A Re-analysis of the Effects of Task Decomposition and Organizational Structure on Performance and Robustness

Helen Gaylard
Centre for Policy Modelling
Manchester Metropolitan University

1 Introduction

Computational models of organizations evaluate the impact of different organizational designs on organizational performance. Organizational design comprises a number of variables, each computational organization being specified in terms of a unique combination of values for each of these variables. Organizational performance is modelled by assigning each formal organization specified the same series of tasks. Evaluation of each organizational design is in terms of its comparitive accuracy in carrying out the tasks assigned. A further aspect of performance is robustness. This is a measure of the extent to which organizational performance can be maintained under sub-optimal conditions. The robustness of each design is measured by repeating the simulation of tasks having degraded each organization in the same way; for example, by absenting *n* personnel from each.

Here we will be concerned with just two aspects of organizational design, organizational structure and task decomposition scheme. Organizational structure relates to the levels of personnel in the organization and the reporting relations amongst these. The task decomposition scheme refers to the allocation of tasks amongst the lowest level agents, or task analysts, in the organization. An important concept relating to both of these variables is that of *redundancy*. There is redundancy in the organizational structure if agents at one level report to more than one agent at the level above. There is redundancy in the task decomposition scheme to the extent that each piece of information needed to perform a task is available to more than one analyst. While a high level of redundancy in the organizational design might be considered inefficient, a certain level is clearly desirable if performance is not to deteriorate markedly under sub-optimal conditions.

Carley & Prietula [1] (henceforth as C&P) describe the simulation and evaluation of a number of organizational designs specified in terms of these two variables, under both optimal and crisis conditions. Their organizational designs incorporate three organizational structures, the *Centralized Hierarchy*, the *Dual Command* hierarchy, and the *Team*. The first two of these are distinguished from the latter in that they are hierarchical structures, with a layer of middle managers, or *Assistant Executive Officers* (AEOs), mediating between the task analysts and the CEO. The Centralized Hierarchy, unlike the other structures, incorporates redundancy in reporting relations, since each of its task analysts may report to two middle managers. Organizational structure can thus itself be broken down into two variables, redundancy in reporting relations and the structural variable, hierarchical or flat. Nine unique organizational designs are generated by combining each of the three organizational structures with three task decomposition schemes, *Segregated*, *Blocked*, and *Distributed*. These are shown in figure 1.

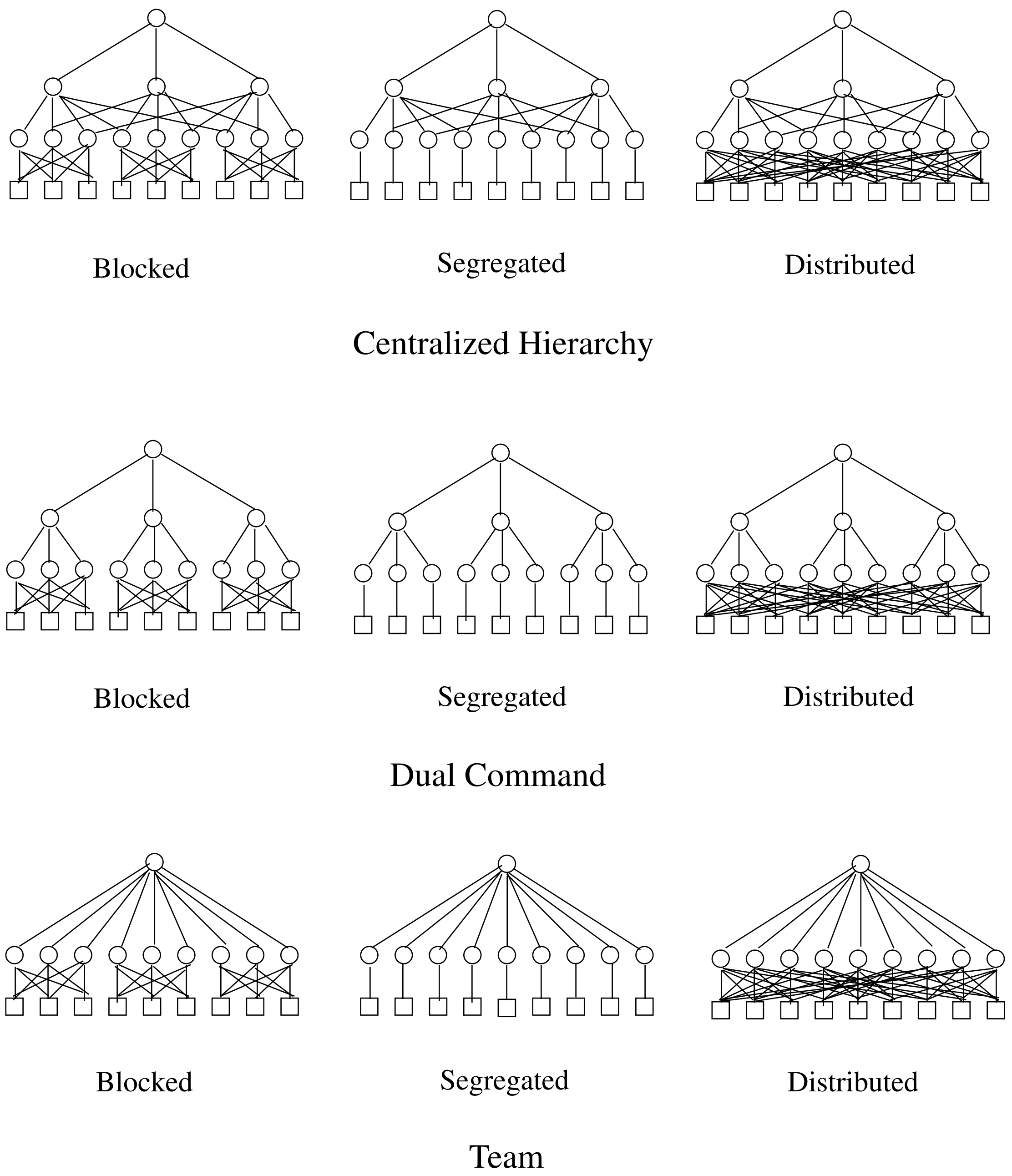


Figure 1: Categories of organisation as defined by C&P in [1]

