J. ENG. DESIGN, 2000, VOL. 11, NO. 3, 211–224

Evaluation of design process alternatives using signal flow graphs

OLA ISAKSSON[†][‡][*], SVEN KESKI-SEPPÄLÄ and
STEVEN D. EPPINGER[¶]

The introduction of new design activities into an established product development process may involve more work in the initial stages of development, yet this extra effort may reduce the need for more expensive and time-consuming redesign activities later in the project. We have studied a case where more intensive use of computational simulations in the early design phase means that fewer hardware tests are needed because the designs can be analytically evaluated in advance of physical testing. Total development lead time and cost can thus be significantly reduced. This paper addresses how to evaluate alternative design strategies and methods with respect to their impact on the development process time. This is achieved by analysing the design process using signal flow graphs. The technique has been applied to jet engine component development projects at Volvo Aero Corporation in Sweden. We have found that evaluating alternative processes using signal flow graphs not only is helpful to assess the effect of introduction of new or improved design activities on the development process, but also is a means to facilitate the discussion of process improvement alternatives and trade-offs for an organization.

1. Introduction

When alternative product development methods become available, we need a rational way to choose among the alternatives. There are several reasons to introduce new activities within a product development process; these motivations broadly include reducing development lead time and cost, increasing process flexibility, improving product quality, increasing product performance, and reducing product cost. Suggested process improvements must be evaluated with respect to these goals. One important effect of a proposed new design technology may be to alter the completion profile of a product development process in terms of completion probability versus duration. Consider the following scenario, which motivated this research.

Volvo Aero Corporation (VAC) in Sweden is a company that develops, produces and maintains aero engines, most often in collaboration with other engine companies. The aero engine industry is constantly seeking to improve the ability to predict technical performance and properties of their products. Computational methods, such as finite element analysis and computational fluid dynamics (CFD)

[†] Division of Computer Aided Design, Luleå University of Technology, S-971 87 Luleå, Sweden.
[‡][*] Volvo Aero Corporation, S-461 81 Trollhättan, Sweden.
[§] Department of Machine Design, Royal Institute of Technology, S-100 44 Stockholm, Sweden.
[¶] Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02142, USA.
[*] To whom correspondence should be addressed.

Journal of Engineering Design
ISSN 0954-4828 print/ISSN 1466–1387 online © 2000 Taylor & Francis Ltd
http://www.tandf.co.uk/journals

are increasingly used to perform these predictions of product behaviour during development. However, these new design methods and strategies affect the flow and timing of activities within the product development process.

Decisions to introduce new methods and tools are often based on 'soft' information such as management skill or personal experience (Naudé *et al.*, 1997; Upton, 1997). Objective evaluation methods have mainly been restricted to software and hardware performance tests, and not focused on the effect that the tools may have on the development process. Making sound decisions for the introduction of new simulation technologies into the design processes therefore contributes to the ability of maintaining and improving competitiveness. To maintain control of actions in development situations is a key to competitive advantage (Wheelwright and Clark, 1992). In analogy with evaluating product performance using computer model simulations, evaluation of process performance using process simulation techniques is addressed in this paper.

Analysis of design process alternatives requires the options to be carefully understood and described. We argue that deeper understanding gained from analysing the design process improves the decision basis for evaluating and selecting the best process alternative.

In this article, a method for evaluation of design process alternatives has been applied to development projects at VAC. The application area of interest is development of mechanical components for high-temperature applications in jet engines. Development of jet engines and jet-engine components requires extensive testing for both design and qualification purposes. Physical hardware testing in engines is both expensive and time consuming. It can be argued that simulation techniques can reduce the number of hardware tests needed since these simulations reduce the risk of late re-designs. Ideally, no re-design is necessary when the design is tested in the engine. The trend is to make several design iterations using numerical simulation tools, and to use physical testing mainly for verification and qualification.

1.1. Design process alternatives

There are many possible strategies to design any product, and the most capable firms continuously seek alternative methods in an effort to improve their development processes. Several authors (Steward, 1981; Kuisak and Wang, 1993; Eppinger *et al.*, 1994; Smith and Eppinger, 1997) have investigated how to understand and re-sequence existing design activities within complex development processes to avoid large design iterations, i.e. iterations involving many activities and/or activities with long duration.

Another approach to improve the product development process is to introduce new design tools and methods. Examples include new simulation techniques, prototyping methods, testing procedures, or analysis programs. In this paper, we compare alternative ways to utilize CFD analysis within the development process for jet engine components at VAC.

Two principally different CFD simulation support strategies have been investigated and compared with a reference case where no CFD simulation has been used. First, computational simulations can be used to enhance the quality of 'traditional' evaluation tests, such as engine rig tests. In this case, the simulation support is made entirely in parallel with an existing design activity, as an add-on activity. Second, computational simulations can be used directly in the definition of a product, i.e.

prior to any hardware manufacturing. In this case, the simulations may become lead-time critical while they also can affect the hardware design.

