A Framework For Mixed Initiative Agent-Based Contracting

John Collins, Maksim Tsvetovat, Corey Bilot, Rashmi Sundareswara, Tim Lee, Maria Gini
Department of Computer Science and Engineering,
University of Minnesota

Bamshad Mobasher
School of Computer Science, Telecommunications, and Information Systems,
DePaul University

Abstract

We describe an approach to agent-based contracting that combines the advantages of automated negotiation and evaluation of alternatives, with the realities of human decision-making and authority. The MAGNET architecture for multi-agent automated contracting provides a secure market-based infrastructure for complex negotiation, including an extensible ontology layer that defines the terms of discourse for inter-agent communication. A generalized evaluation capability in the agent has the ability to either support human decision-making, or to operate autonomously. We review the background from economics that leads us to the conclusion that for many realistic domains, such a mixed-initiative decision-making approach is preferable to a fully-automated one.

1 Introduction

The University of Minnesota's MAGNET (Multi-Agent Negotiation Testbed) system is an innovative agent-based approach to complex contracting and supply-chain management problems [3]. The MAGNET system comprises a set of agents who negotiate with each other through a market infrastructure using a finite, leveled-commitment protocol. It is designed to support the execution of complex plans among a population of independent, autonomous, heterogeneous, self-interested agents. We call this activity *Plan Execution By Contracting*.

Plan Execution by Contracting is designed to extend the applicability of agent negotiation to new domains, where schedules for production and delivery affect the cost and feasibility of services and products, and where monitoring the performance of task execution is an essential part of the process. This is especially important for application domains such as logistics, dynamic planning and scheduling, and coordination of supply-chain management with production scheduling [17]. In these applications, factors such as flexibility, ease of use, quality, risk, and performance can be more important than cost [8]. In many of these domains, fully autonomous behavior is impractical or unacceptable due to risk factors that cannot easily be quantified [1], or simply because decision making is the responsibility of a person. In other domains, where specifications and business relationships are in place among the negotiating entities, fully autonomous behavior may be very acceptable.

We describe an approach to automated contracting that makes effective use of the capabilities of a community of automated agents, by using them to support human decision-making. The level of autonomy is adjustable; agents may act as communication conduits with some useful evaluation functions, or they may be given full authority to make commitments on behalf of their principals within a circumscribed domain. Agent interactions are mediated through an independent market infrastructure which, among other services, provides a domain ontology, the plan execution by contracting protocol, and authentication services.

This paper is organized as follows. Section 2 describes the market infrastructure in MAGNET, and the interaction of agents within the MAGNET markets and the human decision makers. Section 3 describes the structure of the Ontology that defines the terms of discourse within a specific market. Section 4 describes the evaluation functions of a Magnet contracting agent. Section 5 presents a short discussion on the economic realities behind the limits of realistic autonomous behavior for these agents. Finally, we conclude with a section on related work and some directions for future research.

2 The MAGNET architecture

The MAGNET architecture is a distributed set of objects that can support electronic commerce in a variety of domains, from the simple buying and selling of goods to situations that require complex multi-agent negotiation and contracting.

Each *Market* within the MAGNET environment is a forum for commerce in a particular commodity or business area. Each market includes a set of domain-specific services and facilities, as shown in Figure 1.

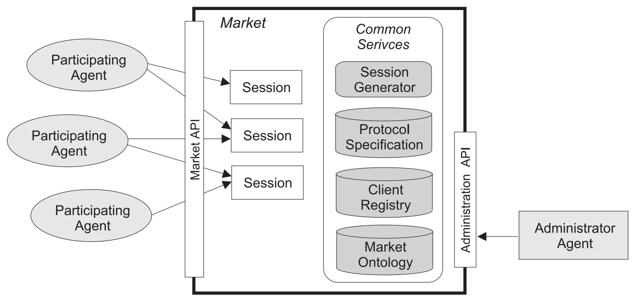

Figure 1: The Structure of a MAGNET Market

An important component of each market is a set of current *Market Sessions* in which the actual agent interactions occur. Agents participating in a market may do so as either session initiators, or as clients, or both. Each session is initiated by a single agent for a particular purpose, and in general multiple agents may join an existing session as clients. The session enforces the protocol rules, and maintains its internal state according to the protocol activity and the passage of time. The architectural components of MAGNET and the current implementation are described in [3].

