Contractual Agent Societies: Negotiated shared context and social control in open multi-agent systems

Chrysanthos Dellarocas
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
Dell@mit.edu

Abstract

Information systems for supporting the fluid organizations of the 21st century must be correspondingly open and agile, able to automatically configure themselves out of heterogeneous system components, accommodate the dynamic exit and entry of hitherto unknown participants and maintain system stability in the face of limited trust. This paper introduces the concept of Contractual Agent Societies (CAS) as a metaphor for building such open information systems. CAS are open information systems where independently developed agents configure themselves automatically through a set of dynamically negotiated social contracts. Social contracts define the shared context of agent interactions, including ontologies, joint beliefs, joint goals, normative behaviors, etc. In addition, they specify classes of associated exceptions (deviations from ideal behavior) together with associated prevention and resolution mechanisms. A research agenda for developing the infrastructure that will enable the construction of practical CAS is discussed. Significant aspects of that infrastructure include a language and ontology for representing social contracts, as well as the definition of agent architectures capable of negotiating social contracts and adapting their behavior accordingly.

Keywords:

Open systems, electronic institutions, exception handling, electronic contracts

1. Introduction

An increasing number of application domains are characterized by the need for organizations that have never worked together in the past to become partners in a transaction, task or mission and to begin to work together effectively in a matter of hours, minutes, or even seconds. Information systems for supporting such domains must be correspondingly open and agile, able to quickly (and, ideally, automatically) configure themselves out of heterogeneous system components. Examples of this trend include international coalition forces and disaster recovery operations in the military domain [Coa99] and open marketplaces and virtual supply chains in the electronic commerce domain [Fis96, Tsv96].

Multi-agent systems (MAS) represent one of the most promising approaches for creating open information systems because of their ability to use multi-agent coordination protocols to dynamically self-organize themselves as their problems and constituent agents change [Jen96]. Nevertheless, most agent architectures proposed to date are closed, in the sense that their agents cannot easily interoperate with agents developed for different architectures.

One important reason why automatic run-time interoperability is difficult is the existence of significant implicit (hard-coded) shared context among the members of multi-agent architectures. Members of a closed multi-agent system rely for their interaction on a number of agreements on policies, protocols, shared facilities, etc., typically made during design time and incorporated in the agent code. In order for heterogeneous agents to dynamically interoperate, this implicit *social interaction context* has to be explicitly communicated among them and possibly renegotiated at run-time.

Furthermore, because of limited trust and lack of control over the actions of independently developed agents, open systems raise issues of stability and control. In such partially controlled multi-agent systems [Bra96] it is important to develop social control mechanisms that can either discourage agents from violating their agreed upon interaction behavior, or detect and resolve violations as they occur.

Based on ideas from the study of human organizations and human societies, we propose a new metaphor for building open multi-agent systems, which we call contractual agent societies. Contractual Agent Societies (CAS) are open systems where independently developed agents configure themselves automatically and coordinate their behavior through a set of dynamically negotiated *social contracts*, which define the shared context of agent interactions, and a system of *social control*, which is responsible for avoiding, or detecting and resolving exceptions, that is, deviations from the desirable system behavior.

The following sections describe our vision of CAS in detail, lay out a research agenda for achieving this vision, report on our current progress in building prototype CAS architectures and discuss related work.

2. A Motivating Scenario

This section makes our vision of automatic run-time interoperability concrete through a motivating scenario drawn from the domain of electronic commerce. Following that, the Contractual Agent Society metaphor is introduced as a powerful tool for identifying what is needed to achieve this kind of interoperability.

Our market-based economy is characterized by a proliferation of different kinds of markets with widely different rules and regulations. Human buyers and sellers are entering and leaving marketplaces at will, in search for quality merchandise and services, as well as a “good deal”. In each marketplace, they adapt their behavior according to the applicable rules and regulations. These rules and regulations specify the context of interactions within that marketplace.

Let us imagine that we could build systems in which software agents are capable of moving from marketplace to marketplace, dynamically adapting their behavior according to the explicitly stated rules of the marketplace and interacting with other, independently developed agents without the need for manual code modifications. Then, scenarios, such as the following could take place:

An electronic investor agent A is interested in locating and forming a virtual partnership with another investor agent B and a reliable stock intelligence agent C. The idea is that A and B will exclusively hire the services of agent C for a minimum time interval T. A and B will jointly pay the "salary" of agent C. A is prepared to cover up to 2/3 of the salary in exchange for getting priority in the handling of its requests.