1.2. Modelling design processes

The flexibility and quality of product development processes can be evaluated by modelling various features of the design process. Krausse and Raupach (1998) have incorporated resource constraints into a process model, and have addressed the coupling between resource management and development project planning. Adler *et al.* (1995) studied the effect of queuing where there is competition between projects for development resources. Eppinger *et al.* (1997) introduced signal flow graphs (SFGs) as a flexible tool to capture the effect of varying task success probabilities and task durations on the lead times of development projects. Andersson *et al.* (1998) extended the functionality of SFG models by incorporating activity cost into SFG analysis in combination with a Monte Carlo simulation implementation.

To predict features such as success probabilities and lead times of the design process, the process must be modelled with sufficient accuracy. In order to capture the impact of the proposed design process modifications, the resolution of the model must be detailed enough to resolve the relevant design activities, yet general enough to enable comparison between alternative processes.

To predict the lead-time distribution, we chose to use the SFG method, as suggested by Eppinger, *et al.* (1997). We demonstrate by example that the necessary input data are obtainable and that the analytical results are helpful to evaluate the design process alternatives.

1.3. The role of design iterations

The mapping of an industrial development process into an analytical model highlights a number of challenges. Foremost, the role and interpretation of design iterations must be carefully considered since the process modelling inevitably includes simplifications and idealizations.

Development projects primarily follow a company-defined development process. The description of this process dictates the nominal sequence in which the many design activities are performed. In reality, the need to iterate (repeat) an earlier design activity is common. These iterations, i.e. return to a previous design stage, are generally not represented within the prescribed design process and cannot be represented by standard project management tools and techniques. Furthermore, we have no models of changes in information quality as iterations occur.

To accelerate the iterative design process, Smith and Eppinger (1997) suggested that either fewer iterations should be conducted or that iterations should be carried out faster. One conclusion from a study at General Electric (Singh *et al.*, 1992) was that many short iterations early in the development process were preferable to a limited number of longer iterations. In the study, the total development lead-time was shortened from 18 to 7.5 months and the product quality was improved. Increased and/or improved use of simulation tools was suggested to improve design iterations in both of these studies.

In a discussion about the role of design iterations, it is helpful to distinguish between different types of design iteration. We describe design iterations using two dimensions: repetitive versus evolutionary, and intentional versus unintentional. An explanation of these types is provided in table 1 and examples given in table 2.

Term	Explanation
Repetitive	The design criteria and activity remains the same, whereas the design or design parameters are changed.
Evolutionary	If, when analysing the result of an iteration, new information alters the set-up of the design activity. New parameters may turn out to be important.
Unintentional	Unplanned iteration, required when an activity has to be repeated although the 'go ahead' criteria have been met.
Intentional	Planned iteration, e.g. first and second prototype of the iteration needed to converge, or meet the design criteria.

Table 1. Terms used to describe design iterations

	Intentional	Unintentional
Repetitive	E.g. Planned parameter study of wall thickness options.	E.g. Redesign when the intended design does not meet specifications.
Evolutionary	E.g. Planned simulation for identification of critical features, leading to new design criteria or a new simulation approach in the next iteration.	E.g. Redesign due to neglected effects of thermal expansion, leading to new specifications and new design tasks.

Table 2. Examples of the four types of design iterations.

2. Signal flow graph modelling

The signal flow graph is well known as a method for circuit and systems analysis in electrical engineering and for modelling discrete-event systems. Eppinger *et al.* (1997) show that SFG is a powerful tool to model and analyse product development processes, and they provide a more detailed explanation of this method.

The signal flow graph is composed of a network of direct branches that are connected by nodes (see figure 1a). The branches represent the design activities, and the nodes represent states of the process and may involve a probabilistic choice as to which subsequent branch to follow. A special type of branch is one which goes back to a previous state of the process, i.e. design iteration. The simplest form of design iteration is the direct repetition of an activity, i.e. return to the same state. (see figure 1b). If the duration and/or probability are different for subsequent iterations and/or we wish to limit the possible number of iterations, another approach to model the process must be used, i.e. adding new nodes and branches that represent subsequent iterations (see figure 1b). The branch jk depicts the activity when going from node j to node k. Each branch is associated with a quantity known as the *branch transmission*, P_{jk}. The branch transmission includes the probability to execute the task, p_{jk}, and the lead time for the task, t_{jk}.

$$P_{jk} = p_{jk} z^{t_{jk}}$$

The transformation variable, z is used to separate time and probabilities when branch transmissions are multiplied. This is convenient for analytical purposes, since the probabilities are multiplied in the expression and times added in the exponent.

