ICEB 2006 Proceedings

International Conference on Electronic Business (ICEB)

Fall 11-28-2006

Multi-Behavior Agent Model for Supply Chain Management

Pascal Forget

Sophie D'Amours

Jean-Marc Frayret

Multi-Behavior Agent Model for Supply Chain Management

Pascal Forget[1, 2]; Sophie D'Amours[1, 3]; Jean-Marc Frayret[1, 4]
[1]FOR@C Research Consortium, Université Laval, Quebec, Canada
[2]Doctoral candidate, pascal.forget@centor.ulaval.ca
[3]Director of Research and Administration, sophie.damours@forac.ulaval.ca
[4]Associate Director of Research, jean-marc.frayret@forac.ulaval.ca

Abstract — **Recent economic and international threats to occidental industries have encouraged companies to rethink their planning systems. Due to consolidation, the development of integrated supply chains and the use of inter-organizational information systems have increased business interdependencies and the need for collaboration. Thus, agility and the ability to deal quickly with disturbances in supply chains are critical to maintain overall performance. In order to develop tools to increase the agility of the supply chain and to promote the collaborative management of such disturbances, agent-based technology takes advantage of the ability of agents to make autonomous decisions in a distributed network. This paper proposes a multi-behavior agent model using different decision making approaches in a context where planning decisions are supported by a distributed advanced planning system (*d*-APS). The implementation of this solution is realized through the FOR@C experimental agent-based platform, dedicated to the supply chain planning for the forest products industry.**

Keywords—**Supply chain management, agent architecture, agent-based planning systems, lumber industry.**

I. INTRODUCTION

Recent economic and international threats to occidental industries have encouraged companies to rethink their planning systems in a way to quickly react to and correct deviance from established plans, respond to demand, reduce inventory and exchange information promptly throughout the supply chain [20], in other words become more agile. Agility can be described as the association of flexibility, which is the ability to react to changes by presenting different solutions, and high responsiveness, which is the ability to react in a timely fashion. Global organization forces have recognized that performance is not the feature of a single firm, but the complex output of a network of interconnected firms [27]. Efforts have been deployed to increase supply chain performance as a way to stay competitive with international consortiums. Developed mainly to improve efficiency between partners by increasing coordination and communication, supply chain management has been studied in multiple ways.

For years supply chains have been (and are still mostly) managed in a hierarchical way, where demand plans (customer orders in a context of dynamic demand) are calculated locally and transmitted to suppliers. This sequential planning gives full autonomy to each company and organizational unit involved, but no effort is invested in synchronizing plans and using partner capacity. In fact, the only synchronization tool is the actual demand plan sent to suppliers in order to improve demand forecast and reduce the bullwhip effect. In a context where agility is put forward, where the need to react to changes in a fast manner is increasingly important, exchange of information between partners is crucial to insure plan synchronization and a high degree of agility when faced with disturbances.

The distributed decision making paradigm provides an interesting approach to both increase agility and permit local correction of the plan. This is done by keeping planning decisions distributed, yet use close collaboration mechanisms between organizational units to insure coherence and synchronization of actions. Agent-based technology provides a natural platform that takes advantage of the autonomy of agents and their ability to make decisions in a distributed context, using collaboration and goal-driven decisions. A distributed agent-based Advanced Planning and Scheduling system (*d*-APS) could maintain a real-time plan by re-planning locally and allow for collaboration between agents to deal with disturbances. Plan adjustment in a short period of time leads the way to agile supply chains and increased global performance.

In this paper, we provide a literature review on supply chain planning and how disturbances are handled in such complex environments. We present different uses of agent-based technologies in supply chains and different agent architectures proposed in literature. Then, we describe the experimental agent-based planning platform developed by the FOR@C Research Consortium, which is dedicated to supply chain planning for the forest industry. Our contribution in this paper is to propose a multi-behavior agent model geared with tools designed to improve agility in supply chains. We detail a Multi-behavior agent meta-model, which represents the different behaviors available to the agent, to plan manufacturing activities and deal with disturbances in a distributed collaborative context. Finally, we present a behavior scenario involving a specific disturbance and we suggest an implementation strategy into the FOR@C experimental platform, with the double objectives to prove feasibility and increase supply chain performance.

The North American lumber industry represents a perfect context for this technology. In fact, this industry is already highly distributed, where many business units interact in all production levels. The main advantage of this industry is the large amount of stochastic disturbances in many aspects of the supply chain, mainly due to the highly heterogeneous aspect of the resource, uncertain process output, production of co-products and by-products, price variation in the spot market and demand variation in commodity markets.

II. LITERATURE REVIEW

A. *Planning in supply chains*

Global supply chains involving different companies represent an important planning challenge. Partners do not exchange private information easily and are reluctant to share a common database [42]. When organizational units are part of the same company, which can be called an internal supply chain or intra-organizational supply chain, centralized information and planning systems are sometimes used. Gathering information in a centralized management system and redistributing plans can insure synchronization and optimization of plans. Decision support systems, such as Advanced Planning and Scheduling (APS) systems are sophisticated sets of decision support applications using operational research (OR) techniques to find optimal solutions to complex planning problems [18]. However, even in an internal supply chain, when the number of organizational units grows, planning problems become more complex and hard to handle. Also, because of the quantities of information only available locally and the time it takes to plan the entire supply chain, plans are sometimes not feasible and the supply chain shows low reactivity. In fact, currently available software solutions generally do not provide the necessary support to network organizations and are clearly insufficient in planning and coordinating activities in heterogeneous environments [6, 41a]. Moreover, planning, scheduling and traditional control mechanisms are insufficiently flexible to react to rapid changes in production modes and client needs [26]. In fact, traditional systems have not been developed to work in decentralized, dynamic and heterogeneous environments.

In recent years there has been a new trend of management systems emerging. Because coordination cannot be implicitly transmitted from a top level, collaboration and coordination mechanisms are needed to insure synchronization and consistency throughout the supply chain. This opened the way to an entire new research domain, which is supply chain management (SCM), where researchers are interested in coordination and decision making between supply chain partners to optimize the supply chain performance [44].

