working overtime, or hire more people to the team. However, these options can results dynamic feedback responses. Hence, if those dynamic responses can be understood before development, a more accurate and feasible plan could be designed and planned for project development.

Considering these circumstances, the aim is to develop a decision support method that will ascertain to design feasible project plan for the improvement of project performance by predicting and understanding dynamic responses of model-based decision-making.

2 Proposed Methodology

2.1 Overview

The overview of the proposed methodology is given below.

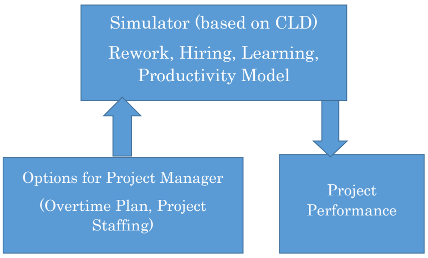

Simulator (based on CLD)
Rework, Hiring, Learning, Productivity Model
Options for Project Manager
(Overtime Plan, Project Staffing)
Project Performance

Fig. 1 Overview of the Proposed Methodology

This figure 1 represents the combination of input, output, and basic process to obtain the desired objective.

2.2 Options for Project Manager

2.2.1 Hiring Plan

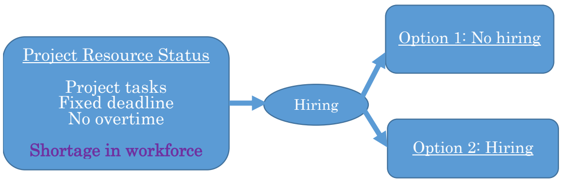


Fig. 2 Options Considerations for Hiring

While planning about hiring, based on the project resources and to get desired number of workforces to the development team, two different options can be chosen – hiring and no hiring, as shown in figure 2.

2.2.2 Overtime Management Plan

For overtime management, two plans have been considered:

(i) Interval-based Overtime: For this overtime plan, workdays with overtime has been considered within 50 days. After workdays with overtime, an average of 30 workdays without overtime is required. The nominal fraction of one workday (AFMDP) is 1 without overtime and 1.35 in average with overtime. This option corresponds to a software development project with milestones of 8-12 weeks intervals[1)].

(ii) Continuous Overtime: For this overtime plan workforces work overtime in a continuous way based on necessities until the project is finished. AFMDP is predicted 1.2 as average value for workdays with overtime since overtime is happening continuously. This option assumes an improvement project of cloud service.

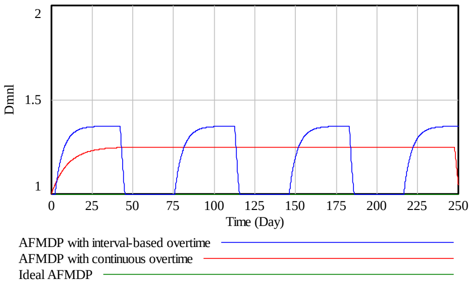


Fig. 3 The AFMDP with Overtime

2.3 Project Management Model

2.3.1 Overview in Causal Loop Diagram (CLD)

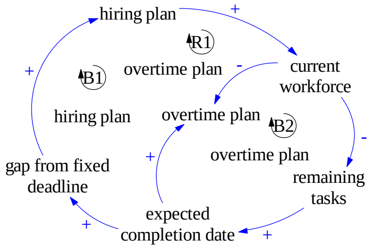


Fig. 4 CLD for Hiring and Overtime Management

The major considerations to analyze project development schemes are human resource, controlling, and planning. Considering these factors, the basic causal loop diagram (CLD) has been designed.

In figure 4, the feedback loop B1 shows hiring scheme and other loops, B2 and R1 represent the process of planning overtime.

