A BDI Assignment Protocol with New Cooperative-Concession Strategies

Kiam Tian Seow, Kwang Mong Sim, Yew Soon Ong and Endang Purwanto Sulaiman

***Abstract*— This paper addresses the collaborative linear assignment problem (CLAP) for a class of allocation applications. CLAP entails using agents to seek a concurrent allocation of one different object for every agent, to optimize a linear sum efficiency function as their (soft) social goal. Anchoring in the standard framework of automated negotiation allows an original BDI negotiation model for CLAP to be conceptually separated into a BDI assignment protocol and an adopted strategy. Facilitated by this conceptual separation, the contributions of this work are: (i) providing a rigorous analysis of the protocol and demonstrating its salient properties, and (ii) formulating new strategies using a novel idea of cooperative concession. Four different strategies for a negotiation agent and the arbitration agent provide sixteen arbitration-negotiation combinations running with the protocol, and these are empirically assessed for their performance profiles in negotiation speed and solution quality. Important findings, including the stability of the protocol in producing better than good enough global allocations, and the strategic influence of cooperative concessions on performance, are examined. The significance and practicality of the work in relation to existing work are also discussed.**

***Index Terms*— Intelligent Agents, BDI Models, Automated Negotiation, Cooperative Concessions, Reasoning Systems.**

I. Introduction

Central to many real world applications in a non-centralized environment is the fundamental problem of assigning or allocating objects to agents. The object can be a task to assign or a resource to allocate. Perhaps the most basic is the linear (sum) assignment problem (LAP) which deals with the question of how to *concurrently* assign N distinct objects to N distinct agents on a one-to-one basis, with maximizing a summation objective function as the optimal *goal*. LAP manifests itself in a diverse range of interesting applications, either as an allocation problem or a subproblem of co-allocation. Examples include personnel management (*assign tasks to persons*), vehicle transportation (*assign passenger requests to taxis*), manufacturing (*assign jobs to parallel machines*) and telecommunication (*match sending and receiving stations*), for which centralized algorithms have been applied [1].

Our research aims to develop techniques to address distributed versions of these LAP applications, emerged to exploit recent advancement in computer and internet technology that has made it possible to have situated agents collaboratively plan the assignments *by* themselves. This is in contrast to a centralized algorithm planning *for* them [1]. Solved this way, the basic problem is termed a collaborative LAP (CLAP) (Section II). While the centralized approach was acceptable in the past, it limited active involvement of distributed agents in incremental planning or problem solving.

This paper addresses and discusses the solutions of CLAP in the framework of automated negotiation [2], where negotiation is viewed as a process of several agents searching for a solution called an *agreement*. The search process is realized via a negotiation mechanism (or algorithm) implementing a negotiation model comprising of a high level *protocol* and a set of *strategies*. In general, given a protocol specifying the ‘rules of interactions’, different strategies can be designed for individual agents to select their own preferences among available choices at each step. Whereas a protocol is *public* in that it is agreed and followed by all the agents in a negotiation process, the strategies used by individual agents are more *private* in that their full details may be hidden from other agents.

In addressing CLAP as a distributed agent problem, an agent attempting to reach an optimal as-

K.T. Seow is with the Division of Computing Systems, School of Computer Engineering, Nanyang Technological University, Republic of Singapore 639798. Email: asktseow@ntu.edu.sg

K.M. Sim is with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Kowloon, Hong Kong. Email: bsim@comp.hkbu.edu.hk

Y.S. Ong and E.P. Sulaiman are with the Division of Information Systems, School Computer Engineering, Nanyang Technological University, Republic of Singapore 639798. Email: {asysong,Y060016}@ntu.edu.sg

signment solution (or agreement) faces the basic issue of deciding what action to perform. In so doing, each agent needs to reason about its beliefs and preferences as well as its collaborating agents', mediating through interactions among the agents during planning. The goal of LAP becomes the joint social goal of these agents. One agent negotiation mechanism MA^3 [3] realizing this implements a Belief-Desire-Intention (BDI) negotiation model developed for CLAP. Essentially, the mechanism involves agents negotiating under the regulation of an arbitration agent. In seeking the social goal, each agent reasons communicatively and submits its *intention* - an object exchange proposal with another agent that will increment the social gain if executed - for arbitration in a finite number of negotiation rounds.

Anchored in the framework of automated negotiation, in the original MA^3, every negotiation agent as well as the arbitration agent adopts a simple greedy strategy 'embedded' in a BDI assignment protocol (Section III-A). The strategy is *greedy* because it asserts that only an object exchange proposal associated with the *highest* social gain is selected in an arbitrary round. Conceptually separating the protocol and the strategy opens up the opportunities for formal analysis of the protocol and development of new strategies for CLAP, along with comparative evaluations of their performance.