B. *Dealing with disturbances in supply chains*

A major difficulty in supply chain planning is dealing with disturbances in an efficient way. In fact, disruptions and uncertainties have been a problem since the beginning of systemized manufacturing and remains an important subject [5]. Disturbances can take different forms, such as change in demand, machine breakdown, late delivery, employee sickness, etc. In a dynamic environment, as in a production plant, as soon as a plan is released, it is immediately subject to random disruptions that quickly render the initial plan obsolete [2].

The traditional way to avoid disturbance related problems is to keep large inventories. In fact, inventory exists more or less as an insurance against uncertainty [13]. While costly, this approach considerably reduces flexibility, because stocked products must be sold even if demand has changed. In contrast, less stock means reducing the overall inventory investment, freeing up available cash flow and improving end-customer service [13].

Keeping low inventory requires close collaboration with partners to ensure precise information on needs. These companies develop business interdependencies since the behavior of one can influence another. In a highly dependent network of entities, when activities are tightly planned, disturbances can have important repercussions throughout the supply chain. For example, a major mechanical breakdown in a strategic third-tier supplier can reduce supply availability for several days, which can have tremendous impacts on the whole supply chain translating in a delay for the final client. Another example is a quick change in demand pattern. When such change happens, every demand plan exchanged between each partner must be updated. If it is not done in a very short period of time, inventories will pile-up and money will be wasted. To counter these problems and their repercussions on the supply chain CPFR (*Collaborative Planning, Forecasting & Replenishment*) methodologies are used and forecasts are prepared jointly.

Much work has been done for dealing with disturbances and uncertainty in a production context. Aytug et al. [5] present a literature review on production scheduling facing uncertainties in the context of a shop floor. Some researchers have presented works on Reactive Scheduling [e.g. 24], which is dedicated to the continuous adaptation of the schedule in a real-time context, with the objective of minimizing perturbations to the initial schedule. Confronted with disturbances, other researchers have worked on finding approaches to modify plans while minimizing impacts on performance using OR techniques [e.g. 2, 3, 7] and artificial intelligence (AI) techniques [e.g. 38]. Replanning is about repairing or starting a new plan in order to adapt to a new context.

Robust scheduling is another approach to deal with disturbances, where the objective is to build a schedule with the best worst-case performance [e.g. 12]. Publications have also presented classifications, management frameworks and the system requirements of disturbances [11, 13, 32, 17].

C. Agent-based system in supply chains

The new trend of distributing decisions resulted in the development of planning systems with agent-based architectures. These approaches are rooted in multi-agent technologies, coming from the AI domain [45]. Agent-based systems focus on implementing individual and social behaviors in a distributed context, using notions like autonomy, reactivity and goal-directed reasoning [9]. The emergence of agent-based systems has represented a real breakthrough in the research world, including researchers from various domains, such as biology, sociology, transportation, management, production, logistics and the military. Agent-based systems are computer systems made of a collection of agents, defined as intelligent software with specific roles and goals, interacting with each other to make the most appropriate decision according to the situation, in order to carry out their part of the planning task [25]. Distributed planning demonstrates many advantages over central planning. For complex problems, sub-problems are easier to solve than centralized problems. Also, because decisions are distributed to different entities, reactivity to changes is increased and the feasibility of plans is likely to improve. The challenge here is that plan performance is linked to agent collaboration capabilities to find acceptable compromises.

Agent-based technology has already been applied to different areas in supply chain management. Parunak [30] presents industrial applications and case studies of agent-based systems, and Shen & Norrie [40] describe more than 30 research projects addressing scheduling, planning and control. More recently, Caridi et al. [10] present a survey and a classification of the different application domains of published multi-agent projects, denoting their degree of maturity.

More specifically, agent-based planning systems have been proposed to manage supply chains and deal with disturbances. Montreuil et al. [27] present the NetMan architecture, an operation system for networked manufacturing organizations that aims to provide a collaborative approach to operations planning. Although the authors created an architecture able to manage unplanned events, they do not present specific behaviors to solve problems following disturbances. Based on intelligent holons, Fletcher et al. [16] present a conceptual architecture of a lumber processing system to improve flexibility and fault tolerance. The ProPlanT multi-agent platform [31] gives decision-making support and simulation possibilities to the manufacturing process. With meta-agents and production agents, they use negotiation, job delegation and task decomposition instead of classic planning and scheduling mechanisms. Building on these research works, we propose to extend the representation of coordination mechanisms in order to increase supply chain agility.

D. Agent architectures

Agents can be designed in various ways, following the internal description of their functions and the connections between them. The architecture of an agent has a direct impact on its behavior and how it reacts when confronted with different situations. Several classifications of architectures are proposed in the literature [e.g. 9, 41]. Basically, three main architectures are prominent: reactive, deliberative and hybrid agents. Reactive and deliberative agents represent extreme cases of behaviors, whereas hybrid agents are positioned somewhere between the two.

A reactive architecture basically links specific inputs to specific outputs. For example, for a specific observation in the environment, the agent has a pre-determined action. These agents have no internal representation of their world and no symbolic representation of knowledge. Although this architecture can perform very well in simple environments, an agent can show a lack of intelligence and adaptability in a more complex world. An evolved reactive architecture is presented by Brooks [8], which is the *subsumption architecture*, also called *behavior-based architecture*. Instead of a single specific reaction to an input, the reactive agent is decomposed into behaviors which are small independent processes that can be triggered, and where some cancel others. Instead of implementing a simple reactivity mechanism, the agent shows an emergent intelligent behavior, resulting from adaptation to its environment. The main advantage of this architecture is the fast adapted response, because no complex processing is needed and different behaviors are available. The disadvantage is the difficulty in creating objective oriented behaviors that follow long term goals and strategies.

