A Distributed Coordination Model for Battle Staff Tasks

Luc Lamontagne
Defence Research Establishment Valcartier
2459 Blvd. Pie XI north
P.O. Box 8800
Courcelette, Québec, Canada
G0A 1R0

Abstract

This paper presents a framework for the distributed coordination of individual tasks of Battle Staff members. The application problem, identified following a preliminary analysis of checklists used in the Canadian NORAD Region Operations Control Center, is described. A preliminary model for the coordination of partially ordered and related tasks executed by a group of battle staff members is developed. This intuitive distributed coordination model leads to a series of activities alternating between selection of intended tasks by an individual and acceptance of this task by the group. Agreement is reached when coordinating agents accept that the established individual sequencing of the intended tasks represent a compromise between individual preferences without violating group constraints.

1. Introduction

This work is directed toward the understanding and design of decision aids for the coordination of battle management tasks executed in organizations such as NORAD command posts. At the present time, very few tasks have been automated and most are still being executed by the Battle Staff individuals. Battle staff members exchange information, conduct individual activities and cooperate on common tasks to accomplish their common mission. Decision aids designed for such organizations should not only provide individual services to their main users but also help them manage their interactions with each other. The principal aim of this study is to illustrate that coherence and improvement of group performance can be reached by adapting individual procedures/activities using distributed coordination techniques.

In this paper, we present an intuitive model for battle staff group coordination. The model to be investigated should ideally be incorporated in distributed decision aids to coordinate BS members activities in command posts where a functional hierarchy, the members roles, and the individual procedures are already established. Decision aids including such a coordination scheme may act as agendas recommending sequenced tasks (actions) to be undertaken by members. Such systems may easily be combined with a graphical representation of military checklists.

The military application problem motivating this study of group coordination is described and the distributed decision aids environment is formulated in section 2. In section 3, a description of the coordination model is presented in two parts: the first part illustrates deliberations leading to hypothetical task selection, and the second part is centered around the interactions between coordinators to reach an agreement. Section 4 is a discussion of further aspects related to group coordination. Due to the classified nature of NORAD operations, the content of this paper makes no direct reference to the ROCC Battle Staff procedures.

2. Problem description

2.1 The military application

While conducting activities that are part of a battle management process, each member of a Battle Staff is assigned a series of tasks to execute. These tasks come from the series of responsibilities describing the roles members play in the organization hierarchy of the Battle Staff. Tasks are divided in two types: context dependent tasks and persistent tasks.

This arbitrary division is deemed necessary to represent the various roles of Battle Staff members.

A checklist may be considered as a partially ordered procedure to be followed by an officer under a change of alert status. It is used as a memo to keep track of the implementation of battle management operation plans. A subset of the tasks assigned to an officer may be found as action items on individual checklists. We refer to these tasks as context dependent since their execution is subject to their relative position in the checklist and depends on the checklist current status implementation of other members of the organization.

The remaining tasks, which we call persistent, are not included within checklists. We make the distinction between context-dependent and persistent tasks mainly because some BS members do not make use of checklists. Among the persistent tasks, some are executed on a periodical basis. For example, the Weather Officer is responsible for assessing almost every hour the meteorological categories of various sites. Some other persistent tasks are executed upon request from other members of the battle staff. One interesting property of the persistent tasks is that their execution normally takes place more than once in the battle management process; the number of times they are selected varies from one situation to another.

Both context-dependent and persistent tasks might necessitate direct interaction between Battle Staff members. However it is important to note that their assignment to individuals is static. In the NORAD context, BS members have dissimilar expertise and a series of dedicated checklists. In general, a task can only be executed by a single officer. This is not always the case for every military organization; for example, in Rescue Coordination Centers (RCC), checklists are associated with individual cases rather than with individuals.

The overall list of tasks to be executed over a period of time by a battle staff member is the aggregation of the context-dependent (action items) and persistent tasks. In figure 1 relationships describing high-level interactions between agent tasks are illustrated in hierarchical manner to represent the organizational structure of the Battle Staff. For the purpose of our study, we assume that individual task flows are centered around the execution of the checklists action items, with possible inclusion of the persistent tasks.