Within this architecture, a MAGNET agent has four basic functions: planning, negotiation, execution monitoring, and resource management. Within the scope of a negotiation, we distinguish between two agent *roles*, the *Contractor* and the *Supplier*. A Contractor is an agent who has a plan to satisfy some goal, and needs resources outside its direct control in order to carry out that plan. The plan may have a *value* that varies over time. In response to a *call-for-bids*, some set of Supplier agents may offer to provide the requested resources or services, for specified prices, over specified time periods.

The bidding portion of the MAGNET negotiation protocol is specified as a 3-step process that begins when a Contractor agent issues a Call For Bids. The Call For Bids specifies a set of tasks that must be performed, along with time and precedence constraints. Suppliers then may reply with bids, and the contractor accepts the bids it chooses with bid-accept messages. Each bid may specify one or more tasks, including prices and time constraints for the individual tasks, as well as a *discount* for acceptance of the whole bid. We have avoided the need for open-ended negotiation by means of bid break downs and a time-based decommitment penalty [3].

The interactions involved in the basic bidding and execution cycle among the contractor, supplier, and market session are illustrated in Figure 2. A detailed description of this protocol can be found in [9]. During the execution phase, the interactions can be much more complex than indicated here, since either party may decommit from a contract (and pay a penalty), and the contractor is continuously monitoring and repairing its plan by replanning and rebidding when events fail to proceed according to expectations. Finally, the customer and suppliers finalize financial settlements and the session is terminated.

An important component of the contractor agent is a *planner* that converts a top-level goal into a partial-order plan. The planner receives the domain model from the *Market Ontology*. The domain model includes terms for the products or services within the domain, as well as terminology for quality, quantity, features,

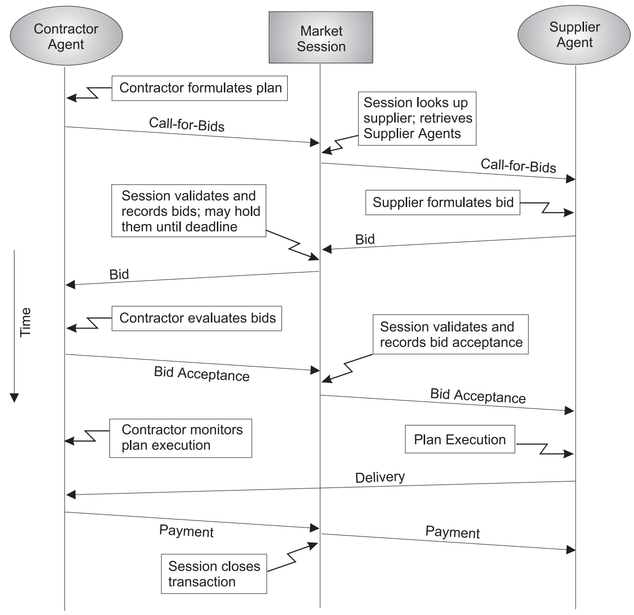

Figure 2: A Typical Session-Mediated Negotiation

terms and conditions of business, etc. and operators to be used by the planner. The generated plan, in turn, is used to construct a call-for-bids within the specified market.

In a mixed initiative system, where human decision makers may need to interact with the contractor agent, it is essential to have a flexible model for the market ontology that can support both automated generation of plans as well as their interactive construction based on human input. In Section 3, we propose such a flexible model based on a unified contract specification system (CSS).

The bid evaluation process is another component of the system which must provide support for a human decision maker. The Contractor agent must compose a set of bids into a feasible plan, and award contracts to the respective Supplier agents. The agent itself can perform this function autonomously, or it may provide its evaluation functions to a human user to assist in exploring the space of possibilities and reaching a decision. In Section 4, we detail a flexible bid evaluation mechanism which provides the core component of our mixed initiative system.