Agent A does not know beforehand any agents who might agree to play the roles of B and C. Fortunately, a number of open agent-based marketplaces exist on the Web. The purpose of these marketplaces is to enable agents to locate and form relationships with other agents. Each marketplace provides different facilities and supports very different rules of interaction.

CNET is an electronic marketplace that supports a variant of the contract net protocol [Smi80] in order to help agents locate one another. Marketplace CNET offers the following facilities to its members:

- Matchmaker agent. All new members must register themselves with the matchmaker. To locate another member of the marketplace, members must send a RFB (Request For Bids) message to the matchmaker, describing the requested service. The matchmaker then broadcasts the request to all potentially eligible members. Interested members may then contact the sender directly by sending it a BID message. The matchmaker is free of charge for all members in "good standing" (see below).

- Notary agent. Once an acceptable bid has been received, the two parties can start communicating directly, or else negotiate and form a contract through the notary service. The marketplace charges a fee for the formation of contracts. The benefit of forming contracts is that the marketplace then offers a number of "legal" guarantees. For example, if a contract is unilaterally canceled by one of the parties, the notary service informs the reputation agent. Also, if a contract is breached, the notary informs both the reputation agent and the matchmaker. Members responsible for breaching more than N contracts lose their "good standing" with the marketplace. As a consequence, they are banned from further use of the matchmaker.

- Reputation agent. The reputation agent stores information about the history and status (completed, canceled, breached, etc.) of all contracts formed by members of the marketplace. Members of the marketplace may consult the reputation agent for a fee in order to decide the trustworthiness of other potential partners. The reputation service receives its data directly from the notary agent and is, therefore, completely under the control of the marketplace.

If agent A decides to use marketplace CNET, the following would be a legal sequence of actions that it would have to perform in order to form the desired partnership with agents B and C within the context of that marketplace:

- Get information about the rules and conventions of marketplace CNET
- Register itself with the matchmaker
- Send an RFB message to locate a possible partner B
- Send an RFB message to locate an information agent C
- Receive bids
- Check the reputation of prospective partners B and C

- Negotiate and form a partnership contract with B and C
- Start transacting!

Suppose that agent A is unsuccessful in locating appropriate partners within marketplace CNET. Its other alternative is to visit marketplace AUCT, which is based on the auction model. Marketplace AUCT offers the following facilities[1]:

- Auction broker. Seller agents register with the auction broker. Buyer agents bid for hiring the services of available sellers using the English auction protocol. Sellers specify the terms of contracts they would be willing to accept beforehand. The only item that is up for bidding is the price.

- Notary agent. The notary agent automatically creates a contract between the winner of an auction and the respective seller agent. It charges both parties a fee for the contract and provides legal guarantees identical to those of marketplace CNET.

- Reputation agent. Works in a very similar way to the reputation agent of marketplace CNET.

Agent A would have to behave in a very different way within marketplace AUCT in order to form the desired partnership. The following is one plausible sequence of actions:

- Agent A informs itself of the rules and regulations of marketplace AUCT
- Agent A queries the auction broker about a list of stock intelligence agents which are currently available for exclusive hire
- Agent A uses the reputation agent to inquire about the trustworthiness of these agents
- Agent A selects a stock intelligence agent and bids for it
- Agent A wins the auction and (by default) forms a contract with C
- Agent A drafts a contract that offers the spare capacity of C for hire and registers it with the auction broker
- Agent B successfully bids for the spare capacity of C
- A new contract is formed between A, B and C

Agent-based systems capable of supporting the above scenario will bring agent-mediated electronic commerce and virtual organizations to a whole new level. However, the scenario presupposes a number of capabilities that current agent-based systems still lack:

- The ability for marketplaces CNET and AUCT to describe to agents A, B and C sufficient information about their rules of interaction, regulations, facilities and “legal guarantees”. The above plain English description of each marketplace provides a good illustration of the richness of the information that may need to be communicated.
- The ability for agents A, B and C to understand this information (possibly negotiating some of its terms) and adapt their behavior accordingly
- The ability of marketplaces CNET and AUCT to act as “legal authorities” capable of enforcing the contracts formed by their members and “punishing” (e.g. through reputation loss and eventual banishment) potential violators.

