INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15-18, 2005

ROBUST PLANNING OF DESIGN TASKS USING SIMULATION

T. L. Flanagan, C.M. Eckert, R. Keller and P.J. Clarkson

Abstract

Planning of large scale design projects is difficult due to complex interdependencies between tasks and uncertainties concerning task duration, rework likelihood, rework behaviour, requirement changes and resource availability. This paper examines how different task rework behaviour affect project duration and considers the impact of varying task orders subject to resource constraints. Results of design project simulation, generated using the signposting approach, show how modifications to task rework behaviour have a major impact on project duration and that appropriate task ordering can reduce task rework.

Keywords: planning, uncertainty, rework, iteration, risk management, design process improvement, robust design, simulation

1. Introduction

Project risk, in terms of time, cost and quality is a major concern in industry [1],[2]. In addition formal project risk assessments are increasingly required by governmental agencies such as the Ministry of Defence in the UK. Nevertheless, deterministic project plans, which do not explicitly capture risks due to task failure, are commonplace. Modelling and predicting rework behaviour is also a challenge [1],[3],[4]. While experienced planners are aware of many different project risks, constraints imposed by planning software 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 buffers provides an effective means of reducing project risk but failure to accurately specify these buffers can lead to inefficiencies and difficulties in controlling the project.

Uncertainty in design projects arises from a number of sources [2]. Task durations, task order, task outcome, rework likelihood, resource availability, sales volumes, supplier performance, changing requirements and even such diverse issues as governmental decisions on taxation policy are often uncertain during the project planning phase. Although these uncertainties are widely acknowledged by both industry and academia, their effects, especially in combination, are poorly understood. In addition to these inherent project uncertainties, the use of plans to represent design projects introduces further uncertainty as the interpretation of plans is subjective.

The word robust can be defined in terms of the ability to yield approximately correct results despite the falsity of certain assumptions or inaccuracy of certain parts of the input [5]. In this light, robust plans [6] can be considered as plans which are insensitive to different uncertainties described above. A large number of uncertainties exist in design projects and the characteristics of robust plans which are less sensitive to uncertainties, are ill defined.

This paper illustrates how simulation of variations to project characteristics can provide insight into the impact of project uncertainties. Three different patters of rework (described in section two) were simulated and their impact on project duration was analysed.

The paper begins by describing the research methodology adopted and outlines key industrial challenges based on case study observations. It then proceeds with a review of relevant literature. Next, the arguments for model generation and simulation are presented as a means to gain insight into these industrial challenges followed by details of the simulation project. After a discussion of the simulation results, the paper concludes with a summary of core conclusions.

2. Exploring planning challenges in industry

During the last year, an extensive case study was undertaken with a diesel engine manufacturer in order to better understand planning problems in industry. Data from 30 one-hour interviews, observations in meetings, company documentation, industrial project plans and process documents were analysed with a view to obtaining a better understanding of planning in industry. Other case studies performed by the Engineering Design Centre at Cambridge confirmed the generality of observations made during this study [7].

Our studies highlighted the following project planning challenges in industry:

- Characterisation of robust project plans
- Determining an appropriate level of detail in plans
- Planning milestones and decision points between gateways
- Quantifiable project risk assessment and contingency planning
- Planning for rework in the event of task failure

While all these issues are very important, we have decided to focus on the first and the last points as an extensive analysis of the complete list of challenges would be beyond the scope of this paper. We also note that these two issues are highly interlinked; construction of a robust plan requires a thorough understanding of task failure and rework.

2.1 Different rework scenarios

The case study also highlighted difficulties in modelling rework during the design process. At first, managers in the company were reluctant to acknowledge the existence of unforeseen rework but closer examination of the issue exposed its importance as a project success factor. It also became clear that rework could assume several different forms, with regards to the time it takes to carry out the iteration. Based on this observation we decided to investigate the impact of three different patterns of rework on project duration.

Parallel iteration: Sometimes, tasks fail to produce the expected output and iteration is performed in parallel to other tasks which follow the original plan in so far as is possible. For this research we assumed that the rework of failed tasks does not restrict the availability of resources to the ongoing tasks but that the lower information state, (see section 3.3) resulting form task failure can influence the task order. This scenario could arise if a supplier redesigns a subcontracted component.

Accelerated iteration: The second iteration model assumes that tasks can be accelerated during rework and that they take only a fraction of their original time (e.g. 20%) due to learning affects but that they do affect resource availability. This model is appropriate for some design tasks where minor modifications are required based on information which was not available at the time when the task was first performed.