Due to the nondeterministic nature of the battle management environment, the order of task execution is expected to be kept dynamic to cope with varying conditions or periods of increased tension. This assumption results from a preliminary analysis of the checklists. The order in which the action items are presented on paper does not necessarily represent a strict order of execution and hence indicates that a partial temporal order between tasks should be considered.

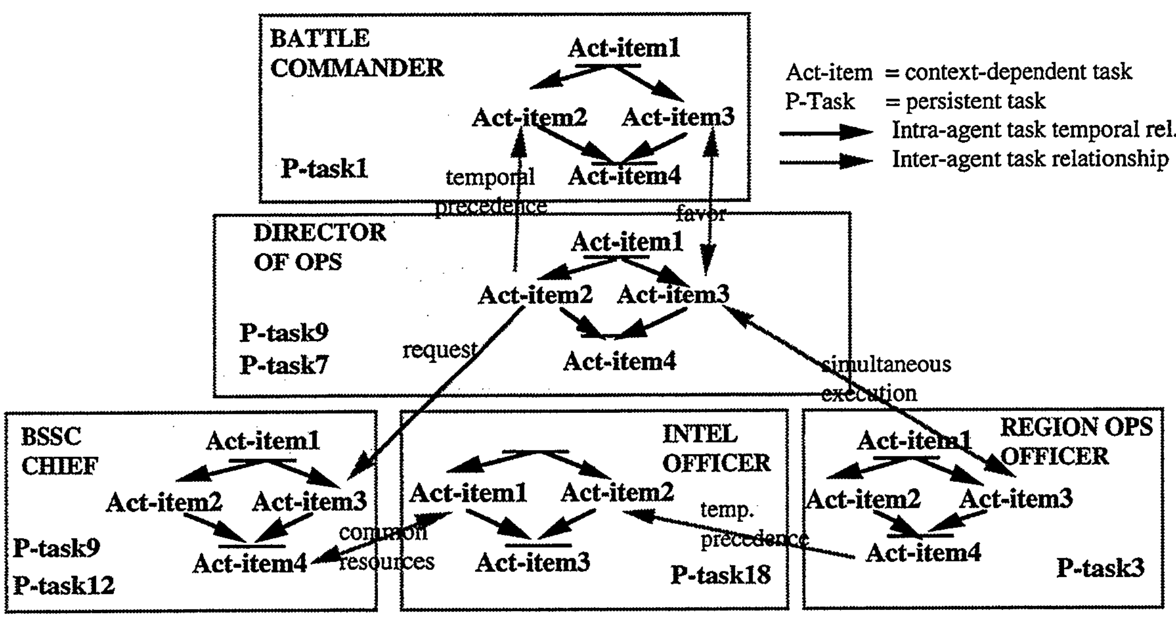

Figure 1 : Relationship Graph of Battle Staff Tasks

Moreover, comparison between various members checklists reveals many analogies, cross-reference or other relationships between tasks and even similar task patterns. It is expected that modification or shuffling of tasks during the course of action will take place to accomodate other members expectations, avoid creating conflicts between procedures, and improve group performance.

2.2 The distributed framework

Decision aids built for organizations where humans exchange information, conduct individual activities and cooperate on common tasks to accomplish their common mission pose some problems [4] [8]. Too often, automation efforts are limited to providing tools in response to individual data manipulation needs, thereby ignoring an integrative group approach. Battle staff members are still left with the responsibility to execute additional tasks, to interact with other BS members as well as with the decision aid system.

In this paper, BS members are mapped in a one-to-one fashion with agents responsible for coordinating their activities with other members activities (figure 2). Coordination of group tasks and of checklists action items is accomplished through exchanges among users and decision aids.

