Real-Time Co-ordinated Scheduling using a Genetic Algorithm

G. Coates[1], A. H. B. Duffy[2], R. I. Whitfield[2], W. Hills[1], P. Sen[3]

[1]Engineering Design Centre,
University of Newcastle,
Newcastle upon Tyne, NE1 7RU
graham.coates@ncl.ac.uk

[2]CAD Centre,
University of Strathclyde,
Glasgow, G1 1XJ

[3]Department of Marine Technology,
University of Newcastle,
Newcastle upon Tyne, NE1 7RU

Abstract

Real-time co-ordination is an emerging approach to operational engineering management aimed at being more comprehensive and widely applicable than existing approaches. Schedule management is a key characteristic of operational co-ordination related to managing the planning and dynamic assignment of tasks to resources, and the enactment of the resulting schedules, throughout a changeable process. This paper presents the application of an agent-oriented system, called the Design Co-ordination System, to an industrial case study in order to demonstrate the appropriate use of a genetic algorithm for the purpose of real-time scheduling. The application demonstrates that real-time co-ordinated scheduling can provide significant reductions in time to complete the computational design process.

1 Introduction

The operational engineering management of the design process of large made-to-order products can be complex, expensive and time-consuming due to the involvement of many resources and tasks, and large quantities of data, information and knowledge. The complexity is compounded by the fact that resources exhibit varying proficiency with regard to the undertaking of a variety of multiple inter-related tasks. Furthermore, due to unforeseen circumstances, resources may not perform as intended and/or scheduled tasks may not progress as expected, the outcome of which influences the performance of the design process. In order to account for such deviations in performance and/or progress, real-time scheduling must be considered. Furthermore, re-scheduling should only be performed if the benefits of doing so outweigh the status quo.

Design co-ordination is a relatively new approach to engineering management with its emphasis on timeliness and appropriateness [1]. More specifically, design co-ordination has been described as involving the effective utilisation of resources in order to carry out tasks for the right reasons, at the right time, to meet the right requirements and give the right results [2]. Further, based on this description, it has been reported that design co-ordination is the concept of the appropriate activities being performed, in a certain order, by a set of capable agents, in a fitting location, at a suitable time, in order to complete a set of tasks [3].

At an operational level of management [4], design co-ordination in real-time has been identified as being comprised of five key characteristics: coherence, communication/interaction, task management, schedule management and resource management [5]. This paper focuses on schedule management since design co-ordinated scheduling in real-time is of significant importance in that it directly relates to the time the design process can be completed in.

2 Real-Time Design Co-ordination System

Real-time operational design co-ordination enables multiple inter-related tasks to be undertaken and completed by allocating and utilising multiple resources, of varying proficiency, in an optimised fashion in accordance with multiple schedules in a coherent, appropriate and timely manner, within the dynamic and unpredictable design process. Furthermore, real-time operational design co-ordination facilitates the improved performance of the design process to be achieved and, in addition, sustained. The term *real-time* is used since in-situ operational design co-ordination is continuously in operation. As such, when an event occurs that causes the performance of the design process to be degraded, appropriate adjustments can be made to resume improved performance. This involves resource allocation and utilisation being adjusted and tasks being re-arranged and re-distributed appropriately. Consequently, the improved performance of the design process can be maintained. An important feature of real-time operational design co-ordination is that it also ensures that adjustments only occur if appropriate and, if so, periods of resource adjustment and task re-arrangement are utilised effectively.

The Design Co-ordination System (DCS) is an agent-oriented architecture that incorporates an approach to operational design co-ordination [5]. As such, the DCS is aimed at the real-time operational design co-ordination of a computational design process. With regard to the approach, operational tasks are initially defined based on the goals to be accomplished. Planning enables knowledge of the outstanding tasks and the resources available to be considered for scheduling. As a result of scheduling, a schedule is derived, which is the basis for the direction and undertaking of tasks. Prior to a task being undertaken, dependency relationships must be satisfied. In addition, any necessary information must be managed such that it is made available to allow the task to be completed. Monitoring facilitates the detection of deviations between the actual and expected performance of a resource. Similarly, monitoring enables deviations in the progress of schedules to be detected. If the deviation in resource performance or schedule progress significantly degrades the performance of the design process, forecasts are made for expected resource performance and/or task durations are revised. In addition, and only if appropriate, planning and scheduling are repeated in order to produce a more suitable schedule. In periods of transition between successive schedules, tasks continue to be completed and resources utilised in an optimised manner. Monitoring is conducted throughout the duration of the design process such that at any time, and if appropriate, the operational course of action can be adapted with respect to the prevailing circumstances.