3 The Market Ontology

To facilitate the negotiation process and provide a common representation of objects and concepts among agents, a MAGNET market provides a *market ontology* as a unified contract specification system (CSS). The roles of CSS are:

- *Descriptive:* to provide a description of a process that has already occurred;
- *Prescriptive:* to provide a recipe describing how a process can occur;
- *Semantic:* to provide a semantic model, define objects and activities and establish the scope of the system;
- *Communicative:* to enable interoperability between different agents in the marketplace.

3.1 The Elements and Structure of an Ontology

The CSS is composed of several major classes of elements:

- *Atoms* are the basic units of trade in the system. An atom can represent a physical object, a unit of a resource, or a unit of work done. A market ontology contains a fixed set of atoms which define the terms of discourse in a particular market.

  There are several main types of atoms, characterized by their behavior in the system.

  - *Abstract* atoms are descriptions or specifications of atoms that can exist but have not been produced yet. The main use of abstract atoms is in planning.
  - *Concrete* atoms represent real objects, as opposed to descriptions of objects. The part that distinguishes concrete atoms from abstract is a cryptographic kernel that certifies an atom as being real.
  - *Consumable* atoms represent raw materials, or other objects that can be consumed in process of making other objects.
  - *Capital* atoms represent capital investment - such as tools, buildings and assembly lines.
  - *Whole* atoms represent units of goods that cannot be divided (or $Utility(\frac{atom}{d}) \leq \frac{Utility(atom)}{d}$). An example of such atom is a representation of a clarinet, whose utility as a whole is that of a musical instrument - but decreases drastically as one sells its parts.
  - *Fractional* atoms represent divisible goods, where $Utility(\frac{atom}{d}) = \frac{Utility(atom)}{d}$. An example of such is oil which can be measured and sold in any amount.

    Money can be represented by fractional atoms. In fact, any fractional atom can be used as a medium of exchange, if agents in the market agree to such use.

- *Actions* Actions represent all activities that are possible in the system. An action is defined as

  $$A = \{P, [T], [S]\}$$

  where P is a set of input an output ports that define required materials and tools, as well as the result of the action, T is an optional set of time constraints, and S is the state.

  A port specifies the type of atom that is required to commence the action or the type of atom produced as the result of this action. There are several types of ports that define what happens to an atom that has been passed through a given port.

  Consume is an input port that signifies a resource to be used up in process of completing an action. The atom passing through a Consume port will not exist after the action has started. *Produce* is an output port that creates an atom which did not exist before the performance of the action. *Use* represents a resource that is exclusively used for performance of the action, and is released once the action has concluded.

  Two other types of ports, *Specify* and *ToSpec*, allow for dynamic creation of atoms in the system. *Specify* produces an abstract atom specifying the action to occur at some later time, and *ToSpec* performs an action specified by such abstract atom.

  Within a particular market, actions defined in the ontology represent the production capabilities of actual or potential suppliers in that market.

3.2 Example

Let us suppose Alice's Musical Instruments (A) builds a new clarinet plant. Figure 3 shows a sample manufacturing plan using the CSS model. The plan shows purchasing raw materials, production and sale. The purchase of raw materials (tasks A_1 and A_2) represent contracts with external suppliers, Bob's Wood (B) and Carol's Metals (C). In order to make their services known, suppliers B and C join the raw materials market and register themselves as suppliers of wood and metal parts. When A starts seeking raw material suppliers, she retrieves the *abstract atoms* describing wood and metal parts and solicits using the MAGNET Call-For-Bids protocol. Eventually, she makes contracts with B and C, at which point both B and C start

