IFAC/IFORS/IIASA/TIMS

The International Federation of Automatic Control
The International Federation of Operational Research Societies
The International Institute for Applied Systems Analysis
The Institute of Management Sciences

SUPPORT SYSTEMS FOR DECISION AND NEGOTIATION PROCESSES

Preprints of the IFAC/IFORS/IIASA/TIMS Workshop
Warsaw, Poland
June 24-26, 1992

Editors:
Roman Kulikowski
Zbigniew Nahorski
Jan W.Owsiński
Andrzej Straszak

Systems Research Institute
Polish Academy of Sciences
Warsaw, Poland

VOLUME 2:

Names of first authors: **L-Z**

SYSTEMS RESEARCH INSTITUTE, POLISH ACADEMY OF SCIENCES

SUPPORT SYSTEMS FOR DECISION AND NEGOTIATION PROCESSES
Preprints, IFAC/IFORS/IIASA/TIMS Workshop, June 24-26, 1992, Warsaw, Poland

A NEGOTIATION-BASED MODEL OF CO-OPERATIVE DECISION-MAKING *

STAN SZPAKOWICZ [1], ZBIG KOPERCZAK [1], GREGORY E. KERSTEN [2]

[1] Department of Computer Science, University of Ottawa, Ottawa, Ontario, Canada
[2]School of Business, Carleton University, Ottawa, Ontario, Canada K1S 5B6

Abstract: Co-operative decision-making (CDM) can be viewed as information exchange in a network of agents. We propose to model agents by means of identical decision analysis and support tools, in the manner developed in the NEGOPLAN project. The goal of working out a commonly acceptable project organization is reached in a series of brief two-side negotiations. We hope to achieve partial automation of CDM, and experimental validation of the agent network connections. Decision-making is couched in terms of the exchange of elements of representations, and the gradual convergence upon a unified representation. We present the COON system—an extension of NEGOPLAN from negotiation support to simulation and support of CDM—and use it on a case study taken from an actual project management situation.

Keywords: Co-operative decision-making, Decision support, Sequential decisions, Project management, Restructurable modelling, Rule-based systems

1. An Overview of the Method

Co-operative decision-making (CDM) can be viewed as information exchange in a network of agents. Offers or positions (in the negotiation sense) could be exchanged in various network topologies. We propose to model agents by means of identical decision analysis and support tools. The communication channel is narrow—only negotiating positions—and the goal is to converge upon a commonly acceptable project organization. This is reached in a series of brief two-side negotiations, with little bargaining and much information passing. We hope to achieve partial automation of CDM, and experimental validation of the topology: testing what connections allow the agents to reach a global agreement most effectively.

Agents are modelled in the manner developed in the NEGOPLAN project (KER91, KER88). An agent has full knowledge of her own decision problem: its environment, its hierarchical decomposition, rules of change (if a change of perspective is required), and a catalogue of reactions to other agents' positions (KER90).

A decision problem is represented in NEGOPLAN as a specialized AND/OR tree that captures the hierarchical decomposition of the decision-maker's principal goal. Tree nodes represent subgoals. The lowest-level, non-decomposable subgoals—we call them *facts*—are assigned

* We thank the employees of the Government of Canada who participated in our research and consulted us on the case study; they wish to remain anonymous. This work was partially supported by grants from the Natural Sciences and Engineering Research Council of Canada.

logical values (true or false) in accordance with a fact's current status in the decision domain. Facts annotated with logical values—we call them *metafacts*—are communicated to the other agents as this agent's position. *Response metarules*, triggered by metafacts, may create new metafacts. This is used to model the agents' reactions to a position. Changes in the problem representation caused by such reactions are modelled by *modification metarules* that may modify or completely reshape the goal representation. A forward-chaining engine is used to apply metarules of both kinds (MAT89).

We have extended the NEGOPLAN approach by separating the goal tree into the *private* and *public* tree*. A private tree represents such goals of the agent as departmental or personal goals. (Goals of this kind need not be operational.) The whole tree is not shown to other participants of the decision process, but its leaves may be revealed. A public tree represents the status of the project: its rationale, constraints, budget and schedules. This tree is known to all agents, but each agent can only modify its clearly defined parts. Private goals influence generation of public goals, but they seldom change themselves, perhaps during reorganization or when the corporate culture changes.