The development of new strategies in this paper for determining local intentions and arbitrating them in MA^3 is based on a novel idea called *cooperative concession* which deviates from the usual definition of *concession*. By the conventional idea of concession in automated negotiation, each negotiation agent considers whether to give up (or concede) an object if an exchange is to be agreed upon [2]. However, by cooperative concession, whether an agent should concede is considered by another agent, depending on which object the former is holding that the latter believes it might exchange its current selection for, to increment social gain. So intuitively, the proposed idea of cooperative concession asserts that *'I' consider whether 'You' should give in* and conversely, instead of the usual definition of *'I' consider whether 'I' should give in*.

Following, the main contributions of this paper are: (i) the novel idea of cooperative concession strategies and their formulations for determining local intentions and arbitrating them in BDI assignment negotiation (Section III), (ii) a formal analysis of the BDI assignment protocol that exposes three salient properties, namely solution guarantee, simplicity, and stability in some specific sense (Section IV), and (iii) a detailed study of the empirical performance of all possible arbitration-negotiation combinations of strategies, along with a discussion on the comparative effectiveness of these strategy combinations and the stability of the BDI assignment protocol for CLAP (Section V). Discussions in relation to existing multiagent work examine the significance of the work (Section VI), with concluding remarks in Section VII.

II. The CLAP Framework

The $N \times N$ CLAP framework [3] entails using N agents to negotiate for an efficient concurrent allocation of N different objects, with one different object for every agent. The original framework [3] is defined with *task* agents negotiating for different *resources*; here it is slightly generalized conceptually to *agents negotiating for objects*, but with no change in mathematical formulation. Thus, in this framework, there is a team of agents $\mathcal{A} = \{a_0, a_1, \cdots, a_{N-1}\}$ of size $N \geq 2$, and a set of different objects $\mathcal{O} = \{r_0, r_1, \cdots, r_{N-1}\}$ of size N. Initially, agent $a \in \mathcal{A}$ only has knowledge of the A-QoS (*application quality-of-service*) it can offer for each object, defined by $d[a, r]$ for all $r \in \mathcal{O}$. Formally, the objective of $N \times N$ CLAP is to find the particular (total) assignment

$$\begin{aligned} \Pi : \mathcal{A} \rightarrow \mathcal{O} \text{ such that for } a_i, a_j \in \mathcal{A}, \\ i \neq j \text{ implies } \ \Pi(a_i) \neq \Pi(a_j) \end{aligned} \tag{1}$$

a one-to-one mapping of agents to objects that (approximately) maximizes the total A-QoS S_{tot},

$$S_{tot} = \sum_{i=0}^{|\mathcal{A}|-1} d[a_i, \Pi(a_i)] \tag{2}$$

$\Pi(a) \in \mathcal{O}$ refers to an object selection by agent $a \in \mathcal{A}$ (under an arbitrary permutation of Π); and $\max\{S_{tot}(2)\}$ defines the (ideal) social goal[1] of the agents. An assignment or allocation set (or simply assignment) corresponds to one permutation of Π

[1] Note that this global optimum is not the same as the sum of individual local optima (each being the largest A-QoS value $d[a, r]$ among all objects in $\mathcal{O}$ for an agent $a \in \mathcal{A}$), unless their corresponding selections form a permutation of Π (1).

(1), and can also be equivalently represented as containing elements of the form $(a, \Pi(a)) \in \mathcal{A} \times \mathcal{O}$.

In attempting to reach an agreement (social goal), the basic issue a negotiation agent $a \in \mathcal{A}$ faces when holding an object selection is deciding what alternative selections to exchange it for, as detailed in Section III.

III. BDI Negotiation Model for CLAP

This section presents an overview of the BDI assignment protocol and the strategies that constitute the proposed negotiation model for $N \times N$ CLAP.

