Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management
Istanbul, Turkey, July 3 – 6, 2012

Systems Dynamics Application in Project Management

Zeynep Ocak
Department of Systems Engineering
Yeditepe University, Istanbul, Turkey

Abstract

Project management provides a highly fertile area of study for system dynamics. Continuous tendency of change in project conditions, together with the nature of human behavior result with difficulties for implementation of traditional forecasting applications. Thus, due to these complicated relationships within the project, using system dynamics for control of the project progress is more accurate and safe by all means. In this study, a post-mortem analysis (modeling after the project is done, and asking "what if" questions) is performed on a particular task of a project in a manufacturing company's quality laboratories.
Using dynamic systems modeling via Stella software, what happened in real life is simulated and further analysis is performed with different approaches.

Keywords
Systems dynamics, Project failure management, Workforce utilization.

1. Introduction

Today's business environment is complex and requires faster decisions, better allocation of scarce resources, and a clearer focus. An organisation consisting of mix of large and small projects faces new challenges in resource planning, prioritisation and monitoring. Short time to complete and low cost requirements of projects demand effective and efficient project oriented organisations. As progress in projects is not always as what is foreseen and because new projects need to be started up all of a sudden during the year due to fast changing results an adequate project control process is essential.

This need for a mechanism of control turns out to be the pedestal for procedures and guidelines of Project Management. Although the existing knowledge is solid and functional for Project Management, the life itself may present situations which are not covered by the literature; or the advised procedure and solution may produce different outcomes for same specific cases. Thus, Project Management is one of the most important - yet a poorly understood - areas of management. Delays and cost overruns are the rule rather than the exception in most projects, and Project Management suffers from numerous problems of costing and scheduling. Cost overruns are not that rare. Delays can last until the point the market conditions are changed. The factors that run in the project itself are also inter-relevant: adding resources to a delayed project with the hope that this will eliminate the delay, will delay the project even longer. Moreover, changing demands of the end-user and ripple / knock-on effects even add more unpredictable results. The results are almost common for all issues above: lower profits, later completion dates, reputation loss, productivity falls, and cost rises. [8]

With all these almost chaotic relationships and hard-to-measure numbers, project management turns out to be a very suitable field for approaches using system dynamics. There are 3 major advantages of using systems dynamics approach in Project Management: [9]

- The results can be calculated correctly and certainly.
- Multiple factors can be analyzed at the same time.
- Simulations can be run under any conditions. Experimentation with extreme factors or constraints, or impossible assumptions can be performed.

In conclusion, a simulation with a dynamic system model is capable of revealing the underlying conditions and causes of an existing or a potential problem, clarifying this peculiar characteristic of projects. Thus, dynamic system modeling can be used before project initiation, during project lifecycle, or after project completion – a postmortem analysis. [2]

The basis of this study is performing an analysis on a project using system dynamics. The project contains detailed information about an exemplary failure. A managerial assumption caused more trouble; trying to complete a training session in a much shorter time than it is planned for. The final results were longer project completion time, more cost overruns, and exhausted personnel. Utilizing the system dynamics approach, we were able to see what went wrong, and solve the problem by asking "what if" questions on the model.

2. Understanding Project Dynamics

The nature of a project inherits many different relationships and syndromes. 4 major structures are common for almost all projects, which are [4]

- Project Features
- The Rework Cycle
- Project Control
- Ripple and Knock-on effects

Project features: System dynamics focuses on modeling features found in actual systems. In projects these include development processes, resources, managerial mental models, and decision making. Modeling important components of actual projects increases the ability to simulate realistic project dynamics and relate directly to the experiences of practicing managers. [1]

The rework cycle: The rework cycle is the most common and crucial feature of projects. Basically, it states that all projects run with "Undiscovered Work" and this work (which should pile up in "Work Done" stock in order to project completion) returns to initial work stock, creating more work [4]. This behavior of projects is the basic reason for all almost all delays and budget overruns. An illustration of this cycle is shown on Figure 1.