C&P equate task decomposition with redundancy in task allocation, which they term *information redundancy*: “The distribution of subtasks ... defines the amount of redundancy at the lowest level and is referred to as the task decomposition scheme.” ([1], p.74) In the case of the segregated scheme there is no redundancy, as there are an equal number of subtasks and analysts and just one subtask is allocated to each analyst. In the blocked scheme, each analyst performs three subtasks, and thus there is redundancy as each subtask is in this case allocated to three analysts. The distributed scheme involves the highest level of information redundancy, with each analyst being allocated five subtasks. In addition to redundancy, a further distinction can be made between the blocked and distributed schemes. Under the blocked scheme, groups of three analysts are allocated three shared subtasks, whereas, under the distributed scheme, subtasks are distributed across analysts such that each is allocated a unique set of subtasks.

The major finding of the simulation and evaluation of these nine designs can be summarised as follows. Task decomposition scheme, rather than organizational structure, is the primary determinant of organizational performance: “For any type of organizational structure, in terms of performance under either optimal or crisis conditions, a segregated structure performs the worst and the distributed structure performs the best, with the blocked structure situated between the two performance extremes.” ([1], p.78)

C&P conclude that: “decomposing the task in a distributed fashion such that there is redundancy, but not total equivalence or overlap in task information, leads to the highest performance both under optimal and crisis conditions.” ([1], p.78) This implies that it is the distributed allocation of tasks, rather than the high level of information redundancy associated with it, which is considered the salient aspect of the task decomposition scheme. Another interpretation of the results is, however, possible. The relationship between task decomposition scheme and organizational performance can also be expressed in terms of information redundancy, since a higher number of subtasks per agent is associated with higher levels of performance.

A method for evaluating these alternative interpretations of the success of the distributed task decomposition scheme involves repeating the organizational experiments while controlling for the level of information redundancy. The segregated task decomposition scheme has, by definition, a fixed level of information redundancy. However, the distributed scheme can be specified so that, as in the blocked scheme, each analyst is assigned three subtasks. If the success of the distributed scheme is due to distributed allocation of subtasks, then this advantage should be maintained when information redundancy is controlled. However, if it is due to a higher level of information redundancy, then the advantage may be lost when a comparable level of information redundancy is maintained. The two interpretations are not necessarily mutually exclusive.

2 Formally Defining the Organizational Variables

C&P separate organizational design into the two variables of organizational structure and task decomposition scheme. The work described here extends the analysis to incorporate explicit formal definitions of the concepts involved in specifying a task decomposition scheme and organizational structure. Some of these are already implicit in the discussion above. The purpose of these definitions is to allow us to specify, for instance, distributed task allocation schemes with different levels of redundancy for comparison with blocked task decomposition schemes.

Organizational structure is broken down into hierarchical structure and redundancy in reporting relations. For our present purposes, the definition of hierarchical structure remains intuitive. However, the possibilities for reporting redundancy are delimited as follows. Any agent will normally report to at least *1* agent at the level above it in the organization, and to a maximum of n agents, where n is the number of agents at this higher level.

Task decomposition scheme is separated into the constituent elements of task allocation method and information redundancy. The two allocation methods which we will discuss here are blocked and distributed. There is no information redundancy where each subtask is allocated to only *1* analyst. Information redundancy increases as subtasks are allocated to a greater number of analysts, with the maximum possible being m, the number of analysts. There is no need to distinguish segregated as a method of task allocation, as this is equivalent to distributed allocation with no information redundancy.

Taking into consideration the interaction between organizational structure and task decomposition scheme, a further distinction can be made. Where the organizational structure is a team, and the task decomposition scheme is blocked, then it makes no difference which groups of analysts are allocated which blocks of subtasks, since each analyst in a team occupies an equivalent position in the organizational structure. However, where the organizational structure is a dual command hierarchy and the task decomposition scheme is blocked, then the group of analysts allocated a block of subtasks all share the same middle manager [1, 2]. An alternative allocation of each block of subtasks is, however, possible given such an organizational structure. Distributing each block of subtasks across a group of analysts reporting to *different* managers gives us a hybrid variant of task allocation, which we will term *Distributed Blocked*.

Formalising the distinction between blocked and distributed blocked task allocation requires some further definitions. We will restrict the use of the term *block* to refer to blocks of *subtasks* allocated to the same analysts, rather than to these analysts. *Analysts* which report to the same manager or managers will be referred to as a *group*. A team without middle managers may thus constitute a group, as may those analysts in a hierarchical organization who report to the same middle manager or managers. In the latter case, if there is such a high level of reporting redundancy that all analysts report to all middle managers, then all analysts will constitute a single group as in team structures.

Given these definitions, we can in turn define blocked and distributed blocked task allocation as follows. Where a block of subtasks is allocated to a group of analysts, the task allocation is blocked. Where a block of subtasks is allocated to analysts in different groups, the task allocation is distributed blocked. This distinction is only really salient for those organizational structures with multiple groups. Where there is a single group of analysts all sharing the same reporting relations, the two allocation schemes can be considered equivalent. Distributed blocked task allocation is also clearly distinguishable from distributed task allocation. Whereas distributed blocked task allocation distributes the same block of subtasks to several agents, distributed task allocation assigns a unique but overlapping set of subtasks to each agent. This is shown in figure 2.