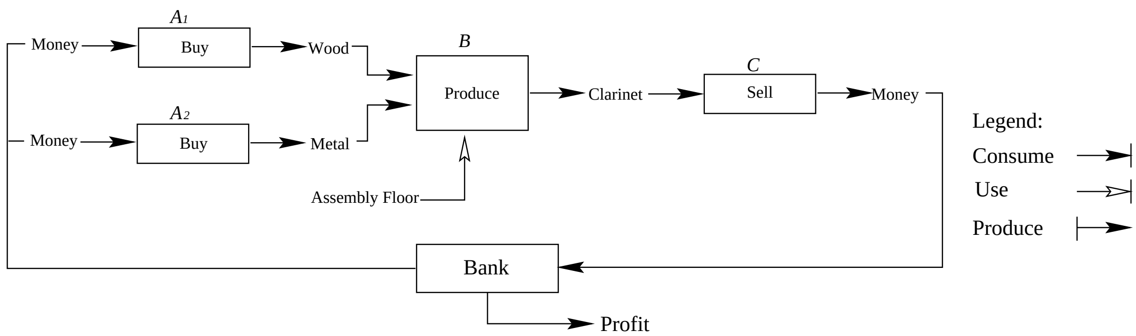

Figure 3: A sample plan using the CSS model.

supplying *concrete atoms* of raw materials (as well as the physical goods corresponding to the concrete atoms) in exchange for *fractional concrete atoms* of money.

While the production step (B) is seen as one action on this plan, in A's internal representation it is actually a complex process of manufacturing a clarinet. However, the world outside Alice's plant does not need aware of the internal complexity. Alice can hide it by only presenting simple input-output relationships to the ontology - or expose the whole process (which would make business sense if she wanted to make not only clarinets but also clarinet parts for other manufacturers). Similarly, suppliers B and C only have to expose simplified representation of their operation to their customers.

Finally, in order to sell the finished product, Alice must enter the market for musical instruments and find an abstract atom in its ontology that adequately describes her clarinets. Then, she proceeds to register herself as a supplier of clarinets. In Alice's internal representation, the sale of clarinets is represented by

$$C = \{consume(clarinet), produce(money)\}.$$

However, the external representation for her customers must be the reverse, or

$$C' = \{consume(money), produce(clarinet)\}.$$

Alice can also register herself as capable of producing clarinets out of raw materials for a fee,

$$C'' = \{consume(wood), consume(metal), consume(money), produce(clarinet)\},$$

or build a clarinet at your site,

$$C''' = \{consume(money), use(assembly\ floor), produce(clarinet)\}.$$

3.3 Issues

There are a number of issues that have to be resolved in order for the CSS ontology to be useful in the real world, including

- *Authentication issues* arise when atoms in an electronic commerce system must represent real physical goods. Every physical object is unique in the sense that one cannot make a perfect atom-by-atom copy of it (and even if that was possible, it won't be made out of the same atoms). An electronic representation of such physical object must also be unique, even though data objects can be freely copied. One way to achieve that uniqueness is to produce physical objects that carry a serial number, and include an encrypted version of the same serial number in its electronic representation.

- In CSS, *money* is just one type of a fractional atom. Thus, nothing prevents some market from declaring its universal medium of exchange as something else (shells, colored beads or crude oil). In fact, nothing prevents the economy from developing multiple and parallel mediums of exchange, or from building a barter-based economy. This makes bidding and evaluation of bids a much more complex process, and therefore agents or markets may need to specify the types of "currency" allowed in exchanges.

4 Bid Evaluation

At the conclusion of the bidding cycle, the Contractor agent has a set of bids that must be composed into a feasible plan and awarded to the respective Supplier agents. The agent itself can perform this function autonomously, or it may provide its evaluation functions to a human user to assist in exploring the space of possibilities and reaching a decision. In general, the result of this process is expected to be a combination of bids that provides coverage of all tasks, allows for a feasible schedule, and minimizes a combination of cost and risk factors.

When the subject of negotiation is a complex plan with many individual tasks and a rich set of temporal constraints, there is a tradeoff to be considered, between specifying the plan in such a way as to guarantee that any non-overlapping combination of supplier bids will compose feasibly (all temporal constraints satisfied), or giving suppliers more flexibility in an attempt to reduce costs and potentially tighten up the schedule. If the latter approach is taken with respect to a complex plan, the evaluation process becomes a challenging proposition for both humans and automated agents.