An agent belongs to one or more groups. She uses NEGOPLAN's response metarules to infer reactions of the remaining group members. This happens in a competitive situation. In a co-operative situation, response rules deal with the issues more typical of CDM: development of partial solutions, their aggregation, and the synthesis of the agents' individual positions into a position of the whole group or its subgroup.

In the remainder of the paper we present the COON system—an extension of NEGOPLAN from negotiation support to simulation and support of CDM. The new approach is illustrated with a case study that was run on a prototype of COON; we discuss one experiment.

2. Co-operation and Negotiation

2.1. Background

Decision-making has been traditionally considered in the context of a multi-value, multi-variable problem for which a single measure is sought to determine one solution. Goal decomposition, a powerful problem-solving paradigm in Artificial Intelligence, breaks the overall goal down into qualitative elements that are mutually related but may exist independently. Distributed problem solving aims at having a co-operative solution found by nodes in a network jointly engaged in a four-phase process: problem decomposition, subproblem distribution, subproblem solution, and answer synthesis (DAV83, CAM83). The focus often is on interaction between nodes that integrate individually reached subproblem solutions into an overall solution (DUR87). These networks are typically used in distributed sensor networks (LES81), distributed air traffic control (CAM83), and distributed robot systems. To increase the performance of a network, nodes rely on their local views to generate and exchange partial solutions (LES83).

Research on CDM is usually restricted to situations when the nodes—agents—decompose the problem, solve, aggregate and synthesize subproblems (LYN90, SMI81). Structuring the

* The idea of public and private trees is due to Daniel Schwabe and Hugo Fuks.

problem is left to the users (MAH86) or a structure known *a priori* is built into the network's knowledge (SAT86). CDM, as well as group decisions, are concerned with problems with an evolving structure (SHA88), such as disaster management, decision making in unknown environments, financial decision making. Such problems have a structure developed and modified in response to input from decision making agents and to changes in the decision environment (including other stakeholders) (KER90). Problem structuring may be assisted by learning-by-being-told, or by aggregation of known atomic structures into larger structures. We propose the latter approach, and the enhancing of the four-phase process into problem identification, identification and construction of subproblems, subproblem solution, aggregation of subproblems' representation so that a structure of the overall problem is obtained, problem analysis, answer synthesis. The proposed process of CDM with a varying number of co-operating agents is illustrated in Figure 1.

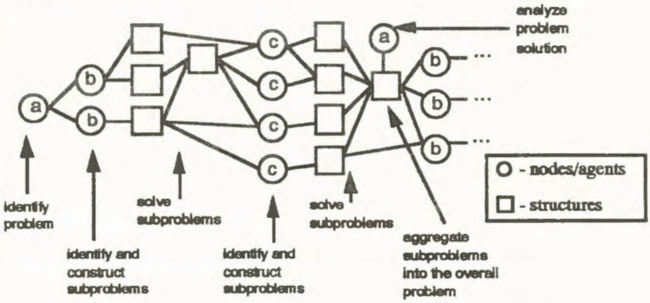

Fig. 1. An example of co-operative problem structuring for a three-level structure

We assume a semi-hierarchical organization of co-operating agents: there is a lot of horizontal exchange of information, each level has decision making powers, and agents can move between levels. A high-ranking originator of the decision problem transmits information to managers who may individually decompose the problem and develop its representations. They discuss and co-ordinate their activities, request their subordinates tò perform certain activities, and inform the originator about work in progress.

2.2. The COON system

The experimental CDM system COON (CO-Operating Negoplans) consists of a number of identical copies of NEGOPLAN, and a simple "communications harness". Each agent has a separate knowledge base with a complete representation of the *private* (personal and departmental) goals that may be non-operational, in contrast with the common *public* goals to be developed by COON. These are initially "atomic"—undeveloped—and one of COON's objectives is to build their commonly acceptable detailed representation.

Problem solving with COON begins when a supervising agent receives a general problem specification from her superior. If she decides that the problem requires team work, she chooses the team members, opens communication channels with them, and sends them information about the problem at hand.