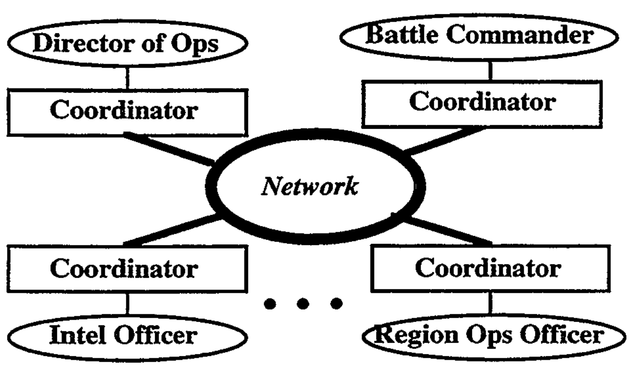

Figure 2: Individual Decision Aids for Group Coordination

A distributed environment is appropriate because of the distributed nature of the battle staff activities, the dissimilar expertise of the BS members, and the functional hierarchical structure of the Battle Staff organization [6]. Furthermore, as partial automation of the command post activities progresses in the future, the work load associated with task execution will be increasingly distributed between the decision aids and their users. Centralized management of hundreds of procedures including dozens of high-level tasks, which can be divided into many subtasks, would make global coordination rather difficult and/or degrade individual performance levels.

3. A Model for the Coordination of BS Member Tasks

In this section, we present a coordination model based on a decentralized scheme. We view coordination as a way to ensure coherence of group task executions in order to improve overall performance in spite of the agents limited knowledge of each other.

Coordination and execution activities are conducted by a group of agents. An agent is either task coordinator (also called coordinator) or task executor (executor). To each coordinator is assigned a task executor. A coordinator is responsible for determining the next tasks, which we call intentions, to be implemented by the associated task executor. A task becomes an intention when its future implementation is agreed upon by group coordinators. Upon completion of a task, executing agents receive from their associated coordinator the next intention to be implemented. In our model, an intention can be revoked if necessary; however a task being executed cannot be interrupted.

Coordination is conducted in an incremental fashion. Each coordinating agent determines a candidate task as its next intention. Coordinator selected tasks are submitted to other coordinators for acceptance. Whenever an agreement is reached, the process starts again for the determination of the next intentions.

The coordination model makes the following assumptions:

- A fixed number of agents is present in the model, hence no reduction or addition of coordinators or task executors is considered;
- Individual responsibilities are assigned and remain static during the coordination/execution process. No role merging is allowed;
- A relative ranking (hierarchical) of the agents is established;

- Procedures, checklists and the underlying tasks are predetermined;
- Each coordinator has some knowledge for identifying tasks relationships.

A task is characterized by the following parameters: name, type, associated agent, expected start time, expected duration, needed resources (type and amount).

The coordination process consists of two levels: The local intention selection and the distributed coordination protocol. The local intention selection is an individual process which determines the next steps. The distributed coordination protocol corresponds to the management of the interaction between coordinators in order to reach agreements.

3.1 Local Intention Selection

The local intention selection process helps answer the question: given the current status of the other agents executed tasks and intentions, what should be a coordinator's next intention? This process consists of: candidate tasks identification, relationships detection and candidate task selection.

Candidate task identification

At the candidate task identification level, the following information is available to a coordinator:

- its context-dependent tasks, formulated as plans with temporal dependencies, and its persistent tasks;
- its task currently being executed;
- its intentions already accepted by other coordinators;
- other agents tasks being executed and intentions.

The candidate tasks set is the combination of the persistent tasks, and the context-dependent tasks not violating any temporal precedence with current intentions and execution. A hypothetical start time, corresponding to the expected completion time of the last agreed upon intention, is assigned to each candidate task.

Relationship detection

The relationship detection consists of identifying the relationships between candidate tasks and other agents task executions and intentions.

Relationships between tasks [10], can either negatively or positively affect concurrent execution. Negative relationships may generate potential inconsistencies in group activities, while positive relationships may constitute a basis for improving the synergy between two individuals.