A. The BDI Assignment Protocol

The proposed *protocol* divides the reasoning process into negotiation rounds, and in each round, performs negotiatory means-end reasoning, where the *end* is to increase the social value, i.e., the total allocated A-QoS (2), using the *means* of object exchange between two negotiation agents. In each round, each agent locally accesses and directly acts only on its own row of A-QoS data, and determines its *belief* set - the information or evidence that indicates all the possible options - the alternative objects - a negotiation agent can exchange its current object selection for to achieve its end. Every agent then begins *negotiating* by communicating with one another to acquire A-QoS data from any agent whose current object selection is in the agent's belief set. In collaborating, any such agents will respond with the required A-QoS values, using which the agent would deliberate to determine its own *desire* set - the means of exchanging its current object selection for options (that survive the deliberation) with the respective agents (currently holding on to these options). Each desire is an exchange means for the agent to achieve its end. As a final step in a negotiation round, the agent will adopt a *strategy* that must[2] select a (local) desire as its *intention*, which it would then use as the basis for an object exchange proposal. All the agents' object exchange intentions would undergo arbitration to decide which two agents to proceed with the object exchange, before negotiation is concluded, and the next round begins. The negotiation process terminates when simultaneously, all negotiating agents have no (more) intention to exchange objects.

[2]This rule is necessary to ensure that any strategy will always select an available exchange means that exists.

B. Strategies for Determining Intentions

Negotiation strategies: Three strategies for a negotiation agent are possible with the BDI assignment protocol in a negotiation round:

1) In the N-Greedy strategy, the agent will select the best (local) desire - the one that offers a net exchange gain that is the highest from the agent's perspective as its intention.
2) In the N-MinCon strategy, the agent will select a (local) desire that offers a net exchange gain along with its exchange partner conceding the least from the agent's perspective as its intention.
3) In the N-MaxCon strategy, the agent will select a (local) desire that offers a net exchange gain along with its exchange partner conceding the most from the agent's perspective as its intention.

Arbitration strategies: Similarly, three strategies for arbitration are possible in a negotiation round. The following describes the strategies for arbitration done through a dedicated (arbitration) agent:

1) In the A-Greedy strategy, the (arbitration) agent will select an intention with the *highest* exchange gain, among all intentions gathered, for object exchange.
2) In the A-MinCon strategy, the agent will select an intention with the exchange partner of the proposing agent conceding *the least*, among all intentions gathered, for object exchange.
3) In the A-MaxCon strategy, the agent will select an intention with the exchange partner of the proposing agent conceding *the most*, among all intentions gathered, for object exchange.

Relative to Greedy, MaxCon tends to encourage more A-QoS information requests in subsequent rounds, whereas MinCon tends to encourage less. An important motivation of this research is to study, with Greedy as the base case, how different arbitration-negotiation combinations including these cooperative-concession strategies impact performance in terms of negotiation speed and solution quality.

Random strategy: In the simulation study (Section V), we also introduce a *random* 'strategy' for the arbitration and negotiation agents; an agent is said to be adopting a random strategy if it nondeterministically selects one of the three proposed before negotiation begins. So, a total of sixteen *arbitration-negotiation* strategy combinations are investigated.

C. Formalization of BDI Concepts & Cooperative Concession

To formally ground and combine the BDI concepts and strategies of cooperative concession for CLAP, the following CLAP-specific data structures are formally defined, in such a way that they can be naturally interpreted as a negotiation agent's beliefs, desires with concession information, and intentions computed in an arbitrary round of negotiation. In these definitions, the current object selections of all agents refer to those made under an arbitrary permutation of Π (1).

Definition 1 (Belief Set B_i): Given that an agent $a_i \in \mathcal{A}$'s current object selection is $r^i \in \mathcal{O}$. Then its (current) belief set B_i is given by

$$\mathsf{B}_i = \{r \in \mathcal{O} \mid d[a_i, r] > d[a_i, r^i]\} \tag{3}$$

If $\mathsf{B}_i \neq \emptyset$, this means that agent $a_i \in \mathcal{A}$ has at least one alternative object selection $r \in \mathsf{B}_i$ that may lead to increase in total A-QoS (2) (when made in exchange with an agent whose current selection is $r \in \mathcal{O}$).

Definition 2 (Desire Set D_i and Concession C_{ij}): Given that an agent $a_i \in \mathcal{A}$'s current object selection is $r^i \in \mathcal{O}$ and its belief set is B_i, $\mathsf{B}_i \neq \emptyset$. An arbitrary agent $a_j \in \mathcal{A}$ whose current object selection is $r^j \in \mathcal{O}$ is said to accept agent $a_i \in \mathcal{A}$'s beliefs B_i if $r^j \in \mathsf{B}_i$. To generate the desired exchange options or desires D_i, agent $a_i \in \mathcal{A}$ broadcasts its beliefs B_i and current selection $r^i \in \mathcal{O}$, and an arbitrary agent $a_j \in \mathcal{A}$ who accepts the beliefs would respond with a pair of A-QoS values $d[a_j, r^j]$ and $d[a_j, r^i]$, so that for each of the $|\mathsf{B}_i|$ responses received, the corresponding object exchange option $[(a_i, r^j), (a_j, r^i), \rho, C_{ij}] \in \mathsf{D}_i$ (i.e., is agent $a_i \in \mathcal{A}$'s desire) if $\rho > 0$, where ρ is defined by

$$\rho = -d[a_i, r^i] + d[a_i, r^j] - C_{ij} \tag{4}$$

with

$$C_{ij} = (d[a_j, r^j] - d[a_j, r^i]) \tag{5}$$

defining the *cooperative concession* of agent $a_j \in \mathcal{A}$ for agent $a_i \in \mathcal{A}$.