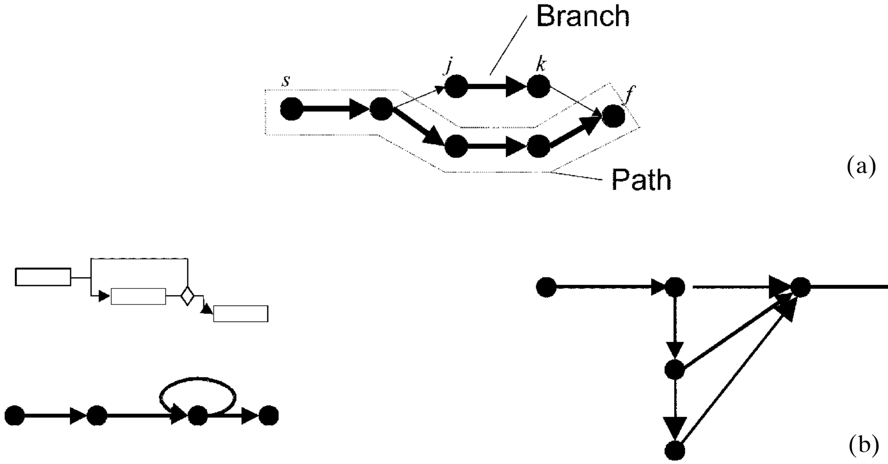

Figure 1. (a) A SFG schema of a simple process . A branch (*j* to *k*) and a path (*s* to *f*) are highlighted. (b) Different ways to model a design iteration. Left, the representation of an infinite SFG 'self loop'; right, a finite number of 'new' design activities, i.e. iterations.

The *path transmission* is then computed as the product of all branch transmissions along a single path. Using this representation is analogous to a discrete sampled data system for which analysis methods are well established.

The *graph transmission*, T_{sf}, is the sum of the path transmissions of all possible paths between a starting node *s* and a finishing node *f*. The graph transmission can be derived analytically, even for complex, iterative situations (Eppinger *et al.*, 1997). This expression is then differentiated and evaluated at $z = 1$ to yield a series of terms of the form $p_i t_i$, summed over all paths, resulting in the expected value of the lead time. The expected value of the lead time, $E[L]$ is defined as the mean lead time.

$$E[L] = \left.\frac{dT_{sf}}{dz}\right|_{z=1}$$

Depending on the lead-time variance, i.e. the distribution of possible lead times and their probability, it is possible to yield much shorter or longer lead times than the mean lead time. It could also be that the mean lead time is not a possible lead time.

We have used a numerical method (programmed using Matlab$^{\text{TM}}$) to identify all possible paths with a lead time below a given value. Since existence of iterations gives an infinite number of possible lead times, the result of the numerical method is a truncation. The accumulated probability is an indicator of the reliability of the answer as it asymptotically approaches unity.

2.1. Modelling inputs and assumptions

By necessity, there are many assumptions made when a SFG model is used. The first and major assumption is that the design process can be described as a number of design activities and design iterations. The second assumption is that it is possible to gather lead-time data for the design activities. How iterations affect the lead time for the repeated activity is difficult to assess. Using project records is a way to obtain these data. Third, we assume that the task durations are accurately represented by deterministic quantities, so we assume that stochastic variations and queuing are negligible.

A fourth assumption is that we know the path probabilities. In situations where the process has been executed many times, statistical data may be available, and probabilities may be obtained with some confidence. However, it is our experience that the design process is often evolving and may be poorly documented. Where no documented data are available, or where the model is used to predict future effects, empirical data may be gathered through interviews.

2.2. *Comparing alternative design processes*

The alternative design processes should be described using the same level of abstraction, and the differences clearly differentiated. We have found it helpful to define a generic base model where the differences of interest are expressed as parameters or as added/removed activities. Criteria for evaluation must also be defined; in our case, it is the probability distribution of lead time to competition.

The alternative processes are then modelled as signal flow graphs, where values of lead time and success probability are assigned to every activity and iteration. The SFG analysis yields the distribution of the possible lead times as well as the expected lead time. Taking the accuracy of input data into account, the most appropriate design process can then be chosen.

2.3. *Simple example*

To illustrate the methodology, we consider a simple design process analysed using the SFG method. This simplified development process is described as three design activities, as shown in figure 2: preliminary design, test and evaluation, and detail design. We are most interested to model the effects of the iteration following the test and evaluation activity. This iteration results from the go/no-go decision taking place after testing. Each of the activities may involve many sub-tasks but this level of detail is not modelled. To simplify the analysis, only design iterations over one design activity are considered.