[1] In the following discussion, we will refer to agents who offer their services for hire as *seller agents* and to agents who are interested in hiring the services of other agents as *buyer agents*.

3. Contractual Agent Societies

The Contractual Agent Society metaphor provides a powerful tool for identifying the computational elements needed to achieve the above vision in a comprehensive way.

The concept of Contractual Agent Societies has been inspired by the work of a number of organizational theorists, economists and interactionist sociologists, who model organizations and social systems after contracts. From a contractual perspective, organizations are seen to be sets of agreements for satisfying diverse interests of self-interested individuals [Cye63, Jen76]. Social order, therefore, emerges out of continual negotiation over the rights and duties of the participants [Str78].

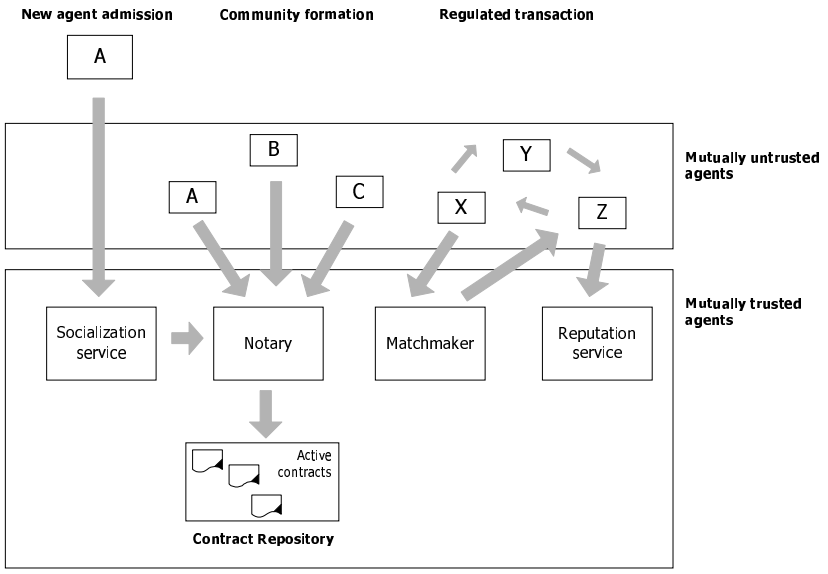

Figure 1: Conceptual architecture of a Contractual Agent Society version of marketplace CNET.

A Contractual Agent Society (CAS) is a multi-agent system where coordinated social activity emerges out of a set of negotiated *social contracts* (social norms) enforced through mechanisms of *social control* (social institutions).

Contractual Agent Societies define a general set of principles for developing heterogeneous multi-agent systems rather than a specific architecture. These principles can be summarized as follows:

- Social contracts specify all elements of a CAS that govern the interaction of a member with the rest of the society. Intuitively they define the rights and obligations of an agent relative to the society. They include beliefs, values, objectives, protocols and policies that two or more agents agree to obey in the context of a social relationship. The rich literature on elements and processes of social systems (see, for example, [Loo60, Gid96]) is a useful source of insights for identifying and representing the various elements that need to be included.
- New agents are admitted to an existing CAS through a process of *socialization*. During the process of socialization, the applicant agent negotiates with existing society members (or their representative) the terms of a social contract that defines the membership of the new agent in the society. As a result of the negotiation, the social contracts of existing members may have to be renegotiated as well.

- Members of a CAS may form additional communities (sub-societies) within the context of a CAS. Communities are formed by negotiating additional social contracts, which define the terms of agent interaction within the community. Such contracts must inherit all policies negotiated between the community members and the CAS.
- A mechanism of social control may be negotiated as part of the social contract. The mechanism defines various classes of exceptions (deviations from the agreed "normal" behavior) and may specify sanctions for some or all of them. In addition, it specifies a mutually acceptable party (typically the system infrastructure) who is given the authority to enact the mechanism and its associated sanctions.

As an example, Figure 1 depicts one possible conceptual architecture of marketplace CNET according to the principles of CAS.