Within the DCS, a collection of agents act as members of a multi-functional team operating in a co-ordinated fashion in order to satisfy the objective of ensuring that the specified inter-related design tasks are completed in a structured manner with respect to time, and the allocation and utilisation of the available resources within the computer network environment. This involves agents taking the opportunity to complete tasks concurrently when and where appropriate. However, the emphasis is placed on operational design co-ordination by ensuring that agent actions are conducted appropriately with respect to the time and order that they are performed.

Seven different types of agent are employed within the DCS, namely Co-ordination Manager, Information Manager, Task Manager, Resource Manager, Scheduler, Resource Monitor and Activity Director. Each of the agent types fulfils a particular role and is capable of performing various activities. The behaviour of all agents is complimentary in that they assist each other in order to satisfy the overall design objective of completing the computational design process in a co-ordinated and improved manner. Thus, consistent with Lesser, the collection of agents can be described as a co-operative or benevolent agent society [6].

In any application of the DCS, the number of agents of type Co-ordination Manager, Resource Manager and Scheduler is fixed at one. The number of agents of type Information Manager, Task Manager, Resource Monitor and Activity Director is dependent on the number of analysis tools to be used in the computational design process and/or the number of available resources in the computer network environment. An analysis tool is an individual software module that is executed, which uses input to produce corresponding output. Each execution of an analysis tool uses different input information, and is defined as a task within the context of the DCS. A resource is a machine in the computer network environment on which analysis tools can be executed. The number of Information Managers is equivalent to the number of different analysis tools to be used. The number of Task Managers is equal to the product of the number of analysis tools and the number of resources since a Task Manager exists for each tool on each resource. Each resource being utilised in the computer network environment is allocated a Resource Monitor and an Activity Director.

The agent configuration within the DCS is shown in Figure 1. In addition, the aspects of operational design co-ordination of each type of agent and communication links are shown.

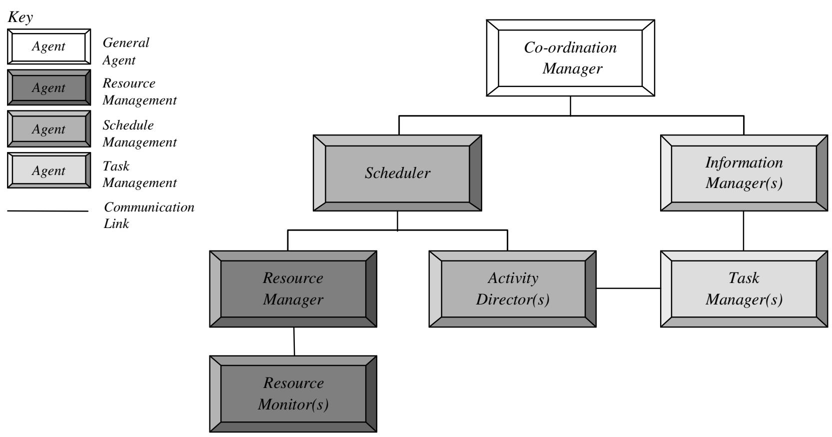

Figure 1: Configuration of DCS agents

At the outset of the operation of the DCS, the Co-ordination Manager facilitates communication links between other agents that need to interact in order to achieve a common goal. Once all appropriate agents are introduced, the Scheduler invokes a genetic algorithm (GA) [7,8] to derive a schedule from which the original schedule models are based. These schedule models enable the optimised utilisation to be made of the available resources with respect to the outstanding tasks. The general purpose GA based tool was developed to handle large combinatorial multiple criteria problems in various engineering fields. Further, several novel concepts of the GA were introduced, namely a Pareto population, adaptive niche sizing and neural network preferencing [8].

Activity Directors orchestrate Task Managers in accordance with their respective original schedule models such that their designated tasks can be completed. Task Managers operate such that prior to executing their tasks, the appropriate task input information is requested from their related Information Manager. Task Managers complete their tasks and inform their related Activity Director and Information Manager, which stores the resulting task output information. Activity Directors inform the Scheduler as tasks are completed and when they have completed their original/revised schedule models. Throughout the computational design process, Resource Monitors observe and analyse the monitored efficiency of their associated resource and inform the Resource Manager of any significant change, i.e. forecasts of resource efficiency. The Resource Manager is responsible for ensuring that knowledge of resources is maintained at all times and informing the Scheduler if a new schedule may be required. On instruction from the Resource Manager, and if appropriate, the Scheduler again invokes the GA in order to derive the revised schedule models. The process of determining whether or not re-scheduling is appropriate involves consideration of future expected performance of resources, outstanding tasks and the performance characteristics of the GA.