In contrast, deliberative agents use their knowledge about their environment and their internal goals to plan and execute actions. They translate information from the world into symbolic knowledge, which they use to update their internal data base. The BDI (Belief-Desire-Intention) architecture [33] is a well-known example of a deliberative architecture, where the agent uses its knowledge about the world (belief) and its goals (desires) to build a plan of action (intention). The advantage of this architecture is the possibility to plan a sequence of actions, in order to meet long term goals. The agent can understand a complex environment and take an appropriate decision following a set of specific inputs. The disadvantage is the slow reaction time in dynamic environments, where situations can change while the agent is processing to find a suitable action. Also, the problematic of knowledge representation is highly complex and is an entire research domain where researchers study new approaches for decades [e.g. 29].

Hybrid agents fit in between these extremes to find an optimal balance of these behaviors. Many authors presented such architectures. The InteRRaP architecture [28] is a layered-based model, composed of three different layers: a behavior layer, a plan layer and a co-operation layer. For a new situation, the agent first tries to find a rule in the behavior layer, that represents the reactive part of the agent. If no rule is known, the agent uses its second layer, the plan layer, where deliberations are executed to build a plan to solve the problem. If no solution is found, the agent uses its last layer, the co-operation layer, where it collaborates with other agents to find a feasible solution. Hybrid agents try to compile advantages of both reactive and deliberative architectures, using the best behavior in each situation. The main disadvantage is the difficulty for the designer to co-ordinate the different layers in order to see an emergent coherent and intelligent agent behavior [9].

E. Hybrid agent architecture in supply chain planning

Several architectures and agent models have been adapted in supply chain context, specifically to improve supply chain performance by planning activities and reacting to disturbances. The variety of possible disturbances, their stochastic distribution and their interactions make this environment highly complex. This is why it is necessary to use deliberative behavior to react to a situation with the best action possible. Also, because the context of supply chains necessitates immediate reaction to changes, fast replanning and instant reply to customer, there is a need for agility only available through reactive behaviors. This is why hybrid agents exhibit the most potential in a supply chain context.

As presented earlier, the InteRRaP architecture provides an interesting approach able to react and deliberate when confronted with disturbances, using different capability levels. The agent can build action plans, depending if an event requires a reactive response, local planning or collaboration for planning. The Agent Building Shell (ABS) [17] is a collection of reusable software components and interfaces needed for any agent involved in a supply chain management system. The ABS is geared to handle perturbations caused by stochastic events in a supply chain. In this architecture, most of the efforts have been focused on defining communication and collaborative aspects. This is done through timely dissemination of information and coordinated revision of plans across the supply chain. The tri-base acquaintance model (3bA) [25] is a collaborative wrapper added to an agent. It provides the possibility of dealing with events in a global perspective instead of resolving problems only in a local view. This is accomplished by using information about other agents without the need of central facilitator. The authors present an example of applications in supply chains and they define the social knowledge needed to increase the efficiency of agents.

From this review, we intend to propose a new hybrid agent model able to deal with disturbances and increase agility in a supply chain context. Our objective is to present an agent model able to use different behaviors following different types of situations, in order to react in the more efficient way and to improve the global supply chain performance. This Multi-behavior agent model is particularly designed to be implemented in a d-APS system, such as the FOR@C experimental platform.

III. FOR@C EXPERIMENTAL PLATFORM

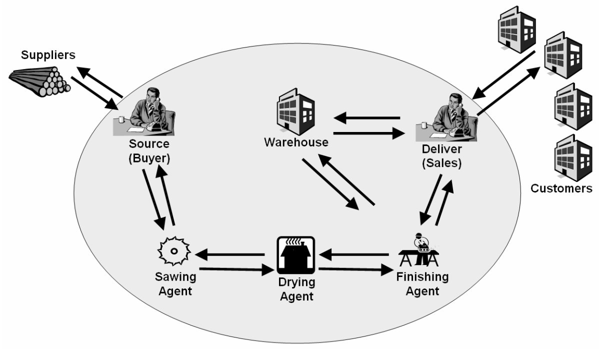
For many years, the planning processes in the North American lumber industry have never been questioned. Due to the highly heterogeneous nature of the resource (i.e. trees) and the inherent complexity of forecasting production throughput, the dominant thinking was to produce the maximum volume with the resource available (push production). Because of the commodity nature of the final product and the standards of sizes and grades, production is oriented towards large batches [19] to take advantage of economies of scale. This industry can be characterized by large inventories, low flexibility and low agility. The recent economic and international threats to the lumber industry have encouraged companies to rethink their planning processes to be able to react quickly to deviance from the plan, respond to demand, reduce inventory and exchange information promptly throughout the supply chain [20]. In order to compensate for the lack of control over the stochastic elements relevant to lumber production, an increase in the exchange of information between the different production centers is necessary, as is to the ability to react quickly in a coordinated manner to changes.

With the purpose of developing a new planning approach for the lumber supply chain, the FOR@C Research Consortium of the Université Laval (Quebec, Canada) has developed an experimental planning platform built on an agent-based architecture for APS, with interaction mechanisms inspired from FIPA (Foundation for Intelligent Physical Agents) standards. This architecture combines agent-based technology with OR techniques to take advantage of the ability of agents to integrate distributed decision problems, and the ability of OR to solve complex decision problems [19]. Because of the distributed context of the supply chain and the use of agents, this platform can be described more precisely as a distributed APS (or *d-APS*), where the first issue is to plan and coordinate all supply chain operations. This platform allows the different production centers to plan and correct deviance independently in line with their proper needs, all the while maintaining feasibility by collaborating with partners.

A. Description of a planning unit

The agent-based architecture presented by FOR@C is based on the natural division of the planning domains. Planning units divide activities between specialized production planning agents: a sawing agent, a drying agent and a finishing agent. This functional distribution is inspired by the SCOR model proposed by the Supply Chain Council [43]. Each of these agents is responsible for supporting the planning of its production center in terms of production output each day. Other agents are also part of the architecture, such as the deliver agent, source agent and warehouse agent. This paper focuses particularly on production planning agents. Figure 1 presents an example of a planning unit, including external exchanges with suppliers and customers.

