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

IMPROVING THE PERFORMANCE OF A NATURAL GAS COMPRESSOR DESIGN PROCESS

Kian Leong Ong, Md Nor Musa and Ainullotfi Abdul-Latif

Keywords: Concurrent engineering, design process, design management, resource planning, CNG compressor.

1 Introduction

Many modern-day design tasks are knowledge-intensive, interdependent and complex. The interdependencies among design tasks give rise to complex information flows as the execution of a design task may create new information or conditions that affect other interdependent tasks. As information flow is also random, and most of the coupled tasks are executed concurrently, design decisions made using incomplete or imperfect information are re-visited in what is termed *design iteration* [1]. Design iterations were found to account for an average of 33% of total project development time in Intel Corporation's semiconductor division [2]. Consequently, modelling, analysis and improving the performance of design iteration are of great importance in the management of design projects.

2 Background of the study

Classical representations of design processes exist, such as the digraph, Project Evaluation and Review Technique (PERT) and Critical Path Method (CPM), Structured Analysis and Design Technique (SADT), Integration Definition or ICAM Definition (IDEF), *Petri net*; however, they cannot explicitly model design iterations [1]. The compact *Design Structure Matrix* (DSM), first introduced by Steward [3], depicts the task dependencies and the design iterations or information loops in a matrix form.

Based on the DSM, Smith and Eppinger [4] identified the 'controlling features' of coupled, concurrent design tasks, i.e. those elements of a coupled design problem which require the greatest number of iterations to reach an acceptable solution. They postulated a numerical DSM called the Work Transformation Matrix (WTM), in which the measure of the strength of dependency between tasks is the percentage of rework created for one task by work performed by other coupled tasks. McDaniel [5] expanded the WTM to test the impact of different work policies on reducing the lead-time of a design process. Yassine et al [6] extended McDaniel's model to the systems (managerial) level, by managing the exchange of information among coupled design tasks. Although their models can accommodate internal disruptions such as policy changes, these and other analytical methods are inadequate in improving concurrent design tasks which are dynamic.

Smith and Eppinger [4] suggest manipulating the work transformation matrix (state matrix), $\boldsymbol{A}$, to improve the stability of a design process. But as the elements of the state matrix represent the task dependencies, these are not easily changed since these dependencies are governed by the causal connection in information flow. Moreover, it is both tedious and error-prone to try to manipulate the parameters to bring about stability in the face of disturbances. Another way suggested by Smith and Eppinger [4] is by shortening the time required to complete the rework at a stage of iteration. This implies that more resources have to be employed by each design task before the project begins. Browning [7] discusses how to achieve faster and fewer iterations by introducing integrated engineering tools into the design team, improving assumptions used as well as intra-group and inter-team coordination. However, whether faster or fewer iterations, the quality, gain, or productivity of each iteration should not be compromised. Faster iterations will only be advantageous if the quality of work produced is not compromised. The same with fewer iterations. Yassine et al. [6] and Mihm et al. [8] explores some other strategies and mitigation actions which management can take to improve unsatisfactory product development processes, such as minimising delays in information exchange, ignoring low priority interactions, limiting system size, applying robust design methods and modularisation. However, since these and others are merely qualitative suggestions to management, they are unable to determine the amount of additional resources required.

Intuitively, pumping in more resources is a realistic way to improve the performance of a design process. Resource allocation has been identified as a managerial level for controlling the completion rate of product development process, but this is an expensive solution. Thus, it immediately raises the questions of the amount of additional resources and the impact of disturbances even after these additional resources are assigned to the design tasks. Based on the WTM, Lee et al [9] proposed a state space model called Non-Homogeneous State Space (NHSS) model which is able to monitor and control coupled, concurrent design tasks dynamically. Besides controlling design processes in the face of unexpected disturbances, the NHSS can facilitate the planning of resources required to improve the performance of a design process before it begins. This paper demonstrates the application of the NHSS model to resource planning of the design process of a natural gas compressor. Using a case study of the development of a natural gas compressor, two resource allocation schemes are discussed under different scenarios subject to different process improvement objectives.

The ensuing section briefly describes the NHSS model. Section 4 discusses the proposed resource allocation schemes. Section 5 discusses the analysis of the case study with a brief explanation of each scenario under the scenario analysis, describing the background and assumptions employed as well as the managerial interpretation of each scenario. This description is followed by a discussion of the analytical results, especially the general patterns and trends.

3 Non-homogeneous state space (NHSS) model

The DSM is a square matrix that maps out the *information links* among constituent design tasks [3]. A design process comprising n tasks is represented as an $n \times n$ matrix. The non-zero elements a_{ij} denote the information dependency between task i and task j: task j in column j supplies information to task i in row i.

Concurrent design iteration can be expressed by the state equation [9]