3 Industrial Case Study

The case study provided from engineering industry involves the use of ten analysis tools employed in the computational turbine blade design process. The analysis tools enable the calculation of stress and vibration characteristics of turbine blades. Each of the analysis tools is executed multiple times. As such, a total of one hundred and thirty one analysis tool executions, i.e. tasks, are involved in the computational turbine blade design process.

Table 1 indicates the dependencies between each of the analysis tools. That is, a non-diagonal element of unity within the matrix signifies that the execution of the analysis tool represented in the particular column must precede that of the analysis tool in the corresponding row. An element equal to zero indicates that no dependency relationship exists between the respective tasks. In addition, values of the datum duration, in seconds, for an execution of the analysis tool associated with the corresponding row and column are shown in the diagonal elements of the matrix.

Table 1: Analysis Tool Dependency Matrix

	A	B	C	D	E	F	G	H	I	J
A	97	0	0	0	0	0	0	0	0	0
B	1	6	0	0	0	0	0	0	0	0
C	0	1	6	0	0	0	0	0	0	0
D	0	0	1	14	0	0	0	0	0	0
E	0	1	0	1	1	0	0	0	0	0
F	0	1	0	0	0	2	0	0	0	0
G	0	0	0	0	0	1	1	0	0	0
H	0	0	0	0	0	1	1	1	0	0
I	0	0	0	1	1	0	0	0	1	0
J	0	1	0	1	0	0	0	0	0	11

In this paper, the case study is presented with an emphasis on determining whether or not re-scheduling in real-time is appropriate at the point when considered during the computational turbine blade design process. That is, if creating and enacting a revised schedule would result in the process being completed in less time than adhering to the original schedule.

With regard to the DCS, the need to consider re-scheduling is relayed to the Scheduler by the Resource Manager as a result of a Resource Monitor indicating that the performance of its associated resource has deviated below a defined threshold. Such a deviation will cause the sequence of tasks to be completed on the resource to be delayed, which may impact the completion of dependent tasks scheduled to be undertaken utilising other resources. In order to establish whether re-scheduling is required, the Scheduler assesses whether it is more economical time-wise to continue with the current schedule or, alternatively, re-schedule a proportion of the outstanding tasks and complete the revised schedule. As such, the decision whether or not to re-schedule is based on the Scheduler determining estimated times to:

- complete the current schedule, T_{CCS},
- derive a revised schedule, T_{DRS}, and,
- complete a revised schedule, T_{CRS}.

That is, re-scheduling is performed if $T_{CCS} > T_{DRS} + T_{CRS}$, otherwise it is not. If re-scheduling were performed, then during this period the outstanding tasks not re-scheduled would be able to be completed in accordance with the interim schedule models.

Prior to determining any of the three estimated times, the Scheduler requests that each Activity Director suspend administering their associated original schedule models. At the point of suspending the computational design process, twenty seven tasks had been completed. Thus, determining the time estimates requires the consideration of the one hundred and four outstanding tasks within the original schedule models of each Activity Director.

3.1 Estimated Time to Complete the Current Schedule

In order to determine the estimated time to complete the current schedule, the Scheduler supplies each Activity Director with the relevant up-to-date resource forecasted efficiency, R_{FE}. This efficiency is expressed as a coefficient between 0 and 1. The Activity Directors then apply their respective forecasted efficiency to the cumulative datum duration, ΣT_{DD}, of the outstanding tasks within its original schedule models to determine an estimated time to complete the model. Table 2 shows the cumulative datum duration of outstanding tasks in each original schedule model, and the existing forecasted efficiency of the corresponding resource, which are used to determine an estimate of the time to complete the original schedule models, ΣT_{ED}.

Table 2: Estimated Times to Complete Current Schedule Models

Resource R_I	ΣT_{DD} (seconds)	R_{FE}	$\Sigma T_{ED} = \Sigma T_{DD}/R_{FE}$ (seconds)
1	27	0.991	27.2
2	35	0.992	35.3
3	23	0.990	23.2
4	33	0.418	78.9

Each Activity Director provides the Scheduler with an estimated time to complete their associated original schedule model. The Scheduler then determines that the original schedule model with the greatest estimated completion time, indicated by the shaded cells, corresponds with resource $R_I = 4$. That is, the resource that experienced the significant reduction in forecasted efficiency to 0.418. Thus, if the original schedule models continue to be adhered to under the prevailing forecasted efficiency, it is estimated that they would be completed in approximately 79 seconds, i.e. $T_{CCS} = 79$ seconds. This estimate is considered conservative since the tasks undertaken utilising resource $R_I = 4$ may delay dependent tasks to be completed utilising other resources.