FIGURE 1
PLANNING UNIT FROM THE FOR@C PLATFORM



The workflow used in a planning unit to plan the internal supply chain upon the receipt of a new demand plan (from outside the PU) is divided in two distinct planning phases: the infinite supply plan and the finite supply plan. During the first phase, deliver agent receives a demand plan from one or many customers. These customers can be part of the same company or different companies. Upon reception, the deliver agent sends a demand plan to the warehouse agent to verify if products are in stock. For non-available products, it sends a demand plan to finishing agent. Using this demand plan, along with resource constraints and lead times the finishing agent builds its plan considering infinite supply and transmits it to the drying agent. Again, using the demand plan, local constraint and considering infinite supply, the drying agent transmits its preferred plan to the sawing agent.

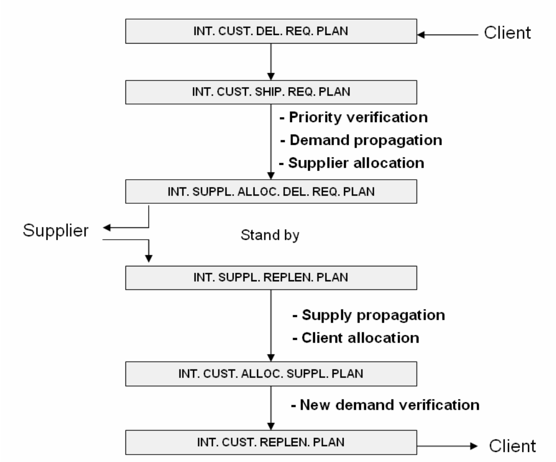
This process continues until suppliers outside the planning unit receive the infinite demand plan. When suppliers answer the demand plans, the source agent receives a supply plan and starts a return loop. This represents the second phase of the planning process, the finite supply plan. The process is largely the same, however plans are built with finite supply, which is the information transmitted by the immediate supplier. For further information the reader is invited to read [19].

If an event occurs in the internal supply chain operations, any agent can initiate a collaboration with its internal clients and suppliers by sending a revised demand or supply plan. This can be triggered by an agent who needs some products to fulfill inventory, lost production or new demand. This explains why agents are also responsible for continuously monitoring their environment and reacting to disturbances. Because of the interaction context, an agent's environment is also made up of all messages received from other agents specifying a new or modified requirement plan, a new or modified replenishment plan, a contingency situation, or a high priority requirement to process.

B. *Actions and task flows*

Each planning agent disposes of objects which can be modified by local actions or actions from other agents. Actions are made possible by task flows, which are sequences of tasks, usually triggered by specific events. A planning agent's standard task flow is the planning protocol (see Figure 2), which is triggered upon reception of a new demand plan from a client. Here, objects are represented by boxes and actions are presented in bold characters. This protocol is divided in two segments. The first concerns modifying a requirement plan, creating a production plan with resource constraints and infinite supply, allocating demand to different suppliers and waiting for an answer. The second concerns receiving supply propositions, updating the production plan with a finite supply, allocating production to clients and modifying a replenishment plan. Optimization algorithms are deployed in the production planning (demand and supply propagation) and allocation tasks to suppliers and clients.

FIGURE 2
CURRENT PLANNING PROTOCOL



Validation of the model was carried out with the collaboration of a forest products company in Canada and real data was used. A supply chain configuration has been developed in order to address the planning of drying and finishing activities inside one plant. This configuration included different types of data, such as production processes, products, orders, on-hand inventory, selling prices, resource costs, forecasted supply, capacity and on-going work. This test covered 100 products, distributed on two dryers and one finishing line, in a planning horizon of 6 weeks. Fifteen agents, more than 600 products, approximately 80 exchange protocols, 100 tasks and 50 task flows were involved. This architecture is a major step toward an improved coordination process for planning requirements.

The current implementation is composed of agents mainly using reactive task flows. These task flows normally in a waiting position and are "fired" when a specific event is noticed. Our objective is to give the agent the possibility of choosing between task flows, by adding specific abilities and knowledge. The agents envisioned in the proposed approach can also exhibit proactive behavior by not only reacting to changes in their environment, but by realizing actions to improve its performance, the performance of a group of agents or the entire supply chain. In this context, we suggest that the notion of goals is fundamental for an agent to adopt the best behavior, following specific situations.

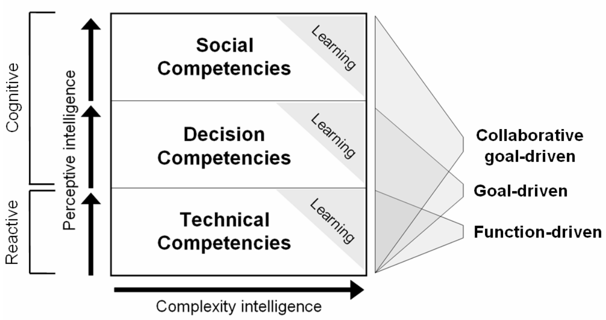
IV. MULTI-BEHAVIOR AGENT MODEL

A. *Agent conceptual model*

Agent-based planning systems, such as the FOR@C experimental platform, represent a promising way to develop new planning systems in the supply chain. The next step is to develop an agent model to describe the characteristics needed to enhance current production planning agents. Facing disturbances, these agents use reactive task flows, triggered by specific messages (from partners or disturbances). To deploy agents with behaviors adapted to different situations and environments, we must give the agent the ability to make choices and the capability to evaluate these choices following its goals and the state of its environment. The agent conceptual model must present the competencies needed in order to show behaviors adapted to a dynamic supply chain context. Inspired by [23] and [35], we define competency as the underlying attributes of an individual determining his capacity to complete successfully a task within a given environment. All competencies can be classified into three categories, which are *attitudes*, *abilities* and *knowledge*. Attitudes are the tendencies to act in a consistent way, following how an individual thinks and feels. Abilities are capabilities to perform specific tasks with the appropriate tools or techniques. Knowledge is defined here as the explicit understanding of information. In other words, the agent knows what the impact of the information is and how this information can be used, both having a direct impact on its behavior [29].