If $\rho > 0$, it means that there is a net exchange gain if agent $a_i \in \mathcal{A}$ gives up its current selection $r^i \in \mathcal{O}$ and selects $r^j \in \mathcal{O}$, and in exchange, agent $a_j \in \mathcal{A}$ gives up its current selection $r^j \in \mathcal{O}$ and selects $r^i \in \mathcal{O}$. Thus, any desire $d \in \mathsf{D}_i$, when carried out, will definitely lead to an increase in total A-QoS without violating Π (1). Quite naturally, it provides the motivation for agent $a_i \in \mathcal{A}$ to want to exchange its current object selection.

If the cooperative concession C_{ij} agent $a_i \in \mathcal{A}$ receives from an agent $a_j \in \mathcal{A}$ is greater than zero, this means that agent $a_j \in \mathcal{A}$ will concede its object for agent a_i's with a local A-QoS decrease if the object exchange with agent $a_i \in \mathcal{A}$ takes place. If $C_{ij} < 0$, it means that the agents will mutually benefit with a local A-QoS increase should the exchange occur.

Definition 3: [Intention I_i] Given that an agent $a_i \in \mathcal{A}$'s desire set is D_i, $\mathsf{D}_i \neq \emptyset$. Then, agent $a_i \in \mathcal{A}$'s intention I_i is determined according to the strategy adopted:

1) N-Greedy strategy

$$\begin{aligned} I_i = [(a_i, r^j), (a_j, r^i), \rho, -] \in \mathsf{D}_i,\ \text{for which} \\ \rho = \max\{\rho' \mid [-, -, \rho', -] \in \mathsf{D}_i\ \} \end{aligned} \tag{6}$$

2) N-MinCon strategy

$$\begin{aligned} I_i = [(a_i, r^j), (a_j, r^i), -, C_{ij}] \in \mathsf{D}_i,\ \text{for which} \\ C_{ij} = \min\{C' \mid [-, -, -, C'] \in \mathsf{D}_i\ \} \end{aligned} \tag{7}$$

3) N-MaxCon strategy

$$\begin{aligned} I_i = [(a_i, r^j), (a_j, r^i), -, C_{ij}] \in \mathsf{D}_i,\ \text{for which} \\ C_{ij} = \max\{C' \mid [-, -, -, C'] \in \mathsf{D}_i\ \} \end{aligned} \tag{8}$$

Agent $a_i \in \mathcal{A}$'s decisive stance or intention is I_i or it is said to have no intention if either $\mathsf{B}_i = \emptyset$ or $\mathsf{D}_i = \emptyset$, in which case $I_i = nil$, where $nil = [-, -, 0, -]$.

Finally, in the role of arbitration, an intention $I = [-, -, \rho, -]$, $\rho > 0$, is similarly selected in accordance to an adopted strategy, but over all the agents' intentions (or the lack thereof communicated as a nil intention) $I_i \in \mathsf{I}$ gathered.

The negotiation process will terminate following a negotiation round when all agents have no (more) intention to exchange objects and so submit nil intentions, discovered through arbitration.

With the above formalization, a distributed agent algorithm that realizes the BDI negotiation model (consisting of the BDI assignment protocol and the set of strategies proposed) is presented in Section III-D. This algorithm is referred to as a Multi-Agent Assignment Algorithm (MA^3), and handles the simple role of arbitration through a dedicated agent.

D. Distributed Agent Algorithm MA^3

MA^3 assumes that $|\mathcal{A}| = |\mathcal{O}| = N$, and consists of an arbitration agent (or arbiter) and a team of agents, $a \in \mathcal{A}$. Each negotiation agent $a \in \mathcal{A}$ has initial A-QoS (local) knowledge, i.e., $d[a, r]$ for each object $r \in \mathcal{O}$. Agent $a \in \mathcal{A}$ initially selects an object $r \in \mathcal{O}$ according to (a permutation of) $\Pi : \mathcal{A} \to \mathcal{O}$ (1). The arbiter then initiates negotiation.