If a human is making the final decision, it is difficult for the agent to predict which cost, temporal, and risk factors are most important. For that reason, we provide a set of evaluation criteria that can be traded off against each other by setting a few coefficients. We also provide the ability to add constraints (e.g. Joe's Plumbing isn't going to work here again no matter how low his bids are!!) to the problem description.

For each bid, the Contractor has the option of selecting the entire bid and paying an overall discounted price, or selecting a subset of the task bids from the bids. Timing information for a bid includes early start, late finish, and duration for each task in the bid. Bids that cover multiple tasks are required to specify prices for each of the individual tasks, as well as a (possibly discounted) price for the entire set of tasks. The semantics of a bid is that a supplier is willing to perform the task or combination of tasks for the bid price, starting at any time in the time window specified in the bid.

Because of the wide expected variability in problem size, and because of the need to be able to limit the time spent evaluating bids, the evaluator uses an adaptive strategy. If the problem complexity is low, a systematic search is used. Bids that are incompatible with each other, either due to overlapping coverage or temporal infeasibility, are marked. If task coverage remains possible after this step, then combinations that meet task coverage and temporal feasibility constraints are evaluated for cost and risk.

If the combination of plan size, number of bids, and number of tasks per bid is beyond the systematic-search threshold, then a heuristically-guided simulated-annealing search [11] is used to generate bid combinations for evaluation. Here is a general outline of the search procedure:

- *choose a node*: A previous partial or complete solution is chosen as a base for the next expansion. A node is a partial or complete mapping of tasks to bids. Each has a set of numeric *evaluations* that represents the costs of the mapped bids, a risk value that estimates recovery cost, and penalties for unmapped tasks and infeasibilities. Unmapped tasks are assigned the average cost of all bids for that task. The root node is the mapping of all tasks to no bids.

- *choose an expansion*: Expansions are made by adding a bid or individual task-bid to a node. A number of *bid selectors* are available to focus the search on different types of improvement. Each one selects either a full bid or a bid component, which is added to the node. If one or more tasks mapped by the selected bid were already covered by other bids, those bids are removed from the node's mapping.

  The following bid-selection methods have been implemented and tested. Note that the feasibility and cost improvement methods have significant complexity costs associated with them.

  - *Random Bid, Random Bid Component*: Choose a bid or a bid component at random, and attempt to add it to the node. The ratio of bids to bid components is adjustable. This method is fast ($O(1)$) and promotes general exploration.
  - *Coverage Improvement*: Choose a bid or bid component that covers a task that is not mapped in the node. The probability of choosing a bid component is equal to the coverage factor of the node. If a bid component is chosen, the coverage factor will increase; if a bid is chosen, the probability of improving coverage is $P_{ci} > 0.5$. This method is also $O(1)$ if the set of unmapped tasks and the set of bids per task is stored.

- *Feasibility Improvement*: The mapping is scanned to find tasks that have negative slack

$$b.t_{es} + d_a > b.t_{lf},$$

  are constrained by their bids rather than by predecessors or successors, and could be moved in a direction that would relieve the negative slack. They are sorted by their potential to reduce infeasibility, and saved. The untried bid or bid component with the highest potential to reduce infeasibility is chosen. Note that when a bid is chosen, there is no guarantee that it will not introduce other infeasibilities.

- *Cost Improvement*: Choose the (untried) bid or bid component that is responsible for the maximum positive deviation from the average price, and replace it with a lower-priced bid that covers at least the task with the highest positive cost deviation.

These selectors can also be composed together and used to generate focused improvement for a given node.

- *evaluate the resulting node*: If a valid node was produced by the expansion, it is tested for coverage, feasibility, cost, and risk. Cost and risk are measured in the same units, while coverage and feasibility involve arbitrary penalty factors. The penalties must be set high enough to drive the search toward high coverage and low infeasibility, and low enough that nodes can carry some penalties and yet remain in contention for further expansion. Experimentation has led us to rate a non-covered task at about 3 times the average cost (as determined by the set of bid received) for the task, and infeasibility so that a task that has negative slack equal to half its duration is penalized by about 4 times its average cost. The risk evaluation is an approximation of the expected cost of recovering from failure of a given bid mapping. A detailed description of this portion of the evaluation is presented in [2].