Integrating agent technology and OR tools, the conceptual model (Figure 4) is composed of three distinct layers, describing the different competencies required for supply chain planning. Other agent architectures present a three-layer approach, such as InteRRaP, but the model presented here is a conceptual model. The objective of this conceptual model is to describe the basics capabilities in order to serve as a guideline for further developments, instead of a precise arrangement of functionalities.

FIGURE 4
AGENT CONCEPTUAL MODEL



The bottom layer of the agent model is the *Technical competency layer*. This decision layer includes all reactive tools, tasks and existing task flows, such as OR tools and algorithms, conversation protocols, negotiation protocols and queries. Goals in this layer are related to minimizing effort (computer processing) while maximizing results (optimization functions included in tools). Usual planning processes and reactive corrective actions to face specific disturbance are known and used when needed. An agent strong in this layer but weak in the others would show *function-driven* behavior. Current agents deployed in the FOR@C platform exhibit such behavior. When they face an event, they send a new demand plan to suppliers and then a new supply plan to clients. At this point, a superior reasoning behavior could be achieved by giving new possibilities to deal with disturbances, other than just starting a global re-planning protocol. Sometimes, different tools can be used to deal with the same situation. The agent would be greatly advantaged to have capabilities of analyzing the situation more deeply allowing it to make a clever choice.

This is where the *Decision competency layer* permits the evolution from the reactive behavior to a cognitive behavior. It includes the explicit knowledge of local goals and the progress toward these goals at any time. Geared toward the optimization of the goals it has been assigned to, the agent is primarily concerned by a set of performance metrics that represents what the systems designer has developed. In brief, the agent only knows the impacts of its decisions in terms of this set of metrics. Here, when a disturbance occurs, the agent has the capability to choose which task, task flow, optimization algorithms or complete plan could fit better, according to its own goals. The agent must have a representation of its goals and mechanisms to update and measure the achievement toward these internal goals.

An agent strong in the decision layer and technical layer would present a *goal-driven* behavior. This additional competency clearly gives some advantage to the agent, but it is still unaware of the impact of its decisions on its partners, or on the supply chain. It needs a broader conception to be able to take decisions in the interest of the majority.

The *Social competency layer* fills this gap by integrating the welfare of the collective through collective goals. The agent is now aware of the impacts of its decisions on other agents and on the whole supply chain. While choosing actions to correct deviations from plan, we want the agent to possess the ability to capture the entire potential of the network and be able to minimize impact on others. This layer includes mechanisms to obtain and update collective goals. Collective goals include other agent goals and network tactical goals (i.e. specific product, client selection, supplier selection). If the agent cannot have direct access to other agent goals or collective goals, it must be able to anticipate them. It needs to have the ability to use collaboration protocols with anticipation of other agent reactions. With this competency layer, the agent can choose which task, task flow or plan responds best to collective goals. Agents covering the three previous layers exhibit a *collaborative goal-driven* behavior.

Imbedded in each layer, the learning competency gives the agent the potential to increase its knowledge in each competency layer. A specific action or sequence of actions that showed positive results in a situation could be learned and remembered for the next occurrence. The idea is to further push the articulation of the human decision-making process in our agent model. Various works have presented learning in agent-based systems as a way to improve the performance of manufacturing systems and supply chains. Shen et al. [39] present an interesting literature review on the subject and propose learning techniques adopted in the MetaMorph project. They distinguish learning from history (case-based reasoning) and learning from the future (by simulation). Alonso et al. [4] argue that learning is the most crucial characteristic of intelligent agent systems and present different learning perspectives and techniques. Although this subject is not detailed in this article, it will be studied in the near future, with the objective to be fully implemented in new FOR@C agent architecture.

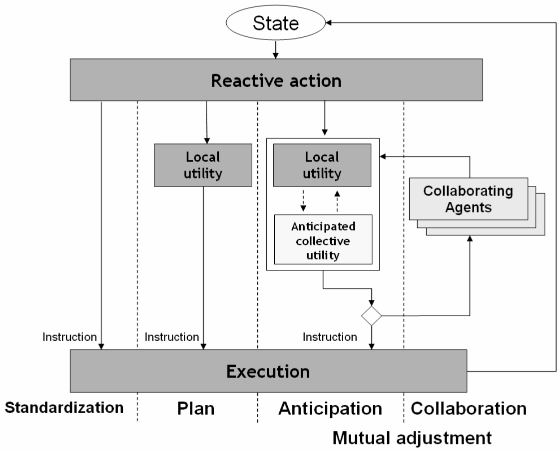
B. Multi-Behavior Agent Meta-Model

The agent conceptual model presented in the above sections gives the basic requirements we believe are necessary in a planning agent involved in a dynamic supply chain. These competencies are quite general and there is need to clarify how they interact to describe a global behavior able to complete the planning tasks. Pursuing this objective, we developed a behavior meta-model (Figure 5). The meta-model presents four basic behaviors to react to a new state in a planning context. Inspired by coordination mechanisms presented in [20], these planning behaviors are *Standardization*, *Plan*, *Anticipation* and *Collaboration*. The two last behaviors are both *Mutual adjustment* behaviors. Depending on the type of disturbance and on the context of the environment, a different behavior can be suggested.

When the system reaches a new state (top of Figure 5), the agent evaluates the situation and selects the appropriate behavior. This reactive action concerns recognizing the type of disturbance and choosing, depending on the type (i.e. new demand requirement, low inventory report, machine breakdown, etc.) and on the context (i.e. short delay of respond, low machine occupancy, etc.) the preferred behavior. It searches in a protocols repository, where events, task flows and protocols are linked together.

If a standardized action is known, a reactive instruction is chosen and executed. This is the simplest reaction available, where a single action or sequence of actions is require for correcting the situation. We call this the *Standardization behavior*. In many situations, humans do not make decisions about what to do, as they just act in a natural way [22]. The same principle is applied here, where a routine behavior is implemented to insure fast reaction time to standard input.