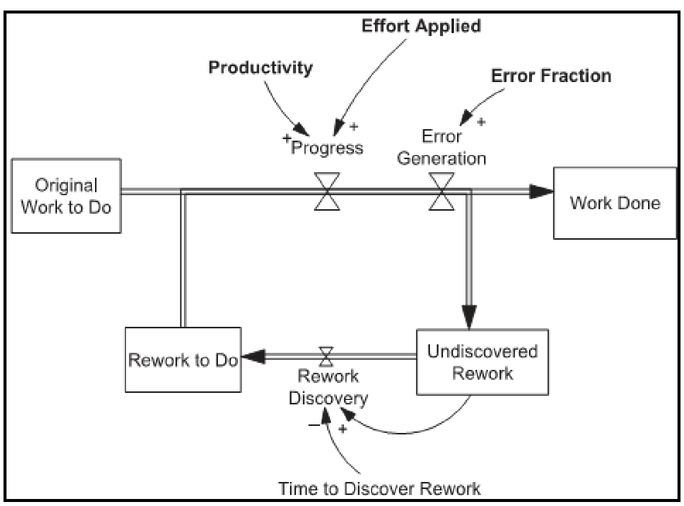

Figure 1: The Rework Cycle [4]

Project control: Modeling, analyzing, and improving the control of dynamic systems is the objective of applying system dynamics in many domains. Since project managers seek to deliver on time, on budget, and within the requirements, modeling the controlling feedback loops through which management attempts to close gaps between project performance and targets directly applies one foundation of system dynamics to project management. [6]

Ripple and knock-on effects: These effects are the result of actions that took place during the project life-cycle. These actions are generally quick fixes that the applier intends to save the project, but sometimes they simply turn out to be bad ideas. In other words, altering or adding/removing factors can create unintended consequences beyond perceived future. [3]

2. Modeling the Case Project with System Dynamics

This study focuses on the analysis of a project: installing a new computer system for six Quality laboratories in a manufacturing plant in the US. This project suffered the problem of delays expanding more than 1 year and a cost overrun of almost $200.000. Project began with the decision of replacing the old computer system used in the laboratories. This system keeps track of inventory and flow of incoming materials and inspection test results with accept/reject or deviation permit decisions. The company had been using and old, non-specific system for this purpose – with frequent collapses. Also, because the old system was not fully capable, a service fee was being paid to a 3rd-party company for computing and necessary information. Due to these reasons, the company management had decided on the replacement of the system in order to increase efficiency and lower the costs. During the system installation, a set of training courses were implemented to teach the staff how to use the new system. However, things went out of plan and poor managerial skills caused more harm than good for solving these problems. Two key points drove the problems in the model:

- Training sessions were developed for 60 days, but it was decided to complete the training in 30 days. In other words, it was falsely predicted that the 60-day program can be completed within 30 days.
- Laboratory personnel were not skilled enough. Previous system was very old, and the new system was a cutting-edge technology for the staff.

2.1 Development of the Dynamic Model

Dynamic model was simulated using the computer software package `STELLA`. STELLA is preferred for this research due to its ease of use as object-oriented modeling environment. STELLA is intuitive to use and is mathematically rigorous. Figure 2 presents the screenshot of Stella model which is used for analysis of the case. "Work to Do" and "Work Done" stocks represent the amount of training to be done and is done. The flow, named "Progress Rate" has two parameters, "Staff Skill" and "Staff Fatigue". By common sense, the more staff learns, the easier will be the trainings, so "Staff Skill" has a positive effect on the progress. However, "Staff Fatigue" is a function of "Pressure" which originates from the time available ("Time Remaining" Stock). The hidden problem on the failure of these trainings is the time pressure. The company employees frequently complained that the trainings were too rushed. However, the responsible manager refused to change the plan, and tried to complete the work within predetermined interval – 30 days. Later in the process, the expert responsible for the trainings is fired. Now this failure will be analyzed with the help of this model.