3.2 Estimated Time to Derive a Revised Schedule

In order to ensure that the appropriate tasks can be undertaken and completed during the period of re-scheduling, the Scheduler estimates the execution time of the GA based on the number of tasks and resources to be considered. At the outset of the operation of the DCS, the Scheduler was provided with knowledge of the relationships between the parameters that influence the execution time of the GA. Based on an empirical study, Figure 2 presents the relationship between the number of tasks to be scheduled, n_{TS}, for a number of resources to be utilised, n_R, and the execution time of the GA, T_{GA}. Furthermore, the information presented in Figure 2 was derived under conditions representative of the actual use of the GA during the operation of the DCS, that is:

- the GA was executed on the machine that would be used for scheduling in the turbine blade design process,
- the forecasted efficiencies of the resources were all set to unity,
- the durations of the tasks and dependencies between them were set in accordance with the case study, and,
- tasks were removed from consideration for re-scheduling in a manner representative of how they would be completed in the case study.

The three curves shown in Figure 2 are modelled using the regression equations shown such that the Scheduler can estimate the execution time of the GA based on the number of tasks to be re-scheduled. The estimated execution time limits of the GA are also shown, which are based on the maximum number of tasks that can be scheduled as dictated by the case study.

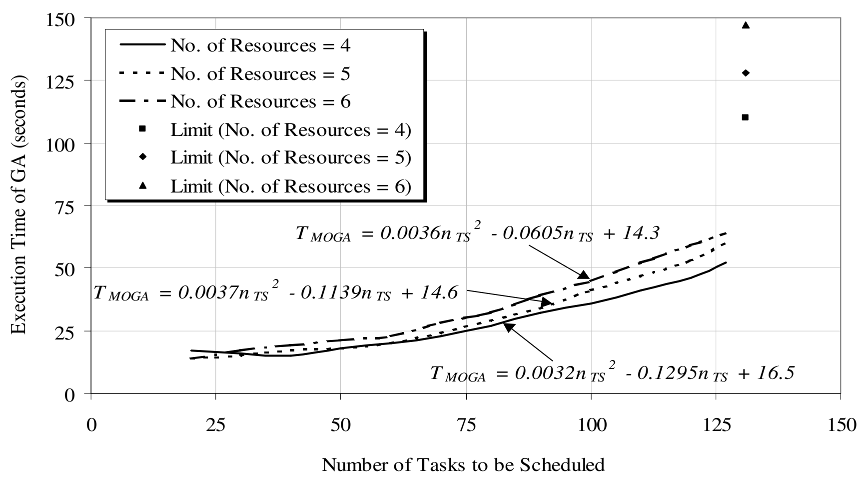

Figure 2: Estimated Execution Time of GA

In Figure 2, the maximum number of tasks that can be re-scheduled is 131, which corresponds to the number of analysis tool executions. The number of tasks to be scheduled, shown on the curves in Figure 2, ranges from 20 to 127. Using the GA to re-schedule beyond these limits would be uneconomical. That is, it would be inefficient in terms of time to re-schedule less than 20 tasks. Further, the upper limit is set at 127 since the datum durations of the first four tasks to be completed are significantly greater than that of all others.

Determining the estimated time to derive a revised schedule simultaneously involves establishing the most appropriate number of outstanding tasks to re-schedule while the remainder are completed. In order to determine the optimum number of tasks to re-schedule, a three-step iterative procedure is applied. Step 1 involves using empirically derived characteristics of the GA to determine its estimated execution time, T_{DRS}, given the number of tasks to be re-scheduled, n_{TRS}. Based on the time estimate from Step 1, Step 2 entails using the original schedule model for each resource in order to determine the number of outstanding tasks that could be completed during re-scheduling, n_{TCRS}. Step 3 involves deducting the cumulative number of outstanding tasks able to be completed utilising all resources determined in Step 2 from the number of tasks considered for re-scheduling in Step 1. The results from the application of the procedure are shown in Table 3. The procedure converges on the number of tasks to re-schedule such that the time taken to re-schedule them is as near-coincident as possible with the completion time of the remaining outstanding tasks, T_{TCRS}. Thus, the idle time of each resource is minimised.