The marketplace itself consists of a set of homogeneous and mutually trusted agents, including the matchmaker, the socialization agent, the notary agent and the reputation agent. In order for (possibly heterogeneous and untrusted) agents A, B and C to join the marketplace, they would first have to negotiate social contracts with the socialization agent. These contracts would specify details such as the available facilities of the marketplace (matchmaker, notary, reputation agents), the protocols for interacting with each of them, and the social control policies of the marketplace (contract cancellation policy, contract breach policy). Social contracts will be validated by the notary agent and stored in its contract repository.

Once "inside" the marketplace, agents A, B and C will make use of the matchmaker in order to locate one another. After they locate one another, they will use the exact same language they used to interact with the socialization service in order to negotiate a new social contract, which will define their partnership (as a community within the marketplace). The negotiation will be mediated by the notary agent, who will also be responsible for storing the contract and resolving potential disputes. The new contract defines the terms of the partnership. Because the new contract is formed within the context of marketplace CNET, it inherits all the policies of the marketplace, such as the sanctions that CNET imposes for contract cancellation and contract breach. Once the new contract has been completed, agents A, B and C can begin transacting. The notary service, on the other hand, will be keeping an "eye" of them and will apply the prescribed sanctions in case the contract is canceled or breached.

The metaphor of Contractual Agent Societies reduces the challenge of interoperability and control of heterogeneous multi-agent systems to that of achieving the following two technical objectives:

- Development of expressive languages for representing the various elements of social contracts
- Development of agents who are capable of negotiating social contracts and adapting their behavior to the terms of those contracts.

In the next section, we propose a pragmatic research agenda for achieving both these objectives.

4. A Research Agenda for developing Contractual Agent Societies

The development of an expressive language and ontology for representing social contracts is central in making the Contractual Agent Society (CAS) vision a reality. Of course, such a language will not be of much use, unless agents can understand it and adapt their behavior accordingly. The development of agent architectures with such capabilities is, therefore, a second important element of the CAS vision.

The following sections describe each of the above components of our vision in more detail.

4.1 Languages and ontologies for representing social contracts

Contractual Agent Societies use social contracts as a vehicle both for communicating shared context among heterogeneous agents, as well as for enabling social control. Intuitively, a contract is a joint commitment of a number of parties to form a social relationship and adapt their current and future behavior in accordance with the contract clauses. A contract consists of one or more contract clauses. A contract clause defines general commitments, which apply to all of the contracting parties, or *role-specific* commitments, which apply only to a subset of the contracting parties, according to their role in the relationship.

Definition: A contract clause is a relation **C**(*x, c, b, s*) where *x* is the set of contracting agents, *c* is the context group, *b* is the body of the contract and *s* is the contract state transition graph.

The above definition of contract is related to the notion of social commitment [Cas95, Jen93, Sin99]. One important distinction, however, is the fact that in CAS, there is no guarantee that the contracting parties will abide by the rules of a contract. Therefore, some mechanism of social control is needed to discourage deviation from the agreed behavior and, if needed, to "punish" violators and restore order. The notion of a contract state transition graph, explained below, provides the connection between a contract clause and social control.

Contracts are first-class objects. One important attribute of a contract is its *state* (created, discharged, canceled, revised, delegated, expired, breached, etc.). From a social perspective, some states are desirable, some are undesirable and some are neutral. The essence of social control is to provide incentives that influence contract participants to maintain a contract in one of the desirable states, while refraining from causing the state of the contract to enter one of the undesirable states.

Incentives are operationalized through *sanctions* (positive and negative) that are enacted whenever a contract enters certain states. Typically, sanctions reward or punish agents that are deemed responsible for the state change. To enable social control, a contract clause needs to specify a *state transition graph* defining the possible states of the clause, the preconditions for each state transition and the associated sanctions. For example, Figure 2 shows the state transition graph of the contract formed by agents A, B and C in the scenario of Section 1.2.1. The graph is represented as a set of state transition rules, such as the one shown in Figure 3.

In the case of contracts with multiple clauses, each clause may have its own state transition graph. In addition, the entire contract has a distinct state transition graph, which is usually a function of the transition graphs of its clauses (e.g. "a contract is breached if any of its clauses are breached").