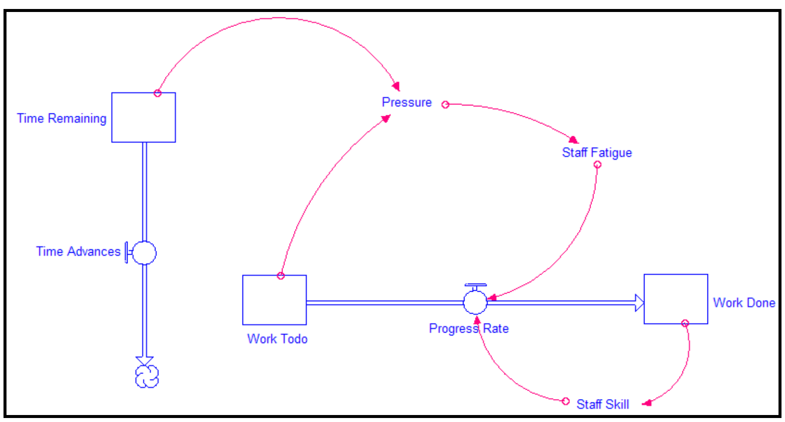

Figure 2: Dynamic Model of the System

The functions of each component follow as:

Time_Remaining(t) = Time_Remaining(t - dt) + (- Time_Advances) * dt
INIT Time_Remaining = 30

OUTFLOWS:
Time_Advances = 1
Work_Done(t) = Work_Done(t - dt) + (Progress_Rate) * dt
INIT Work_Done = 0

INFLOWS:
Progress_Rate = Staff_Skill-Staff_Fatigue
Work_Todo(t) = Work_Todo(t - dt) + (- Progress_Rate) * dt
INIT Work_Todo = 100

OUTFLOWS:
Progress_Rate = Staff_Skill-Staff_Fatigue
Pressure = 0.1+LOGN(1.0001+(Time_Remaining*Work_Todo))
Staff_Fatigue = 10-(Pressure)
Staff_Skill = 4+LOG10(1.0001+Work_Done)

- "Work To Do" is a stock containing 100 units of "work", flowing to the "Work Done" stock with a dynamic rate.
- Progress Rate is the difference between staff's skill and fatigue. From the paper, it's known that complaints from the staff being trained has stopped sessions, so enough fatigue is able to block the progress.
- Pressure is a logarithmical function of remaining time and work to do.
- Fatigue is calculated with pressure and a constant, 10. To explain, the initial "energy" that staff begins with is constant, and then lowers with more pressure applied.
- Staff Skill is a combination of a logarithmical function and a constant. Obviously, staff has some knowledge and experience with computer systems (the constant), and more training they get, more skill they develop.

After finalizing the model and the functions, the system is simulated for 30 days (the guessed interval by project management staff) and the results in the analysis section are obtained.

2.2 Analysis of Results

Table 1 presents results of running the simulation for 30 time units (days).

Table 1: Simulation Results

Days	Work Todo	Progress Rate	Work Done
0	100.00	2.15	0.00
1	97.85	2.56	2.15
2	95.29	2.75	4.71
3	92.54	2.86	7.46
4	89.68	2.91	10.32
5	86.77	2.94	13.23
6	83.83	2.94	16.17
7	80.88	2.93	19.12
8	77.95	2.91	22.05
9	75.04	2.88	24.96
10	72.16	2.84	27.84
11	69.32	2.79	30.68
12	66.53	2.73	33.47
13	63.81	2.66	36.19

14	61.14	2.59	38.86
15	58.56	2.51	41.44
16	56.05	2.42	43.95
17	53.63	2.33	46.37
18	51.30	2.22	48.70
19	49.08	2.11	50.92
20	46.97	1.99	53.03
21	44.98	1.86	55.02
22	43.13	1.71	56.87
23	41.42	1.55	58.58
24	39.87	1.37	60.13
25	38.50	1.16	61.50
26	37.34	0.92	62.66
27	36.42	0.61	63.58
28	35.80	0.20	64.20
29	35.60	0.00	64.40
Final	35.60	0.00	64.40

Rapid decrease on the amount of progress rate is obvious. The amount of work done is barely more than the half of the complete work, so the behavior of the model includes characteristics of burnout cycle. Graphical representations show a clearer view (see figure 3 and figure 4):