Table I: Types of task relationships used for modeling

TYPE	NAME	EXAMPLE
Negative	temporal precedence	recall(agt1) & receive-briefing-from(agt1)
	task request	Direct(ag1,determine RWstatus) & determine(RWstatus)
	common resources	consult(charts) & prepare(breifing)
	simultaneous execution	discuss-with (agt1,Wxstatus) & discuss-with (agt2,Wxstatus).
positive	favor exchange	call(agt1, status)

In our current effort to model BS task relationships, four different types of negative relationships are retained: temporal precedence, task request, common resources, and simultaneous execution. Inconsistencies arise when temporal precedence is ignored, task requests arise under hierarchical authority (inter-agent commitments), common resources are scheduling constraints to prevent overutilization, simultaneous execution necessitates interactions between officers in order to jointly assess/deliberate/agree.

The only positive relationship considered in this work is favor exchange [10] which is possible when more than one agent have the capability to execute a given task. Agreement between officers should then be sought in order to avoid redundancies. In general, it is expected that favors are made by inferiors to accomodate superiors (task delegation); however in this model, task favors are assigned following dynamic group deliberations.

The identification of relationships is based on the task type, the expected start time and duration of task, and on the nature and amount of resources needed for execution.

Candidate Task Selection

Candidate task selection is a computation done locally by each coordinator in order to determine the next intention to be proposed to other coordinators. This selection is based on the list of candidate tasks previously identified and their detected relationships. The intention choice process consists of a conflict avoidance step and a task ranking step.

Conflict avoidance consists of either candidate task elimination, or candidate task parameter modification depending on the relationships. Temporal precedence and simultaneous execution conflicts can be avoided by shifting candidate task start time. If avoidance is revealed impossible due to prior selections, a group intention revision procedure must be initiated as explained in section 3.2. Common resources constraints can be relaxed by either reducing the amount of resources or shifting start time. Favor tasks are kept as candidate tasks to be ranked; however in the case where they are being executed by another agent, they are eliminated since task executions cannot be revoked. Candidate tasks involved in a request relationship are kept for later ranking.

Candidate intention selection is represented as a decision problem based on a multicriteria ranking of the modified candidate tasks, resulting from the conflict avoidance step. In this current model, task ranking is done according to the following criteria:

- the nature of the tasks;
- the type of identified task relationships;
- a cost assigned to the task modifications;
- the agents involved in the relationships.

A local intention selection based on such criteria can be seen as a compromise between what is locally most preferable and what is the least constraining for group activities. One difficulty associated to this formulation is the determination of trade-offs among various hypothetical situations. For example, one would have to determine if it is more preferable to do a favor to a superior or anticipate a request relationship with inferiors. Similar considerations should be given to ensure that execution of the persistent tasks is eventually favored over context dependent task. Such trade-offs can only be determined in a domain specific study and require for further investigations.

3.2 Distributed Coordination Protocol

When a candidate intention is locally selected, it is submitted for other coordinators approval, who upon receipt can either:

- accept sender intention;
- accept sender intention and change their own intentions;
- accept and relax their own intentions to accomodate the sender, through a temporal shift or a resource reduction modification.
- refuse sender intention (bring sender to change its intention);

If a proposal is refused, the proposing agent initiates a new local selection process using alternative candidate intentions. Counterproposals by the receiving coordinator are not admissible in this model, since they can only be made when extensive knowledge of other agents is available.

An agreement is reached when a group of intentions is accepted by all agents. However, general acceptance of additionnal intentions or modifications may require a global revision of previoulsy accepted ones, since they may create new conflicts. Such discrepancies may result from the asynchronous nature of inter-agent communications. It is also possible that execution time delays postpone the detection of negative and positive relationships. In such cases, a group intention revision procedure must be declared.

A revision procedure forces the group to backtrack (backjump) over their already determined intentions and reinstantiate new ones where necessary. This backtracking (backjumping) can be seen as a search over the possible group intention space. A group revision can be declared by any coordinating agent whenever a deadlock arises. For example, the instantiated inversion of two tasks involved in a temporal precedence creates a deadlock.