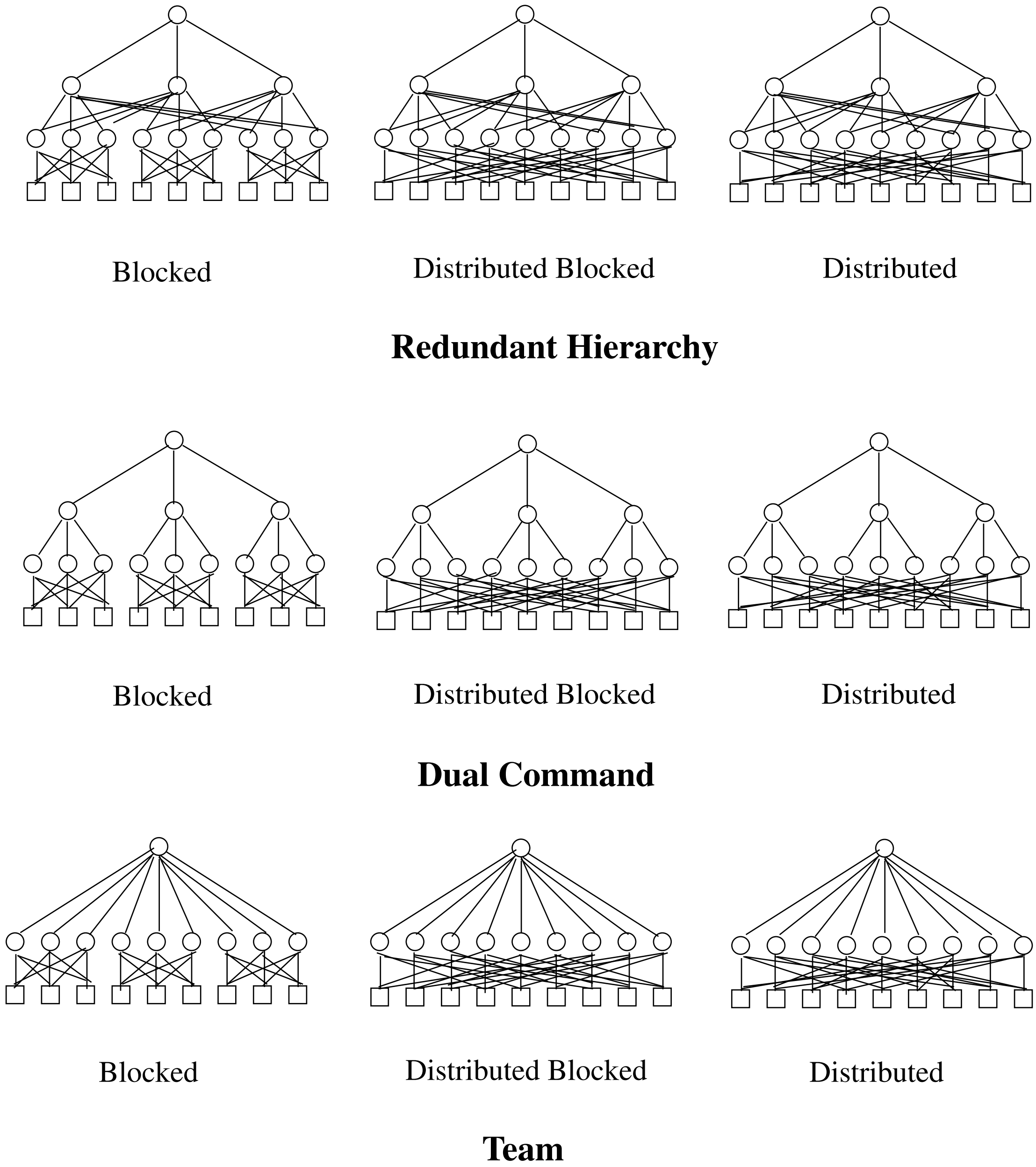


Figure 2: Alternative suggested organisational categories

3 The Organizational Designs

Given the preliminary definitions outlined above, we can specify a set of organizational designs suitable for distinguishing between the impact of task allocation and information redundancy. The specific issue addressed will be whether the advantage for distributed task allocation over blocked task allocation is maintained when the level of information redundancy is controlled. We introduce the hybrid task allocation scheme, distributed blocked, which shares blocks of subtasks with blocked task allocation but distribution of subtasks across groups of analysts with distributed task allocation. Comparing the performance of this with that of the other task allocation schemes may further elucidate upon the nature of any advantage for one task allocation scheme over another.

The number of organizational designs simulated in our model is determined by the number of task allocation schemes investigated multiplied by the number of organizational structures to which each is applied. We are concerned primarily with the three task allocation schemes, blocked, distributed, and distributed blocked. Information redundancy is controlled for comparability of the different means of task allocation to a level of three subtasks per agent. In order to ensure the consistency of our results with those of [1], we add to these the further task decomposition schemes employed by them, distributed with information redundancies of one subtask per analyst (i.e., segregated) and five subtasks per analyst. Note that it is not possible to test the blocked and distributed blocked allocation schemes with equivalent levels of information redundancy to these. This gives us a total of five task decomposition schemes.

The primary reason for combining each task decomposition scheme with several distinct organizational structures is to distinguish evaluations of them which are consistent over different organizational structures from those specific to a particular organizational design. Following [1], we specify three organizational structures.