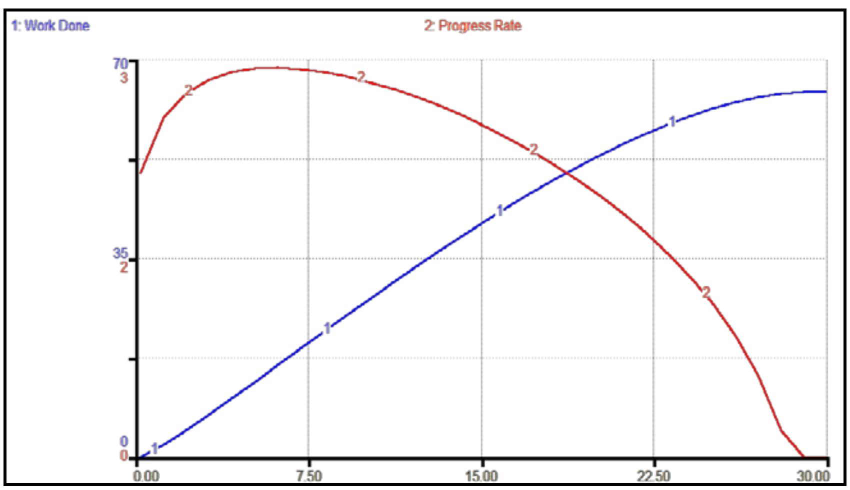

Figure 3: Comparison of “Work Done” vs. “Progress Rate”

Figure 3 depicts the change of “Work Done” and “Progress Rate” over 30 days. Notice that the progress rate is sharply descending after a point, blocking the flow, hence the completion of the process. “Work Done” is stuck before reaching to even 70, way lower than the necessary amount of 100. Figure 4 represents the reason of this situation. When “Staff Skill” and “Staff Fatigue” converters are compared, the reason under the problem is visible. Fatigue function climbs up rapidly with an exponential behavior, and overtakes skill function. At that point, progress stops and the amount of performed work can no longer move on. This is what leads the staff to complain about the trainings and the replacement of teaching expert. This graph is sketched for 40 days instead of 30, for easier viewing.

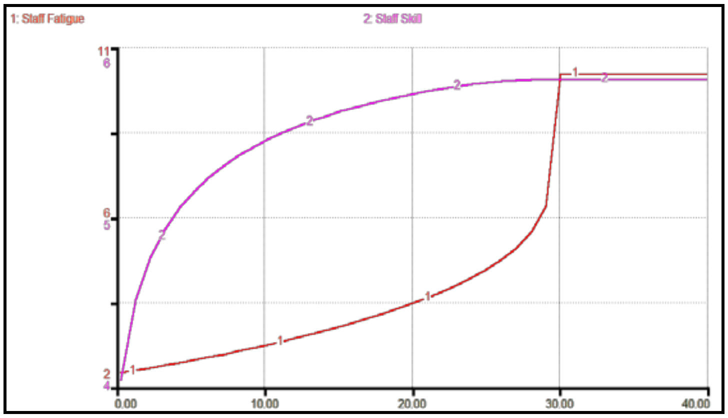

Figure 4: Comparison of “Staff Fatigue” vs “Staff Skill”

2.3 Model Validation

Before performing further work, it must be ensured that the model produces sensible outcomes. Although the results make sense at this point, it is a necessary step to check whether or not the model is proper and appropriate for advanced analysis. The literature contains some guidelines to build confidence on the model, the following being the key criteria [5]. Satisfying these restrictions, our model is believed to be valid. [3]

- The behavior of the model must be plausible
- Causal loop diagram corresponds to the problem statement
- Equations, and direction of linkages are consistent with the model
- Unit consistency
- Conversation of flow

2.4 Policy Analysis

Less pressure: It is known that haste is the key element of failure of the training sessions, so trying a longer interval has the most sense in terms of an analysis. The training program was designed for 60 days, so changing the “Time Remaining” stock from 30 to 60 and re-running the simulation produces the results shown in Figure 5.