- *stopping criterion*: The search ends when some number of expansions have been attempted without finding an improvement. The best feasible and covered solution, if any, is returned. Clearly this number needs to be larger as problem complexity increases. For this experiment we used a value of $40 \log pb$ where p is the number of tasks in the plan, and b is the number of bids received.

Use of the evaluator in a mixed-initiative setting involves exposing to the user the ability to adjust the weighting factors, control over the bid selectors used, the ability of the search to return multiple proposed solutions, and a facility that allows the user to add constraints to the problem and re-run the search. The multiple-solution approach is enabled by providing multiple sets of weighting factors for coverage, feasibility, cost, and risk. For each set of factors, the search tracks the best solution encountered and returns the resulting set of proposed solutions. User controlled constraints can include forced inclusion or exclusion of particular bids or vendors, and additional temporal constraints.

For example, if a manager were using a MAGNET-based market system to choose suppliers for an upcoming manufacturing run, she might be less concerned about cost than about other issues such as minimizing the overall makespan or leaving adequate slack around a task that has been hard to schedule accurately in the past. If the evaluator were to record all generated solutions that met some minimum quality threshold, an OLAP browsing tool could be used to explore the parameter space of potential solutions along multiple dimensions such as overall cost, minimum slack, overall makespan, or vendor rating.

5 Limitations of Automatic Risk Evaluation Strategies

If MAGNET is to produce plans that are acceptable to a human decision-maker (DM), then MAGNET must consider not only the expected value of a plan of action, but also the risk associated with the plan. At first glance, it may seem intuitive that MAGNET should select the plan that maximizes expected value. Paradoxically, maximizing expected value may lead to plans that few human DM's would find acceptable. This observation is hardly new to the field of economics. Early economic ideas by Nicholas Bernoulli showed the weakness of maximizing expected value as a sole criterion for choosing a gamble. Bernoulli constructed a gamble, dubbed the St. Petersburg paradox, that very few people would accept, even though the expected value of the gamble approaches positive infinity. Bernoulli thus argued that people choose a gamble on some basis other than expected value [1].

To produce plans that are more acceptable to a human DM, MAGNET will incorporate concepts from expected utility theory (EUT). EUT states that a gamble will be acceptable to a DM based, not on the expected value of the gamble, but on the expected utility of the gamble. The expected utility of a gamble is the sum, for each possible outcome, of the probability of the outcome and the utility of the outcome. The utility of an outcome is based on a von Neumann-Morgenstern utility curve, a curve that is a subjective mapping of a level of wealth to a level of utility. Thus the shape of the DM's utility curve will guide MAGNET in its choice of a plan that is acceptable to the DM. By contrast, the plans produced by an expected value approach implicitly assume that a DM derives equal utility from each successive dollar gained or lost. This is the implicit assumption that Bernoulli argued was incorrect [1].

On a more intuitive level, EUT will allow MAGNET to model risk aversion. Risk aversion results when the shape of a DM's von Neumann-Morgenstern utility curve is increasing, but concave. This corresponds to a DM who always prefers more wealth to less wealth, but who derives less additional utility for each successive increment in wealth. A DM with a utility curve of this shape will often reject a fair gamble. Such a DM is termed "risk averse". The risk-averse DM will reject a fair gamble because the DM will derive less utility from the wealth gained than from the wealth that could potentially be lost, even though the amount of wealth gained or lost may be equal [10].

Borrowing concepts from EUT, while resolving some issues, raises others. Is a utility curve something that a DM can intuitively understand and input into the MAGNET system? If not, can MAGNET present a series of questions to the DM, using the answers to derive a curve? Will MAGNET derive a utility curve by observing and learning from the choices of the DM? Or will it be some combination of the above?