1. The *Dual Command* structure, a hierarchy in which there is no reporting redundancy, is an identical organizational structure to that used in the original simulations.
2. A *Team* is also used, but in this case with a managing CEO rather than team voting. In practice, this makes no difference to the results for this structure, for reasons which will be explained below.
3. The third organizational structure employed is a hierarchy with a reporting redundancy of two managers per analyst, which we term the *Redundant Hierarchy*[1]. This differs from C&P's Centralized Hierarchy[2] only insofar as every analyst reports to two managers, as compared with only a subset of analysts (2 out of 3) reporting to more than one manager in the original model. The reasons underlying this choice of structure will be detailed below. Another *apparent* difference is that in the Centralized Hierarchy no two analysts in the same block share two managers. However, this observation conflates the notion of task allocation block with that of reporting group (see previous section for this distinction). In both the Centralized and Redundant Hiercharchies, there are analysts which share the same managers, but there are three for each pair of managers in the Redundant Hierarchy while there are only two for each pair of managers (and one for each single manager) in the Centralized Hierarchy.

Taken together, the five task decomposition schemes and three organizational structures outlined give us a total of fifteen organizational designs.

4 The Organizational Task and Subtasks

The basic organizational task is the same as that specified in the original model, to determine whether there are more ones than zeroes in a binary string (the length of which is equivalent to the number of analysts in each organizational design modelled). Each bit in the string is equivalent to a subtask. Analysts decide whether there are more ones than zeroes and report their decisions to te next level. Managers perform the same kinds of decisions but their data is in the form of analysts' reports.

Whereas in the original model there is learning, the model reported here is simpler in this respect. The binary string input is always randomly generated so that there is an equal chance

[1]Equivalent to the Matrix-2 organizational structure of [2].

[2]Equivalent to the Matrix-1 organizational structure of [2].

of each bit being a zero or one. In this context it is appropriate for agents to make decisions simply on the basis of the data to which they are currently exposed. In contrast, in the learning model, feedback on agents' prior decisions might lead them to favour certain choices over others.

The inclusion of learning facilitates exploration of such issues as the locus of institutional memory ([1], p.73), which are beyond the scope of this paper. It also widens the range of crises which can be explored, to include, for instance, replacement of experienced personnel with new personnel. Since the aim of this paper is to isolate the effects of task allocation scheme from those of information redundancy, a simpler model in which the effects of both of these can be isolated from the effects of learning is appropriate.

As agents have no memory relating to past choices in the model reported, it makes no difference whether individual agents in a team register their votes or whether they all report to a common manager: the majority opinion determines the outcome either way.

5 The Organizational Crisis

Performance under crisis conditions will practically be at least as important a consideration in evaluating alternative organizational designs as is performance under ideal operating conditions. C&P found that, while their distributed task decomposition scheme consistently led to the best performance under both optimal and crisis conditions, the blocked task decomposition scheme was associated with the smallest deterioration in performance under crisis conditions. The robustness of the blocked task decomposition scheme is a further result which may be attributable either to blocked task allocation or to a lower level of information redundancy.

Four types of crises are explored in the model reported here:

1. A number, i, of information links from subtasks to their analysts might be missing. (Information redundancy would be expected to afford some kind of protection against this kind of crisis, since the links lost would be a smaller proportion of the total.)
2. A number, c, of communication breakdowns might take place between agents and their superiors. (Here, reporting redundancy would be expected to be beneficial.)
3. A number, l, of the bits needed to perform subtasks might be lost (and thus unavailable to *all* of their analysts). The number, i, of missing information links resulting, will be determined by the level of information redundancy.
4. A number, a, of agents might be absent (so that their decisions are unavailable to *all* of their superiors). The number, c, of (effective) communication breakdowns, will be determined by the reporting redundancy. (Here, either information or reporting redundancy might be expected to mitigate the effects.)

5 Implementation of the Model

The model is implemented in SDML (for *Strictly Declarative Modelling Language*), a logic-based language with object-oriented features which is optimised for multi-agent modelling [3].

The model is designed to allow for easy, interactive specification of a wide range of organizational designs at the start of a simulation. Organizations corresponding to these are then automatically generated by the model and their performance in (randomly generated variants of) the organizational task is simulated over the number of time periods specified by the user. The fifteen organization simulation described is just one example of the many possible simulations the design of the organizational model allows.

Organizational designs may be specified in one of three ways. Those used in previous simulations may have been saved, in which case they may be filed into the current simulation. The user may choose to custom design the organization. In this case, starting from the top level of the organization, the user is asked to specify, for each department, how many agents and sub-departments are directly contained within it. This is the only way of specifying certain irregular organizational designs that cannot be specified summarily, this being the third method.

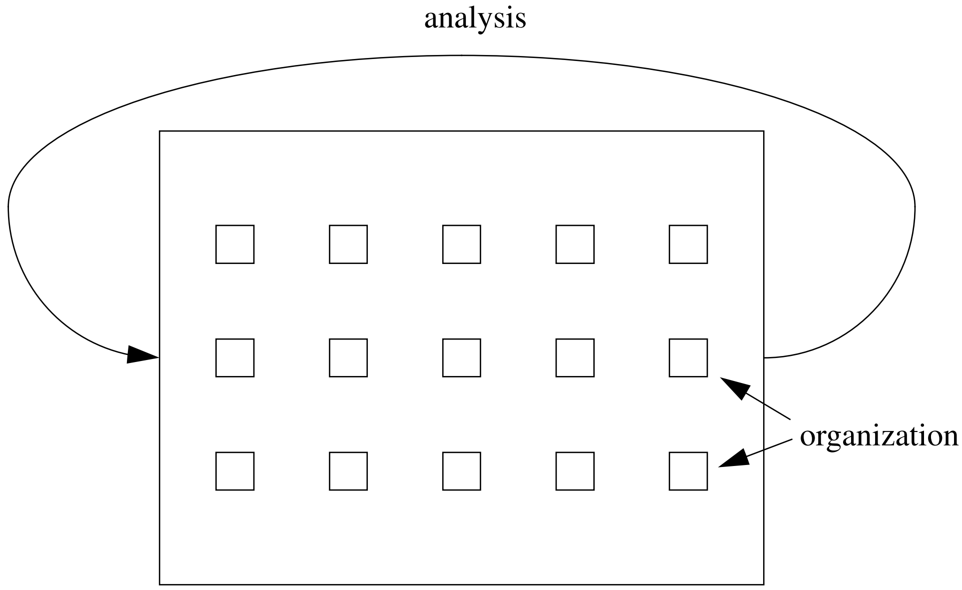
All organizations in the simulation reported were generated from summary data. The variables specified for each organization were: the overall number of agents and the breakdown of these into top-level manager, middle managers (if any), and task analysts; the number of managers to which each analyst reports (a measure of reporting redundancy); the task allocation scheme; and the number of subtasks allocated to each analyst (a measure of information redundancy). One of the reasons the Redundant Hierarchy was chosen as an organizational structure is that, unlike the Centralized Hierarchy, it has a regular structure that is easily specified using summary data. It is also easier to to give a meaningful analysis of the results of simulations which compare regular structures.