For the example design process, a simple SFG model is shown in figure 3. The model includes the lead times and success probabilities of the tasks. The process begins with preliminary design and the first execution of test and evaluation, taking 45 and 25 days, respectively. After the first ‘test and evaluation’ activity, there is only a 30% possibility to proceed with the 30-day detail design activity. More likely (70%), a test and evaluation iteration is necessary, taking 10 days, after which there is an 80% possibility to go ahead with the 30-day detail design task. If more iterations are needed, these also take 10 days to conduct, and the possibility to proceed is 80% after each iteration. There is an infinitely small possibility of an infinite

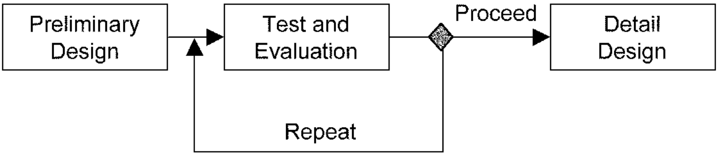

Figure 2. A simple, iterative, design process model.

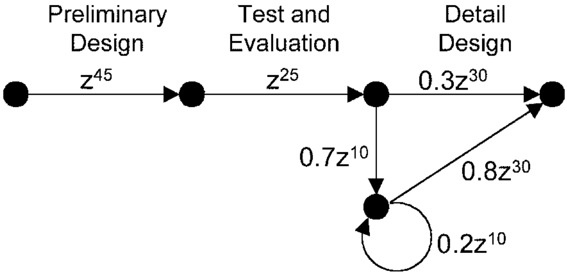

Figure 3. A SFG model of the simple design process.

number of iterations, but the accumulated completion probability asymptotically approaches unity.

The result from the numerical SFG computation is a (truncated) list of all possible outcomes, i.e. a lead time and a probability for each possible path through the signal flow graph. This list can be used to calculate the expected lead time and to create a histogram of probability for different lead times. However, we find that the most interesting result graph is of the cumulative probability of completing the design process as a function of lead time. This result plot is presented in figure 4.

3. Application – jet-engine component design

Development of a jet-engine component consists of a number of *development stages*—typically, a preliminary design stage, a detail design stage, and a qualification

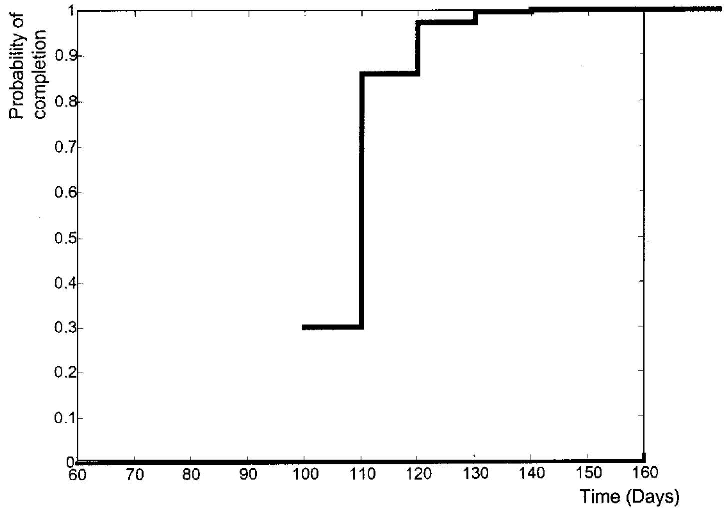

Figure 4. Cumulative probability of completion as computed from the SFG model.

stage. The development stages are divided by critical decision gates, where stage repetition, project continuation, or termination is decided. Each development stage consists of a number of *design phases* where a design phase typically consists of *design activities* followed by *evaluation activities*. The number of design phases scheduled within a stage represents the intentional (planned) iterations. A development stage with two design phases and two activities within each phase is illustrated in figure 5.

The lead time and success probabilities of the development stages are critical for progress of the development project. Failure to meet stage exit criteria has a major impact on the development project and may terminate the project or lead to expensive and time-consuming re-design iterations. Iterations within each development activity would have less severe consequences.

Use of computational simulation support such as CFD in design is a relatively new design process option. Historically, such support has not been possible. As CFD was introduced in development, these methods were mainly used for confirming evaluations and were not used for critical-path design activities. Experience and confidence was gained, and lately CFD has been used for design support, i.e. as a central activity directly affecting the component design (and the critical path).

We now compare three different component design processes for a single development stage at VAC. In these three processes, differing levels of computational support are used. Associated with each design activity are the appropriate task times and iteration probabilities. We compare the processes based on the overall expected lead time and success probability.

3.1. *Design process model*

To compare several alternative design processes, we define a parameterized process on an abstract level valid for all of the situations of interest. Changing the various parameters represents differences between the processes.