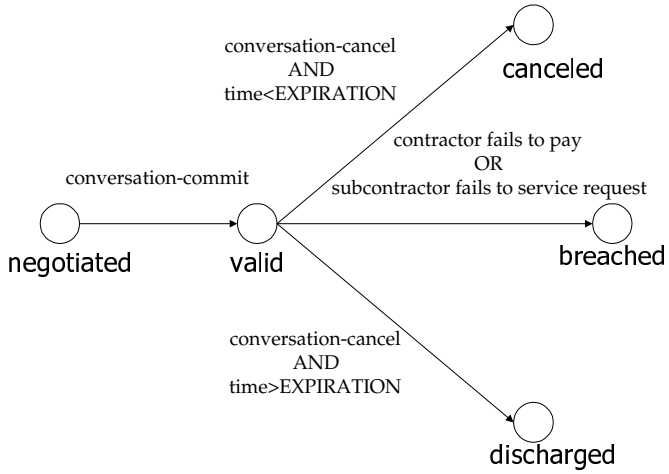

Figure 2: Example contract state transition graph.

```
; The following rule governs the cancellation of the contract. The contract
; is considered cancelled if any of the contracting parties sends a cancel
; message to the notary before the agreed contract expiration time. The
; notary then enacts conversation-cancel, which informs all parties of the
; cancellation. Finally, it reports the party who initiated the cancellation
; to the reputation server.

(transition-rule tt2
     :current-state 'valid
     :next-state 'canceled
     :triggered-by '(cancel :sender (contracting-party ?x)
                            :content (contract THIS-CONTRACT))
     :preconditions (and (< (current-time) CONTRACT-EXPIRATION-TIME)
                         (successfully-completed conversation-cancel))
     :sanctions  ((transmit (contract-canceled :to REPUTATION-AGENT
                            :content ((agent ?x) (contract THIS-CONTRACT))))))
```

Figure 3: Example state transition rule.

Because of limited trust among the contracting parties, the authority for enacting the state transition graph and its associated sanctions is conferred to a mutually trusted *context group*, which is normally distinct from the contracting parties. The context group commits to maintain an authoritative representation of the current state of a contract and apply the associated sanctions in accordance with the specified contract state transition graph.

Finally, the body *b* of a contract can be either:

- a set of contract clauses **C'**(*x'*, *c'*, *b'*, *s'*), where $x' \subseteq x$ and $c' \subseteq c$, or
- a primitive clause

A *primitive clause* represents an element of the social relationship that all contracting parties promise to adhere to. One important objective of our work is to better understand what the various classes of useful social elements are, as well as to develop elegant notations and ontologies for representing them. Once again, we are finding that the work of sociologists on identifying the core elements and processes of

social systems [Loo60] can provide useful insights. Some examples of "elements of social relationships" that would be needed to describe various aspects of the social contract of agents A, B and C include:

- Beliefs. Factual agreements that contracting parties commit to add to their belief repository. For example:

```
(beliefs
    (CONTRACT-EXPIRATION-TIME 100000)
    (SUBCONTRACTOR-SALARY 100)
    (SALARY-FREQUENCY 100)
    (SALARY-PAID-BY-A 70)
    (SALARY-PAID-BY-B 30))
```

- Objectives. Outcomes that all contracting parties agree to achieve or maintain. For example: "the subcontractor will respond to any message sent by A or B within time interval t".

- Social (organizational) values. Jointly held statements of what is considered important in this relationship. This can be formulated by assigning utility values to the various contract objectives or simply by specifying a partial ordering indicating preferences. For example, the statement "requests by A will always be given higher priority than requests by B" can be expressed by a preference relationship between two previously defined objectives:

```
(PREFERRED (RESPOND-ON-TIME A) (RESPOND-ON-TIME B))
```

- Conversation protocols (norms). Descriptions of legal message types and sequences for specific conversations defined by the contract.

- Policies. Other restrictions on behavior, especially restrictions that refer to other social elements. For example: "all contracts formed within marketplace CNET must inherit the standard state transition graph (i.e. the standard policies regarding contract breach and contract cancellation) supported by the marketplace"

4.2 Principles and architectures for Contractual Agents

Contractual Agent Societies (CAS) will allow agents with different internal architectures to interoperate through the negotiation of social contracts. Although the specific internal architecture of CAS agents is not defined, in order for two or more agents to participate in this scheme, they must:

- support a mutually compatible protocol for negotiating social contracts
- be capable of adapting their behavior in order to obey the terms of the agreed social contract

The above two capabilities constitute the minimum "cost of admission" for agents in CAS. One important objective of our work is to propose agent architectures capable of satisfying the above requirements and explore how such architectures extend or otherwise relate to a number of established agent architectures, such as BDI [Bra88, Geo87]. To increase the practical impact of our ideas, we would also like to explore how existing agents can be extended with capabilities that enable them to participate in CAS.