Global variables were (in addition to the number of organizations to be simulated) the number of subtasks in the organizational task (9 bits) and the length of the simulation (100 time periods, or *analyses*).

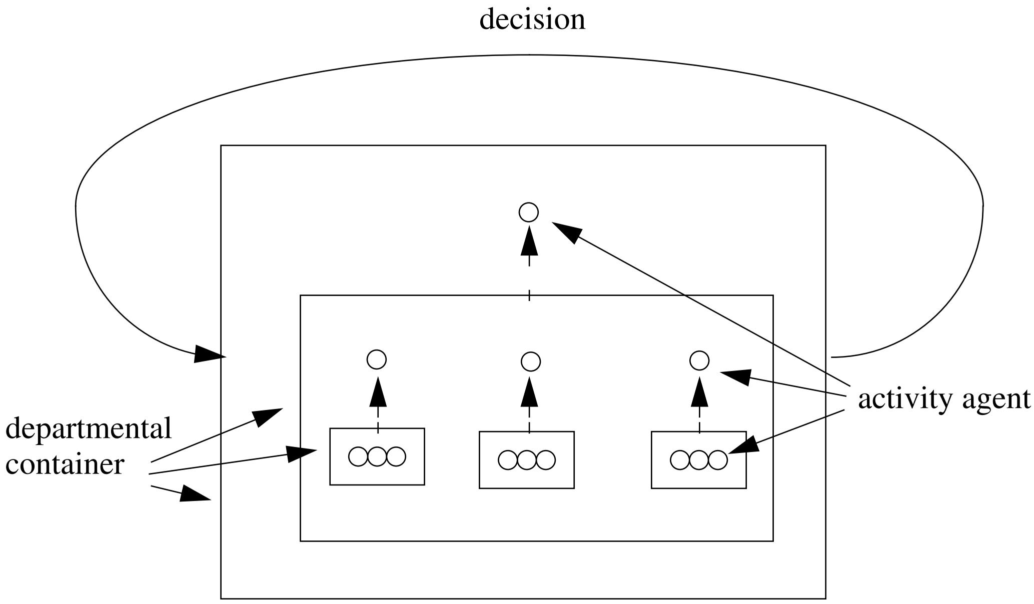
6 Structure of the Model

All of the rules used by the model are contained in the rulebases of three user-generated SDML types, which are subtypes of (i.e., inherit from) the in-built type *Agent* (or one of its subtypes). These types are the *Organizations Container*, the *Departmental Container*, and *Activity Agent*. The first two are named *containers* because they are composite agent types, which means that instances of these agents may contain other agents nested within them. The *Container Hierarchy* is another term for the structure of a model written in SDML.

The model contains a single, persistent instance of the type Organizations Container. Its rules interact with the user at the beginning of the simulation to dynamically generate the organizations to be simulated, which are then contained within it. It also generates a new version of the shared organizational task, arranged into indexed subtasks, at the beginning of each analysis. At the end of the simulation, it evaluates the performance of each organization within it.



Organizations Container



An Organization

Figure 3: Representation in SDML

All other agents in the model are generated dynamically at the start of the simulation. Organizations are instances of the type Departmental Container, as are the departments contained within them. All managers and analysts are instances of the type Activity Agent and are contained within instances of the type Departmental Container. Each organization, department, and agent is given a unique name by which it may be referred. The user chooses the organization name, and those of its constituents are generated automatically from this, e.g., DualCommandBlockeddept-3dept-1agent2. Each instance inherits the rulebase of its type, so this represents a very flexible and efficient way of generating organizations.

The organizational structure is represented by the container hierarchy, reflecting the organization's hierarchical structure, together with permanent clauses asserted to the databases of departments, specifying to which manager(s) agents within the department report. Departmental Containers can thus be seen to transparently represent the concept of reporting groups. The appropriate container hierarchy is generated automatically from the user's inputs concerning types and numbers of agents and their reporting relations. This is shown in figure 3.

The task decomposition scheme is represented by permanent clauses asserted to the databases of task analysts specifying which of the subtasks they will analyse. The subtasks are allocated automatically in accordance with the task allocation scheme and level of information redundancy specified by the user.

During each analysis, the main rules to fire are those of the Activity Agent rulebase, which contains the rules used by task analysts and managers. Within each organization, agents operate at a lower time level than the time period analysis. They operate within the time period *decision*. During the first decision within any analysis, task analysts analyse their subtasks and report the results of the analysis to their managers. Any agent who receives a report in one decision analyses the data it received in the next decision, and in turn reports the result to their own manager. When the top-level manager has arrived at a result, this triggers a rule which specifies that there will be no more decisions within this analysis. When the organizations have finished firing their rules, the Organizations Container goes on to generate the task for the next analysis.

7 Implementation of Crises

A series of user-directed queries allow the user to specify any crises at the start of the simulation. The possible crises offered are those described in a previous section. For each crisis chosen, the user specifies the number of crises of the type, e.g., the number of task analysts absent. The same crises are applied to each organization, with the exception that some crises are not applicable to all organizations, e.g., middle managers missing is inapplicable to teams.