The three alternative design processes are illustrated in figure 6. The model describes a development stage including two design phases (D1 & D2). As engine test activities are expensive and time consuming, including hardware construction, the two engine tests (T1 & T2) are planned and scheduled in advance. Even if the

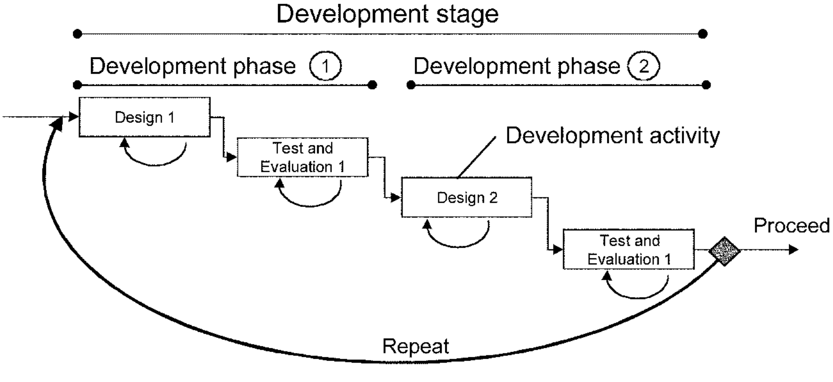

Figure 5. A development stage with two development phases.

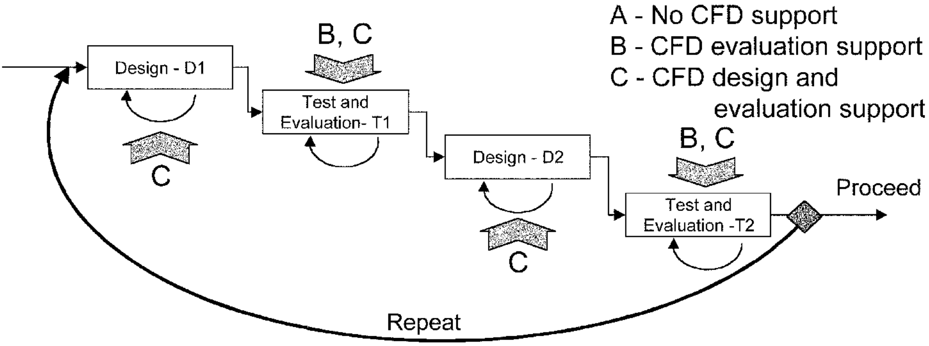

Figure 6. Three alternative design processes. Case A involves no CFD support, case B uses CFD for evaluation purposes and, in case C, CFD is used in design and evaluation.

first engine test shows that the design does not meet all the requirements, only minor design changes are possible within the same test activity. The fact that the second engine test already is planned allows larger design changes, based on the first test result, to be included in the second engine test. The two development phases can thus be seen as intentional/evolutionary iterations (as defined in tables 1 and 2).

In process alternative A, no CFD is used during development. In process alternative B, CFD is used only to support evaluation of engine rig tests. CFD does not become lead-time critical, because it does not directly affect hardware design. In alternative C, CFD is used directly in the design process, so CFD iterations in the design activities are included in the model.

Activities D1 and T1 comprise the concept development phase of the project. D1 is the first design activity, defined as the period from the programme launch, through concept generation and definition until the design is frozen. CFD parameter design iterations, $D1_n$, are considered only in alternative C. No iterations are defined for cases A and B, since these are not relevant for comparison.

T1 is the first engine test activity, which includes generation of manufacturing information (drawings), manufacturing of hardware, instrumentation and rig set up, engine rig test and evaluation of test results. Iterations $T1_n$, representing fixes, but no new testing hardware, take place in cases A, B and C.

Activities D2 and T2 comprise the refinement phase of the project. D2 is a design activity using input from D1 and T1 activities. T2 is the last test activity, followed by the critical decision to go-ahead to the next development phase and execute another test iteration, or to repeat the entire phase.

Ti_1 and Ti_2 are the incremental modifications within engine test, without new hardware manufacturing. Ti_1 and Ti_2 occur for all three alternatives. Di_1 and Di_2 are the CFD design iterations only. These CFD iterations do not include new major design, only parameter modifications of the CFD model.

3.2. SFG model

A signal flow graph capable of representing cases A and B is shown in figure 7. Using null probabilities, the signal flow graph capable of representing C, shown in figure 8, is able to also model alternatives A and B. The SFG model is only

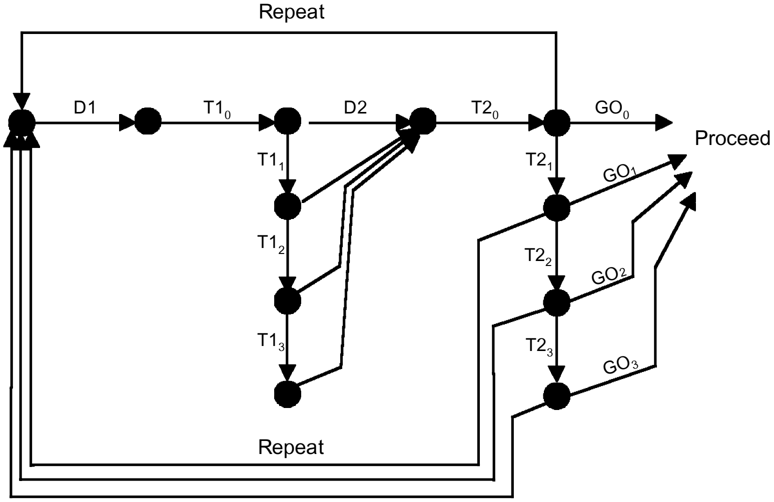

Figure 7. SFG model of processes A and B.