Agents talk in groups. There is a topology of allowable connections and communication lines. An agent who is currently active broadcasts her position to the members of her group. The intended recipients read the position and react to it. (In the COON prototype, one recipient at a time is activated manually.) If this particular set of metafacts does not concern an agent, even if she is in the group, she simply keeps her problem representation intact. One of the recipients receives the control signal and generates her position, broadcasts it, and so on. An agent analyzes her position and compares it with those of other agents. The comparison may be qualitative (e.g., another agent's solution is used to determine if the important goals are achieved) or quantitative (with the use of measures on solution elements) or mixed. If the agent views two solutions as interchangeable or very similar, then she proposes an agreement. This is rejected, accepted, or accepted after modifications.

Agents have their own copies of the public goals. Each broadcast position may cause a redefinition of a public goal, and increase the agent's understanding of this goal's nature.

An agent may operationalize a goal in her own terms that need not be identical with the perspective of other agents. For example, a political decision-maker may care little about budget details, so long as a limit is not exceeded. The existence of many perspectives may lead to inconsistencies: a fact may be deemed both true and false, or two facts may be true but mutually exclusive in the domain. Public goals are accumulated step by step in such a way that no conflicts exist between the private and public goals of one agent. Conflicts between private goals of various agents do not matter.

Long-term results of the work reported in this paper will allow the simulation of organizational structures with generic projects, experimental (on-line) global validation of connections within organizations. It will be possible to develop structures for various typical tasks and verify *ad-hoc* structures (e.g., for handling emergency situations). The system will allow to keep track of information flow, and will automatically produce schematic progress reports. It will reduce the need for face-to-face meetings, and will enable local on-line validation of structures without involving the management.

3. Case Study: Design of a Voice Recognition System

We illustrate our approach by modelling project management procedures of a Canadian government department. It was involved in negotiation and decision making in a recent software project aimed at improving telephone answering service (e.g. to inquiries about welfare payments, or about new laws and regulations). The caller is usually presented with a choice from a few relevant areas, typically handled by an operator who directs the caller to different extension.

Governmental projects are initiated by deputy ministers (DM). DM evaluated last year's performance and decided that the department should lower the number of user complaints. That was found too high because the volume of calls steadily increased in the last five years without changing the number of operators. DM asked the Program Evaluator (PE) for a solution that would not require more operators—due to the freeze on hiring.

PE communicated the request and the constrains to the Program Implementer (PI), asking for an idea of a system that provides faster service with fewer operators than the traditional approach, is cheaper, and promotes the introduction of new technologies. The latter was dictated by PE's mandate to keep the department's infrastructure up to date. PI was a technical person responsible for implementing new projects developed in-house, subcontracted, or purchased as turnkey solutions. Three ideas were proposed: low-tech, easy-to-make, and high-tech. The first (increase the number of telephone lines and have operators use fast text retrieval techniques) was rejected as calling for staff increase. The easy-to-make solution (use touch-tone telephones, lead the user through several layers of menus, and have an operator answer a specific question) was rejected because most customers has rotary or simulated rotary lines: upgrading to touch-tone could require a substantial investment.

The high-tech solution was to have the customer walk through menus by asking her to *say* a one-digit number to choose the next menu. This required no heavy investment, but involved a substantial technological risk due to the immaturity of the voice recognition technology. Still, PI chose this solution, and produced a short description and preliminary cost-time estimates. PE accepted the idea, and consulted with PI to adjust the cost estimates to conform with the governmental guidelines. PE then submitted a project proposal to DM for approval which was granted after requesting modifications in estimates. PE introduced the changes and informed PI about the approval. PI submitted a detailed budget to DM who altered the time estimates. PI negotiated with DM an increased person/year allocation to compensate for the shorter development time, and the budget was accepted by DM.

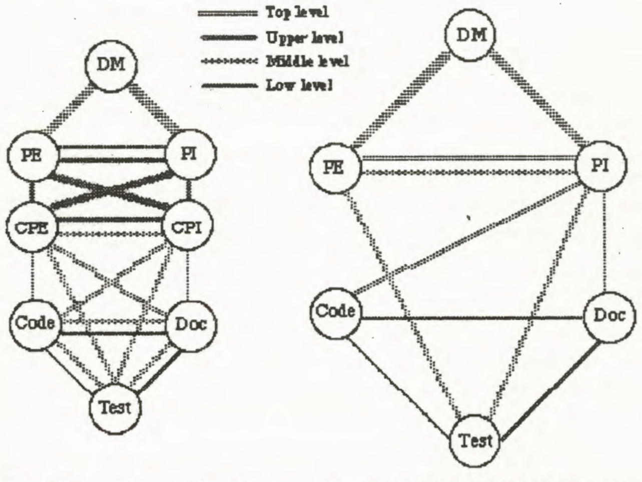

Fig. 2. (a) The full connections between agents. (b) The simplified connections.