The crises specified are recorded by means of permanent clauses asserted to the Organizations Container's database. The exact agent or task affected by any crisis is chosen randomly by the Organizations Container at the start of each analysis and recorded by a transient clause on the appropriate database. The rules of Activity Agents check for these clauses during the simulation and their behaviour is affected in the appropriate way. For example, an agent will not report to a particular manager if there is a clause asserted to the database stating that there has been a communication breakdown between the two of them.

In the original model, crises were initiated after a certain amount of learning had taken place in the model. As there is no learning in the model reported here, crises can be simulated from the start of the simulation. In order to analyse the effects of different types of crises on different organizational designs, as well as the overall robustness of the alternative organizational designs, we simulated the organizations under optimal conditions for 100 analyses and then repeated this simulation for each type of crisis applied.

The crises reported upon here are the following: 1 subtask missing; 1 task analyst missing; 1 middle manager missing (where applicable); 1 subtask unavailable to 1of its analysts; 1analyst available to 1 of its managers; and 1 middle manager (where applicable) unavailable to the CEO - which is actually equivalent to 1 middle manager missing.

While these represent only a modest and preliminary subset of the range of possible crises to be explored within the framework of the model, they do give some interesting and consistent results.

8 Analysis of Results

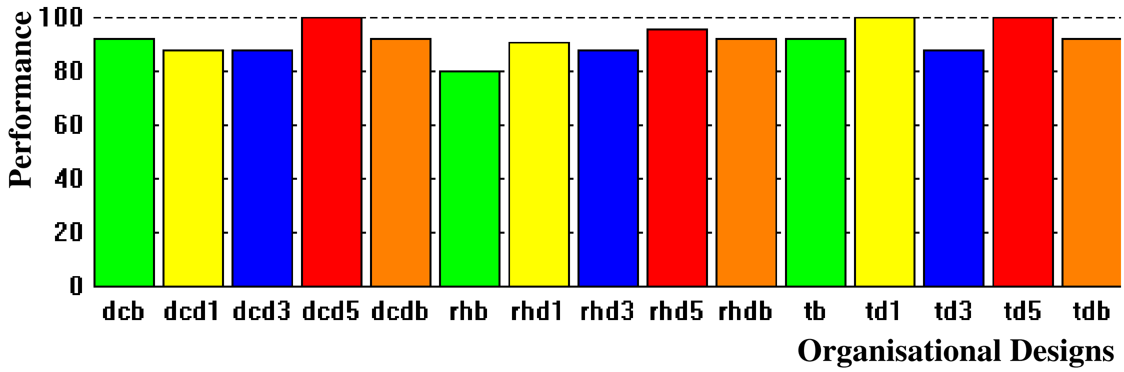


Figure 4: Performance of organisational designs under optimal conditions

The bar chart above (figure 4) summarises the performance of the fifteen organizational designs under optimal conditions. The abbreviations used are “dc” for dual command, “rh” for redundant hierarchy, “t” for team, referring to the organizational structure, and “b” for blocked (with 3 subtasks per agent), “d” for distributed (with 1, 3, or 5 subtasks per agent, as specified), and “db” for distributed blocked (with 3 subtasks per agent), referring to the task decomposition scheme.

The results, like those of the original simulation, confirm that it is the task decomposition scheme rather than the organizational structure which is the major determinant of organizational performance. Performances across two of the organizational structures, the team and the dual command hierarchy, are identical for all of the task decomposition schemes except Distributed(1) . The performance of the redundant hierarchy is also identical with that of the others for two of the task decomposition schemes, Distributed(3) and Distributed Blocked.

The most striking observation which can be made with respect to the impact of organizational structure is that redundancy in reporting relations is not necessarily beneficial. More than twice as many errors are made by the redundant hierarchy with a Blocked task decomposition scheme as with the other organizational structures with the same task decomposition scheme. The reasons for this are clear. While three identical reports will always be made to the manager in the Dual Command hierarchy, the manager in the Redundant Hierarchy will sometimes be presented with three reports that there are more 1s than 0s and three reports contradicting this claim. In such circumstances, managers will simply have to make a random decision.

There is no reason to suppose that increasing redundancy in reporting relations will generally lead to a deterioration in performance. Increasing the reporting redundancy to 3 managers per task analyst, which is the highest level of redundancy possible, would improve the information available to managers and thus the organization’s performance. One possibility which must be considered is that the organizational task used in the simulation is too abstract to be adequate to explore the effects of reporting redundancy.

With respect to the task decomposition scheme, the superiority of Distributed(5) is confirmed. However, when the level of information redundancy is fixed at a level of 3 subtasks per analyst, the Distributed Blocked task allocation scheme consistently gives a higher level of performance than Distributed(3), as does the Blocked scheme for two of the organizational structures. The hypothesis that distributed task allocation results in improved organizational performance is disconfirmed. Rather, the advantage of the task decomposition scheme Distributed(5) seems to derive from its high level of information redundancy. As with reporting redundancy, however, the relationship between information redundancy and organizational performance is not a straightforward one.