considering 'go ahead' or 're-design' as options after the completed second test and evaluation activity. The 'no go' decision is not modelled.

Each of the branches in the SFG model is assigned parameter values for probability and lead time for their transmission functions pz^t, as listed in table 3. For simplicity, in figure 8, some of the branches leading to the first iteration of each activity are not labelled with indices; however, their probabilities are

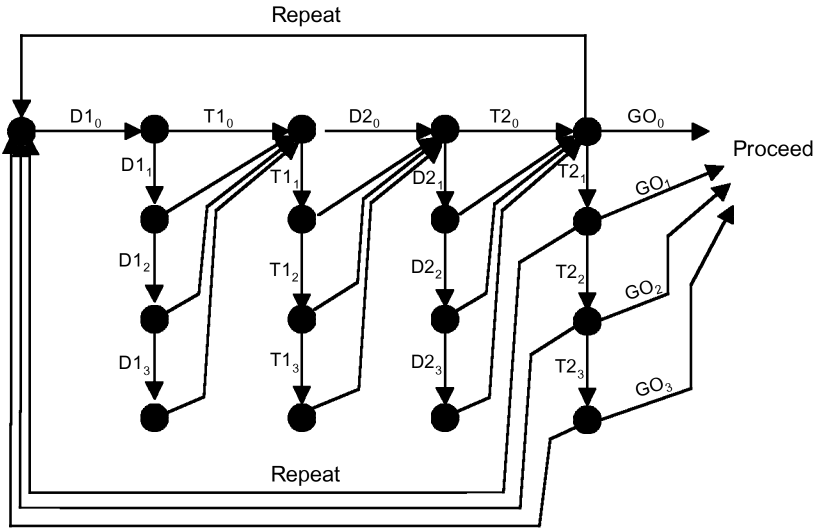

Figure 8. A SFG model capable of representing all three design alternatives by parameter variation.

Parameter	A	B	C	Parameter	A	B	C
$D1_0$	$1.00z^{45}$	$1.00z^{45}$	$1.00z^{75}$	$T2_0$	$1.00z^{150}$	$1.00z^{150}$	$0.70z^{150}$
$D1_1$	$0.00z^{0}$	$0.00z^{0}$	$0.50z^{10}$	$T2_1$	$0.70z^{5}$	$0.70z^{5}$	$0.10z^{5}$
$D1_2$	$0.00z^{0}$	$0.00z^{0}$	$0.35z^{10}$	$T2_2$	$0.50z^{5}$	$0.50z^{5}$	$0.05z^{5}$
$D1_3$	$0.00z^{0}$	$0.00z^{0}$	$0.00z^{10}$	$T2_3$	$0.10z^{5}$	$0.20z^{5}$	$0.00z^{5}$
$T1_0$	$1.00z^{150}$	$1.00z^{150}$	$0.50z^{150}$	GO_0	$0.10z^{0}$	$0.15z^{0}$	$0.80z^{0}$
$T1_1$	$0.90z^{5}$	$0.90z^{5}$	$0.50z^{5}$	GO_1	$0.30z^{0}$	$0.40z^{0}$	$0.90z^{0}$
$T1_2$	$0.20z^{5}$	$0.20z^{5}$	$0.20z^{5}$	GO_2	$0.55z^{0}$	$0.60z^{0}$	$0.95z^{0}$
$T1_3$	$0.00z^{5}$	$0.00z^{5}$	$0.00z^{5}$	GO_3	$0.55z^{0}$	$0.60z^{0}$	$0.95z^{0}$
$D2_0$	$1.00z^{30}$	$1.00z^{30}$	$1.00z^{45}$				
$D2_1$	$0.00z^{0}$	$0.00z^{0}$	$0.30z^{10}$				
$D2_2$	$0.00z^{0}$	$0.00z^{0}$	$0.10z^{10}$				
$D2_3$	$0.00z^{0}$	$0.00z^{0}$	$0.00z^{0}$				

Table 3. Parameter values pz^t for the three design process alternatives. p is the probability to take a branch, t is the time duration of a branch. Note that for cases A and B, some branches are omitted ($p = 0$).

complementary to the other branches—regardless of the number of exit branches from a node, the probability sum is one. The lead times for these branches are equal to the $T1_0$, $D2_0$ and $T2_0$, respectively, as the lead time of the first test or design activity is independent of previous design iterations.