FIGURE 5
MULTI-BEHAVIOR AGENT META MODEL



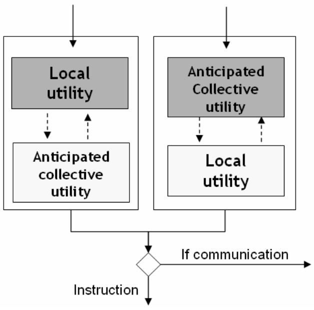
For specific disturbances, more than a standardized action is needed. When the agent recognizes such a situation it must build a plan adapted to the context, where the utility of the different possibilities of actions or sequences of actions is evaluated. Utility can be defined as the degree of usefulness of a state to an agent [15, 34]. When alternative actions are possible to an agent, it must choose the action leading to the state with the highest utility. Utility theory is used to represent and reason about preferences. At this stage, the agent reasons about its utility, using its goals, as defined by its designer. The agent calculates the contribution of choosing a specific action over another. This is *Plan behavior*.

The utility evaluation capability is basically about comparing choices and selecting the best among them. Different characteristics of each action can increase or decrease utility, i.e. probability of success, execution time, perturbation among partners and optimality of solution. Depending on the environment and on goals, utility for an agent can vary.

Other disturbances necessitate an even more complex approach to decision making, especially when the decision will impact other agents in the network. Such situations call for mutual adjustment from the planning agent, where it must take care of the utility of its collectivity (sets of interrelated agents, part of the global supply chain), not only itself. When no full disclosure is possible amongst agents, they can use an approximate anticipation of the impact of its decision on the collective utility. This is the third behavior called *Anticipation behavior*. Anticipation is the conception of a partial or complete model of the partner's reasoning. By taking into account a partner's decision model, the agent can improve its outcome by being closer to the optimal solution. The agent chooses an action following its local utility and then uses an anticipation of the collective utility of the same decision, in order to check if the action is desirable for the collectivity of agents (or the entire supply chain). Anticipation of partners in supply chains has been studied in a hierarchical relation [37] (also called upstream planning).

It is possible to invert the importance of local utility versus collective utility, by changing the anticipation mode (Figure 6). In this way, an agent becomes totally dedicated to the collectivity. The agent first searches for an action that is the best for the collectivity, then it checks the local utility of this decision, to make sure the decision is desirable. If it is not the case, the agent must find another solution. As the first model could be compared as a *realistic behavior*, the latter can be viewed as an *altruistic behavior*. Readings on anticipation can be found in [36, 37].

FIGURE 6
ANTICIPATION MODES



When communication is possible with the collectivity, the agent can adopt a collaboration behavior. This fourth behavior, the *Collaboration behavior*, includes a feedback loop provided by a communication channel between the planning agent and collaborating agents. Collaboration between planning partners in supply chains has been studied in distributed relations [1, 43]. After the anticipation mechanism has been used, a message including the decision for action is transmitted to the agent collectivity for feedback. Therefore, corrections can be made on real opinion or simulation from other agents and not depending only on anticipation of the impact of decisions. When a negotiation loop is introduced in distributed networks such as supply chains with no concrete hierarchy, convergence mechanisms must be followed to insure a potential solution. Dudek and Stadtler [14] propose a negotiation-based scheme between two supply chain partners with exchange of proposals and local associated costs.

This Multi-behavior agent meta-model represents the general case where an agent has to face a new state in its environment. Because the agent has to meet specific objectives, like production rates, client satisfaction, etc, it must react to correct the situation. This model covers all kinds of disturbances met in supply chains and describes what kind of reaction can be taken.

C. *Advantages of the Multi-behavior agent*

Compared to a reactive agent, the Multi-Behavior Agent presents many advantages. Although reactive behavior is still available for quick standardized well-known reactions, it is possible to use a deliberative decision process to apply the best action possible. One of the main advantages is the possibility to deal with actions that have impact on more than one goal. For example, it could allow for reaching a compromise between two different local goals (i.e. minimize inventory vs. maximize production output) or between a local goal and a collective goal (maximize local performance vs. maximize client satisfaction). Also, mutual adjustment behaviors permit agent decisions to confront its anticipation of the collectivity reaction or with direct negotiation to find a compromise profitable for the supply chain.

Another advantage is the possibility to adjust the behavior following external factors. For example, when a client sends a demand plan and requests an acceptance or a refusal in a short time frame, the agent is able to use its standardized behavior fastest respond. In this case, instead of replanning the production plan (that would take a certain amount of time); it would just check available time in the current schedule. In contrast, if a large amount of time is available, the agent would take time to send new demand plans to suppliers to check the possibility of a positive response to the new plan. This example is detailed in the next section.

Moreover, the possibility to anticipate collective goals when communication is not possible (or too long to achieve) represents an appreciable advantage, as better decisions can be taken with limited knowledge. Otherwise, when communication is permitted, negotiation protocols permit a convergence to a compromise that would increase collective performance instead of only individual performance.

Although this description of advantages seems promising, it is still based on a conceptual model. A proof of concept is needed and performance measurements must be accomplished to claim any real advantages. This requires the implementation of the Multi-Behavior Agent in a real-world supply chain context, where manufacturing activities must be planned and confronted with stochastic disturbances.

V. IMPLEMENTATION OF THE MULTI-BEHAVIOR AGENT

In order to implement the Multi-behavior agent, it is necessary to develop different scenarios where a production planning agent is confronted to a specific disturbance. Examples of these scenarios are inspired from the lumber industry, i.e. a major kiln breakdown, out of stock report, unmet harvest and new demand plan. In a supplier/client relationship, we detail the actions of these agents to solve the problem. Only then will it be possible to design and implement an agent able to reproduce the behaviors denoted in the scenarios.

A. *Behavior scenario*

The scenario we retain here is the reception of a new demand plan to the planning agent from a client. A demand plan is formed of different product orders, requested at different dates. Because the planning agent must respond to his client in a very short delay (because he may lose the sale), he gives a delay of one hour to provide the client with an offer. Following this new state, the planning agent must decide if the fulfillment of the new demand is possible or an alternative proposition can be produced, considering the modifications needed in its own production plan, the availability of resources and the delivery dates requested.