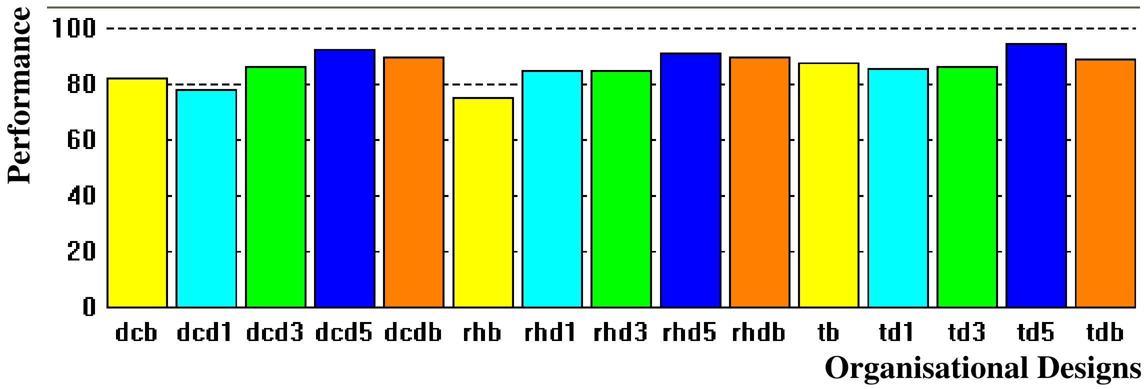


Figure 5: Performance of organisational designs under crisis conditions

The chart above (figure 5) illustrates the average performance of the organizational designs over the crisis conditions simulated. Four of these crises applied to all organizational types, while two, the absence of a middle manager and the communication breakdown between manager and CEO applied to the hierarchical structures only and not to teams. Comparing teams with the other organizational structures would, therefore, involve a further comparison, of the average performance of teams with the average performance of the hierarchical structures in the kinds of crises which affect teams. This would give a fairer indication of relative performance in the common kinds of crises, but it is also fair to consider teams to be robust insofar as they are not prone to some of the potentially most damaging kinds of crises which affect the other organizational structures.

The analysis of performance under the crisis conditions simulated can be summarised as follows. Distributed(5) retains its overall advantage as a task decomposition scheme. When the level of information redundancy is controlled, Distributed(3) task allocation still gives better performance than Blocked task allocation in the hierarchical organizational structures, whereas the situation is reversed for teams. The advantage for Distributed(3) over Blocked is largely due to the susceptibility of the Blocked scheme to two closely related crisis types, the absence of a middle manager and a communication breakdown between a manger and the CEO. However, the newly introduced hybrid task allocation scheme, Distributed Blocked, gives in better performance than either the Distributed(3) or Blocked schemes across all organizational structures.

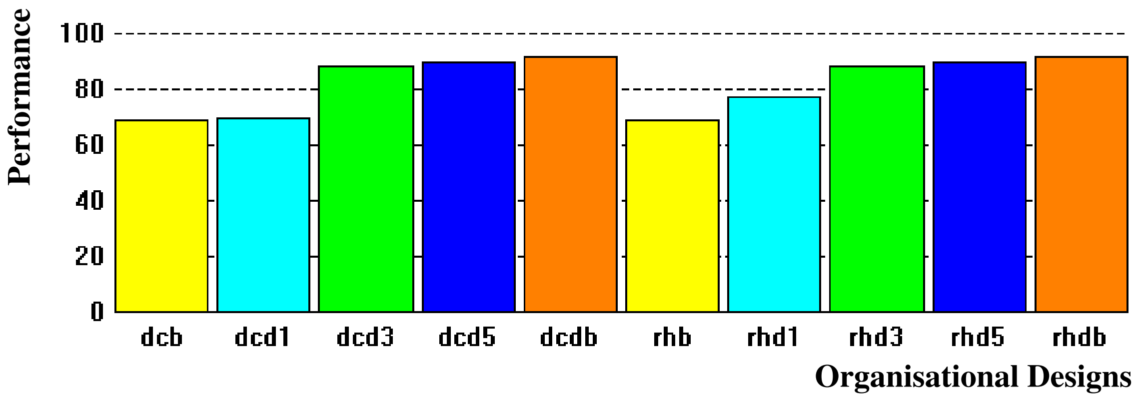


Figure 6: Performance of organisational designs under manager-related crises

The chart above (figure 6) illustrates the performance of the different task decomposition schemes under manager-related crises. The results are identical for each of the hierarchical organizational structures for all of the task decomposition schemes except Distributed(1). In this case the Redundant Hierarchy outperforms the Dual Command structure, suggesting that reporting redundancy may to some extent compensate for a lack of information redundancy. While there is a serious deterioration in performance under the Blocked scheme, performance is very little affected when task allocation is Distributed with the same level of information redundany, and *the high performance levels associated with optimal conditions are actually maintained under the Distributed Blocked task allocation scheme*. This means that, in these kinds of crisis conditions, the Distributed Blocked scheme even outperforms Distributed(5).

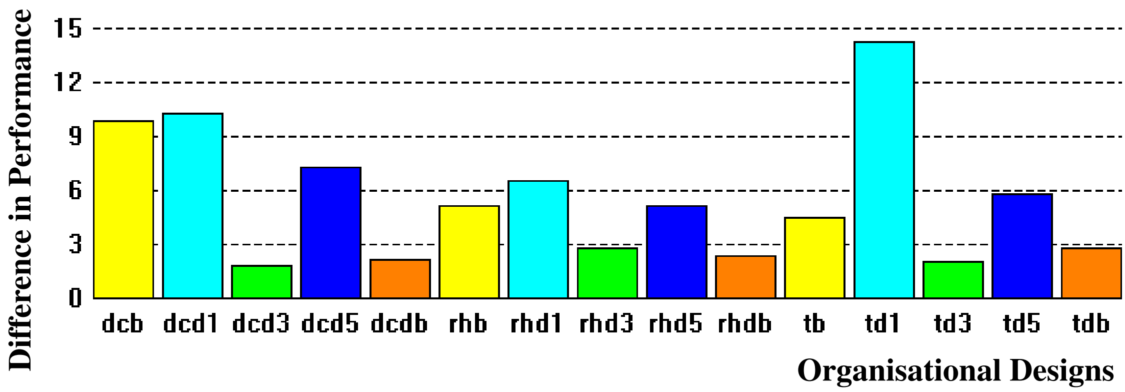


Figure 7: Deterioation of performance of organisational designs from optimal to crisis conditions

The chart above (figure 7) shows to what extent performance deteriorates between optimal and crisis conditions. Taking this as a measure of robustness, the Distributed(3) and Distributed Blocked task allocation schemes fare best. C&P found the smallest deterioration in performance in crisis conditions for the Blocked task allocation scheme. This result is confirmed if the Blocked scheme is compared with Distributed(5), but disconfirmed when the level of redundancy is kept constant over the task allocation schemes.