In the general case, creating an agent, which is capable of adapting its behavior according to any possible term of any social contract seems to be a difficult problem. Nevertheless, we believe that we can identify useful classes of agents with restricted adaptation and negotiation capabilities and provide specifications and toolkits for building them.

At the limit, any agent can be turned into a CAS agent by manually constructing a non-negotiable social contract that describes the agent's hard-coded interaction capabilities and by adding an interface through which the agent communicates that contract to other CAS agents. Although other CAS agents cannot negotiate any term of the contract (they have to either accept it as is or reject it) even this simple scheme may allow a rigid legacy agent to participate in a contractual society if the other participants are flexible enough.

It is our hypothesis that between the "completely rigid" agents, such as the one described above, and "completely flexible" agents, lie several useful levels of flexibility for which relatively simple implementations are possible. Agent flexibility can be characterized in terms of (a) the types of social contracts that the agent is capable of supporting and (b) the degree to which the agent is capable of negotiating the terms of these contracts.

We intend to explore the space of agent flexibility, identify interesting regions and provide both architectures for building such agents, as well as guidelines for extending "legacy" agents into CAS agents of the specified flexibility level.

5. Current Status

We have developed a prototype open agent marketplace that embodies the principles of CAS. Our marketplace enables independently developed agents to meet and transact using the contract net protocol. Our emphasis in this initial experiment was in developing the social control aspects of the architecture. More specifically, we have assumed that the interaction protocol itself was fixed, while agents could negotiate the extent to which the marketplace will exert social control.

When an agent joins our marketplace, it must register with a *registrar* responsible for assigning it a *sentinel* that will mediate all of the agents' further interactions with other agents in the system. The agents so 'wrapped' can include problem solving agents as well as components such as matchmakers that support the protocols they enact.

Sentinels are the central element in this approach. They can be viewed as "commitment monitors" whose role is to observe and influence agent behavior as necessary to ensure the robust functioning of the system as a whole. Each sentinel acts as an interpreter for the state transition graphs of all commitments that its associated agents are currently engaged in. As was discussed in Section 4.1, the state transition graph of a social contract describes the characteristic exceptions and associated handlers for the protocol(s) enacted by the agents in that MAS. Sentinels monitor message traffic to develop a model of the commitments their agent(s) are involved in, use the appropriate anticipation and/or detection handlers to uncover when these commitments are violated, diagnose the underlying causes to identify the appropriate avoidance and/or resolution handlers, and enact these handlers to help re-establish the violated commitments, or at least minimize the impact of them having been violated. Ancillary services such as the contract notary and reputation server keep track of global state information such as commitment structures and reliability statistics.

Whenever agents form new contracts, they are capable of negotiating to what extent sentinels will monitor their activities, as well as the range of exception handlers that sentinels will enact in order to detect and resolve deviations from optimal behavior.

Let us consider, for example, how this approach can handle the undesirable effects of non-deterministic agent death (because of bugs or infrastructure issues) in CNET. Sentinels can use the handlers described above to detect and resolve this exception as follows:

- Whenever a contractor sends an award message to a subcontractor, the subcontractor's sentinel makes a note of this commitment and ensures (by periodic polling) that the subcontractor is still functioning. The contract notary is also informed about the commitment between the two agents.
- If the subcontractor appears to have died, its sentinel then instructs the matchmaker to remove the dead subcontractor from its database, and directs the contractor to re-start the bidding process for the task previously allocated to the deceased agent. The sentinel also queries the contract notary to see if the dead subcontractor had awarded any subtasks to other CNET agents; if so, these agents are instructed to cancel these 'orphaned' tasks. Finally, the sentinel informs the reputation server of the subcontractors' death in order to update the reliability statistics for that agent in case it should re-join the MAS at some later time.

We have implemented and empirically evaluated a simplified subset of these services applied to the agent death exception for CNET. Our results have shown that the sentinel architecture substantially out-performed the widely-used 'survivalist' exception handling technique (timeout and retry), with comparable agent complexity. In one experimental condition, for example, the sentinel-based social control reduced average task completion times, when agent death occurs, by a factor of nearly four [Del00].