Slow iteration: Slow iteration assumes that tasks aren't accelerated during rework. Some testing tasks, which have a predefined duration, should be modelled using this approach. Again, it is assumed that these tasks draw resources from the project resource pool.

Analyses of these different rework behaviours were eventually performed by generating a model for a hypothetical design project and simulating perturbations of this model corresponding to each rework scenario. Prior to these analyses, however, we first examined relevant literature on project planning and simulation.

3. Project planning and simulation

In this section, we first consider different techniques and support tools for industrial project planning, as the overall goal of this research was to provide a strategy for robust planning. The final sections of this literature review give an overview of simulation research and an introduction to the Signposting tool, which acts as the simulation engine.

3.1 Project planning

Generic approaches to project planning include program evaluation and review technique (PERT) and the critical path method (CPM) [8], both of which are instances of the Precedence Diagramming Method (PDM). In such models, activities are shown as nodes and arrows between activities represent information or material flow. The major difference between both approaches is that CPM uses a single-point estimate for task duration, while PERT uses a weighted average of lowest, highest and most likely duration. The critical path is the longest chain of consecutive zero-slack tasks in the project – any delays to these tasks will result in project overrun [9].

Microsoft Project™ and Primavera™ are among the most commonly used software implementations of PERT and CPM. While these tools show task ordering and can highlight the knock-on effects of changes to a given task, they suffer from inherent limitations of PDM. At the outset of a design project many precedence are unknown or uncertain; in industry the existence of precedence relationships often becomes clear only when tasks are executed. Neither iteration nor task alternatives can be modelled using PDM and this limits their effectiveness when trying to understand and improve the design process. Another criticism of current tools is that they offer no information on the dependencies between tasks. Often, knowing the nature of the precedence relationships is as important as being aware that a relationship exists but industrial software tools fail to provide this information.

Design projects exhibits many characteristics of complex systems [10] – design tasks are highly interlinked and changes to any task (outcome, delays, increased resource demand) are likely to affect other tasks. Hence, it is extremely difficult to predict the behaviour of such projects. This observation casts a shadow on planning approaches which assume that project behaviour is linearly deterministic: such approaches are perhaps best suited to projects (such as manufacturing) which exhibit less uncertainty that their design counterparts. A more thorough understanding of design project behaviour coupled with improved planning techniques is required to create realistic project plans.

3.2 Simulation

Computer simulation is increasingly being seen as a powerful way to model and explore the dynamic behaviour of some complex systems [11]. Simulation was chosen for this research because the alternative approaches of direct experimentation or mathematical modelling were respectively considered impractical or inadequate. Direct experimentation is unrealistic or impossible because all projects are different while mathematical models cannot adequately capture the dynamic and transient effects of encountered in project management problems [12]. Simulation also offers other advantages including improved scalability, greater control over project variables and richness of output results available. These results, in turn, are easier to access, analyse and interpret than actual project data and were more easily reduced to visual summaries such as histograms and Gantt charts, thus yielding interesting insights into the system being examined. In order to achieve validation, a subset of the simulation results were compared to actual project data.

When simulating, elements of the real system are mapped to abstract elements in the simulation model. Execution of the simulation program represents the passing of time during which the state of these elements changes. Simulation research attempts to map this new state back the real system, arguing that observations of the simulation behaviour can be used to gain insight into the real system [13]. A model of the system is used as input into the simulation. A model is defined as "A simplified or idealized description or conception of a particular system, situation, or process, often in mathematical terms, that is put forward as a basis for theoretical or empirical understanding, or for calculations, predictions, etc.; a conceptual or mental representation of something" [5]. Thus models, including models used for simulation are not intended to be exact representations of a system; rather they should characterise a system sufficiently well such that useful inferences can be made.

Despite the merits of simulation, the validity of simulated models can be difficult to prove. Nonetheless, simulation is an accepted approach in many fields, especially in modelling of physical systems [11] and the soundness of results can usually be established using systematic validation techniques [12],[14]. In some cases, simulation is the only approach that can obtain insight into complex systems and even opponents to the approach must accept the merits of simulation.

The value of simulation in project management has been recognised for over forty years in the context or PERT. More recently, Cooper has investigated the importance of rework discovery time using simulation [15] and Cho and Eppinger have used simulation to model product development processes [16] while O'Donovan has developed a software tool for simulation based on the signposting approach as described below [17]. All of these tools aim to answer "what-if" questions relating to different project uncertainties. Probability density functions are commonly used to compare data from different simulation runs as shown in figure 1 (The hatched area represents project overruns).