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \tag{1}$$

where the index k, the discrete-time variable, represents a finite number of stages of iteration. Each state variable x_i in the state vector $\boldsymbol{x}(k)$ indicates the status of task i at the kth stage of iteration. The state matrix ***A*** is a DSM the elements of which quantitatively denote the interdependencies of tasks, while the input matrix ***B*** represents the proportion of common resources shared by two or more tasks. $\boldsymbol{u}(k)$ denotes the additional resources required by tasks in order for the entire design project to arrive at the desired states. The control input of a task u_i may be the additional resources required by a design task to cope with or to reduce the amount of rework. These additional resources can be acquired through overtime work, hiring new staff, temporary help, outsourcing, technology acquisitions, etc. The unit of measure of a task's state can be one of several, e.g. cost, engineering times, the number of design actions, the amount of rework, etc. As the volume of work is chosen as the unit of measure of state, the state matrix ***A*** is a Work Transformation Matrix (WTM), similar to Smith and Eppinger's [4]. Each of the entries a_{ij} in the WTM ***A*** implies that, as one unit of work is accomplished by task *j,* a_{ij} units of rework are created for task *i*. ***A*** therefore embodies the degree of coupling among the inter-dependent tasks. The state (or work) vector $\boldsymbol{x}(k)$ is an *n*-vector, where *n* is the number of coupled design tasks. Each element of $\boldsymbol{x}(k)$ represents the fraction of the initial work that each task must accomplish after iteration stage *k*. Equation (1) shows that the volume of work that has to be done by a design task is a linear combination of the amount of work generated by other coupled tasks in the preceding stage of iteration *plus* the effect of the control input. The open-loop state space representation of equation (1) incorporating the influence of external disturbances is termed a *non-homogeneous state space* (NHSS) model. A *homogeneous state space* (HSS) model does not consider external disturbances; its response is due only to initial conditions [9]. Thus, the homogeneous state space representation is

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) \tag{2}$$

It is assumed that after each stage of iteration, all rework is completed. Every stage of iteration produces a change in the state vector according to equation (1). To compute the time required to complete all the rework at a particular stage of iteration, a diagonal task times matrix ***P*** is multiplied into $\boldsymbol{x}(k)$ to derive the length of time needed by each task to complete the work during the k^{th} iteration:

$$\boldsymbol{t}(k)=\boldsymbol{P}\boldsymbol{x}(k) \tag{3}$$

where $\boldsymbol{t}(k)$ is the vector of task times for *k*th stage of iteration. Each element of ***P***, P_i, represents the time need by task *i* for a complete execution.

4 Improving the performance of a design process

During the planning stage of a design project, the necessary resources are planned to achieve various managerial objectives subject to constraints; for instance, date of product launch, budget of the project, resource availability, etc. However, opportunities to improve the quality of design and/or hasten the completion will always arise. This immediately raises the question of the amount of additional resources that should be planned in advance. The NHSS model proposed in this thesis is able to analyse the effects of additional resources and determine the required amount of additional resources.

Equation (3) was proposed in section 3 to compute the length of time needed by each task to complete its work during the k^{th} iteration. From this equation, the total time required by task *i* to complete *n* stages of iteration $T_{i,n}$ is therefore

$$T_{i,n} = P_i \sum_{k=0}^{n} x_i(k) \tag{4}$$

In concurrent design, the time to complete n stages of iteration is the time taken by the longest task L to complete n stages of iteration. The longest task, L, is the task with the largest numerical value in $\boldsymbol{t}(k)$, i.e. the task that requires the longest time to complete a stage of iteration. If T_c is the desired total time for the entire design process to complete after n stages of iteration, then

$$T_c = P_L \sum_{k=0}^{n} x_L(k) \tag{5}$$

If the total completion time of a project needs to be shortened, there are two ways of doing so:

1. Allocating extra *initial* resources to task L *before* the process begins, for example, increasing the number of team members for task L, in order to shorten the time needed for a complete execution.

2. Allocating additional resources to inter-dependent tasks *during* the course of the project, for example, implementing overtime, in order to reduce the rework incurred by these tasks.

If extra initial resources are assigned, a new time for the longest task, P_L, is computed from equation (5) based on the desired total completion time. This approach is named *extra initial resource scheme*. The amount of extra initial resources required can be determined through scenario analysis which is demonstrated in the next section.

In the second approach, namely *additional resource scheme*, the required amount of additional resources at every stage of iteration can be determined from the *eigenstructure* and state feedback control (SFC) analysis proposed in [9]. In the analysis, a desired completion state $\boldsymbol{x}_c$ is defined from equation (5) based on the desired total completion time. This approach of expediting the design process is also demonstrated through scenario analysis in the next section.

To improve the quality of design decisions, more iterations should be allowed for more deliberations. More iterations improve the chance that the design will converge to acceptable multi-attribute performance levels [7]. However, more iterations take more time and/or more resources so a trade-off is inevitable. In light of the abovementioned resource allocation schemes, one may allow a design process to undergo more iterations but assure that the completion date is not compromised by assigning more resources. But if no extra resources are available due to budget or other constraints, the project would be forced to either delay its completion or compromise with lower quality of design decisions. The trade-off between faster completion time and better quality of design decisions should be carefully evaluated with other information such as resource availability, product launch time window before the decisions of resource planning can be made.

The scenario analysis of the development of a natural gas compressor discussed in the following sections allows management to observe the effects of resource assignment, the time of completion and the number of stages of iteration under different situations.

5 Design process of a natural gas compressor: A case study

An international petroleum company and Universiti Teknologi Malaysia collaborated concurrently to design a compressor of compressed natural gas (CNG) for refuelling purposes. A new concept of symmetrical wobble plate was employed in the compressor design. A picture of the compressor's conceptual design is shown in Figure 1. While the university focused on the mechanical aspects of design, the company concentrated on the electrical and electronic aspects. This case study focuses on the mechanical aspects. The authors define a design task as beginning with the definition of functions and specifications of the artefact to detailed design and analyses before prototyping.