For each of the process alternatives A, B and C, probabilities and lead times have been assigned for each branch. Probabilities have been assessed through multiple interviews with six persons at Volvo Aero, including project leaders, design engineers, simulation experts and test engineers who are, or have been, involved in the development project studied. They were asked to give their answers on probability levels with no higher resolution than 5 percent units. Lead-time data, t, have been taken directly from the project documentation and have been scaled for proprietary reasons.

3.3. SFG results and evaluation

The expected lead time for the go-ahead decision can be derived from the SFG model. Figure 9 shows the cumulative probability of completing the two-phase development stage.

Using CFD support increases the probability of stage completion without redesign from 52% without CFD support to 68% using CFD to support test evaluations, and to 89% using CFD in design. The time penalty of the extra work using CFD is seen in the lowest possible lead times for the different cases. The first stage is completed in between 345 and 375 days for cases A and B, while case C's first stage is completed in between 390 and 450 days.

The expected value for the mean lead time is 642 days for case A, 501 days for case B and 435 days for case C. The lead time for a minimum of 80% probability of stage completion is more than 1000 days for case A, close to 700 days for case B and 415 days for case C.

Depending on how VAC chooses to define the evaluation criteria, different design processes might be selected. In spite of the extra effort added in the initial design stages, the mean lead time is 15% shorter for alternative C compared with case B. It is notable that mean lead times for both case A and case B are on a 'probability plateau', indicating that the mean lead times are not among the possible lead

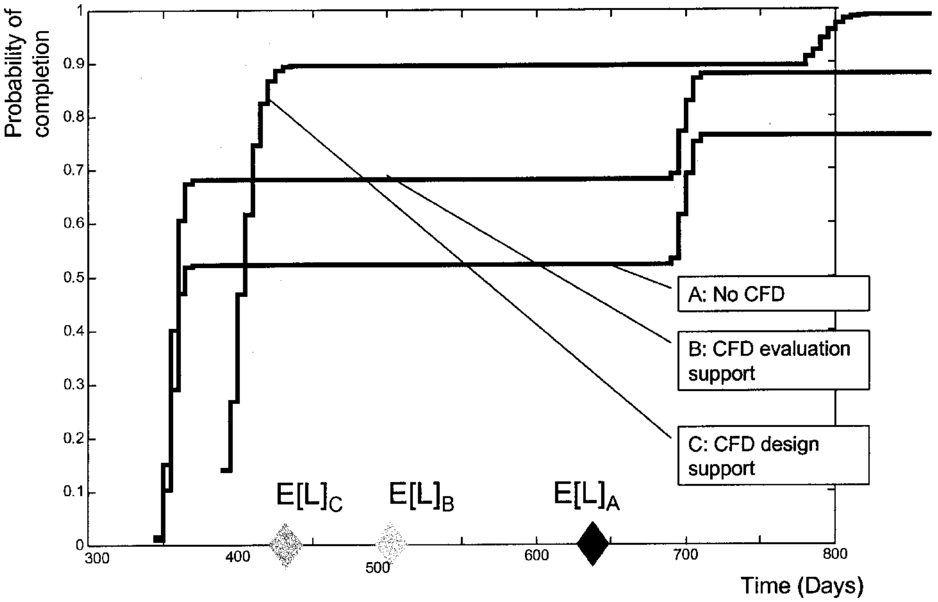

Figure 9. Mean lead times and cumulative probability of lead time ranging over two design phases.

times. This 'probability plateau' is due to the large unwanted iteration including two new engine tests, i.e. it occurs if a re-design decision is made after T2.

If completion probabilities lower than 68% can be acceptable, case B has the shortest lead time, followed by case C, and then case A. If as low as 52% completion probability can be accepted, even case A has a shorter lead time than case C. However, these 'possibly faster' processes have a critical drawback; the slow rise of the right-hand tails of their cumulative probability graphs represent the likelihood of schedule slippage. More pragmatically, for a minimum 80% probability of stage completion, case C is the obvious option, as the 80% probability lead times for cases A and B are 241 and 167%, respectively, of that for case C.

4. Discussion

In this paper, SFG modelling and simulation have been used to analyse alternative design processes. This technique has been used to predict the lead time to completion, taking into account the uncertainties in each activity. Based on the analytical results, the design process options can be readily compared in terms of lead time to completion.

However, there are other important development process metrics to be considered, such as cost, quality and flexibility. Since the SFG analysis is based on uncertainties in each activity, the resulting completion profile makes explicit the reliability of process timing, which is one dimension of process quality. (This is sometimes called schedule risk.) This model can potentially be extended to include activity costs (Andersson *et al.*, 1998) and resource constraints (Krause and

Raupach, 1998), but these have not been the topic of this paper. Using this method gives managers and engineers confidence to adopt new engineering processes more quickly, and thus contributes to a more flexible product development process.

