Leveraging Dependency Structure Matrix (DSM) and System Dynamics in Combination to Reduce Project Rework

Tomas Flanagan, BP London,
Email: tomas.flanagan@bp.com

Scott T Johnson, BP Houston,
Email: scott.johnson2@bp.com

Reiner Dojen, University of Limerick,
Email: reiner.dojen@ul.ie

Leveraging Dependency Structure Matrix (DSM) and System Dynamics in Combination to Reduce Project Rework

Keywords

Rework cycle, Dependency Structure Matrix (DSM), System Dynamics, project modelling, risk mitigation, planning

Abstract

Planning and managing large-scale projects is non-trivial, as evidenced by the large number of projects that exceed budget and schedule targets. In many cases, rework is a key factor in project delays. Accurate rework prediction is challenging – even when the tasks responsible for rework can be identified, the likely project impact is difficult to determine. The work described in this paper examines how Dependency Structure Matrix (DSM) techniques can be leveraged to support and improve System Dynamics applications. It demonstrates how the DSM can be used to identify tasks that are likely to drive rework within a project and exploits System Dynamics to quantify the associated financial and schedule downsides. Using the context of current oil and gas projects, the challenges of managing dependencies between multidiscipline teams working to identify, evaluate, and select a development concept are examined. An overview of DSM fundamentals and approaches used to help with the management of these dependencies is presented. Next a discussion of how System Dynamics can both benefit from DSM analysis and resolve known limitations are considered. The natural integration of DSM and System Dynamics for management of project dependencies is summarized and used as a basis for suggesting new research agendas.

Introduction

Project risk, in terms of cost and duration is a major concern in industry [1],[2]. Nevertheless, deterministic project plans, which do not explicitly capture risks due to task rework, are commonplace. Modelling and predicting rework behaviour is difficult [1],[3],[4] and while experienced planners are aware of many different project risks, constraints imposed by planning software such as MS Project or Primavera bias companies to produce plans which indicate that tasks will complete on time and within budget. This can lead to high-risk plans that have undesirable consequences if tasks fail. In many cases, the use of contingency provides a means of reducing project risk but this introduces inefficiencies as well as reducing transparency and hence introduces difficulties in controlling the project.

Uncertainty in design projects arises from a number of sources [2]. Task order, resource availability, contractor performance and potential changes to requirements are often unknown or unclear during the project planning phase. These factors can act together or in isolation to create project rework which leads to delays and cost overruns.

This paper illustrates how DSMs (Dependency Structure Matrices) can be used to identify potential sources of project rework but argues that finding rework is not enough – a means to quantify the resulting project impact is also required. The paper argues that this challenge can be addressed through system dynamics simulation and that the combined use of systems dynamics and DSM can lead to improved project plans in industry.

The paper begins by outlining key challenges in energy sector project planning. It then proceeds with a brief review of relevant literature on DSMs and SD. Next, the argument for a combined approach which leverages both techniques is presented as a means to gain insight into the planning challenges identified. After a discussion of the merits and limitations of the combined approach, the paper concludes with a summary of core conclusions.

The challenge in industry

When a new oil or gas resource has been discovered, the development project is planned and executed using a stage-gate capital value process to ensure the outcome achieves corporate objectives. Following is a high-level characterization of the phases: *Appraise* to identify at least one feasible development option; *Select* to narrow the development options down to one; *Define* to conclude detailed engineering; and *Execute* to construct and deliver to an operating condition. Multi-discipline project teams are organized to complete each phase. This is challenging for the participants as the work is initially highly conceptual and there are many interdependencies that have to be identified and managed. Traditional project management techniques rely on discussion processes to identify and manage the interfaces. These traditional processes are less reliable when complexity is high and experience is low. This calls for more reliable methods of identifying and managing dependencies.

Feedback dependencies between tasks typically prove particularly challenging. At the project outset, many decisions are taken subject to uncertainty. As work progresses, new information becomes available which either confirms the validity of these assumptions or shows them to be false. In the latter case, the utility of work performed based on invalid assumptions needs to be considered and, in many cases, such work needs to be redone. In engineering projects, revisiting previously completed activities is typically referred to as iteration or rework.

Different types of rework are commonplace within projects. This work focuses on two types (notwithstanding that other authors may include recurring tasks as a separate rework category):

- Iteration takes place to converge on a solution. In aerospace, for example, 7 iterations are typically required to optimise the design to a sufficient performance level (although fewer iterations are typically performed in the energy sector).
- Rework may take place due to errors or undesired/expected test outcomes. If a faulty part is assembled within a product (or sub-system), stripping of the product may be required to replace the faulty part. In such cases, the rework of other tasks can be more of a problem than correcting the initial error.