Now, PE and PI assigned the task of setting a terms-of-reference document to their contact persons, CPE and CPI who negotiated details: frequency of the steering committee meetings, a communication protocol, the availability of experts from the PE group for consultation with the PI team, the number of initial sites. The terms-of-reference document was submitted to PI and PE for approval, and CPI was made the leader of the project. He negotiated milestones with his Testing, Software, Hardware and Documentation team leaders. The negotiators concerned the scheduling and funds appropriations. CPI negotiated for new programmers to be hired, for part of work to be subcontracted, and for each group's deliverables. CPE had an advisory voice in part of the negotiations between CPI and Testing, because testing (to be performed on clients' sites) required much interdepartmental co-operation.

In Figure 2(a) we show four groups of agents and all the possible connections between them.

4. Implementation

4.1 Goal Representation

In section 3 we described the flow of information between the Deputy Minister (DM), the Program Evaluator (PE), the Program Implementer (PI), PE's Contact Person (CPE), PI's Contact Person (CPI), the Testing Group, the Hardware-Software Group and the Documentation Group. For the purpose of our experiment we have identified PI with CPI and PE with CPE. This simplification does not introduce any substantial changes in our case, because DM, PI and PE do not participate in the negotiations after the terms-of-reference stage has been initiated. Also, some connections are never used in decision-making. The simplified arrangement is illustrated in Figure 2(b).

DM's goals are represented by an initial goal tree (in the NEGOPLAN notation, a *rule* expresses the decomposition of a goal into immediate subgoals). Its leaves are the demands presented to the participants. DM has private and public goals; the synchronization subgoal has only a technical significance—it helps in the timing of agent-to-agent exchanges.

```
goals( dm ) <-  private_goals & public_goals & synchronization.
synchronization  <-  cycle( 0 ).
```

DM is concerned with policy issues and economic issues such as reducing departmental spending by 10%. His public goal is the improvement of public enquiry service.

```
private_goals  <-  policy_issues & economic_issues.
policy_issues  <-  reduce_size_of_public_service.
economic_issues  <-  reduce_costs( 10 ) .
public_goals  <-  request( improve_public_enquiry_service ).
```

The actual public goal, `improve_public_enquiry_service`, is "wrapped" in a standard request predicate. This lets NEGOPLAN use generic metarules to deal with different issues.

DM's public goal will initiate the negotiating process: PI and PE will have to react to DM's request and new issues will appear.

PE's private goals can be divided into personal and departmental. He is concerned with keeping his job and advancing his career. His departmental goals are to increase the use of the newest technology in his department and balance his part of the budget. CPE's private goals are to deliver good service to client departments and avoid antagonizing other departments. We have merged CPE and PE in order to simplify the system, so that their goals must be combined. PE's public goals are not specified because the negotiation has not started yet.

```
goals( pe )  <-  private_goals & public_goals & synchronization.
private_goals  <-  personal_goals & departmental_goals.
personal_goals  <-  not lose_job & get_promoted.
departmental_goals  <- increase_new_technology_use &
  balance_budget & maintain_good_relations.
maintain_good_relations  <-  deliver_good_service.
```

PI's private departmental goals are to increase technical expertise in the department and balance his part of the budget. CPI's private departmental goals are to deliver the project and improve his team by increasing its cohesiveness and effectiveness. The personal goals are to keep the job and get a bonus.

```
goals( pi )  <-  private_goals & public_goals & synchronization.
private_goals  <-  personal_goals & departmental_goals.
personal_goals  <-  not lose_job & get_bonus.
departmental_goals  <- increase_technical_expertise &
  balance_budget & improve_team.
improve_team  <-  increase_effectiveness & increase_cohesiveness.
```

The goals of Testing, Coding and Documentation are similar. Their private goals are to keep their jobs and get bonuses. The public goal is not determined yet.

```
goals( testing )  <-  private_goals & public_goals & synchronization.
private_goals  <-  personal_goals.
personal_goals  <-  not lose_job & get_bonus.
```

4.2 Information Flow and the Co-operative Process

CDM participants need to exchange positions in order to establish issues, clarify their meaning and agree on them throughout the negotiation process. We assume that positions do not flow freely among participants, but rather they travel along previously established information paths.