According to the agent Behavior Meta-Model (Figure 5), the first action taken by the planning agent is to evaluate which behavior is the most appropriate depending on the situation. With this short delay, the agent clearly does not have time for mutual adjustment (anticipation or collaboration). The standardization behavior has only a single method to answer question and this is to replan the entire production plan incorporating the new requirements. Because this process takes a few hours to complete and necessitates sending information messages to suppliers and waiting for answers, it is not a feasible solution. The behavior to put forward in this case is planning using local utility evaluation. The agent can now choose between different options. During the time left, the agent can check production resource availability to try to fit the new demand plan into the current production plan. Also, it can find a sub-contractor who would be willing to do the job. Here, a message must be sent with a shorter respond delay than asked by the client. Another possibility is to reassign stocks or on-line productions promised to another client to accommodate the new client. Sometimes, these supply reassignments, applied using OR tools, can find optimal solutions while minimizing delivery delays. Between these choices, the agent must perform a utility evaluation of the options, following the percentage of change of success and the profitability of each. Also, if time permits, more than one action can be successively performed. In this example, by reassigning supplies, the agent finds a way to fit in the client's new demand plan. It updates its production plan and sends an acceptance message to its client. In order to select the best action, researchers have proposed approaches using a shop floor context, using case-based reasoning and heuristic search techniques [5].

This example demonstrates the advantage of this design over reactive and deliberative agent architectures. A purely reactive agent would not have time to send a proper answer to the client because its new production plan would not be completed and the sale would be lost. On the other hand, a purely deliberative agent would send a supply plan different from the client demand plan, requesting a negotiation. This process is not adapted to short delay situation.

B. Simulation plan

In order to prove the concept of the multi-behavior agent and test its performance, implementation and simulation must be undertaken. Implementation will be gradual, and behaviors will be developed successively. As the current implementation of the planning agent is mainly the standardized behavior mentioned previously, little work will be needed for this behavior. The first implementation is the plan behavior, using local goals. This step includes the capability of recognizing situations and matching them to the best possible behavior, the environment context taking in account. It also encompasses the design of a utility evaluation function, where the agent can classify actions following its preferences. At this stage, it will be possible to simulate this *Goal-driven agent* (GD) on the FOR@C experimental platform, by designing a supply chain made of GD planning agents. Performance tests will be possible by comparing key performance factors (i.e. resource use, rapidity of answer, etc.) of the GD supply chain with a reactive agent supply chain.

The second implementation will be the anticipation behavior. This includes the introduction of collective goals in the utility function. Also, the anticipation function must be designed with updated mechanisms. Here, testing will be possible by comparing an *Anticipated Goal-driven* (AGD) supply chain with a GD supply chain or a reactive supply chain. In addition, it will be possible to compare both anticipation mode, stating the realistic behavior (local goals as top-model) and altruist behavior (collective goals as top-model).

The final implementation will be the collaboration behavior. Collaboration protocols will be developed, including convergence mechanisms to insure compromise. Again, comparison of performances will be possible between a Collaborative Goal-driven (CGD) supply chain and other previously implemented supply chains.

VI. CONCLUSION

Supply chain planning agent models using the advantage of reactivity, utility evaluation, anticipation and negotiation, such as the Multi-behavior agent, can be a powerful tool to reach appreciated gains when implemented in a distributed planning system such as the FOR@C experimental platform. Following the conceptualization of the intelligent behaviors and their implementation, future work is needed. For example, we intend to test different agent configurations in real-world planning situations to determine the different situations where specific behaviors react well and those where they react badly. In a different perspective, it will be of great interest to increase research efforts on the learning competency, with both its implications and impacts. A Multi-behavior agent geared with learning abilities would be able to update its utility functions to modify its preference for an action which gave good results in the past. This is highly promising and should lead to an even more agile and performing supply chain.

ACKNOWLEDGEMENT

This work was funded by the FOR@C Research Consortium and the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

[1] C. Abid, S. D'Amours & B. Montreuil, "Collaborative order management in distributed manufacturing", *International Journal of Production Research*, vol. 42, no. 2, 2004, pp. 283-302

[2] R.J. Abumaizar & J.A. Svestka, "Rescheduling job shops under random disruptions", *International Journal of Production Research*, vol. 35, no. 7, 1997, pp. 2065-2082

[3] M.S. Akturk & E. Gorgulu, "Match-up scheduling under a machine breakdown", *European Journal of Operational Research*, vol. 112, 1999, pp. 81-97

[4] E. Alonso, M. D'Inverno, D. Kudenko, M. Luck & J. Noble, "Learning in multi-agent systems", *The Knowledge Engineering Review*, vol. 16, no.3, 2001, pp. 277-284

[5] H. Aytug, M.A. Lawley, K. McKay, S. Mohan & R. Uzsoy "Executing production schedules in the face of uncertainties: A review and some future directions", *European Journal of Operational Research*, vol. 161, 2005, pp. 86-110

[6] A. Azevedo, C. Toscano, J.P. Sousa, & A.L. Soares, "An advanced agent-based order planning system for dynamic networked enterprises", *Production Planning & Control*, Vol. 15, No.2, 2004, pp. 133-144

[7] A. Barua, N. Raghavan, A. Upasani & R. Uzsoy, "Implementing global factory schedules in the face of stochastic disruptions", *International Journal of Production Research*, vol. 43, no. 4, 2005, pp. 793-818

[8] R.A. Brooks, "A robust layered control system for a mobile robot", *IEEE Journal of Robotics and Automation*, vol. 2, no. 1, 1986, pp. 14-23

[9] S. Bussmann, N.R. Jennings & M. Wooldridge, *Multiagent Systems for manufacturing Control: A Design Methodology*, 2004, Springer-Verlag

[10] M. Caridi & S. Cavalieri, "Multi-agent systems in production planning and control: an overview", *Production Planning & Control,* vol. 15, no. 2, 2004, pp. 106-118

[11] L. Cloutier, J.M. Frayret, S. D'Amours, B. Espinasse & B. Montreuil, "A commitment-oriented framework for networked manufacturing co-ordination", *International Journal of Computer Integrated Manufacturing*, vol. 14, no. 6, 2001, pp. 522-534