Table 3: Determination of time to re-schedule and concurrently complete tasks

Item	n_{TRS}	T_{DRS} (secs)	Resources								Σn_{TCRS}
			$R_I = 1$		$R_I = 2$		$R_I = 3$		$R_I = 4$		
			n_{TCRS}	T_{TCRS} (secs)	n_{TCRS}	T_{TCRS} (secs)	n_{TCRS}	T_{TCRS} (secs)	n_{TCRS}	T_{TCRS} (secs)	
1	104	37.6	8	18.16	16	19.15	13	13.13	14	35.89	51
2	53	18.6	8	18.16	15	18.15	13	13.13	6	16.75	42
3	62	20.8	8	18.16	16	19.15	13	13.13	7	19.14	44
4	60	20.3	8	18.16	16	19.15	13	13.13	7	19.14	44

In Table 3, it can be seen that convergence to the optimum solution with respect to concurrent re-scheduling/task completion is reached after four iterations. That is, the Scheduler should re-schedule sixty tasks, estimated to take approximately 20 seconds, i.e. T_{DRS} = 20.3 seconds, according to the regression equation associated with four resources. During the period of re-scheduling, forty four tasks are estimated as being able to be completed utilising the four resources. Based on their most recent forecasted efficiency, resources R_I = 1, 2, and 4 would be utilised for approximately 19 seconds, while resource R_I =3 for approximately 13 seconds. As such, not only has the most appropriate time to re-schedule an appropriate number of outstanding tasks been determined but also the actual tasks to be completed during this period have been identified, i.e. those for inclusion within the interim schedule models.

Furthermore, this concurrent re-scheduling/task completion results in a mean idle time of the resources of approximately 3 seconds. Since the idle time of resources is minimised, thus maintaining their optimised utilisation, then the arrival of the revised schedule is expected to be as close as possible to the completion of the interim schedule models. That is, the difference between the time for the Scheduler to re-schedule and Activity Directors to complete their respective interim schedule models is minimised.

3.3 Estimated Time to Complete a Revised Schedule

Unlike determining an estimate of the time taken to complete the current schedule, an estimated completion time for a revised schedule must be obtained without using a schedule. This is achieved by determining the critical path of the outstanding tasks to be re-scheduled while considering the number of resources available and their forecasted efficiencies.

The Scheduler applies a two-step iterative procedure to determine the estimated time to complete a revised schedule.

3.3.1 Step 1 – Arrange tasks within groups according to their dependencies

Based on the dependencies between the outstanding tasks to be re-scheduled, tasks are arranged within groups such that the groups must be completed sequentially, however, tasks within groups may be completed in parallel. To determine the groups that the sixty tasks to be re-scheduled could be divided into, an assessment of the tasks they are dependent on is made, i.e. whether it/they:

- was/were completed in accordance with an original schedule model,
- will be completed in accordance with an interim schedule model, and/or,
- will be re-scheduled for inclusion with a revised schedule model.

Based on the assessment, forty eight tasks would not be dependent on the completion of other tasks once re-scheduled since:

- they were never dependent on the completion of any other tasks,
- the tasks they are dependent on were completed in accordance with the original schedule models prior to the consideration of re-scheduling, and/or,
- the tasks they are dependent on will be completed in accordance with the interim schedule models during the period of re-scheduling.

Similarly, as a result of re-scheduling, only twelve tasks will be dependent on the completion of other tasks within the revised schedule models.

Consequently, the sixty tasks to be considered for re-scheduling can be divided into two groups, i.e. one group comprising of forty eight tasks and another group consisting of twelve tasks. Further, these two groups must be completed sequentially. That is, the group of forty eight tasks must be completed prior to any of the group of twelve tasks being undertaken. However, tasks within each group may be completed concurrently since they are independent of other tasks in the same group.

3.3.2 Step 2 – For each task group, order tasks and assign to resources
Within each task group, tasks are ordered in descending order with regard to their respective datum durations. For the first task group, the first task is assigned to the resource with the greatest forecasted efficiency. Subsequently, the next task is assigned to the resource that leads to the minimum estimated time to complete the assigned tasks. This process continues until all tasks within the first group are assigned to resources in this manner. Similarly, the second group of tasks are ordered and assigned to resources.

Given that the datum duration of each outstanding task to be re-scheduled is one second, Table 4 presents information regarding how the two groups of tasks identified in Step 1 could be distributed amongst the four resources such that their collective time to completion is minimised.