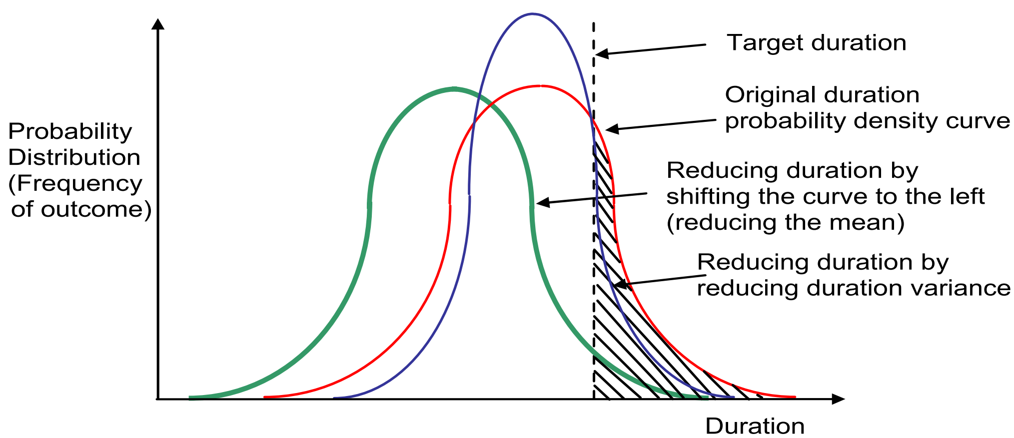

Figure 1. Probability density function for comparing simulation analysis results

3.3 Signposting

Signposting is a dynamic design process modelling approach that can also be used for project planning and simulation. Originally the signposting approach was developed to guide designers to the next task, by showing designers those tasks for which they had appropriate input data to advance the state of the design knowledge [18]. The technique also supports optimum task ordering for the entire project by selecting the most appropriate option from a list of available tasks [19],[20]. The technique is extremely versatile and can model project specific constraints, as well as alternative routes through the design process. Signposting models capture task connectivity due to parameter quality [18] – parameters mature during the design process as they are worked on by different tasks. The state of a parameter is indicated in terms of the subjective confidence that the designer has in its refinement. A task order is implicit in the confidence values and the effect that one task has on another is determined by confidence mapping. Task failure is represented by reduced confidence in the parameter output as opposed to the input and a probability specifying the likelihood that a failure occurs. Partial failure can be modelled by creating a confidence mapping that is lower than required but not below the input level of confidence. Both kinds of failure drive iteration in the model.

A project simulation tool, based on the Signposting approach was developed by O'Donnovan, who also extended the Signposting model to include such features as resource constraints and learning during rework [17]. Extensions to the tool were developed by the authors to facilitate process visualisation. This enhanced tool comprised the engine for the simulation analyses reported in this paper. In parallel to the work reported here, which focuses on model analysis, Wynn has concentrated on improved signposting model elicitation support and representation in the context of aero engine development [21].

4. Generating design project models

Previous simulation work concerning design projects has focused on capturing the real-world projects and exploring variations to these systems [15],[16],[17]. Our approach was to synthesise models of design projects and to investigate variations in abstract project properties and to analyse how the structure of the model itself affects the iteration behaviour of the process.

Risk assessment is concerned with a priori understanding of risk in projects. It is performed at a point when the project does not exist other than through its representation in a model. Even when a project has been executed, it could be argued that it does not have a real objective existence, but is a mental and social construct of its participants (e.g. Checkland [22]), who interact with the project based on its description. The way a project is described and modelled profoundly affects how the risk is assessed and the project carried out. Using generated models, we can look at the structural properties of models, in terms of degrees of detail, number of iterative tasks, and number of parallel tasks to explore their effect on project risk.

4.1 The argument for model generation in design

The use of generated models allows consideration of project properties and uncertainties independently of the constraints of any real-world project. This overcomes concerns about bias during the interpretation of simulation analyses for real-world projects. We can investigate the impact of increased task concurrency, changing project scale, timing of task failure and rework as well as the resetting effect of gateways.