2.3.2 Uncertainty by Rework

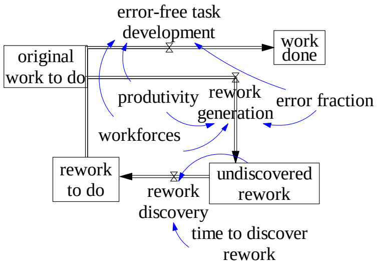
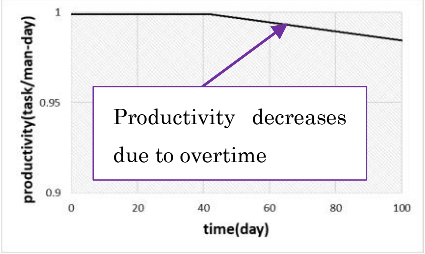


Fig. 5 Basic Rework Cycle

The rework cycle is the most important feature of the System Dynamics project model. From figure 5, it is seen that at the start of the project, all tasks are stored in the stock *original work to do*. Tasks done correctly enters to *work done* stock and never need to do rework. However, work that contains errors, goes to the stock *undiscovered rework* and after detecting tasks go to the stock rework to do that demands the application of extra effort. The error generation happens based on one of the variable, *error fraction* as shown in the figure. The value of error fraction changes over time and the amount of tasks development. The initial value has been considered as 25%.

2.3.3 Non-Linear Behavior of Overtime and Productivity Model

Working overtime increases progress rate but at the same time decreases productivity gradually when the amount of overtime increases and workforces get exhausted. As a result, the actual productivity becomes different from the potential productivity.



Productivity decreases due to overtime

Fig. 6 Impact of Overtime on Productivity

The basic formulation for productivity is

$$productivity = nominal\ productivity * productivity\ loss\ due\ to\ fatigue(impact\ of\ overtime)$$

2.3.4 Project Performance

The parameters used for measuring project performance are

Productivity: Represents the completion of task per time in a project

Project Schedule: Defines the required time of project completion

3 Model Validation

The usefulness of a model depends on how it behaves and this behavior is analyzed through the validation process. For our model, we have performed two different validation process considering hiring and overtime plan separately for different projects.

3.1 Model Validation with Hiring and without Overtime

Table 1. Input Parameters with Cases for Model Validation with Hiring

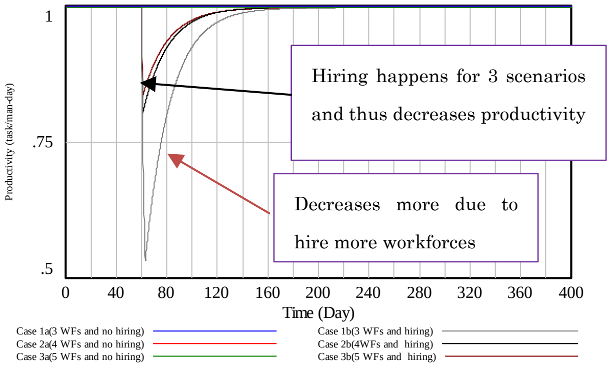
	Input Values					
	Case 1a	Case 1b	Case 2a	Case 2b	Case 3a	Case 3b
Project size(tasks)	1000	1000	1000	1000	1000	1000
Workforces (initial)	3	**3**	4	**4**	5	**5**
Deadline (fixed)(day)	250	250	250	250	250	250
Nominal productivity (task/man-day)	1	1	1	1	1	1
Hiring	No	Yes	No	Yes	No	Yes
Overtime	No	No	No	No	No	No

The first validation has been done considering with hiring and without overtime. According the cases shown in table 1 with input parameters, simulation results have been analyzed to observe the performance behavior and to obtain a feasible plan.

3.1.1 Result Analysis

To analyze the performance behavior of this validation process, the key results with explanation have shown in the following figures.