In its attempt to weigh plans based on expected utility, MAGNET must overcome yet another obstacle: uncertainty. Thus far, we have assumed that the probabilities of various outcomes are known. MAGNET, however, must work in an environment in which these probabilities are unknown.

Certainly, the fields of economics and statistics have tools to aid with the estimation of unknown probabilities, e.g., entropy, the Central Limit Theorem, and Bayes Theorem. The MAGNET architecture also provides a market mechanism to collect data on suppliers and their past performance. These will certainly act as starting points for estimating probabilities connected with the future performance of individual suppliers.

Still, larger challenges remain. These challenges will tend to drive the MAGNET implementation away from a purely autonomous agent that calculates many-parametered complex probability distributions — and more towards a mixed-initiative system which requires the seasoned guidance of a human DM. First, MAGNET may be called upon to operate in markets for which there is little prior data on supplier performance. In the absence of data, MAGNET will need guidance from the DM in the form of subjective probabilities. Furthermore, MAGNET is designed to operate in an environment of heterogeneous self-interested agents. These agents may change their behaviors over time (perhaps abruptly) so that past data, particularly old data, may be less relevant or even irrelevant to estimating the future performance of suppliers. Thus the problem becomes less about estimating unknown probabilities and more about estimating probabilities that are largely unknowable. Furthermore, while some market ontologies may be well defined in that all relevant data can be represented in the ontology, in real-life situations a DM may need to intervene in MAGNET's decisions, providing the type of subjective input and seasoned judgments that an ontology cannot reasonably represent.

6 Related Work

Markets play an essential role in the economy, by facilitating the exchange of information, goods, and services, and there is growing evidence that software agents will play an increasing role as mediators in electronic markets.

Of the essential functions of a market (i) matching buyers and sellers, (ii) facilitating transactions, and (iii) providing institutional infrastructure existing software agents mostly satisfy the need to search for product and price information (see, for instance, [4, 18]), but there is a growing need for agents capable of sophisticated automated negotiations [7].

Automated contracting protocols generally assume direct agent-to-agent negotiation. For example, Smith [16] pioneered research in communication among cooperating distributed agents with the Contract Net protocol. The Contract Net has been extended by Sandholm and Lesser [12] to self-interested agents. In these systems, agents communicate and negotiate directly with each other. In our work, agents interact with each other through a market. To the extent that we require the existence of an external market mechanism as an intermediary, our proposed framework is similar to that of Wellman's market-oriented programming used in AuctionBot [19].

There are a number of groups and projects working on market ontologies. Fox and the group at University of Toronto have developed the TOVE enterprise modeling ontology [5, 6], and the National Institute for Standards in Technology (NIST) group on process specification has developed the Process Specification Language (PSL) [13, 15, 14] which also addresses some of the issues of market ontologies. Both TOVE and PSL are similar in the respect that they use sets of object-action relationships to represent complex plans. However, both of these structures are intended to define internal processes only and provide no mechanism for hiding internal complexity or distinguishing between internal and external representations. They also make no provision for communication of object and action information. CSS enhances the data model of TOVE and PSL to both represent internal business processes and expose external process descriptions to potential customers.

7 Conclusions and Future Work

The MAGNET system offers a framework and capability for multiple agents to reach agreement on complex contracts in a fully autonomous manner. However, in many real-world situations, such autonomous behavior is unlikely to be acceptable to the people and human organizations on whose behalf the agents are working. This is due in part to the difficulty of assessing risk in a way that reflects human preferences, in part to the fact that some of the risk factors are fundamentally unknowable, and in part because many humans are reluctant to delegate authority to commit resources to an automated agent.

To overcome the drawbacks of fully-autonomous behavior, we have shown how the MAGNET framework can be adapted to a mixed-initiative decision support system. The market infrastructure provides a secure environment for inter-agent negotiation, and its ontology provides a standard vocabulary that can be shared by people and automated agents. The contractor agent can use its evaluator to make bid-award decisions, or it can expose the internals of its evaluator as a rich computational tool for a human user.