The generic BDI reasoning mechanism of a negotiation agent and the simple role of the arbitration agent in an arbitrary round of collaborative negotiation can now be described as follows:

MA^3 : Collaborative Negotiation Agent

---

1) If agent believes that there are alternative object selections which may lead to increase in total A-QoS, it would, based on its (local) beliefs, generate the desired exchange options or desires, from which an option in accordance to its adopted strategy will be chosen as its intention.
2) Agent submits its intention (or the lack thereof) to the arbitration agent.
3) Concurrent with Steps 1 and 2, it responds to any requesting agent whose beliefs it accepts, by sending to the requesting agent the A-QoS values as required for computing the requesting agent's desire.
4) Agent changes its object selection (and then acknowledges it), proceeds to next round of negotiation or quit, as decided by the arbitration agent.

MA^3 : Arbitration Agent

---

1) Agent first receives the intentions (or the lack thereof) of all the negotiation agents.
2) If agent sees that all agents have no intention to exchange, it terminates the negotiation by telling all agents to quit.
3) Otherwise, it
   a) selects an intention in accordance to its adopted strategy and instructs the two agents concerned to proceed with the object exchange.
   b) receives acknowledgement of object exchange made as instructed (from the two agents concerned), before telling all agents to proceed to next round of negotiation.

IV. Protocol Analysis

Formally, the BDI assignment protocol induces the permutations of Π (1) for an $N \times N$ CLAP into a negotiation space formalized as an assignment reachability graph $\mathcal{G}$. Essentially $\mathcal{G}$ defines the complete space of possible sequential execution of desires selected as intentions in a negotiation process.

A. Negotiation Space: An Assignment Reachability Graph

For a set of agents $\mathcal{A}$ and a set of objects $\mathcal{O}$, for which $|\mathcal{A}| = |\mathcal{O}| = N \geq 2$, let

$$\mathcal{G} \stackrel{\text{def}}{=} (V, D, \delta, V_o) \tag{9}$$

represent an assignment reachability graph (ARG) in which:

1) V denotes a (nonempty) finite set of states uniquely characterizing the permutations of Π (1), and we write $\Pi(a)|_v$ to denote the object selection of agent $a \in \mathcal{A}$ in state $v \in V$. $|V| = N!$. The total A-QoS (2) in a state $v \in V$ (i.e., a permutation of Π) is denoted by $|v|$ and given by
$$|v| = \sum_{i=0}^{(N-1)} d[a_i, \Pi(a_i)|_v].$$
2) $D \subseteq V \times V$ denotes a finite set of desires.
3) $\delta : D \times V \to V$ is a state transition function (due to object exchange between two arbitrary agents $a_i, a_j \in \mathcal{A}$), such that $\delta(e_{ij}, v) = v' \in V$ iff $\Pi(a_i)|_{v'} = \Pi(a_j)|_v$ and $\Pi(a_j)|_{v'} = \Pi(a_i)|_v$ and the magnitude of $e_{ij} \in D$, $\triangle e_{ij}|_v > 0$, is defined by
$$\begin{aligned}\triangle e_{ij}|_v \;=\; & \{-d[a_i, \Pi(a_i)|_v] + d[a_i, \Pi(a_j)|_v]\} + \\ & \{-d[a_j, \Pi(a_j)|_v] + d[a_j, \Pi(a_i)|_v]\} > 0.\end{aligned}$$
We can interpret $\triangle e_{ij}|_v$ as the increase in total A-QoS if agent a_i and agent a_j exchange their object selections held in state $v \in V$, i.e., $\Pi(a_i)|_v$ and $\Pi(a_j)|_v$, respectively.
4) $V_o \subseteq V$ denotes a finite set of terminal states such that for $v_o \in V_o$, $\delta(e, v_o)$ is not defined for any $e \in D$.

Let D^* contain all possible finite sequences, or strings, over D, plus the null string ε. Then, definition of δ can be extended to D^* as follows:

$$\delta(\varepsilon, v) = v,$$

$$(\forall e \in D)(\forall w \in D^*), \delta(we, v) = \delta(e, \delta(w, v)).$$

For a string $s \in D^*$ that is defined at an arbitrary $v \in V$ (i.e., $\delta(s, v) \in V$) and hence is called an *admissible* string, $|s|$ denotes the length of the string, i.e., the number of elements of set D, assumed nonempty, in string s. $|s| = 0$ if $s = \varepsilon$.