Different project properties (e.g. duration and resource uncertainty, rework likelihood) can be systematically perturbed. The resulting deviations from the plan predicted using simulation as can the consequences of combinations of possible events. Such perturbation analyses allow us to ask different "what-if" questions relating to different project uncertainties. Numerous different scenarios can be modelled and tested in a controlled manner, such that statistically valid results may be inferred. Learning from these theoretical models can then be transferred back to actual projects enabling planners and managers to identify opportunities to relax constraints and direct their efforts to improve process improvement.

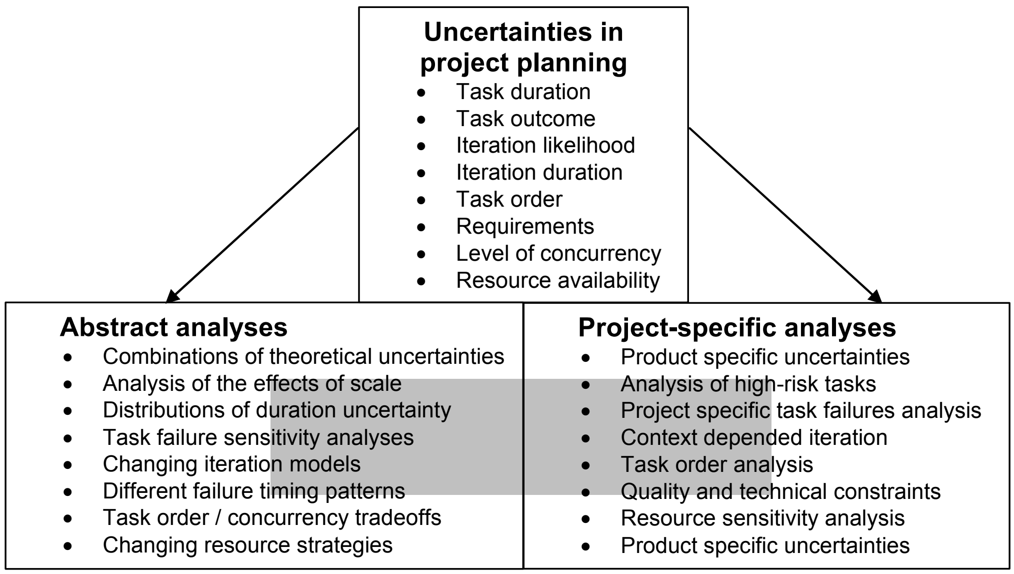

Figure 2. Investigating different project uncertainties

Figure 2 illustrates how insights into different project uncertainties can be gained through both abstract and project specific analyses. Although it is possible and useful to simulate real-world projects, the specific characteristics of the underlying product are likely to constrain the investigations that can be carried out. Further, there are inherent limitations on what can be learned by examining previous projects as all projects are different by definition.

For example, it makes little sense to allocate a normal distribution to the duration of a real-world task if the exact time is known with high accuracy. Conversely, investigating the response of a project to different distributions of duration uncertainty (normal, bimodal) makes sense for an abstract, generated model since there is no real-world instance of the generated tasks. Similarly, it makes sense to vary project properties, such as scale, for a hypothetical project but such variations are not meaningful in the context of a real-world project which has a predefined scale. In addition, the data gathering effort would be immense (many planners and designers are busy and do not have time to perform extensive project post-mortems).

While this paper reports the simulation analysis of generated models, the same simulation functionality for varying model characteristics can be used to systematically investigate the affects of variations in real-world project models (Fig 3). Project specific real-world scenarios can be simulated to determine the likely outcome of possible combinations of different task uncertainties subject to the quality and technical constraints associated with a given product. The same model perturbation and visualisation functionality comprise an important part of the analysis of both real world and generated models.