Table 4: Assignment of re-scheduled tasks

R_I	R_{FE}	Group 1: 48 tasks		Group 2: 12 tasks		Total Time (secs)
		Number of Tasks Assigned	ΣT_{ED} (secs)	Number of Tasks Assigned	ΣT_{ED} (secs)	
1	0.991	14	14.13	4	4.04	18.17
2	0.992	14	14.11	4	4.03	18.14
3	0.990	14	14.14	3	3.03	17.17
4	0.418	6	14.35	1	2.39	16.74

Based on Table 4, with respect to the available resources, the estimated time to complete the revised schedule, T_{CRS}, is approximately 18 seconds. This corresponds to the greatest cumulative time to complete the two groups of tasks, as indicated by the shaded row of Table 4.

3.4 Decision to Re-schedule

An estimated time to continue adhering to and complete the current schedule has been calculated as approximately 79 seconds. Performing re-scheduling, while simultaneously completing interim schedule models, and the time to complete the revised schedule is estimated as approximately 38 seconds, i.e. 20 and 18 seconds as shown in Tables 3 and 4 respectively. Thus, $T_{CCS} > T_{DRS} + T_{CRS}$ leading to the Scheduler taking the decision to re-schedule. Furthermore, the interim schedule models derived, as a result of determining the estimated time to derive a revised schedule, will be administered by the Activity Directors of the associated resources such that tasks can be completed during the period of re-scheduling. In actuality, due to the application of the procedure used to determine the tasks for inclusion within the interim schedule models, their completion occurred within several seconds prior to the arrival of the revised schedule. This near co-incident occurrence leads to the conclusion that the empirically derived characteristics of the GA and the three-step iterative procedure used are reliable.

4 Conclusion

Through the implementation of the approach to operational design co-ordination, the DCS provides a systematic means of simultaneously co-ordinating the various management activities such that resource utilisation can be optimised and design tasks are undertaken and completed in accordance with schedules in a coherent manner. Specifically, this paper has demonstrated that the application of the DCS to the computational turbine blade design process has facilitated real-time co-ordinated scheduling. That is, the schedule management mechanism within the DCS ensures that re-scheduling is only performed if appropriate. The key feature of adjusting in real-time when appropriate enabled benefits to be realised in terms of reducing the time to complete the computational turbine blade design process. However, it is acknowledged that the magnitude of any reduction achieved is dependent on the stage of completion of the process.

With regard to the case study, by deciding to re-schedule, the turbine blade design process was completed in approximately 38 seconds from the point in time when re-scheduling was considered whereas continuing to adhere to the original schedule models would have taken 79 seconds. That is, an approximate reduction of 50% in time to complete the computational turbine blade design process was achieved. As such, relatively significant reductions in the time to complete the design process have been attained as a result of co-ordinated scheduling in real-time.

In order to demonstrate the approach implemented within the DCS further in terms of scalability, future work could involve its application within an engineering organisation where similar significant savings in time could be achieved on a larger absolute scale, i.e. in the order of man weeks or man months. Furthermore, given the generic nature of the approach, it can be applied to any general dynamic re-scheduling problem.

References

1. Duffy AHB, (1995) Ensuring Competitive Advantage with Design Co-ordination. Procs International Conference on Design to Manufacture in Modern Industry; 68-89.
2. Duffy AHB, Andreasen MM, O'Donnell FJ, (1999) Design Co-ordination. Procs International Conference on Engineering Design, **1**; 113-118.
3. Coates G, Duffy AHB, Hills W, Whitfield RI, (2000) A Generic Coordination Approach Applied to a Manufacturing Environment. Journal of Materials Processing Technology, **107/1**; 404-411.
4. Cole GA, (1994) Strategic Management. DP Publications.
5. Coates G, (2001) An Approach to Operational Design Co-ordination. Ph.D. Thesis, University of Newcastle, United Kingdom.
6. Lesser VR, (1999) Cooperative Multiagent Systems: A Personal View of the State of the Art. IEEE Transactions on Knowledge and Data Engineering, **11**(1); 133-142.

7 Todd DS, (1997) Multiple Criteria Genetic Algorithms in Engineering Design and Operation. Ph.D. Thesis, University of Newcastle, United Kingdom.
8 Todd DS, Sen P, (1998) Tackling Complex Job Shop Problems Using Operation Based Scheduling. Procs of Adaptive Computing in Design and Manufacture; 45-58.