With respect to iteration, the goal of successful project management is to:

- ensure that project schedules account for the necessary or planned iteration while simultaneously minimizing the time taken to converge on a solution (by reducing the number of iteration cycles required or by accelerating the speed at which iterations take place) and
- avoid unnecessary rework and minimise the time taken to discover the errors that drive rework.

The subsequent sections discuss two techniques that have been used in industry to plan and manage iteration and suggest the potential benefit of combining both techniques.

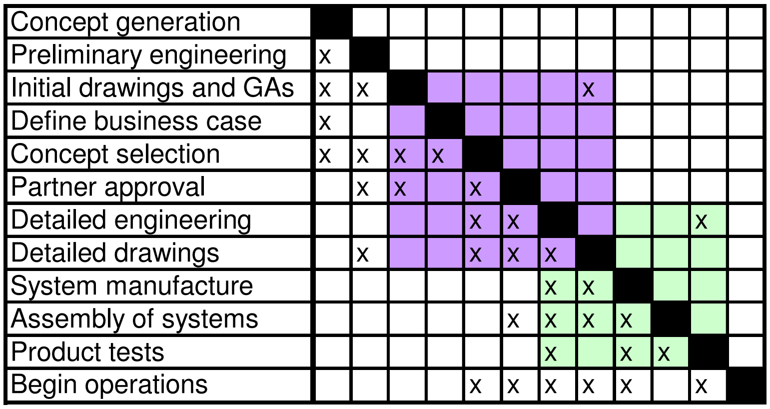
DSM Fundamentals

The Dependency Structure Matrix (DSM) [6,7] is a transposed adjacency matrix that provides "a simple, compact, and visual representation" [7] of system connectivity. DSMs are widely used by engineering researchers and practitioners to both analyse product architecture and project structure.

A DSM consists of identically labelled rows and columns and uses off-diagonal entries (tick-marks) to signify the dependency of one element on another (Figure 1). DSMs have been successfully used to model product, process and organisational connectivity. When used to model the design process, the matrices capture dependency between different tasks and can be reordered to achieve minimum iteration. DSM product models show the connectivity between different components and organisational DSMs represent connections between teams and individuals.

Numerous researchers (see for example [8]) use DSMs as a basis for their work in modelling design processes in order to understand iteration and mitigate against rework by changing the order in which tasks are performed. For example, Eppinger et al. [8] use DSMs to examine processes with the goal of improving the task order. Yassine et al [9] perform simulation analysis based on the DSM method to model rework in the automotive industry and to assess sensitivity to errors in rework probabilities. Cho and Eppinger [10] discuss how iterated tasks may be performed in sequence, in parallel, or with varying degrees of overlap depending on the extent of information dependency between tasks. They note that such models of rework, while useful, are not without limitations such as poor scalability and over-simplifying assumptions.

Figure 1 (below) illustrates the task dependencies for a simplified oil-and-gas sector engineering project. The shaded boxes show where rework cycles are likely to occur. The first (purple) shaded box illustrates a feedback associated poor quality drawings and the second (blue) shaded box shows how testing activities may identify errors and oversights in engineering that require rework. However, the DSM provides little insight into the manner in which these potential rework cycles are likely to impact the project; SD can provide some useful insights into this problem as discussed below.



Concept generation	■											
Preliminary engineering	x	■										
Initial drawings and GAs	x	x	■					x				
Define business case	x			■								
Concept selection	x	x	x	x	■							
Partner approval		x	x		x	■						
Detailed engineering					x	x	■				x	
Detailed drawings		x			x	x	x	■				
System manufacture							x	x	■			
Assembly of systems						x	x	x	x	■		
Product tests							x		x	x	■	
Begin operations					x	x	x	x	x		x	■

Figure 1 Task DSM for an energy-sector project (simplified)

Previous applications of System Dynamics to model rework

Cooper [11] presents the rework cycle as a mechanism to explain major delays, especially in large-scale projects (Fig 2). Based on extensive industrial experience as a consultant working in System Dynamics modelling, he argues that undiscovered rework, creates a falsely optimistic picture of progress and that early discovery of such rework is critical to on-time project completion.

He also points out that late augmentation of resources often does little to improve the situation and that, in some cases, the approach can even exacerbate the problem. Typically, an increased focus on quality assurance early in the project is likely to prove far more beneficial than fire-fighting a project that has gone off track. Similar effects, in relation to the negative impacts of fire-fighting, are reported by Repenning, Gonclaves and Black [12] and Ford and Sterman [13]. They also discuss the "90% syndrome" where project seem to be right on target until they reach 90% completion. An extensive review of system dynamics applications in project management, which discusses the rework cycle, the "90% syndrome", change management and project control, is provided by Lyneis and Ford [14].