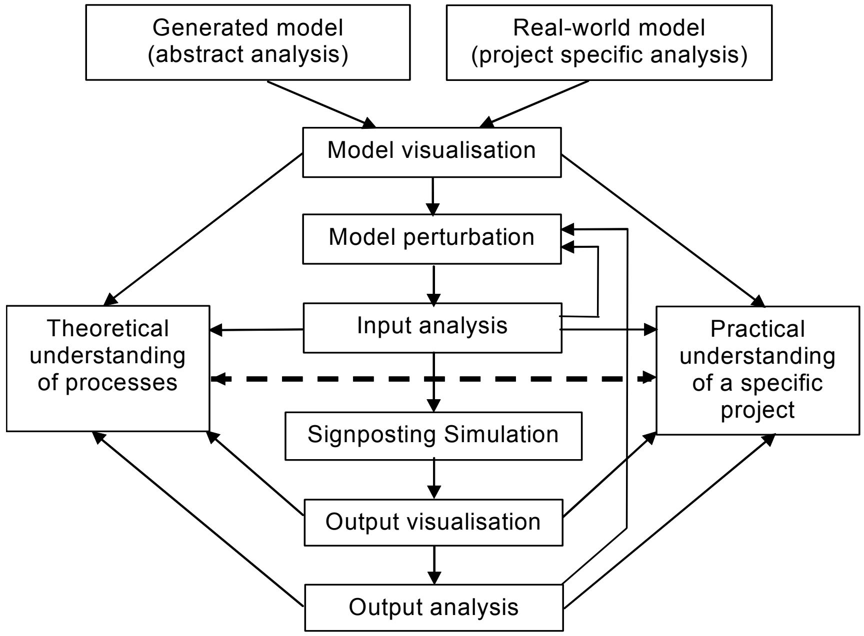

Figure 3. Project model analysis

4.2 Architecture of the model generator

Prior to implementation, the architecture for the model generator was conceptualised as described in figure 4. We chose the signposting approach to represent individual task instances because of its ability to model intricate task behaviour at a high level of detail. The specific properties of the tasks such as cost, duration and failure probability are randomly assigned but constraints can be applied to reflect different industrial contexts. Generated project models are constructed by creating several task instances and linking these together through dependencies. Task alternatives are produced by cloning specific tasks and modifying its characteristics to reflect changes in input information state. Due to the richness of the signposting framework, we can model uncertainty in task duration, number of tasks, connectivity level between tasks and risk of task failure. We are also able to model different types of task failure, as described in the section 5.2.

Both the scope of the model and the system requirements were refined by focusing on the required model outputs and measures, as determined from the objectives. Efforts were also made during the architecture conceptualisation stage to ensure that generated models were compatible with the existing signposting simulation tool.

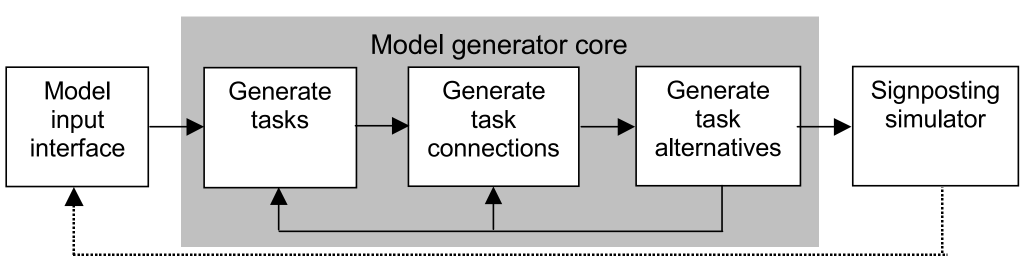

Figure 4. Schematic of the model generator

The model generator and simulator comprise five modules:

- The information input module allows the user to specify characteristics of the model to be generated such as project scale, connectivity level between tasks and information on task risk.
- Based on this information, the model generator generates a number of tasks and assigns properties such as cost and duration
- Next, the model generator assigns dependencies between tasks.
- Finally, task alternatives are created by cloning and perturbing tasks and merging tasks together.
- The result from the model generator (an abstract design project) is fed into the signposting simulator allowing analysis of the project.

Outputs from the Signposting software can be used to guide user inputs in creating further models by perturbation to explore "what if questions". Based on the analysis of several variations in model properties (Fig 5), insights into the project can be obtained by considering different "what-if" questions.

As the simulation software was implemented in java; the simulation language did not significantly restrict what could be modelled. Nonetheless, efforts were made to keep the implementation as simple as possible, while still capturing the complexity of the underlying system [23]. Verification of the code was performed by generating simple models and working through the calculations by hand.