Finally, to define the beliefs B_i, desires D_i and intention I_i of an agent $a_i \in \mathcal{A}$ in a state $v \in V$, we write: $\mathsf{B}_i|_v$, $\mathsf{D}_i|_v$ and $I_i|_v$, respectively; if agent $a_i \in \mathcal{A}$ has no intention in a state $v \in V$, we write $I_i|_v = nil$. We also call a state $v \in V$ an agreement, and $|v|$, an agreement value.

This leads us to formally stating the basic logical relationship among the beliefs, desires and intention of an agent $a_i \in \mathcal{A}$ under an agreement $v \in V$, in accordance to the BDI assignment protocol.

B. Characteristic Axioms of BDI Assignment Protocol

Axiom 1: $\mathsf{B}_i|_v = \emptyset \Longrightarrow \mathsf{D}_i|_v = \emptyset$.

Axiom 2: $(\forall a_i \in \mathcal{A}, \mathsf{D}_i|_v = \emptyset) \Longleftrightarrow v \in V_o$.

Axiom 3: $\mathsf{D}_i|_v = \emptyset \Longleftrightarrow I_i|_v = nil$.

Axiom 1[3] states that in an arbitrary state $v \in V$, if an agent $a_i \in \mathcal{A}$ has no belief, it has no desire. Axiom 2 states that a state $v \in V$ is in V_o provided all the agents have no desire. Axiom 3 states that in an arbitrary state $v \in V$, an agent $a_i \in \mathcal{A}$ has no desire provided it has no intention.

C. Properties of ARG

Below, we present some basic properties of an ARG $\mathcal{G}$ (9). These properties are needed to establish the key properties of the BDI assignment protocol in Section IV-D.

Property 1: If $e \in D$ and $\delta(e, v) = v' \in V$, then $|v'| > |v|$.

Proof: See proof of [3, Property 1,p. 260]. ■

Property 2: ARG $\mathcal{G}$ is acyclic (i.e., $\forall v \in V$, there is no $s \in D^* - \{\varepsilon\}$ such that $\delta(s, v) = v$).

Proof: See proof of [3, Property 2, p.260]. ■

Property 3: $V_o \neq \emptyset$ (i.e., given an arbitrary $v \in V$, $\exists w \in D^* : \delta(w, v) \in V_o$).

Proof: See proof of [3, Property 3, p. 260]. ■

Property 4: $\forall a_i \in \mathcal{A}, I_i|_v = nil$ iff $v \in V_o$.

Proof:

$$\begin{aligned} v \in V_o &\Longleftrightarrow \forall a_i \in \mathcal{A}, \mathsf{D}_i|_v = \emptyset \\ &\qquad \{\text{by Axiom 2}\} \\ &\Longleftrightarrow \forall a_i \in \mathcal{A}, I_i|_v = nil \\ &\qquad \{\text{by Axiom 3}\}. \end{aligned}$$

Hence the result. ■

Property 5: If $v \in V$ is an optimal agreement, then $v \in V_o$.

Proof: We prove by contrapositive reasoning.

$$\begin{aligned} v \notin V_o &\Longrightarrow \textit{not } (\forall a_i \in \mathcal{A}, I_i|_v = nil) \\ &\qquad \{\text{by Property 4}\} \\ &\Longrightarrow \exists a_i \in \mathcal{A} : I_i|_v = [-, -, \rho, -] \neq nil \\ &\qquad \{\text{ by propositional reasoning }\} \\ &\Longrightarrow \exists v' \in V : v' = \delta(e, v) \\ &\qquad \{\text{ by Definition 3 of intention as a desire, which is a transition } I_i|_v = e \in D \text{ of ARG } \mathcal{G} \text{ (9) }\} \\ &\Longrightarrow \exists v' \in V : |v'| > |v| \\ &\qquad \{\text{ by Property 1 }\} \\ &\Longrightarrow v \in V \text{ is not optimal.} \end{aligned}$$

Hence the result. ■

The proofs of the next two properties rely on two elementary results (Lemmas 1 and 2) for a directed graph, denoted G, and some terminology, namely, a bipartite and a colorable graph, and its chromatic number $\chi(G)$; these are presented in Appendix I.

Property 6: The following statements are true and equivalent: 1) $\chi(\mathcal{G}) = 2$ and 2) $\mathcal{G}$ is a bipartite graph.