[12] R.L. Daniels & P. Kouvelis, "Robust Scheduling to Hedge against Processing Time Uncertainty in Single-Stage Production", *Management Science*, vol. 41, no. 2, 1995, pp. 363-376

[13] T. Davis, "Effective Supply Chain Management", *Sloan Management Review*, summer 1993, pp. 35-46

[14] G. Dudek & H. Stadtler, "Negotiation-based collaborative planning between supply chains partners", *European Journal of Operational Research*, vol. 163, 2005, pp. 668-687

[15] P.C. Fishburn, *Utility Theory for Decision Making*, 1970, John Wiley and Sons

[16] M. Fletcher, R.W. Brennan, D.H. Norrie & M. Fleetwood, "Reconfiguring processes in a holonic sawmill", *2001 IEEE International Conference on Systems, Man, and Cybernetics*, vol.1, 2001, pp. 158-163

[17] M.S. Fox, M. Barbuceanu & R. Teigen, "Agent-Oriented Supply-Chain Management", *International Journal of Flexible Manufacturing System*, vol. 12, 2000, pp. 165-188

[18] J.M. Frayret, *A conceptual framework to operate collaborative manufacturing networks*, Doctoral thesis, Université Laval, 2002

[19] J.M. Frayret, S. D'Amours, A. Rousseau & S. Harvey, *Agent-Based Supply Chain Planning In The Forest Products Industry*, Working Paper Dt-2005-Jmf-1, Centor, Université Laval, 2005

[20] J.M. Frayret, S. D'Amours & B. Montreuil, "Coordination and Control in Distributed and Agent-Based Manufacturing Systems", *Production Planning & Control*, vol. 15, no. 1, 2004, pp. 42-54

[21] G. Gardner, *Frames of mind: the theory of multiple intelligences*, 1983, New York: Basic Books

[22] A. Guye-Vuillème & D. Thalmann, "Requirements for an Architecture for Believable Social Agents", in *Agents 2000 Barcelona Spain*, ACM, 2000, pp. 48-49

[23] T. Hoffmann, "The meanings of competency", *Journal of European Industrial Training*, vol. 23, no. 6, 1999, pp. 275-285

[24] R. Kerr & E. Szelke, *Artificial Intelligence in Reactive Scheduling*, 1995, Chapman & Hall

[25] V. Marik, M. Pechoucek, O. Stepankova, "Social Knowledge in Multi-agent Systems", *Multi-Agent Systems and Applications*, *ACAI 2001, LNAI 2086*, 2001, Springer, pp. 211-245

[26] F.P. Maturana, W. Shen & D.H. Norrie, "Metamorph: An Adaptive Agent-Based Architecture for Intelligent Manufacturing", *International Journal of Production Research*, vol. 37, no. 10, 1999, pp. 2159-2174

[27] B. Montreuil, J.M. Frayret & S. D'Amours, "A Strategic Framework for Networked Manufacturing", *Computers in Industry*, vol. 42, 2000, pp. 299-317

[28] J.P. Muller, "The Design of Intelligent Agents: A Layered Approach", Lecture *Notes in Artificial Intelligence, LNAI 1177*, 1996, Springer

[29] A. Newell, "The Knowledge Level", *Artificial Intelligence*, vol. 18, 1982, pp. 87-127

[30] H.V.D Parunak, P*ractical and Industrial Applications of Agent-based Systems,* 1998, Industrial Technology Institute

[31] M. Pechoucek, J. Vokrinek & P. Becvar, "ExPlanTech: Multiagent Support for Manufacturing Decision Making", *IEEE Intelligent Systems*, vol. 20, no. 1, 2005, pp. 67-74

[32] L. Pryor & G. Collins, "Planning for Contingencies: A Decision-based Approach", *Journal of Artificial Intelligence Research*, vol. 4, 1996, pp. 287-339

[33] A.S. Rao & M.P. Georgeff, "An abstract architecture for rational agents", In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, 1992, Boston, MA, pp. 439-449

[34] S. Russel & P. Norvig, *Artificial Intelligence: A Modern Approach*, Second Edition, 2003, Prentice Hall, New Jersey

[35] F. Salvador & C. Forza, "Individual competences and training needs for mass customization", In *3rd Interdisciplinary World Congress on Mass Customization and Customization*, 2005, Hong Kong

[36] C. Schneeweiss, *Distributed Decision Making*, Second Edition, 2003, Springer-Verlag

[37] C. Schneeweiss & K. Zimmer, "Hierarchical coordination mechanisms within the supply chain", *European Journal of Operational Research*, vol. 153, 2004, pp. 687-703

[38] E. Szelke & G. Markus, "A blackboard based perspective of reactive scheduling", in *Artificial Intelligence in Reactive Scheduling*, 1995, Chapman & Hall

[39] W. Shen, F. Maturana & D.H. Norrie, "Enhancing the performance of an agent-based manufacturing system through learning and forecasting", *Journal of Intelligent Manufacturing*, vol.11, no. 4, 2000, pp. 365-380

[40] W. Shen & D.H. Norrie DH, "Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey", *Knowledge Information Systems,* vol. 1, no. 2, 1999, pp. 129-156

[41] W. Shen, D.H. Norrie & J.P. Barthès, *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*, 2001, Taylor & Francis, London

[42] H. Stadtler, "Supply chain management and advanced planning – basics, overview and challenges", *European Journal of Operational Research*, vol. 163, 2005, pp. 575-588

[43] S. Stephens, "The Supply Chain Council and the Supply Chain Operations Reference (SCOR) model: Integrating processes, performance measurements, technology and best practice", *Logistics Spectrum*, vol. 34, No.3, 2000, pp. 16-18

[44] T.J. Strader, F.R. Lin & M.J. Shaw, "Simulation of Order Fulfillment in Divergent Assembly Supply Chains", *Journal of Artificial Societies and Social Simulation*, vol. 1, no. 2, 1998

[45] G. Weiss, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, 2003, The MIT Press, Cambridge, Massachusetts