Our next round of experiments will take this paradigm one step further, supporting agents who, in addition to the social control mechanisms present in the system, are also able to negotiate the details of the core interaction protocol itself.

6. Related Work

The CAS metaphor views multi-agent systems as societies of heterogeneous, self-interested participants, where coordinated behavior emerges through a nexus of social contracts, enforced by a system of social control. This viewpoint is influenced by the work of organizational theorists and economists, who employ a similar contractual perspective in order to explain human organizations and firms [Cye63, Jen76]. Similar ideas have been proposed by interactionist sociologists, notably Ansell Strauss [Str78]. Strauss's "negotiated order" approach shows that many organizations that seem to be stable, functionally ordered systems are, in fact, products of continual negotiation over the rights and duties of the participants.

Several researchers have addressed the need for introducing social concepts in the design of multi-agent systems. Gasser [Gas91] describes some of the sociological issues underlying multi-agent systems. The concept of social commitment has been studied by a number of researchers, notably Castelfranchi [Cas95] and Singh [Sin99]. Our notion of social contract relates to their work, in that it defines a contract as a set of social commitments, which involve a witness or context group. We extend the definition of Singh in that social contracts specify a number of different elements of social interaction (beliefs, goals, actions) as opposed to just goals. In fact, identifying and developing representations and ontologies of the various elements of shared interaction context is an important objective of our research. Furthermore, our definition defines the role of the context group more precisely and relates it to contract enforcement. Jennings [Jen93] has introduced the notion of convention, as ways of reasoning about commitments. The state transition graph associated with social contracts extends Jennings' notion of convention and introduces the notion of sanctions. Whereas Jennings assumes that agents will follow conventions, no such guarantee is assumed by our approach. In contrast, social contracts introduce the notion of social control, defined as a contract, on behalf of the context group, to enact the state transition graph and its

associated sanctions. A final, and important, distinction of the nature of our work is that our objective is not only to propose theoretical definitions of contracts and commitments, but also to develop practical languages and ontologies for representing their various elements.

Languages for expressing contracts and commitments are not very useful, unless agents are capable of understanding them and adapting their behavior accordingly. This requires the development of theories and architectures of social or *normative* agents. Early work in normative agents has been of an experimental nature and for the purpose of social simulation [Car94]. In these types of systems, norms are built-in constraints in the agent's architecture. Shoham and Tennenholtz [Sho95] propose the idea of computational social laws for the purpose of reducing agent coordination or transaction costs. However, in their work, these laws are "computed" off-line and built into the agent architecture.

To this date relatively little work has been done in the direction of normative agents, which are capable of explicit communication of their norms and corresponding adaptation of their behavior [Con99]. Krogh [Kro96] argues for the necessity of agents with normative positions in open architectures, such as the Internet. Castelfranchi et. al. [Cas99] define deliberative normative agents as agents that are able to behave deliberately on the basis of explicitly represented norms. However, once again, whereas [Cas99] only presents conceptual architectures for building such agents, our intention is to develop operational prototypes. One additional novel aspect of our proposal is our definition of the space of agent flexibility (see section 1.3.2) and our intention to identify useful regions and agent architectures within that space.

Barbuceanu [Bar99] developed systems where agents coordinate by exchanging information about obliged and forbidden behavior at run-time. He has developed languages for representing agent obligations and conversation policies, as well as operational prototype systems. Our work is similar, in spirit, to his approach and similarly focuses on developing languages and operational prototype systems. Our notion of contract generalizes his notion of obligations, interdictions and conversation policies. In addition, our work emphasizes the specification and development of practical mechanisms for contract enforcement in the face of limited control over the system participants, whereas [Bar99] simply mentions the existence of "costs" associated with violations and does not specify any mechanism for enforcement.

References

[Bar99] Barbuceanu M., Gray T., Mankovski S. The Role of Obligations in Multiagent Coordination. Applied Artificial Intelligence 13 (1-2), Jan.-March 1999, pp. 11-38.

[Bra99] Bradshaw J., Greaves M. chairs. Workshop on Specifying and Implementing Conversation Policies. Third International Conference on Autonomous Agents, Seattle, WA, May 1-5, 1999.