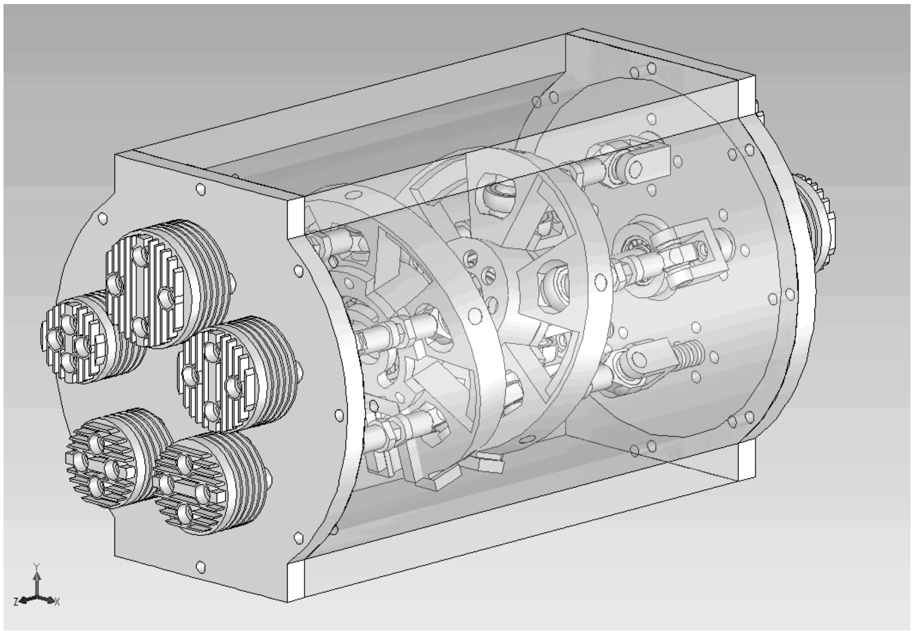

Figure 1. Conceptual design of the symmetrical wobble plate compressor.

The design of mechanical system comprises 12 subsystems: Wobbling Mechanism, Transmission Mechanism, Compression System, Driver System, Gas Dynamic, Cylinder & Casing Assemblies, Lubrication System, Maintenance, Anti-rotating Mechanism, Storage & Piping System, Thermodynamic & Cooling and Motion Work. The 12 sub-systems were designed concurrently as the 8 design engineers from manufacturing, R&D and mechanical engineering were known to be able to work with each other. These 8 engineers were the main resources assigned to the project.

The 8 engineers were interviewed for all the information necessary to construct a numerical Design Structure Matrix (DSM). For instance, they were asked to spell out the precedence relationships among the 12 design tasks and to estimate the cycle time of a complete iteration of each task. An iteration is a design cycle during which some design action is taken to finish the work that was generated by other tasks during the immediately-preceding stage of iteration. In a *complete* iteration, the remaining work is 100%; for example, at the initial execution of a design task. A state matrix, similar to the WTM, was obtained as shown in Table 1. In Table 1, the element in 3rd row and 1st column means that task 1's relationship to task 3 is such that task 3 (Compression System) has to redo 30% of its work after task 1 (Wobbling Mechanism) undergoes a complete iteration. An initial HSS analysis of all 12 design tasks can now be undertaken.

Table 1. State matrix of the symmetrical wobble plate compressor design

Task name	Time (Hour)	ID	1	2	3	4	5	6	7	8	9	10	11	12
Wobbling mechanism	140	1	0	0	0.20	0.04	0.20	0.03	0.04	0	0.04	0.01	0	0.05
Transmission mechanism	140	2	0	0	0	0.05	0	0	0.06	0	0	0	0	0
Compression system	336	3	0.30	0	0	0.02	0.24	0.01	0.14	0.07	0	0	0.20	0
Driver	84	4	0.16	0.06	0.14	0	0.05	0.01	0.09	0.05	0.01	0.01	0.07	0.09
Gas dynamic	280	5	0.03	0	0.05	0.03	0	0.03	0.17	0.17	0	0.01	0.17	0
Block & casing	196	6	0.02	0	0	0.01	0.03	0	0.04	0.03	0	0.01	0.05	0
Lubrication system	420	7	0.16	0.06	0.14	0.08	0.27	0	0	0	0	0	0.20	0.05
Maintenace	140	8	0	0.15	0.07	0	0.08	0.05	0	0	0	0.03	0.17	0
Anti-rotating mechanism	112	9	0.02	0	0.04	0.01	0	0	0.04	0	0	0	0	0.25
Storage & piping system	420	10	0.02	0	0.03	0	0.01	0.01	0	0.03	0	0	0.05	0
Thermodynamic & cooling	560	11	0.07	0	0.10	0.07	0.22	0.07	0.40	0	0	0.05	0	0
Motion work	224	12	0.15	0.08	0.07	0.09	0.21	0	0.20	0	0	0	0	0

5.1 Natural response of design tasks

According to the eigenstrtucture analysis [9], the state matrix's eigenstructure predicts a stable and convergent natural response of all 12 design tasks (see Figure 2). It can be seen from Figure 2 that, assuming that all 12 design tasks started off with 100% of work, the remaining work of all tasks decreased after every stage of iteration, except task 11 (Thermodynamic & Cooling) at its first stage of iteration. There are several reasons for this. The Thermodynamic & Cooling is an important subsystem that determines design parameters of other sub-systems, e.g. Compression System, Gas Dynamic and Lubrication System; but at the same time, it is constrained by heat generation and dissipation of these sub-systems. However, it started together with other subsystems based on some assumptions. After the first execution, information of other subsystems was available and the assumptions were reviewed. This resulted in a major review of Thermodynamic & Cooling since it is strongly linked to 7 subsystems. The slowest task was task 11 (Thermodynamic & Cooling) with the largest value in the critical eigenvector – 0.463. The earliest possible completion of the entire project therefore hinges on task 11.