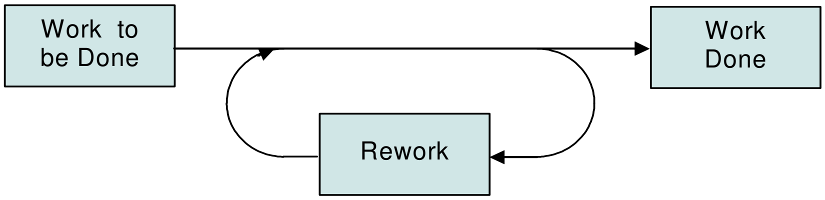


Figure 2 The Rework Cycle [11]

The concepts described by Cooper were used by a ship-building company to argue in court that the rework impacts (in terms of cost and schedule) associated with changes to requirements where considerably larger than client had anticipated and that the client rather than the ship-builder was responsible for these delays. The court's ruling underscores the plausibility of SD as an approach for quantifying the impacts of rework.

SD and DSM Integration: The value of a combined approach

The above sections described how both DSM and Systems Dynamics have been successfully applied in industry to mitigate against the undesired effects of iteration. However, while both techniques offer valuable insights for the project planner neither approach offers a complete view of the impact that rework is likely have on the project. Specifically, the DSM constitutes a structured approach that shows which tasks are likely to drive rework but provides little guidance on the likely delays and cost overruns that will be in incurred. In contrast, System Dynamics typically takes a more holistic approach – it typically ignores the task-specific information that can be gleaned from the DSM but quantifies the overall effects of associated rework due to multiple different tasks in combination. SD handles non-linearities associated with feedbacks and shows how delays in the detection of rework can provide a false sense of optimism that the project is on schedule when, in fact, undiscovered rework will lead to major delays in downstream tasks.

Given the merits and limitations of both techniques it seemed obvious that a combined approach to dealing with rework would likely prove valuable in project planning and management. Firstly, the DSM is used to identify:

1) whether rework was likely to take place in the process
2) which tasks were likely to drive rework
3) the cycle time for rework (based on the number of tasks involved in a given cycle)
4) how changing the task order would likely reduce rework
5) whether other strategies such as extra testing or quality assurance (QA) work could reduce rework.

This information from the DSM exercise is then used to inform System Dynamics (SD) modelling and analysis:

1) to decide whether a SD model is required (in cases where no rework is identified using the DSM, for example, SD modelling may not be necessary)
2) inform the SD model focus (when in the project is rework likely to occur, what tasks are likely to cause rework)
3) calibrate the model (in terms of rework discovery time, rework cycle time)
4) provide assurance that the SD model deals with all of the rework risks and issues identified from the DSM analysis.

Once constructed the SD model can be used to evaluate the strengths and weaknesses associated with alternative plans, particularly in relation to rework mitigation. By quantifying the effects of rework, SD analysis shows:

1) how changes to the task order are likely to reduce rework
2) at what stage in the project additional resources provide the maximum benefit
3) when increased testing makes sense and
4) to what degree expediting rework cycles is likely to impact cost and schedule.

Figure 3 (below) graphically represents results from actual project data concerning the effects of rework on performance. Results show that the actual project performance was accurately predicted by the Vensim SD model but that the project deviated significantly from the plan. The deviation was due to undiscovered rework associated with engineering drawings. The model also correctly predicted the 5-month delay between the point where the error takes place (month 4) and the time when rework occurs (approx month 9).