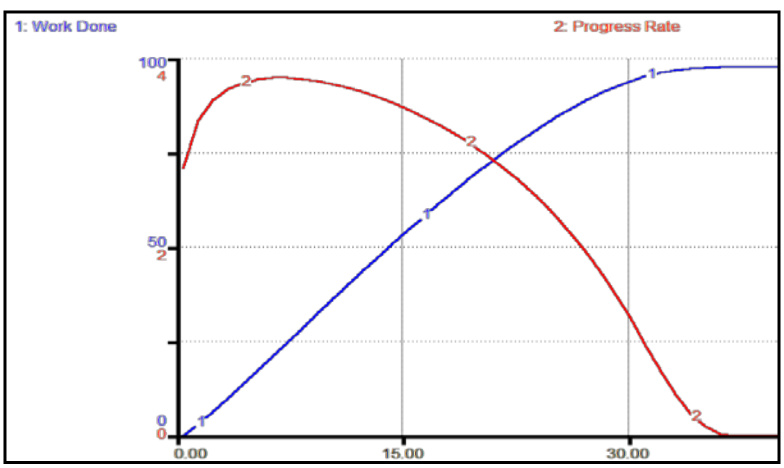

Figure 5: Comparison of “Work Done” vs. “Progress Rate” for “less pressure” policy

This policy produces 97.97 of completed work – close to total amount of 100. It’s important that the “Time Remaining” stock is what causes the “Pressure” on the staff, so staff performs better, with lower amount of pressure – hence fatigue. 37th day is the date where the completed work stops climbing, so it makes sense that not pushing the staff harder will actually yield a highly satisfactory date of completion. Figure 5 reveals that progress rate declines severely again, but this time almost all of the work is completed before the progress stops.

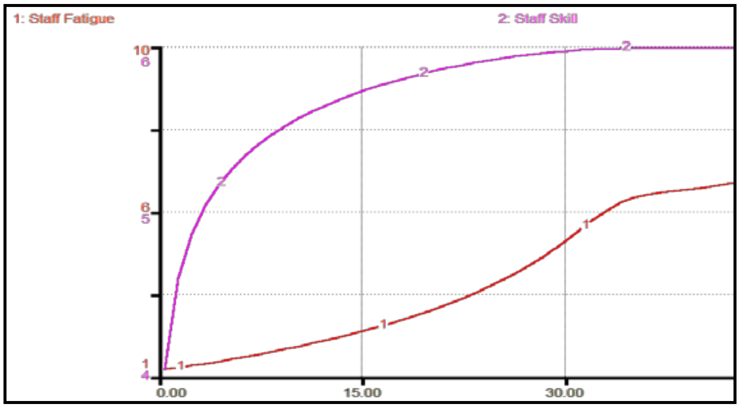

Figure 6: Comparison of "Staff Fatigue" vs. "Staff Skill" for "less pressure" policy

As one can easily notice in Figure 6 that fatigue has a lower rate of climbing, so it cannot overtake the skill function before work completion. This policy comparison shows that pushing harder is a critical mistake. Analysis show that the performance would be significantly better with less pressure applied on the trainings.

More skill: Since the staff's ability of learning is a function of a constant of 4, and the logarithmical function of the training performed, beginning with a higher level of "base knowledge" (a higher constant) is supposed to produce better output. The results below are obtained after varying the function of "Staff Skill" convertor – increasing the constant to value of 6. This time the amount of completed work reaches up to 98.71, and final value is obtained on 26th day. Having a staff with more experience on computers promises a more successful training session than taking easy with a relatively inexperienced staff. Figure 7 and Figure 8 show that the fatigue is still far from blocking the progress. As predicted, a more "skilled" staff performs better results.