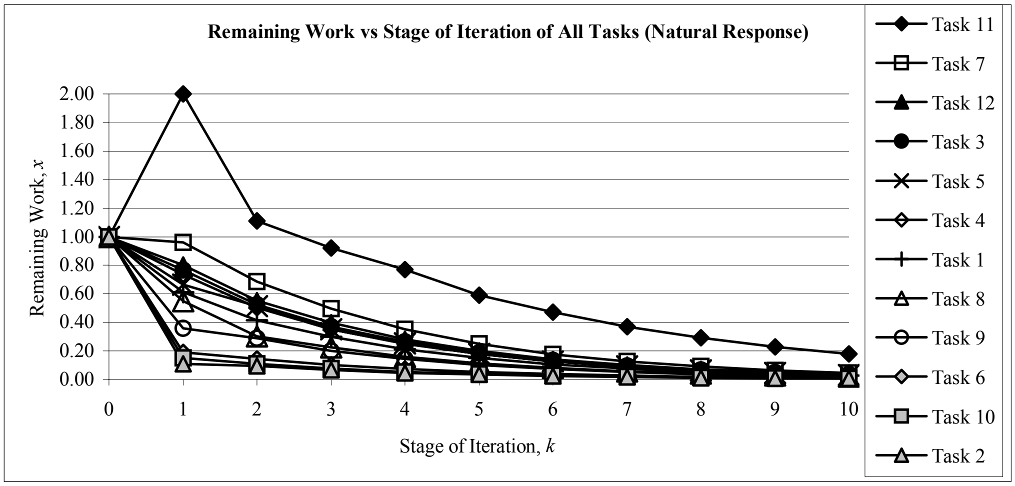

Figure 2. Natural response of the CNG compressor design

Table 2 shows the predicted natural response of the entire mechanical system design, given that task 11 is the slowest-converging task. The mechanical system design is deemed to be completed before the fabrication of the first prototype, i.e. by week 25, after 12 stages of iteration. However, from Table 2, the remaining work of task 11 by week 25 is approximately 12%. If delivery of the design is enforced on week 25, this means that only 88% of task 11 will have been completed. Thus, the predicted completion state of task 11 was set at $x_{c,k=12}$ = 0.12. That is to say, at the time of scheduled completion, the remaining work of each and every of the 12 development tasks should not exceed 12%. Since the first prototype was intended to be built with production-intent parts, i.e. parts having the same geometry and material properties as those in actual production, but not necessarily fabricated by the same manufacturing process, 12% of remaining work is considered acceptable.

Table 2. Natural response of the design process

Stage of Iteration, *k*	**Remaining work of the slowest task (%)**	**Cumulative time elapsed (hours)**	**In week #**	**Status**
0	100	120	3.0	
1	200	360	9.0	
2	111	493	12.3	
3	92	603	15.1	
4	77	695	17.4	
5	59	766	19.1	
6	47	822	20.6	
7	37	867	21.7	
8	29	902	22.5	
9	23	930	23.2	
10	18	952	23.8	
11	15	969	24.2	
12	12	983	24.9	Predicted completion

Once the natural response of the development task are analysed, management can now consider if the development process should be improved using the two resource allocation schemes discussed in section 4. The implication of each resource allocation scheme on the time of completion, number of iteration and resource requirement of development tasks can be exposed through scenario analysis. A base scenario is established based on the system characteristics predicted by the HSS analysis in this section and some assumptions (will be discussed in section 5.2) in order to compare with other scenarios.

5.2 Scenario analysis

We begin with an explanation of the two primary plots of the results of analysis: a graph of design status of the slowest design task (task 11 – Thermodynamic & Cooling), expressed as a percent of remaining work, and a graph of total resource utilisation. Each data point in the graphs represents the design status or total resource utilisation before a stage of iteration. Some assumptions were made to establish a base scenario by which other scenarios would be compared. The natural response presented in section 5.1 is the base scenario. It was assumed to be free of external disturbances and:

1) The desired product content and component cost, investment and quality levels are known beforehand.

2) Tasks were performed with the goal of staying on schedule, not getting ahead of schedule.

Since there was no external disturbance, the behaviour of the design process is due to the interplay among the design process structure, resource allocation policy and scheduled deadline. The task dependencies are inherent features of the design process due to the decomposition of design tasks, regardless of the policies used to manage them or who performs the design task. With the attributes specific to the base scenario identified, the behaviour of the natural gas compressor design such as the natural response and the convergence rate of tasks, was considered as in section 5.1. With the performance and process characteristics of the base scenario established as a reference, the following discussions consider the impact of various process improvements on the overall design process.

The effect of the two different resource allocation schemes (increasing initial resources *before* the project starts and allocating additional resources *after* the project starts) to achieve: (i) faster time to completion, and (ii) higher quality of design decisions, were analysed using the NHSS model. Hence, 4 scenarios will be discussed (see Table 3): scenario 1, extra initial resource allocation to the longest task with faster time to completion (if 5 weeks faster than the base scenario); scenario 2, additional resource allocation to tasks with faster time to completion (if 5 weeks faster than the base scenario); scenario 3, initial resource allocation to the longest task with faster time to completion and with higher quality of design decisions (if 3 more stages of iteration than scenario 1); scenario 4, additional resource allocation to tasks with faster time to completion and with higher quality of design decisions (if 3 more stages of iteration than scenario 2). The longest task of the system development is also the slowest task, i.e. task 11 since it possesses the largest value in $\boldsymbol{t}(k)$.

The extra initial resources in scenarios 1 and 3 could be acquired through hiring new programmers or engineers to be the team members of task 11. However, the assigned extra manpower should join the development process in the very beginning and then work together with the existing project participants through out the entire process. The additional resources required in scenarios 2 and 4 may be sourced through overtime. To examine the effect of resource allocation schemes on different process improvement objectives, the design process was simulated under the conditions stated in Table 4. Note that the desired completion states and stages of iteration described in Table 4 are the key parameters of the eigenstructure assignment proposed in [9].