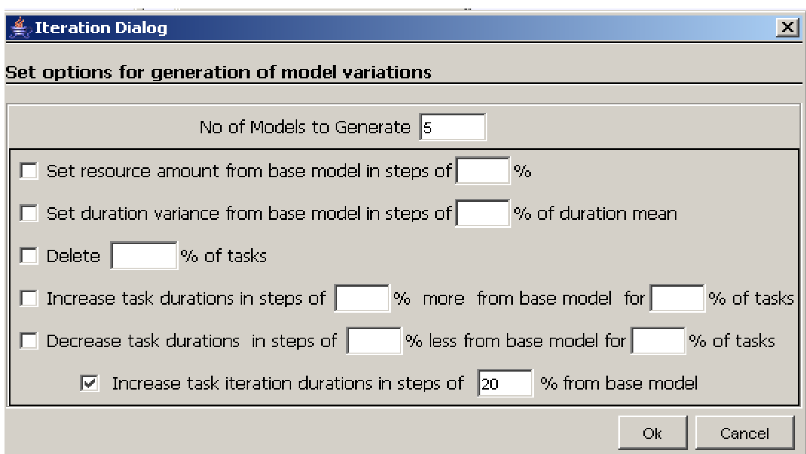

Figure 5. Generating model variations

5. Insights for the design of robust projects

Our research so far yields two important results for robust planning of design projects. Firstly, this work shows how project respond to different task behaviour during rework. This reflects real-world project where the level and form of rework varies significantly for different tasks. Secondly, the results show how more robust plans can be created by considering the likely consequences of different task failure modes (as described below) when determining the task order.

5.1 Analysis of different rework behaviours

The effects of different iteration behaviour models (parallel iteration, accelerated iteration and slow iteration as described below) on project duration are shown in figure 6. All three histograms are based on the same model of task connectivity and resource availability but different task rework behaviour is considered each scenario.

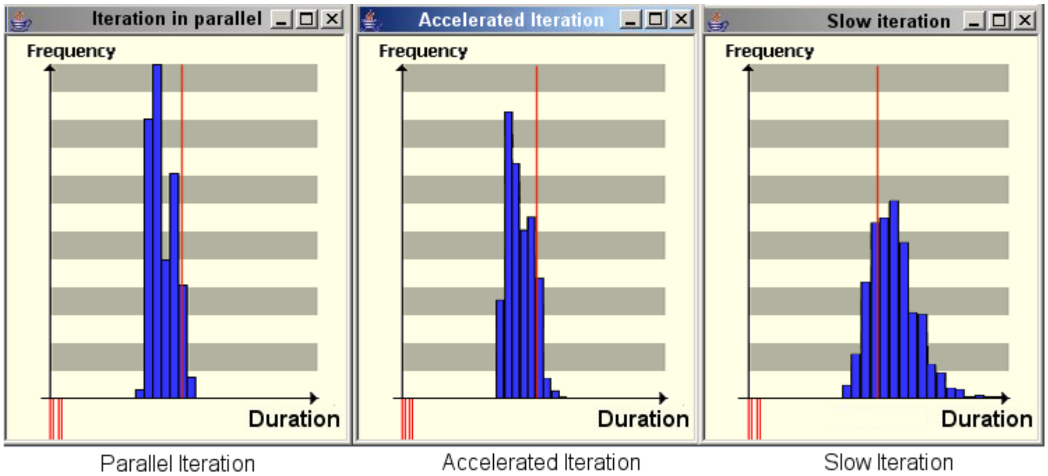

Figure 6. Rework behaviour model has a strong influence on project duration

The results, based on 3000 simulation runs, show that the choice of iteration model has a major influence on the project duration, even when deterministic (as opposed to probabilistic) task durations are used. Although it makes intuitive sense that project duration will increase in response to longer rework times, the results highlight the importance of correctly predicting the type of iteration in design project planning. The target duration for the project, represented by the vertical red line, can be used as a reference point for comparing the different simulation results. The mean and standard deviations for project duration are presented in Table 1.

Table 1 Comparison of different iteration behaviours

	Duration mean	Standard deviation
Parallel iteration	159.80	14.06
Accelerated iteration	167.61	15.99
Slow iteration	198.91	30.62

Parallel iteration: On the left hand side of the figure, the probability distribution for project duration subject to parallel iteration is shown. (This is modelled by allocating a rework tasks zero duration). While this model is an oversimplification of reality, it underscores the importance of iteration when estimating project duration: as all tasks durations are modelled as deterministic, all project duration variability is due to rework. The reason for this increased duration is discussed below (section 5.2).

Accelerated iteration: The second iteration model shown in figure 6, assumes that tasks take only a fraction of their original duration (20%) due to learning affects in rework. This model is appropriate for some design tasks where minor modifications are required based on information which was not available at the time when the task was first performed. Because this model of rework influences the resource pool, project duration suffers when compares to the simpler model. However, the impact is not dramatically worse than the zero-duration rework model due to the high speed of reworked tasks.