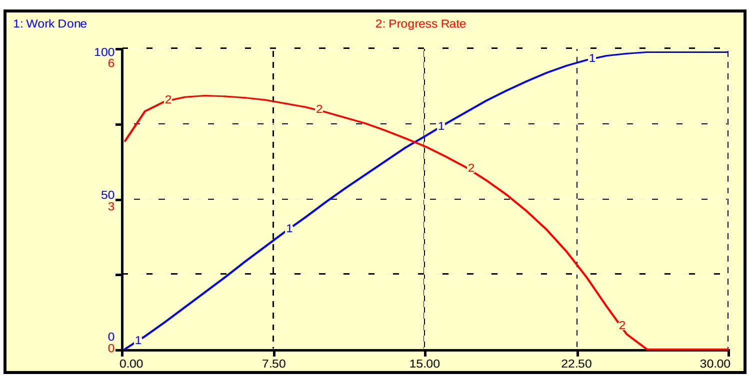

Figure 7: Comparison of "Work Done" vs. "Progress Rate" for "more skill" policy

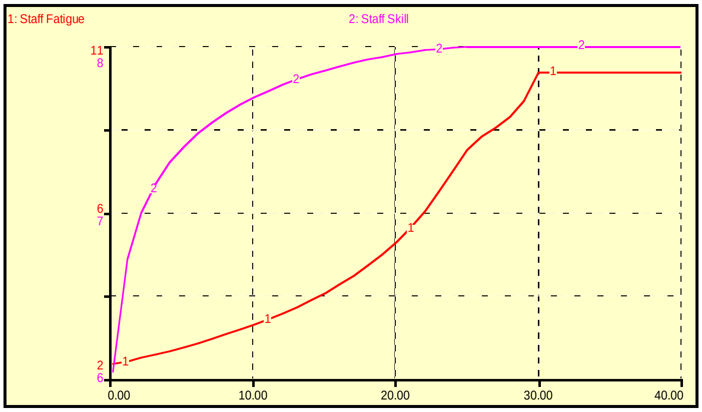

Figure 8: Comparison of “Staff Fatigue” vs. “Staff Skill” for “more skill” policy

3. Conclusion

The dynamic model developed proves that the initial prediction (burnout evolving from fatigue fails the training activities) is a correct interpretation of task failure. The nature of human behavior and capability applies on project management as well, so managers should always keep in mind that workforce is a delicate resource, and must be utilized wisely. Our analysis on different policies reveal that efficiency would be much higher with a gentler pace, especially when relatively lower capability of the staff is considered. Running the model with 60 days (the designed duration for the trainings) of “time remaining” stock instead of 30 reduced the pressure and provided better efficiency, because of staff’s inability to perform with their limited experience on computers. Altering staff’s ability of learning by changing the constant in relevant equation turned out to be the next sensible step, and this policy performed better results by common sense. In the process, these “experiments” on the model strengthens model validity since they produce results consistent with common sense. Outputs and results of trials are consistent with what happened in real life. Trying different models or models with different relationships are promising ideas for application of dynamic system simulations on similar cases. So for future research, it will be a wise approach to perform further analysis with extended models – and with more detailed information if available.

References

1. Duncan W.R., 1996, “A Guide to the Project Management Body of Knowledge”, Project Management Institute.
2. Ford, D. N., Sterman, J. D., 2003, “Overcoming the 90% Syndrome: Iteration Management in Concurrent Development Projects”, Concurrent Engineering: Research and Applications.
3. Lock, D, Project Management, 9th Edition, Gower Publishing Limited, Hampshire, 2007.
4. Lyneis, J.M., Ford, D.M., 2007, “System Dynamics Applied to Project management: a survey, assessment and directions for future research”, System Dynamics Review, Vol 23.
5. Maani K.E., Cavana R.Y., 2007, Systems Thinking, System Dynamics, 2nd Edition, Pearson Education New Zealand.
6. Meredith, J.R., Mantel, S.J, 1995, Project Management: A Managerial Approach, 3rd Edition, Wiley, USA.
7. Rodrigues, A., 1994, “The Role of System Dynamics In Project Management: A Comparative Analysis With Traditional Models”, 1994 International System Dynamics Conference.
8. Sterman, J. D., 2002, “System Dynamics Modeling: Tools for Learning in a Complex World,” IEEE Engineering Management Review, 30**,** 42-52.
9. Sterman, J. D., 2000, Business Dynamics: Systems Thinking and Modeling for a Complex World, Irwin/McGraw-Hill.