To ensure that coordination can be reached at a given time while limiting the expected duration of deliberations during group revision, it is desirable to take advantage of the relative hierarchical ranking among coordinators in order to resolve conflicts [2]. Control over intention revision is sequentially delegated to coordinators following a descending path along the group hierarchy (in our case, the BS hierarchy). The revision procedure is initiated by the highest authority which performs a local selection intention. The selected candidate intention is then broadcasted to all inferiors for acceptance based on their own intentions. When the highest authority intention is agreed upon by all inferiors, this intention is kept constant and the control is passed on to the next ranked authority which repeats the same steps with its inferiors. The process terminates when the control reaches the bottom of the hierarchy ranking.

4. Discussion

A general outline of the distributed coordination model has been presented for Battle Staff group activities. Application of the proposed scheme to realistic examples of Battle Staff operations, and comparison with other proposed distributed coordination scheme [1] [2] [9] and local intention selection are being conducted.

Coordination of group procedures is often modeled as a two-phase process: coordinate all tasks first and then submit the result for task execution. Such a scheme cannot be adopted in dynamic environments involving uncertainty. Trade-offs between complete coordination and immediate execution remain to be studied. Such trade-offs, representing the horizon of intention coordination, are largely dependent on the status of the environment and the progression of group activities. Task refinement and definition will be taken into account in further coordination studies of BS member activities.

With such a distributed scheme, can specific task combinations be imposed on a group of coordinators? Integrity of the authority among coordinators directly depends on the selected combination of parameters such as the relative importance of coordinating agents, the preference orders over tasks and relationships, as well as the frequency of intention revision.

5. Conclusion

A group coordination model of Battle Staff activities, a description of the military context and of the associated decision aid environment have been presented in this paper. The coordination model is designed to favor the selection of intentions locally. The selection process is subject to a series of relationships among group intentions. We proposed five relationships to describe task dependence. Selected intentions are then submitted for acceptance to other coordinating agents. Coordination is reached when proposed intentions are agreed upon by all other coordinators. The distributed coordination protocol also provides a group intention revision procedure to resolve conflicts over sequences of individual intentions. Group intention revision is based on the relative ranking of the coordinators, hence taking advantage of the functional hierarchy structure of organizations like the NORAD ROCC Battle Staff.

As explained in the discussion, future work will look at the validation of the proposed coordination scheme on realistic examples of Battle Staff operations and the incorporation of tasks refinement/definition to the proposed model.

Acknowledgement

The author would especially like to thank Stephen Woods and Irene Abi-Zeid for providing helpful comments relative to this work.

References

1. S. Conry, K. Kuwabara, V. Lesser, "Multistage negotiation for distributed satisfaction," IEEE Trans. SMC, vol. 21, no. 6, pp. 1462-1477, Noc/Dec 1991.
2. E. H. Durfee, T. A. Montgomery, "Coordination as distributed search in a hierarchical behavior space," IEEE Trans. SMC; vol. 21, no. 6, pp. 1363-1377, Noc/Dec 1991.
3. A. Rao, M. Georgeff, "Intentions and rational commitment," Technical Note 8, AAII, February 1990.
4. I. Greif (ed.), "Computer supported cooperative work: a book of readings," San Mateo, CA, Morgan Kaufmann, 1988.
5. M. Klein, "Supporting conflict resolution in cooperative design systems," IEEE Trans. SMC; vol. 21, no. 6, pp. 1379-1389, Noc/Dec 1991.
6. T. Malone, "Modeling coordination in organizations and markets," in Readings in Distributed Artificial Intelligence, pp. 151-158, 1988.
7. S. Minton, M. Johnston, A. Philips, P. Laird, "Minimizing conflicts: a heuristic repair method for constraint satisfaction and

scheduling problems," Artificial Intelligence; vol. 58, pp. 161-205, 1992.

8. J. Rasmussen, B. Brehmer, J. Leplat, (eds.) "Distributed Decision Making: Cognitive Models for Cooperative Work," Wiley, Chichester, England, 1991.

9. K. Sycara, S. Roth, N. Sadeh, M. Fox, "Distributed constraint heuristic search," IEEE Trans. SMC; vol. 21, no. 6, pp. 1446-1460, Noc/Dec 1991.

10. F. von Martial, "Coordinating plans of autonomous agents," Springer-Verlag, Berlin Germany, 1992.