Slow iteration: Slow iteration assumes that tasks aren't accelerated during rework and that this rework draws resources from the project resource pool. Our results show that project duration distribution varies drastically when this model of rework is used.

We also simulated iteration in which task duration during rework is twice the original. This scenario could occur if unforeseen problems arise and require a complete rethinking of the design approach, or where iteration is slow due to poor resource availability. The results were similar to those obtained for the slow iteration model shown in figure 6, but the mean and standard deviation for project duration increased.

The above results show that the high sensitivity of the project to changes in task behaviour during rework. Many planning tools, such as Gantt charts fail to recognise iteration at all; while others, such as DSMs, highlight tasks that will iterate but don't consider the iteration behaviour. This work shows that it is not sufficient to identify the existence of rework; the specific nature of rework must also be considered.

5.2 The importance of task order on project duration

The parallel iteration scenario analysis shows that even when even when the actual rework activities do not affect the resource pool, they can still have a major indirect influence on project duration by modifying the task order. This happens because the lower confidence state, resulting from failed and partially failed tasks, limits the choice of available tasks and ultimately delays the project (section 2.4). An example of this situation would be when a subcontractor provides insufficient or inaccurate data about a component design; inaccurate data leads to rework of dependent tasks while insufficient data delays their execution (Fig 7).

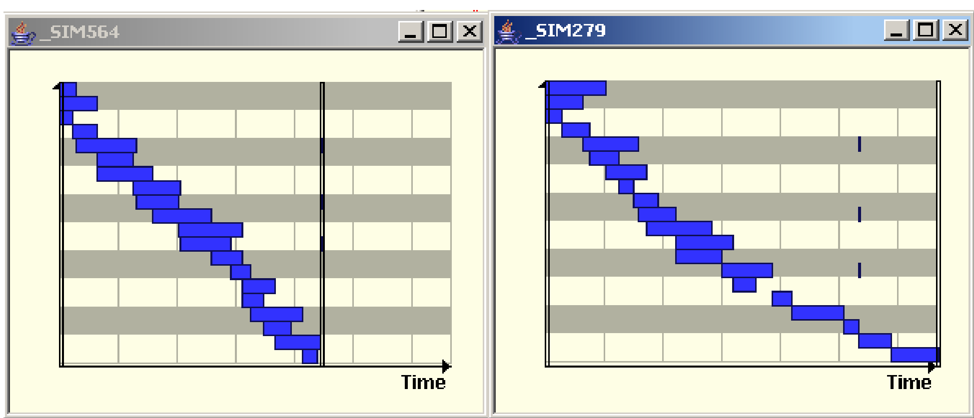

Figure 7. Changing task order due to iteration

In some cases, project duration can be reduced by appropriate consideration of rework behaviour when determining the task order [19]. A simple example helps explain our observations. Consider the design of two components A and B. Task B can create information which drives rework of task A. If task B fails to achieve a satisfactory outcome, rework may be required in both tasks; this is not true of task A. Thus, performing task B before task A is likely to reduce the total rework associated with the project (Fig 8). In both situations, task B fails. However, due to the improved task order, the resulting rework is reduced in the more robust plan as task A is performed only once.

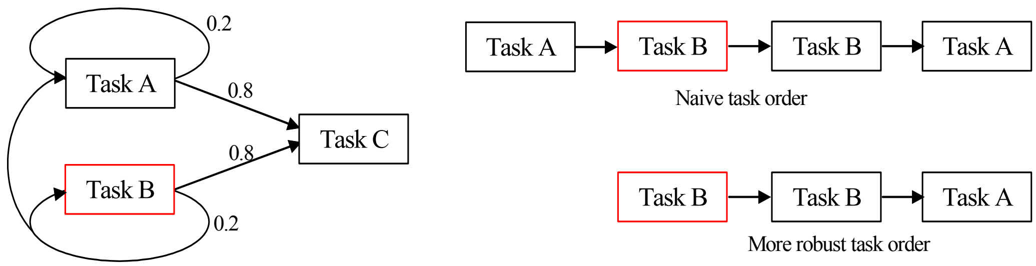

Figure 8. Task order should reflect task failure mode

Hence, it is not sufficient to consider the likelihood of task failure as an indicator of project risk; the rework characteristics of dependent tasks must also be taken into account. This applies for all models of iteration described above but that it will have the biggest impact for slow iteration scenario, as rework duration in slow iteration is longest.