We have used a broadcast method: all agents post their positions in one place, but everyone reads messages only from the group to which she belongs. For example, DM would only read messages from PE and PI, and would be screened from messages posted by Testing, Documentation and Coding. In our implementation, each participant puts her position in a common position directory, and creates links to this position in the directories of the members of his group. A participant who has read a newly posted position removes his link to it. No links to the positions of the members of a group can be found if nothing new has been posted; the agents poll their links periodically.

During the co-operation process a public tree is created. It becomes the documentation of the project. Here is a self-explanatory example of such a public tree.

```
public_tree <- problem_definition & constraints &
  alternative_solutions & chosen_solution & budget & milestones.
problem_definition <- improve_telephone_answering_service.
improve_telephone_answering_service <-
  improve_level_of_service_by(25).
```

```
constraints <-increase_in_manpower(0) & increase_in_new_technology(40).
alternative_solutions <- voice_recognition.
alternative_solutions <- touch_tone.
alternative_solutions <- improve_text_search.
alternative_solutions <- hire_more_staff.
chosen_solution <- voice_recognition.
budget <- budget(duration,12,months,cost,1,million,tech_staff,12).
milestones <- milestones(coding) & milestones(testing) &
 milestones(documentation).
milestones(coding) <-
schedule(coding,duration,6,months,cost,0.2,tech_staff,4).
milestones(testing) <-
 schedule(testing,duration,6, months, cost,0.4,tech_staff,4).
milestones(documentation) <-
 schedule(documentation,duration,4,months,cost,0.2,tech_staff,4).
```

5. Experiments and Results

We have run six NEGOPLAN systems to simulate six members of the co-operating group: DM, PE, PI, Testing, Coding, Documentation. Each NEGOPLAN had a separate goal representation and metarules. We did a complete run of COON with the connections shown in Figure 3. This allowed us to test our broadcast method, and to find out which structures are passed around along these particular connections. The possibility of experimentally determining the information flow (given a topology) is an important result of our preliminary work.

At the present stage of our work, the COON system is led manually through all the steps. This allows us to debug the system. We are working on the automation of the control process similar to that achieved in just two coupled NEGOPLAN systems (KOP91). In that work we had two NEGOPLANs negotiate a settlement in a union-management dispute without the operator's intervention.

6. Future Work

Co-operation between the agents ends with the establishment of a joint position—a version of the public tree accepted by all members or by the group supervisor (e.g., project initiator). In the prototype of COON we did not consider the environment—an essential element of decision processes—that may be carefully modelled in NEGOPLAN. Re-introducing the environment into COON will enhance the decision process, because agreements may have to be revised due to changes in the decision situation. This leads to the issue of different perception of the same events due to the differences in the agents' individual knowledge bases. This may require adding translation and explanation facilities that would compare and analyze individual interpretations of events.

In the long run, we aim for an automated system that dispenses with human intervention when reacting to the changes in the other participants' positions; at present, the position exchange is synchronized manually. We also consider extending the notion of strategy to allow for strategy changes depending on the circumstances and concession levels of the participants (KOP92). Finally, we intend to incorporate the concept of commitment (FUK91) into our approach.

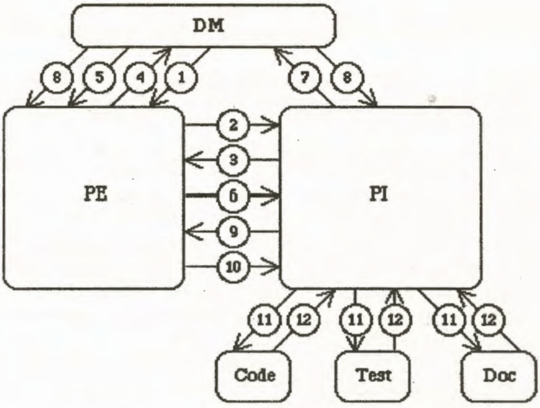

Fig. 3. A sample flow of information between the agents. The numbers on the transitions mean the following items (B, T and M denote the values passed around):