In using this modelling method, it is important to keep in mind the underlying assumptions and limitations of such a technique. The SFG model is an abstraction of a real process, and the analytical predictions are based solely upon the model, not the real situation. The fidelity of these predictions depends to a large extent on how well the model itself represents the situation under study. Parameter uncertainties also affect the accuracy of the results. In our application, some activity timing data used to create the model could be obtained from project records; however, the probabilities had to be estimated by specialists based upon their personal project experiences. To validate our modelling results, managers at VAC are collecting data from current projects for comparison with the timing distributions predicted.

While in this paper we have discussed how to compare and evaluate different product development strategies, it is also possible to approach the inverse problem—to prescribe the necessary performance requirements of new design activities. An insight of the following form could be extremely valuable: ‘We should only adopt this new rapid prototyping method if it reduces the need for confirming experiments by 20% or more’. Of course, the introduction of new design activities may, in turn, affect the timing and success probabilities of other, already existing, design activities. Modelling these secondary effects is essential to assess the overall impact of a proposed change upon the entire development process.

Certainly, important insights are to be gained through the interpretation of the SFG analysis to compare different strategies. However, our experience also shows that even the initial process modelling and data gathering activities do stimulate discussion within the organization, leading to engineering design process improvement.

5. Conclusions

In this paper, we have presented an approach to compare alternative product development strategies with respect to their impact on lead time and project success probability. We argue that the SFG methodology, in spite of the difficulties in generating the appropriate model and obtaining the parameter data, is an efficient method to analyse effects of alternative design strategies on the product development process. An industrial case example illustrates the effect of CFD support activities on project completion time and on the risk of project slippage (due to unplanned design iterations).

We found that the process modelling and analysis also led to managerial insight regarding product development process improvement. This insight can be used for understanding the company’s development process and to discuss the effect of reducing weak links in the process. These results provided a powerful way of explaining the development process at many levels in the organization. The method and results were discussed among engineers as well as at senior management meetings.

We recommend the use of SFG as a tool for predicting lead time and project success probability. The method is particularly helpful to compare alternative suggested product development processes and to quantitatively predict the effects of proposed changes to an established process.

Acknowledgements

We sincerely appreciate the time and assistance provided by the engineers and managers at Volvo Aero Corporation. Financial support for this research was provided by the Swedish Strategic Foundation's national program ENDREA (Engineering Design Research and Education Agenda) and by MIT's Center for Innovation in Product Development (CIPD).

References

Adler, P. S., Mandelbaum, A., Ngyen, V., and Schwerer, E., 1995, From project to process management: an empirically-based framework for analyzing product development time. *Management Science*, **41**, 458–484.

Andersson, J., Pohl, J., and Eppinger, S. D., 1998, A design process modeling approach incorporating nonlinear elements. Proceedings of 1998 ASME Design Engineering Technical Conferences (Atlanta, GA), September.

Eppinger, S. D., Whitney, D. E., Smith, R. P., and Gebala, D. A., 1994, A model-based method for organizing tasks in product development. *Research in Engineering Design*, **6**, 1–13.

Eppinger, S. D., Nukala, M. V., and Whitney, D. E., 1997, Generalised models of design iteration using signal flow graphs. *Research in Engineering Design*, **9**, 112–123.

Krause, F.-L., and Raupach, C., 1998, Simulation of product development processes. Proceedings of the First International Workshop on Intelligent Manufacturing Systems, 15–17 April (Lausanne), pp. 503–512.

Kuisak, A., and Wang, J., 1993, Efficient organizing of design activities. *International Journal of Production Research*, **31**, 753–769.

Naudé, P., Lockett, G., and Holmes, K., 1997, A case study of strategic engineering decision making using judgmental modelling and psychological profiling. *IEEE Transactions On Engineering Management*, **44**, 237–247.

Singh, K. J., Erkes, J. W., Czechowski, J., Lewis, J. W., and Issac, M. G., 1992, DICE approach for reducing product development cycle. *SAE Technical Paper 922139*, pp. 141–150.

Smith, R. P., and Eppinger, S. D., 1997, Identifying controlling features of engineering design iteration. *Management Science*, **43**, 276–293.

Steward, D. V., 1981, The design structure system: a method for managing the design of complex systems. *IEEE Transactions on Engineering Management*, **EM-28**, pp. 71–74.

Upton, N., 1997, Clarifying the business value of computer-based tools. International Conference on Engineering Design (ICED '97) (Tampere), vol. 3, pp. 283–288.

Wheelwright, S. C., and Clark, K. B., 1992, *Revolutionizing Product Development* (The Free Press).