6. Conclusions

Uncertainty in complex engineering projects is a multifaceted issue and resulting problems are a concern in industry. Our research uses the signposting framework to simulate variations in task orders against the backdrop of different task behaviour during rework. Based on simulation results, characteristics of robust plans were identified, which reduce the impact of task failures on the overall project. Simulations of design project plans demonstrated that the rework behaviour of tasks has considerable impact on project duration and that the influence of task failure mode on project success can be alleviated by appropriate task sequencing.

The research also introduced the concept of model generation to explore the properties of design projects; a research avenue which presents numerous opportunities for further research. Such research could focus on exploration of the different industrial challenges described in section three through the execution of additional sensitivity analyses. In addition, combinations of different uncertainties could be simulated to determine their impact on the project and alternative task behaviours during rework could be modelled.

In essence, the results from this research illustrate how project duration in engineering design can be diminished through sensible task ordering and resource allocation. This approach facilitates design project planners in creating more robust plans, where the affects of task failures on the overall project are reduced.

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] Simpson, J., and Weiner, E.C., (Eds) “Oxford English Dictionary”, (Second Edition) Clarendon Press, Oxford, ISBN: 0-19-861186-2, 1989

[6] Policella, N., Smith, S., Cesta, A. and Oddi., A., “Generating Robust Schedules through Temporal Flexibility”, 14th International Conference on Automated Planning and Scheduling, Whistler, Canada, 2004

[7] Eckert, C.M., and Clarkson PJ. “The reality of design process planning”, in ICED03, 14th International Conference on Engineering Design, Stockholm, Sweden, 2003.

[8] Kerzner, H. “Project management: a systems approach to planning, scheduling, and controlling”, 4th ed. ISBN: 0442010850, Van Nostrand Reinhold, New York, 1992

[9] Horowitz, J., “Critical path scheduling: management control through CPM and PERT”, Krieger Publishing, Malabar Florida, ISBN: 0898740886, 1967

[10] Clarkson, P.J. And Eckert, C.M., “Design process improvement - a review of current practice"”, Springer, 2004.

[11] Johnson, J.., “The "Can You Trust It?" Problem of Simulation Science in the Design of Socio-technical Systems”, Complexity, Vol. 6, No. 2, 2001, pp.34-40.

[12] Pidd, M. “Computer simulation in management science”, Third Edition, ISBN: 0-471-93462-3, Wiley, 1992

[13] Zeigler, B.P., “Theory of modelling and simulation”, Wiley – Interscience, New York, ISBN: 0471981524, 1976

[14] Sargent, R.G. “Verification and validation of simulation models”, Proceedings of the 30th conference on Winter simulation, Washington, D.C, ISBN:0-7803-5134-7, 1998

[15] Cooper, K.G., “The Rework Cycle: Why Projects Are Mismanaged”, PMNetwork (February 1993), pp 5–7.

[16] Cho, S-H., and Eppinger, S.D., ”Product Development Process Modelling using Advance Simualtion”# DETC 2001, Pittsburgh, USA.

[17] Donovan, B., Clarkson, P.J. and Eckert, C.M., “The simulated design process”, Proceedings of ICED’03, Stockholm, Sweden, 2003

[18] Clarkson, P.J. and Hamilton J.R., “Signposting: a parameter-driven task-based model of the design process”, Research in Engineering Design, 2000, pp18-38.

[19] Melo, A.F. and Clarkson P.J., “Design process planning using a state-action model”, Proceedings of ICED’01, Glasgow, UK, 2001.

[20] Jarrett, J.P. and Clarkson P.J. “Improving the product by accelerating the process”, Proceedings of ICED’01, Glasgow, UK, 2001.

[21] Wynn, D., Clarkson, P.J. and Eckert, C.M., “Addressing uncertainty in design planning and modelling: a case study of complex aerospace design”, submitted to DTM 2006

[22] Checkland, P., Systems Thinking, Systems Practice, Chichester, John Wiley, 1981

[23] Chwif, L., Barretto, M.R., and Paul, R.J., “On Simulation Model Complexity”, Proceedings of the 2000 Winter Simulation Conference, Florida, USA, 2000

Tomas Flanagan, Engineering Design Centre, University of Cambridge,
Trumpington Street, Cambridge, CB2 1PZ, United Kingdom
Tel.: +44 1223 332828, Fax: +44 1223 766963
Email: tlf21@cam.ac.uk
URL: www-edc.eng.cam.ac.uk/people/tlf21.html