9 Discussion

The results of the simulation can be summarized as follows. By controlling the level of information redundancy across different task allocation schemes, we found that the results reported in [1] were reversed. Blocked task allocation outperformed Distributed task allocation, for all organizational structures, under both optimal and (average) crisis conditons, when the level of information redundancy was set to three subtasks per analyst. Also, performance deteriorated less for the Distributed scheme than for the Blocked scheme under crisis conditions, which is the opposite of the result reported for the original simulation. The original results were duplicated when the Blocked scheme was compared with the Distributed scheme with a higher level of information redundancy of five subtasks per analyst. Also the greater impact of task decomposition scheme than organizational structure on the performance of the different organizational designs was confirmed.

The most dramatic result was that the newly introduced hybrid task allocation scheme, Distributed Blocked, outperformed both the Blocked and Distributed schemes with the same level of information redundancy. It was still outperformed by the Distributed(5) scheme for most of the condistions investigated, but its performance deteriorated less under crisis conditions. Its relative robustness suggests that its performance might be expected to overtake that of Distributed(5) if a wider range of crises were investigated, as was indeed found to be the case in relation to the manager-related crises.

A simple explanation of the superiority of the Distributed Blocked task allocation scheme is possible. Under this scheme the information relating to subtasks is distributed evenly over the reporting groups of analysts so that each of these groups receives exactly the same kind of information, passing the same collection of decisions to the level above. If there are middle managers, each of these will in turn receive the same decisions and so they will be in agreement in the decisions they pass on to the CEO. This contrasts with other task allocation schemes which, in interaction with particular organizational structures, result in sets of decisions which contradict each other being transmitted up the organizational hierarchy. The combination of the Blocked scheme with a Redundant Hierarchy, discussed in an earlier section, is an example of this.

The Distributed Blocked task allocation scheme can be described as *redundancy-preserving*. Redundancy is a measure of information duplication; hence its association with the property of robustness. In the organization with a Distributed Blocked task allocation scheme, each of the reporting groups (and, where applicable, middle managers) is redundant with each of the others. Information redundancy is preserved as information is transmitted up the hierarchy. Each subtask is transmitted three times to an analyst. If there is no reporting redundancy, then each reporting group passes three decisions just once to its manager, but, as these decisions will be the same for each of the groups, each decision is, in effect, passed three times to a manager. Where there is reporting redundancy this serves to further duplicate the already duplicated (in this case, triplicated) information, so that, in a Redunant Hierarchy, each decision is passed to a manager six times.

10 Conclusion

The purpose of the organizational simulations reported here was to distinguish between two possible interpretations of the results reported in [1]. This involved, in the first instance, analysing two components of organizational design, organizational structure and task decomposition scheme, into their own constituent elements. This analysis provided the vocabulary with which to express two competing hypotheses, that the superior performance of the distributed task decomposition scheme was due to its method of task allocation, or that it was due to its information redundancy. These hypotheses were tested by specifying and simulating organizational designs which shared the same level of information redundancy but used different methods of task allocation, distributed or blocked. Since blocked task allocation conflated the notion of a block of subtasks and that of a group of agents reporting to the same manager, a third hybrid method of task allocation, distributed blocked, was added to these. This retained the blocking of subtasks, while distributing these blocks across different groups of agents.

The finding of the simulations was that the distributed method of task allocation did not outperform the blocked method when the level of information redundancy was controlled. However, the newly introduced method of task allocation, distributed blocked, outperformed both distributed and blocked, across three organizational structures and under optimal and crisis conditions. This is due to its redundancy-preserving properties, as explained in the previous section.

C&P reported that task decomposition scheme, rather than organizational structure, had the greatest impact on organizational performance. This finding is confirmed and strengthened. The additional task decomposition scheme investigated here, distributed blocked, is sufficiently robust that organizational structure had no impact upon performance under the crisis conditions investigated when this task decompsition scheme was used. This is despite the fact that certain of the crises investigated were applicable to some but not all of the organizational structures modelled.

Organizational designs employing the distributed blocked task decomposition scheme preserve such high levels of redundancy that they appear to have traded efficiency for robustness. This raises the question of whether the levels of information redundancy and reporting redundancy in the organizational designs modelled are not too high to be of any practical value except in relation to the most high-risk, safety-critical environments. Future simulations will need to investigate how real organizations without such permanently high levels of redundancy can adapt to deal with crises. The development of the organizational model in SDML is the first step towards this. In the same way that it allows us to specify a wide range of organizational designs from a small number of types, it will enable us to model changing organizational structures and relations using these same types in conjunction with SDML's time levels.

References

[1] Carley, K.M. & Prietula, M. (1994). ACTS Theory: Extending the model of bounded rationality. In Carley, K.M. & Prietula, M. (eds.), *Computational Organization Theory*, pp. 55-87. Hove: Erlbaum.

[2] Lin, Z. (1994). A Theoretical Evaluation of Measures of Organizational Design: Interrelationship and Performance Predictability. In Carley, K.M. & Prietula, M. (eds.), *Computational Organization Theory*, pp. 113-59. Hove: Erlbaum.

[3] Moss, S., Gaylard, H., Wallis, S. & Edmonds, B. (forthcoming). SDML: A Multi-Agent Language for Organizational Modelling. *Computational and Mathematical Organization Theory.*