Table 3. Objective and resource allocation scheme of the 4 scenarios

Scenario	**Objective**	**Resource deployment scheme**
1	Shorten project completion time from 25 weeks to 20 weeks (the target completion time 20 weeks is chosen just for the purpose of demonstration only).	Allocate more initial resources to the longest task *before* the project starts.
2	Shorten project completion time from 25 weeks to 20 weeks.	Allocate additional resources to all tasks *after* the project starts.
3	Shorten project completion time from 25 weeks to 20 weeks but with 3 more stages of iteration.	Allocate more initial resources to the longest task *before* the project starts.
4	Shorten project completion time from 25 weeks to 20 weeks but with 3 more stages of iteration.	Allocate additional resources to all tasks *after* the project starts.

Table 4. Specifications of the 4 scenarios of process improvement

Scenario	**Desired completion state, $\boldsymbol{x_d}$, and desired stage of iteration, $\boldsymbol{k_d}$**
1	The number of iterations to completion and the desired completion states of tasks are the same as those for the base scenario, $x_{c,k=12}$, e.g. the remaining work of task 1 is 12% when the project is assumed completed at the 12th stage of iteration. But the time need by he longest task (task 1) for a complete execution is improved from 120 man-hour to 98 man-hour (computed from equation 5.2). $\boldsymbol{x}_{d1} = \boldsymbol{x}_{c,k=12}$, $k_{d1} = 12$, $P_1 = 98$ man-hour
2	The number of iterations to completion and the desired completion states of tasks are the same as those for the base scenario, $x_{c,k=12}$, except the desired completion state of the longest task (task 1) is 0.115 (determined from equation 5.2), i.e. the remaining work is 11.5% when the project is assumed completed at the 12th stage of iteration. $\boldsymbol{x}_{d2} = [0.115\ \ 0.083\ \ 0.076\ \ 0.089\ \ 0.083\ \ 0.053\ \ 0.03\ \ 0.045\ \ 0.047\ \ 0.053\ \ 0.038\ \ 0.026\ \ 0.047\ \ 0.023\ \ 0.049\ \ 0.025\ \ 0.02\ \ 0.025\ \ 0.09\ \ 0.035]^T$, $k_{d2} = 12$
3	The number of iterations to completion and the desired completion states of tasks are the same as those for the base scenario with extended iteration to 15 stages, $x_{c,k=15}$, e.g. the remaining work of task 1 is 6% when the project is assumed completed at the 10th stage of iteration. But the time need by he longest task (task 1) for a complete execution is improved from 120 man-hour to 95 man-hour (computed from equation 5.2). $\boldsymbol{x}_{d3} = \boldsymbol{x}_{c,k=15}$, $k_{d3} = 15$, $P_1 = 95$ man-hour
4	The number of iterations to completion and the desired completion states of tasks are the same as those for the base scenario with extended iteration to 15 stages, $x_{c,k=10}$, except the desired completion state of the longest task (task 1) is 0.057 (determined from equation 5.2) , i.e. the remaining work is 5.7% when the project is assumed completed at the 15th stage of iteration. $\boldsymbol{x}_{d4} = [0.057\ \ 0.042\ \ 0.038\ \ 0.044\ \ 0.041\ \ 0.026\ \ 0.015\ \ 0.022\ \ 0.024\ \ 0.026\ \ 0.019\ \ 0.013\ \ 0.023\ \ 0.011\ \ 0.025\ \ 0.012\ \ 0.01\ \ 0.013\ \ 0.045\ \ 0.017]^T$, $k_{d4} = 15$

Figure 3 shows the progress of the compressor under every scenario in week, which is the progress of the slowest task (task 11) since the progress of the entire development process is hinged on it. Figure 4 indicates the total normalised resource usage by the design process under every scenario. It should be kept in mind that the resource levels indicated in this figure are normalised with respect to the initial resource level required to complete the amount of work of a task for the first time, i.e. a complete execution. Normalisation of resource is consistent with normalisation of work, whereby the absolute units are transparent to the model. This is necessary as there is no single universal measure [4] of design status applicable to all design processes. Table 5 shows the resource usage of each task in all scenarios. The results of the scenario analysis will be discussed in detail in the following sections.