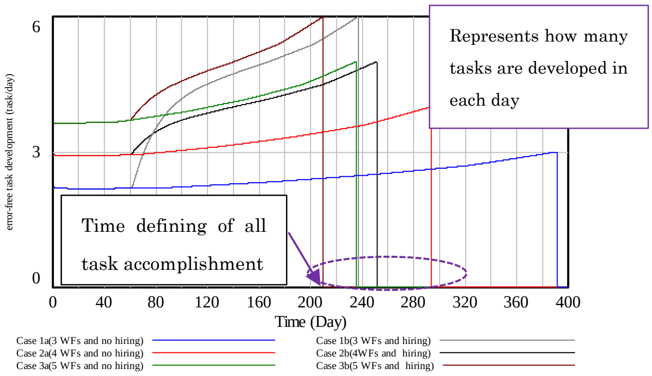
Hiring new workforces affects the productivity since their productivity is low comparing to members who are already in the project. This behavior has obtained through figure 7.



Hiring happens for 3 scenarios and thus decreases productivity

Decreases more due to hire more workforces

Fig. 7 Impact of Hiring on Productivity



Represents how many tasks are developed in each day

Time defining of all task accomplishment

Fig. 8. Error-free Task Development

Figure 8 represents how many tasks can be developed each day. Getting an idea about how many tasks can be developed each day, a project manger gets a rough estimation for different scenarios and would be able to choose a feasible plan for project development.

Again, the usage of rework cycle, creates error-based tasks and based on this error-based tasks, rework to do happens as shown in figure 9.

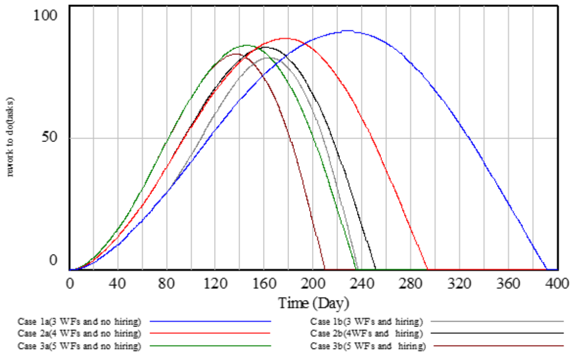


Fig. 9 Rework to Do

The amount of rework is changeable as it is not possible to measure the exact amount. However, through this analysis, at least an estimation can be obtained. Based on the estimation of rework to do, and the error-free tasks development explanation, it would be easier to design a feasible plan for project development considering both duration and cost from the following figure 10.

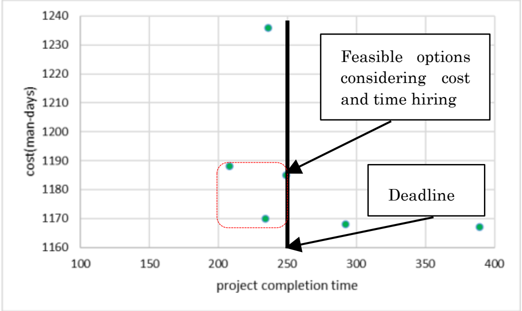


Fig. 10 Project Completion Time with Man-days

3.1 Model Validation with overtime and without Hiring

The second validation has been done considering with overtime and without hiring. The input parameters and the scenarios are given below.

Table 2. Input Parameters

Parameters	Input values
Project Size	750 tasks
Deadline	200 days
Nominal Productivity	1 task/man-day

Table 3. Cases Considered for Overtime

Cases	workforces	Overtime phases
Case 1	3	No
		Interval—based
		Continuous
Case 2	4	No
		Interval—based
		Continuous
Case 3	5	No
		Interval—based
		Continuous

3.2.1 Result Analysis

Working overtime adds extra man-hour to the task development time which increases daily tasks development rate but affects adversely the productivity as shown in figure 11.