Proof: By Property 2, $\mathcal{G}$ is acyclic, implying it has no cycles and hence no cycles of odd length. Thus, by Lemma 1, $\mathcal{G}$ is 2-colorable; therefore $\chi(\mathcal{G}) = 2$. Since by Lemma 2, $\chi(\mathcal{G}) = 2$ iff $\mathcal{G}$ is a bipartite graph, it follows that $\mathcal{G}$ is bipartite. Hence the result. ■

By Property 6, $\mathcal{G}$ is a bipartite graph (2-colorable) implying V can be partitioned into (disjoint) sets of V-0 and V-1, i.e.,

$$V = V\text{-}0 \cup V\text{-}1 \text{ such that } V\text{-}0 \cap V\text{-}1 = \emptyset \tag{10}$$

Together with acyclicity of $\mathcal{G}$ by Property 2, we can further partition V-0 and V-1 individually, with

$$V\text{-}0 = \bigcup_{i=0}^{a} V\text{-}0_{2i} \text{ and } V\text{-}1 = \bigcup_{i=0}^{b} V\text{-}1_{2i+1} \tag{11}$$

for some finite integers a and b, such that with $z \in \{0, 1\}$[4], the following two conditions are true:

[3] Axiom 1 is not used in any proof in this paper, but is presented here for completeness' sake.

[4] Think of z as a binary variable, with $\overline{z}$ denoting *not* z.

1) For $x \neq y, V\text{-}z_x \cap V\text{-}z_y = \emptyset$;
2) $(v' = \delta(e, v)$ and $v \in V\text{-}z_x) \implies \exists! y > x : v' \in V\text{-}\overline{z}_y$.

Property 7: For an arbitrary $v \in V$, $s \in D^*$, if $\delta(s, v) \in V$, $|s| \leq N(N-1) - 1$.

Proof: If s is a null string ε, $\delta(s, v) = v \in V$; and trivially, $|s| = 0 \leq N(N-1) - 1$, since $N \geq 2$.

By (10) and (11), an arbitrary admissible string $s \in D^* - \{\varepsilon\}$ traversing in $\mathcal{G}$ alternates between states in V-0 and V-1 through a sequence of partition subsets, $V\text{-}0_{0^*} \to V\text{-}1_{1^*} \to V\text{-}0_{2^*} \to V\text{-}1_{3^*} \to \cdots \to V\text{-}z_{|s|^*}$, $z \in \{0, 1\}$ and $k^* > j^*$ for $k = j+1$. In traversing from one such partition subset $V\text{-}z_{x^*}$ to another, one of $\sum_{y=1}^{w_x} \left(\binom{N}{2} - A_{xy} \right)$ possible desired exchanges (under II) from the subset is taken, where $w_x \geq 1$ is the number of states in a $V\text{-}z_{x^*}$ and A_{xy} is the number of non-desired exchanges [out of a possible $\binom{N}{2}$] from a state indexed by y in $V\text{-}z_{x^*}$, $0 \leq A_{xy} \leq \binom{N}{2}$. Following,

$$\sum_{x=0}^{|s|-1} \sum_{y=1}^{w_x} \left(\binom{N}{2} - A_{xy} \right) \leq \left(\binom{N^2}{2} - 2N \binom{N}{2} \right) - \sum_{x=0}^{|s|-1} \sum_{y=1}^{w_x} A_{xy} - \beta \binom{N}{2},$$

where β, $1 \leq \beta \leq |V_o|$, is the number of terminal states not in any subset $V\text{-}z_{x^*}$, $0 \leq x \leq |s| - 1$. Simplifying, we get

$$\sum_{x=0}^{|s|-1} w_x \binom{N}{2} \leq \binom{N^2}{2} - (2N + \beta) \binom{N}{2}.$$

Rewriting the left-hand side, we have

$$|s| \alpha \binom{N}{2} \leq \binom{N^2}{2} - (2N + \beta) \binom{N}{2},$$

where

$$\alpha = \frac{\sum_{x=0}^{|s|-1} w_x}{|s|} \geq 1.$$

So in general,

$$|s| \binom{N}{2} \leq \binom{N^2}{2} - (2N + 1) \binom{N}{2},$$

or

$$|s| \leq N(N-1) - 1.$$

Hence the result. ∎

D. Properties of BDI Assignment Protocol

Proposition 1 (Solution Guarantee): The BDI assignment protocol ensures MA^3 always terminates in a finite number of negotiation rounds.

Proof: Starting from an arbitrary state $v \in V$ of an ARG $\mathcal{G}$, by Property 3, MA^3 will take a *finite* number of transitions, one per negotiation round to reach a state $v_o \in V_o$. In this state, a final round of negotiation proceeds during which, by Property 4, the arbitration agent will receive the lack of intentions by all agents and inform them to terminate negotiation. Hence the result. ∎

Proposition 2 (Computational Simplicity): Given an arbitrary $N \times N$ CLAP instance, the BDI assignment protocol ensures the worst-case complexity of MA^3 in terms of the number of negotiation rounds is $O(N^2)$.