This paper has presented a work in progress. The basic MAGNET infrastructure and the evaluator have been implemented and tested, and much of the functionality needed for mixed-initiative behavior is in place. A user interface that allows user interaction with the evaluator has not been built, and effectively communicating such multidimensional evaluation results will be a challenge. The representational adequacy of the CSS ontology approach has yet to be validated in any realistic domain. An ideal test of these ideas might be to compare the relative performance of humans, automated MAGNET agents, and a mixed-initiative approach, on a selection of realistic or real-world problems.

References

[1] Tapan Biswas. *Decision-Making Under Uncertainty.* St. Martin's Press, Inc., 1997.

[2] John Collins, Makesim Tsvetovat, Rashmi Sundareswara, Joshua Van Tonder, Maria Gini, and Bamshad Mobasher. Evaluating risk: Flexibility and feasibility in multi-agent contracting. Technical Report 99-001, University of Minnesota Department of Computer Science and Engineering, Minneapolis, Minnesota, February 1999.

[3] John Collins, Ben Youngdahl, Scott Jamison, Bamshad Mobasher, and Maria Gini. A market architecture for multi-agent contracting. In *Proc. of the Second Int'l Conf. on Autonomous Agents*, pages 285–292, May 1998.

[4] B. Doorenbos, O. Etzioni, and D. Weld. A scalable comparison-shopping agent for the world-wide web. In *Proc. of the First Int'l Conf. on Autonomous Agents*, pages 39–48, February 1997.

[5] M. S. Fox, J. F. Chionglo, and F. G. Fadel. A common sense model of the enterprise. In *Proceedings of the 2nd Industrial Engineering Research Conference*, volume 1, pages 425–429, Norcross GA, USA, 1993.

[6] M. S. FOX and et al. An organization ontology for enterprise modeling: Preliminary concepts. *Computers in Industry*, 1996.

[7] Robert H. Guttman, Alexandros G. Moukas, and Pattie Maes. Agent-mediated electronic commerce: a survey. *Knowledge Engineering Review*, June 1998.

[8] S. Helper. How much has really changed between us manufacturers and their suppliers. *Sloan Management Review*, 32(4):15–28, 1991.

[9] Scott Jamison. A negotiation protocol and market agent model for complex transactions in electronic commerce. Technical report, University of Minnesota, Department of Computer Science and Engineering, Minneapolis, MN, 1997.

[10] John W. Pratt. Risk aversion in the small and in the large. *Econometrica*, 32:122–136, 1964.

[11] Colin R. Reeves. *Modern Heuristic Techniques for Combinatorial Problems.* John Wiley & Sons, New York, NY, 1993.

[12] Tuomas Sandholm and Victor Lesser. Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *1st Int'l Conf. on Multiagent Systems*, pages 328–335, San Francisco, 1995.

[13] Craig Schlenoff, Rob Ivester, and Amy Knutilla. Unified process specification language: Requirements for modeling process. Technical Report NISTIR 5910, National Institute of Standards and Technology, Gaithersburg MD, September 1996.

[14] Craig Schlenoff, Rob Ivester, and Amy Knutilla. A robust process ontology for manufacturing systems integration. National Institute of Standards and Technology, 1998.

[15] Craigh Schlenoff, Amy Knutilla, and Steven Ray, editors. *Proceedings of the Process Specification Language (PSL) Roundtable*, Gaithersburg MD, April 1997. National Institute of Standards and Technology.

[16] R. G. Smith. The contract net protocol: High level communication and control in a distributed problem solver. *IEEE Trans. on Computers*, 29(12):1104–1113, December 1980.

[17] J. M. Swaminathan, S.F. Smith, and N.M. Sadeh. Modeling the dynamics of supply chains: A multi-agent approach. *Decision Sciences*, 1997.

[18] K. Sycara, K. Decker, and M. Williamson. Middle-agents for the Internet. In *Proc. of the 15th Joint Conf. on Artificial Intelligence*, 1997.

[19] P.R. Wurman, M.P. Wellman, and W.E. Walsh. The Michigan Internet AuctionBot: A configurable auction server for human and software agents. In *Second Int'l Conf. on Autonomous Agents*, May 1998.