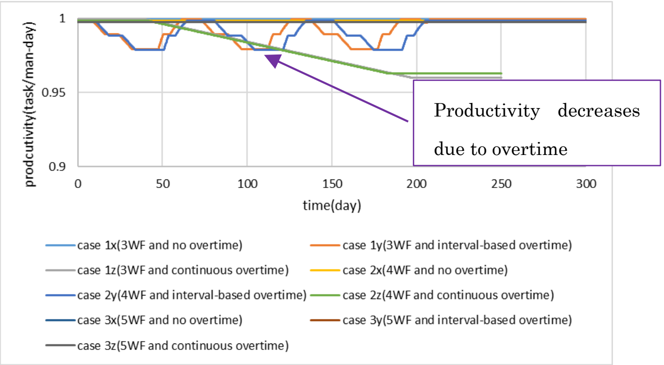


Fig. 11 Impact of Overtime on Productivity

Based on the productivity and with overtime work, task development happens as shown in figure 12.

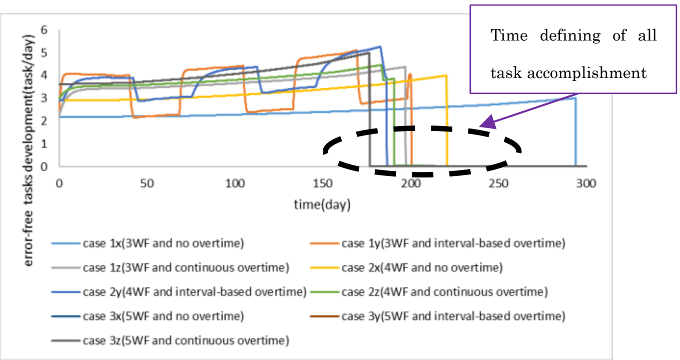


Fig. 12 Error-free Task Development Rate

While considering overtime, figure 12 gives a good assumption which represents how many tasks can be developed in each day and how much time is required for each scenario. This analysis would help to design a feasible plan considering overtime in some extent.

And based on the usage of rework cycle, error-based tasks are generated and referred as rework to do as shown in figure 13. The generation of this rework would help to understand the increase of project schedule.

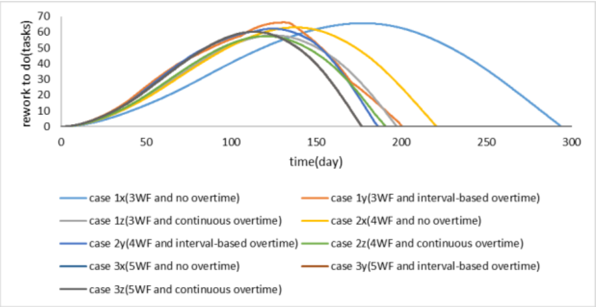


Fig. 13 Rework to Do

Based on rework to do and error-free tasks development, all task are accomplished to complete the project. The following figure 14 shows the project completion time with cost for each scenario, through which a clear understanding would be obtained for feasible project planning when overtime is in consideration.

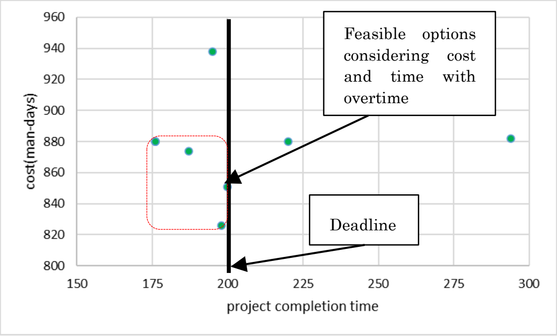


Fig. 14 Project Completion Time with Man-days

3.3 Summary of Validation Process

The purpose of the validation process is to obtain the usefulness of the model and get a very general idea of model-based decision making. These analyses give a better understanding of applying hiring and working overtime separately while designing a feasible plan. From the behavior of the results obtained, it can be stated that it is possible to make a feasible plan based on the given resources and considering hiring or overtime when necessary.