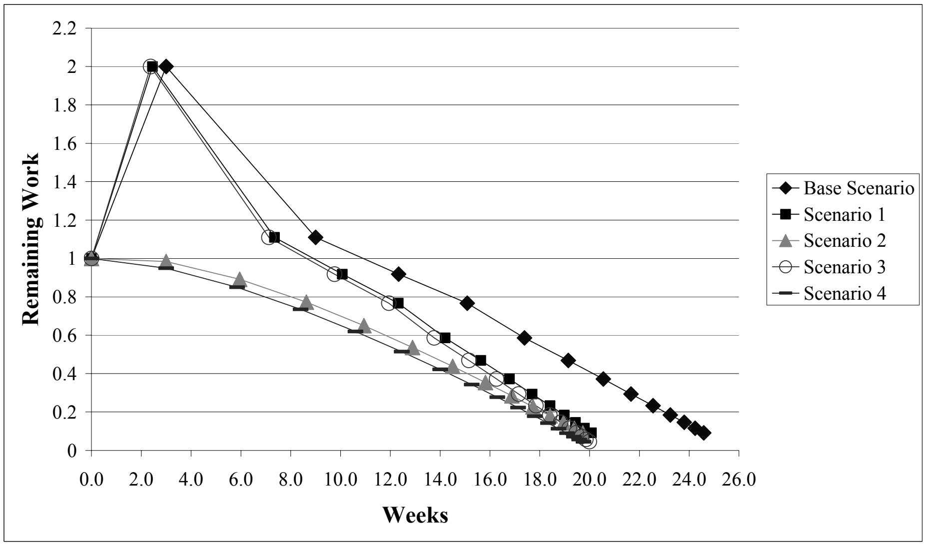

Figure 3. Progression of scenarios

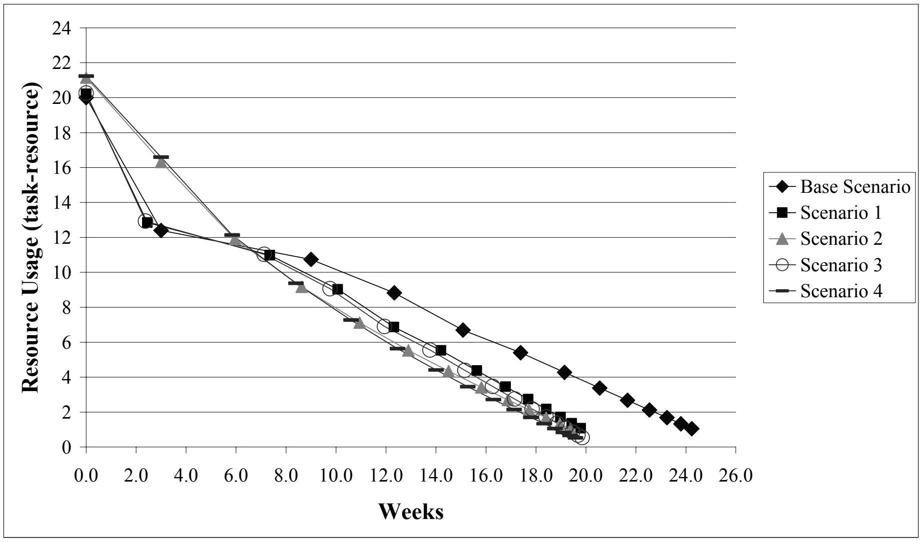

Figure 4. Total normalised resource usage under all scenarios

Table 5. Resource usage of each task under every scenario

Task	Resource usage (task-resource)				
	Base scenario	Scenario 1	Scenario 2	Scenario 3	Scenario 4
1	3.08	3.08	3.73	3.15	3.77
2	5.78	5.78	5.79	5.94	5.94
3	5.45	5.45	5.47	5.59	5.61
4	6.05	6.05	6.05	6.22	6.22
5	5.82	5.82	5.82	5.98	5.98
6	3.93	3.93	4.55	4.03	4.54
7	2.62	2.62	3.20	2.68	3.41
8	3.47	3.47	4.21	3.55	4.24
9	3.63	3.63	4.00	3.72	4.14
10	3.93	3.93	4.19	4.03	4.42
11	8.19	10.03	8.19	10.63	8.42
12	2.49	2.49	2.91	2.54	3.14
Total	54.44	56.28	58.11	58.06	59.83

1. Base Scenario

Since it is intuitive that the time needed to execute a task at a particular stage of iteration depends on the amount of work to be done at that stage of iteration, the time elapsed to complete succeeding tasks at each succeeding stage of iteration is shorter and shorter. Following the first execution at time 0, the work effort increases as predicted by the HSS analysis in section 5.1 (see Figure 3). This is because after the first execution, information of other subsystems was available and a major review of the slowest task (Thermodynamic & cooling) was undertaken since it required information from 7 subsystems. This increase continues until after week 3. The remaining work of the task starts to decline as the task attempts to assimilate the new development information feedback from other tasks. The remaining work diminishes as the tasks are performed expeditiously with more design information. However, even after the completion date, there is still remaining work for tasks which are coupled to task 11. Many adjustments are still required after week 25 as the defined completion is up to the version 1.0 prototype only. If there are no external disruptions or internal design changes, the compressor design process will be completed as scheduled. However, the progress of the development process in this scenario remains as an ideal case since disturbances and changes are inevitable in real life. The total resource usage of the process decreases after its commencement. But the resource usage of task 11 (Thermodynamic & Cooling) decreases only after week 3. This is because the amount of work increased when it receives updated information from other tasks after the initial stage. As the remaining work begins to decline after that, the resource usage of tasks decreases as well (see Figure 4). The resource usage of each task under the base scenario is shown in Table 5. The unit of the normalised resources is defined as *task-resource*, where 1 task-resource represents the amount of initial resources assigned to one task as discussed in section 5. The total normalised resource required for the entire process over 25 weeks is 54.4 task-resources, i.e. 54.4 times the initial resources of one task.

2. Scenario 1

In this scenario, the longest task, task 11, was assigned extra initial resources before the start of the project, with the goal of completing the entire design project in 20 weeks, i.e. 5 weeks faster than the base scenario. As Figure 3 indicates that the rate of design progression improved. The time required to complete each iteration was shortened as extra initial resources were assigned to task 11. For instance, the first execution of tasks ends 0.5 weeks faster than the base scenario. By week 12, the progress was already one stage of iteration ahead of the base scenario. In this scenario, the time required by task 11 to execute a complete design cycle is 458 man-hours (see Table 4), i.e. 102 man-hours faster than the base scenario (560 man-hours). Although this represents an 18.3% improvement in time, 22.5% extra initial resources are required by task 11 before the project starts in order to shorten the time required to perform a complete iteration. The total resource usage by the design process under this scenario is indicated in Figure 4. The total normalised resource usage of each iteration is slightly higher than the base scenario although the time required by each iteration is reduced. The total resource usage of the entire accelerated process is 56.3 task-resource (see Table 5), i.e. 3.50% more than the base scenario of 54.4 task-resources. However, the extra resources are assigned to task 11 only. Therefore the resource usage of other tasks remains the same as in the base scenario. The resource usage of task 11 increase from 8.19 task-resources to 10.0 task-resources. The resource usage of each task shown in Table 5 is the total resources (including additional resources, if any) required by each task to complete the amount of work for all iterations.