[Bra96] Brafman R.I., Tennenholtz M. On Partially Controlled Multi-Agent Systems. Journal of Artificial Intelligence Research 4, 1996, pp. 477-507

[Bra88] Bratman M.E., Israel D.J., Pollack M.E. Plans and resource-bounded practical reasoning. Computational Intelligence 4, 1988, pp. 349-355.

[Car94] Carley K.M., Prietula M.J., editors. Computational Organization Theory. Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.

[Cas95] Castelfranchi C. Commitments: From individual intentions to groups and organizations. Proceedings of the First International Conference on Multi-Agent Systems (ICMAS), San Francisco, California, June 1995, pp. 41-48.

[Cas99] Castelfranchi C., Dignum F., Jonker C.M., Treur J. Deliberative Normative Agents: Principles and Architecture. Proceedings of the Sixth International Workshop on Agent Theories, Architectures, and Languages (ATAL-99), Orlando, FL, July 15-17, 1999.

[Con99] Conte R., Falcone R., Sartor G. Agents and Norms: How to fill the gap? Artificial Intelligence and Law 7 (1), March 1999, pp.1-15.

[Cye63] Cyert R.M., March J.G. A Behavioral Theory of the Firm. Prentice-Hall, Englewood Cliffs, NJ, 1963.

[Del00] Dellarocas C., Klein M. An Experimental Evaluation of Domain-Independent Fault Handling Services in Open Multi-Agent Systems. Proceedings of the International Conference on Multi-Agent Systems (ICMAS-2000), July 2000, Boston, MA.

[Fis96] Fischer K., Muller J.P., Heimig I., Scheer A.-W. Intelligent agents in virtual enterprises. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM'96), Blackpool, UK, pp.205-23.

[Gas91] Gasser L. Social conceptions of knowledge and action: DAI foundations and open system semantics. Artificial Intelligence 43 (1), 1991, pp. 107-138.

[Geo87] Georgeff M.P., Lansky A.L. Reactive reasoning and planning. Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87), Seattle, WA, 1987, pp. 677-682.

[Gid96] Giddens A. Introduction to Sociology. W.W. Norton & Co., 1996.

[Jen93] Jennings N.R. Commitments and conventions: The foundation of coordination in multi-agent systems. Knowledge Engineering Review 2 (3), 1993, pp. 223-250.

[Jen98] Jennings N.R., Sycara K. and Wooldridge M. A Roadmap of Agent Research and Development, Autonomous Agents and Multi-Agent Systems 1 (1), 1998, pp. 7-38.

[Jen76] Jensen M.C., Meckling W.H. Theory of the firm: Managerial behavior, agency costs and ownership structure. Journal of Financial Economics 3, 1976, pp. 305-360.

[Kle99] Klein M., Dellarocas, C. Exception Handling in Agent Systems Proceedings of the Third International Conference on Autonomous Agents, Seattle, WA, May 1999, pp. 62-68.

[Kro96] Krogh K. The Rights of Agents. In Wooldridge M., Müller J. and Tambe M. (eds): Intelligent Agents II - Proceedings of the 1995 Workshop on Agent Theories, Architectures and Languages (ATAL-95), Lecture Notes in Comp. Science, Springer-Verlag, 1996, pp. 1-16.

[Loo60] Loomis C.P. Social Systems: Essays on their Persistence and Change. D. Van Nostrand Company, Inc., 1960.

[Sho95] Shoham Y., Tennenholtz M. Social Laws for Artificial Agent Societies: Off-line Design, Artificial Intelligence 73 (1-2), February 1995, pp. 231-252.

[Sin99] Singh M. An Ontology for Commitments in Multiagent Systems: Toward a Unification of Normative Concepts. Artificial Intelligence and Law 7 (1), 1999, pp. 97-113.

[Smi80] Smith R.G. The contract net protocol: high level communication and control in a distributed problem solver. IEEE Transactions on Computers 29 (12), December 1980, pp.1104-1113.

[Str78] Strauss A. Negotiations: varieties, contexts, processes, and social order. Jossey-Bass, San Francisco, Ca., 1978.

[Tsv96] Tsvetovatyy M.B., Gini M. Toward a virtual marketplace: architectures and strategies. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM'96), Blackpool, UK, pp. 597-613.

[Tsv97] Tsvetovatyy M.B., Gini M., Mobasher B., Wieckowski Z. MAGMA: An agent-based virtual marketplace for electronic commerce. Applied Artificial Intelligence 11 (6), 1997, pp. 501-524.