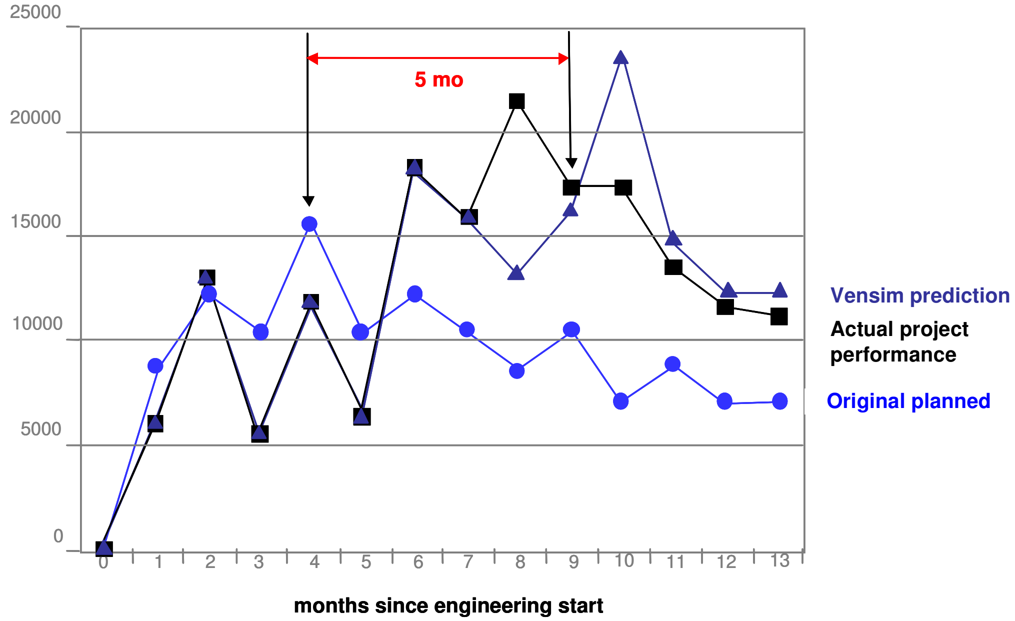


Figure 3 Using system dynamics to quantify the impacts of rework due to errors in drawings on project schedule

The DSM approach outlined above and described in Figure 1 highlights the potential for rework associated with poor quality drawings but that the SD model is required to quantify the effects. Together DSM and SD provide an efficient and effective means to interrogate project plans and hence to reduce likely deviation from cost and schedule targets.

Discussion and key conclusions

Effective project planning is a major challenge in industry and improved tools and techniques are required to overcome the associated problems. This research considered how SD and DSM can be used in combination to provide different but complementary perspectives on project rework, highlighting the merits and limitations of each approach.

In essence, the DSM is an effective tool for discovering rework while SD constitutes a suitable technique for quantifying the likely impacts on the project. Together, they provide a unique capability to inform project plans and mitigate against unnecessary rework. In highlighting the synergy between SD and DSM, this paper identifies a research agenda which is likely to deliver meaningful value to industry. Future work focuses on more thorough evaluation of the SD/DSM approach during the early stages of a major project (greater than $100m CAPEX) in the energy sector.

References

[1] Browning, T.R. and Eppinger, S.D., “Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development”, IEEE Transactions on Engineering Management, 49(4), 2002, pp 428-442.

[2] Browning, T.R., "Sources of Schedule Risk in Complex System Development", Systems Engineering, 2(3), 1999, pp: 129-142.

[3] Yassine, A, Whitney, D. and Zambito, T., “Assessment of Rework Probabilities for Design Structure Matrix (DSM) Simulation in Product Development Management“, DTM 2001, Pittsburgh, USA, 2001

[4] Eppinger, S. D., Whitney, D., Smith, R. and Gebala, D., “A Model-based Method for Organizing Tasks in Product Development“, Research in Engineering Design, Vol. 6, No. 1, 1994, pp. 1-13

[5] Wynn, D., Eckert, C.M and Clarkson, P.J. Applied Signposting: A modelling framework to support design process improvement. ASME DTM 06. Philadelphia, Pennsylvania, US, 2006

[6] Steward, D.V., The Design Structure System: A Method for Managing the Design of Complex Systems. IEEE Transactions on Engineering Management, 1981. 28(3): p. 71-74.

[7] Browning, T.R., Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and new Directions. IEEE Transactions on Engineering Management, 2001. 48(3): p. 292-306.

[8] Eppinger, S.D., D.E. Whitney, R.P. Smith, and D.A. Gebala, A Model-Based Method for Organizing Tasks in Product Development. Research in Engineering Design, 1994. 6(1): p. 1-13.

[9] Yassine, A., D.E. Whitney, and T. Zambito. Assessment of Rework Probabilities for Design Structure Matrix (DSM) Simulation in Product Development Management. in 13th International Conference on Design Theory and Methodology (DTM 2001). 2001. Pittsburgh, Pennsylvania, US.

[10] Cho, S.H. and S.D. Eppinger, A Simulation-Based Process Model for Managing Complex Design Projects. IEEE Transactions on engineering mgmt, 2005. 52(3).

[11] Cooper KG. The Rework Cycle: Why Projects Are Mismanaged, PM Network, 1993.

[12] Repenning, N., P. Goncalves. and L. Black, Past the Tipping Point: The Persistence of Fire Fighting in Product Development, California Management Review, Vol.43, No. 4, pp44-63, 2001

[13] Ford, D. and J Sterman,. The Liars Club: Concealing Rework in Concurrent Development, Concurrent Engineering, Vol.11, pp211-219, 2003

[14] Lyneis, J.M. and D.N. Ford , System dynamics applied to project management: a survey, assessment, and directions for future research, System Dynamics Review, Vol.23, No. 2-3, 2007