3. Scenario 2

In this scenario, additional resources were deployed to tasks that are coupled to the longest task, task 11, after the project has started, with the goal of completing the entire design project in 20 weeks time, i.e. 5 weeks faster than the base scenario. The progress of the compressor design under this scenario is indicated in Figure 3. Faster progress with reduced amount of rework during the early stages of iteration was encountered. For instance, the remaining work after the first execution of tasks is 50% less than the base scenario. The rework incurred by tasks that are coupled to task 11 was reduced as additional resources were assigned to them. By week 9, the design progression is already one stage of iteration ahead of the base scenario. In this scenario, the time required by task 11 to execute a complete iteration remains at 120 man-hours. However, significantly more resources are required by its coupled tasks to reduce the rework incurred by them. This can be seen in Figure 4, where the total normalised resource requirement for the first iteration is almost 4 units (33%) more than in the base scenario. This is because many design decisions are still uncertain during the first execution of tasks, therefore the coupled tasks require relatively more effort to resolve the ambiguous issues. However, the total resource usage at each subsequent iteration is close to or less than the base scenario's as the amount of rework declines with the number of iterations, i.e. when more and more updated design information becomes available. The total resource usage of under this scenario is 58.11 task-resources (see Table 5), i.e. 6.80% more than the base scenario of 54.4 task-resources and 3.20% greater than scenario 1.

4. Scenario 3

In this scenario, extra initial resources were deployed to the longest task, task 11, before the design process starts with the goal of completing the entire design project 5 weeks faster and 3 more stages of iteration than the base scenario, i.e. the task would be reviewed 3 times more than scheduled to further refine its outcome. The progress of the compressor design under this scenario is indicated in Figure 3. As Figure 3 indicates, the rate of design progression increased without affecting the trend of progression. The time required to complete each iteration was shortened as extra initial resources were assigned to task 11 so that it could complete an iteration faster than as in the base scenario and scenario 1. For instance, the compressor design process started the 4th stage of iteration at week 12, rather than at week 15 in the base scenario. By week 15, the design progression is already two stages of iteration ahead of the base scenario. In this scenario, the time required by task 11 to execute a complete iteration is 443 man-hours (see Table 4), i.e. 20.8% faster than the base scenario of 560 man-hours and 3.3% faster than scenario 1 (458 man-hours). However, 26.3% extra initial resources are required by task 11. The total resource usage by the design process under this scenario is indicated in Figure 4.The total resource usage of each iteration is slightly more than the base scenario's although the time required by each iteration is reduced by 20.8% as discussed earlier. This is very similar to scenario 1 except that the number of stages of iteration is increased because task 11 required shorter time to complete an iteration. The total resource usage of the entire process is 58.06 task-resource (see Table 5), i.e. 6.73% more than the base scenario (54.4 task-resource) and 3.13% greater than scenario 1 (56.3 task-resource).

5. Scenario 4

This scenario is similar to scenario 2 with the additional goal of better quality of design decisions. Three more stages of iteration were performed before the compressor design process is finished by week 20. The additional resources were assigned to tasks which are coupled to task 11 during the course of the design process in order to reduce the rework created by them on task 11. By so doing, the time required to complete each iteration was shortened to allow more stages of iteration to be performed. The progress of the compressor design under this scenario is indicated in Figure 3. As Figure 3 indicates, very similar to scenario 2, the rate of design progression increased with reduced amount of rework during the early stages of iteration. For instance, the remaining work after the first execution of tasks is 95% of the initial amount of work, rather than 200% as in the base scenario and 98% as in scenario 2. By week 13, the progress of the design is already two stages of iteration ahead of the base scenario. In other words, by that time the design of the compressor is already reviewed 2 times more than in the case of the base scenario. With the additional stages of iteration, the level of confidence of design information and the quality of design decisions can be expected to be better than the base scenario. In this scenario however, significantly more resources are required by its coupled tasks in order to reduce the rework incurred by them. This can be seen in Figure 4, where the total normalised resource requirement for every stage of iteration is always greater than the base scenario, especially during the first iteration due to the reason discussed in scenario 2. The total resource usage of the entire process under this scenario is 59.83 task-resources (see Table 5), i.e. 9.98% more than the base scenario (54.4 task-resource) and 2.96% greater than scenario 2.

5.3 Summary of the scenario analyses

This section summarises the main findings of the four scenarios (see Table 6).