4 Case Study

After the validation process, a case study has performed applying both hiring and overtime simultaneously. The input parameters and the scenarios for this study are given below.

Table 4. Input Parameters for Case Study

Parameters	Input values
Project Size	1200 tasks
Workforce (initial)	15 person
Deadline	1800 days
Nominal Productivity	0.048 task/man-day

Table 5. Scenarios for Case Study

	Hiring	Overtime
Scenario #1	Yes	Interval-based
Scenario #2	No	Continuous
Scenario #3	Yes	No
Scenario #4	No	Interval-based
Scenario #5	Yes	Continuous
Scenario #6	No	No

4.1 Result Analysis

The impact of hiring and overtime on productivity have shown separately in chapter 3. When these two factors are considered simultaneously, both of them affect productivity as shown in the following figure 15.

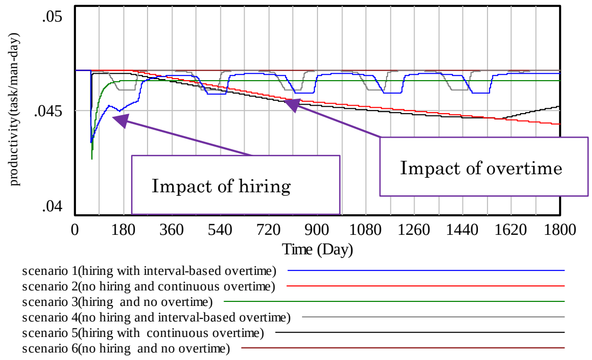


Fig. 15 Impact of Hiring and Overtime on Productivity

The following figure 16 represents project duration with cost for each scenario of the case study. Through this performance, now it becomes easier to understand which scenario requires how much time and cost while considering hiring and overtime both. And based on this performance behavior, it is possible to design feasible plan for the designed project or any other project.

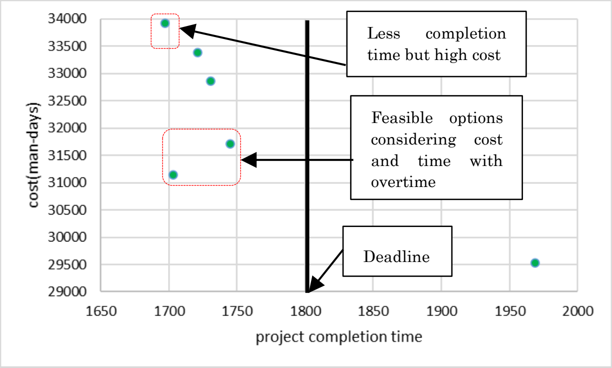


Fig. 16 Project Completion Time with Man-days

4.2 Summary of Case Study

The purpose of this case study is to obtain a conception about applying hiring and overtime simultaneously and its consequences on project performance. To keep the deadline fixed, either hiring or overtime or both can be applied. However, the application of these factors has both positive and adverse effects on project performance. Analyzed scenarios from the case study provides a way of understanding the usage of overtime along with hiring and their impact on project performance.

5 Discussion

The model has been developed considering both static and dynamic behavior for project development. Along with while considering overtime, the usage of continuous overtime besides interval-based overtime has been proposed, explained, and compared the outcome of both cases. These categories of overtime would help us to understand the consequences of overtime in a better way while considering planning and controlling actions. This framework for software development project management has been designed to better management of projects in real-time.

6 Conclusion

The methodologies and approaches we have presented in this research aimed to develop a decision support method for software development project management that allows for both static and dynamic elements to embrace the existing system. This proposed method gives a comprehensible basis of designing and choosing a feasible plan based on the given resources which would support behavioral understanding, prediction, and evaluation of process improvement, project planning, and controlling across a range of alternative scenarios.

References

1) Tarek K. Abdel-Hamid, “The Dynamics of Software Development Project Management: An Integrative System Dynamics Perspective”, Massachusetts Institute of Technology, (January 1984)