1) request(improve_answering_service)
2) idea_needed(improve_answering_service)
3) idea(improve_answering_service, voice_recognition)
4) proposal(improve_answering_service, voice_recognition)
5) accepted_proposal(improve_answering_service, voice_recognition)
6) project_initiation(voice_recognition)
7) budget_proposal(voice_recognition, B)
8) accepted_budget(voice_recognition, B)

9-10) terms_of_reference(voice_recognition, T)

11-12) milestones(voice_recognition, M)

References

CAM83 S. Cammarata, D. McArthur, R. Steeb (1983) "Strategies of Cooperation in Distributed Problem Solving". *Proc IJCAI 1983*, 767-770.

DAV83 R. Davis, R. G. Smith (1983) "Negotiation as a Metaphor for Distributed Problem Solving". *Artificial Intelligence*, **20**, 63-109.

DUR87 E. H. Durfee, V. Lesser, D. Corkill (1987) "Coherent Cooperation Among Communicating Problem Solvers". *IEEE Transactions on Computers*, **36**, 1275-1291.

FUK91 H. Fuks (1991) "Negotiation Using Commitment and Dialogue". PhD thesis, Department of Computing, Imperial College, University of London.

KER90 G. E. Kersten, S. Szpakowicz, Z. Koperczak (1990) "Modelling of Decision Making for Discrete Processes in Dynamic Environments". *Computers and Mathematics with Applications*, **20**(9/10), 29-43.

KER88 G. E. Kersten, W. Michalowski, S. Matwin, S. Szpakowicz (1988) "Representing the Negotiation Process with a Rule-based Formalism". *Theory and Decision*, **25**, 225-257.

KER91 G. E. Kersten, W. Michalowski, S. Szpakowicz, Z. Koperczak (1991)

"Restructurable Representations of Negotiation". *Management Science*, **37**(10), 1269-1290.

KOP91 Z. Koperczak, S. Matwin, S. Szpakowicz (1991) "A Comparison of the Effectiveness of Negotiation Strategies". *Proc VIII Brazilian Symposium on Artificial Intelligence*, Brasilia, Nov. 1991, 119-127.

KOP92 Z. Koperczak, S. Matwin, S. Szpakowicz (to appear) "Modelling Negotiation Strategies with Two Interacting Expert Systems". *Control and Cybernetics*.

LES81 V. Lesser, D. Corkill (1981) "Functionally Accurate, Cooperative Distributed Systems". *IEEE Transactions on Syst., Man and Cybernetics*, **29**, 1144-1163.

LES83 V. Lesser, D. Corkill (1983) "The Distributed Vehicle Monitoring Testbed: A Tool for Investigating Distributed Problem Solving Networks", *AI Magazine*, **4**, 15-33.

LYN90 K. J. Lynch, J. M. Snyder, S. E. Goodman, W. K. McHenry, L. M. Hoopes (1990) "Requirements for Integrated Collaborative Research Systems", *Proc 23rd Annual Hawaii Int'l Conf on System Sciences*, **4**, 72-81.

MAH86 D. E. Mahling, B. G. Coury, W. B. Croft (1990), "User Models in Cooperative TAsk-Oriented Environments". *Proc 23rd Annual Hawaii Int'l Conf on System Sciences*, **4**, 94-100.

MAT89 S. Matwin, S. Szpakowicz, Z. Koperczak, G. E. Kersten, W. Michalowski (1989) "NEGOPLAN: An Expert System Shell for Negotiation Support". IEEE Expert, **4**(4), 50-62.

SAT86 A. Sathi, T. E. Morton, S. F. Roth (1986) "Callisto: An Intelligent Project Management System". *AI Magazine*, Winter, 34-52.

SHA88 M.F. Shakun (1988) *Evolutionary Systems Design: Policy Making Under Complexity and Group Decision Support Systems*. Oakland: Holden-Day.

SMI81 R. G. Smith, R. Davis (1981) "Frameworks for Cooperation in Distributed Problem Solving". *IEEE Transactions on Syst., Man and Cybernetics*, **11**, 61-70.