Table 6. The four scenarios compared

Scenario	Scheduled completion (week)	Scheduled completion (stage of iteration)	% Increase of total resource usage of task 11 over the base scenario	% Increase of total resource usage of the entire design process over the base scenario
Base	25	12	0	0
1	20	12	22.5	3.5
2	20	12	0	6.8
3	20	15	26.3	6.7
4	20	15	0	10.0

- *Resource allocation is a significant determinant of process performance*. The increase of resource requirements is in response to the corresponding work effort needed to expedite the longest task (extra initial resource schemes in scenarios 1 & 3) and the interdependent tasks (additional resource schemes in scenarios 2 & 4). For example, scenario 1 required 3.5% extra resources to expedite task 11, while scenario 2 required additional 6.8% resources to expedite tasks that are coupled to task 11. For scenarios 3 & 4, more resources were required compared to scenarios 1 & 2. However, the amount of extra resources used to complete additional stages of iteration (scenario 3 & 4) is relatively small compared to the amount of extra resources used to improve the completion time (scenario 1 & 2). For instance, compared to scenario 1 which required 3.5% extra resources, scenario 3 only required 6.7% extra resources for 3 more stages of iteration. This is because the process is converging as shown by the stability analysis in section 5.1, i.e. the remaining work is reduced with every succeeding stage of iteration. Thus, the additional stages of iteration (stages 13 to 15) require fewer resources to complete. This result reveals that expediting a design process, rather than allowing better quality of design decisions, requires more resources. This is intuitively true since speeding up a process requires more effort to prevent and resolve potential errors due to rush and careless work. However, with more time, one is able to improve his/her jobs with the least additional resources, though the extra time permitted could be considered as an added cost.

- *The resource allocation scheme determines the total resource consumption by the entire process*. The additional resource scheme (scenarios 2 & 4) requires more resources than the extra initial resource scheme (scenarios 1 & 3) for similar process improvement. For example, scenario 2 required 6.8% additional resources to expedite the process by 5 weeks than the base scenario, while scenario 1 required only 3.5% extra resources for the same process improvement. This is because for the extra initial resource scheme, only the longest task (task 11) required extra initial resources, while in the additional resource scheme more than one task required additional resources. This result suggests that the availability of resources in the beginning of a process, rather than the ability to catch-up on back-logged work, is a key determinant to resource saving. For instance, as can be seen from Table 6, to achieve the same process improvement (i.e. an earlier completion by 5 weeks), the increase of additional resources over the base scenario required by scenario 1 was 3.3% less than scenario 2. This is intuitively true because given sufficient resources in the initial stage, designers or engineers are able to solve the major tasks and finalise uncertain design decisions earlier. This strategy avoids unnecessary rework and thus saves resources. However, the additional resource scheme (scenarios 2 & 4) allows affordable piecemeal additional resources after the project has started. Therefore it is a good strategy for design projects with limited initial resources at the beginning of the project. Another interesting point indicated by the data in Table 6 is that for the additional resource scheme (scenarios 2 & 4), the longest task (task 11) does not require extra resources because additional resources are assigned to its coupled tasks that created its rework. This result suggests that the additional resource scheme is an appropriate alternative to improve the process performance if extra resources are not available for the longest task; for example, the absence of highly qualified staff.

6 Conclusion

The performance of a design process may be improved by expediting the completion time of the entire process and producing better quality design decisions through more iterations. Extra resources are needed in both cases. Using the NHSS model, the amount of extra resources can be determined even before the design process begins. Two resource allocation schemes are proposed to improve the performance of a design process:

1. Allocate extra initial resources to the longest task *before* the start of the design process to shorten the time required to complete its work.
2. Allocate additional resources to inter-dependent tasks *during* the course of a design process to reduce and compensate the rework incurred by these tasks.

A natural gas compressor process was analysed under different scenarios to observe the effect of various ways of allocating resources on the different objectives. It appears that more resources are needed to hasten a design process than allowing better quality of design decisions. The availability of resources at the beginning of a process economises most on resources. However, the additional resource scheme allows piecemeal additional resources to be assigned after the project has begun. This is helpful in cases where initial resources are unavailable at the beginning of the project. Furthermore, the additional resource scheme is ideal for cases where it is difficult to allocate extra resources to the longest task due to the nature of the task. Although the discussed observations are specific to the natural gas compressor design process, the methodology demonstrated in this paper provides insight into the planning of additional resources required to improve a design process before it begins.

References

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

[2] Osborne, S. M., "Product Development Cycle Time Characterization through Modeling of Process Iteration", Master's Thesis, MIT, MA, 1993.

[3] Steward, D.V., "System Analysis and Management: Structure, Strategy and Design", Petrocelli Books, New York, 1981.

[4] Smith, R.P. and Eppinger, S.D., "Identifying Controlling Features of Engineering Design Iteration", Management Science, 43(3), 1997, 276-293.

[5] McDaniel, C.D., "A Linear Systems Framework for Analyzing the Automotive Appearance Design Process", Master's Thesis (Mgmt./EE), MIT, MA, 1996.

[6] Yassine, A., Joglekar, N., Braha, D., Eppinger, S. and Whitney, D., "Information Hiding in Product Development: The Design Churn Effect", Working Paper 4333-02, MIT Sloan School of Mgmt., MIT, MA, 2002.

[7] Browning, T.R. "Use of Dependency Structure Matrices for Product Development Cycle Time Reduction", Proceedings of the Fifth ISPE International Conference on Concurrent Engineering: Research and Applications, Tokyo, Japan, 1998, 89-96.

[8] Mihm, J., Loch, C. and Huchzermeier, A., "Modelling the Problem Solving Dynamics in Complex Engineering Projects", INSEAD Working Paper, 2002. (http://www.insead.fr/~loch/fullcv.htm)

[9] Lee, S.G., Ong, K.L. and Khoo, L.P., "Control and Monitoring of Concurrent Design Tasks in a Dynamic Environment", Concurrent Engineering: Research and Applications, 12(1), 2004, 59-66.

Corresponding author:
Ong, Kian Leong
Universiti Teknologi Malaysia
Faculty of Mechanical Engineering
81310 UTM Skudai, Johor,
Malaysia.
Phone: +60 7 5533333
Fax: +60 7 5566159
E-mail: pj280434@ntu.edu.sg