Proof: By Property 7, the maximum number of transitions in an admissible string of $\mathcal{G}$ is $N(N-1) - 1$. Since MA^3 will execute a transition per negotiation round until the last round when all agents discover their lack of intentions to exchange objects, the negotiation rounds would not exceed $N(N-1)$. In terms of the number of negotiation rounds, it follows that the worst-case complexity is $O(N^2)$. Hence the result. ∎

With Property 4, we can characterize a worst case agreement (when negotiation terminates), denoted $v_{wc} \in V$, as follows:

$$|v_{wc}| = \min\{|v_o| \mid v_o \in V_o\} \tag{12}$$

With Property 5, we can characterize an optimal agreement, denoted $v_{opt} \in V$, as follows:

$$|v_{opt}| = \max\{|v_o| \mid v_o \in V_o\} \tag{13}$$

Given an agreement $v \in V$, its deviation in value from (or error with respect to) the optimal is defined by

$$\epsilon_v = \frac{|v_{opt}| - |v|}{|v_{opt}|} \times 100\% \tag{14}$$

Definition 4: A negotiation protocol is said to be ϵ_v-stable if any strategy utilized with it will converge to an agreement value $|v|$ that is within ϵ_v of (the optimal) $|v_{opt}|$.

Using an ϵ_v-stable negotiation protocol, the implication is that no negotiation agents which can tolerate a maximum of ϵ_v from the optimal agreement value will have any incentive to deviate from their adopted strategies. Importantly, this simplifies agent design as it means that no additional, complex and time wasting reasoning is needed by any agent to speculate about others' strategies to arrive at an agreeable solution.

Let $\epsilon_v = \epsilon_{wc}$ when $v = v_{wc}$, and we arrive at the following proposition.

Proposition 3 (Stability): The BDI assignment protocol in MA^3 is ϵ_{wc}-stable.

Proof: By Property 3, MA^3 will reach a state $v_o \in V_o$ regardless of any strategy used, where it will terminate since by Property 4, the termination condition for negotiation is satisfied, guaranteeing it stays in state $v_o \in V_o$. Hence the result by Definition 4 with $\epsilon_v = \epsilon_{wc}$ in (14). ■

Ideally, $\epsilon_{wc} = 0\%$. As Section V will show, empirically ϵ_{wc} is found to be quite small, approximately 10%, with a high probability of achieving an even smaller deviation of 5% through an appropriate arbitration-negotiation combination of strategies.

V. Simulation Results, Analysis & Discussions

In this section, we present an empirical study of MA^3 to assess the comparative performance of all strategies proposed. The performance is assessed primarily by the solution quality produced and the implementation-independent negotiation speed. The solution quality is measured (and graded) in terms of the various extents (in %) that a solution produced deviates from the optimal one, and the negotiation speed is measured in terms of the number of negotiation rounds needed to converge to a solution. The 'profile' of the performance is gathered together with the various probabilities of interest defined, which include those of converging to these 'graded solutions, and those of the algorithm running at various defined speed levels.

A. Simulation Set-up

To conduct the study, we first prototyped a simulator for the algorithms. The simulator consists of a centralized program running on an Intel® Pentium® personal computer with a 1.8GHz CPU and 512MB (RAM) memory. For a $N \times N$ problem instance, the program generates and inputs each of the $N!$ initial assignment solutions to a reasoning mechanism which computes the agents' object selections which would have resulted from the distributed MA^3 agents executing their adopted strategies. The centralization is aimed at simplifying the code that, importantly, automates and speeds up the experimental (running and data collection) process, but with all the features of the original algorithm retained, except for its distribution.

In principle, MA^3 can handle an arbitrary problem size N. But for a complete simulation, the number of simulation runs required is $N!$ per problem instance. Clearly for a big N, it can become intractably time consuming to simulate for a large number of problem instances. For experimental purposes, we limit $N = 10$, requiring 10! (or 3,628,800) simulation runs per problem instance, along with using an available implementation[5] of one LAP algorithm [4] to produce an optimal total value as a reference solution. This was manageable when we ran the simulator prototype executing all the possible strategy combinations for the same set of 100 *randomly generated* 10×10 problem instances. Despite the limit on N, we note that the simulation results can also provide a base reference for addressing large problem instances decomposed into smaller subproblems for MA^3. Problem decomposition, however, is usually done based on application-specific criteria that are beyond the scope of this paper.

The average simulation result of each variable Z, $Z \in \{\epsilon_{wc}, n_{max}, P_x, P_{wc}, P_{vhi}, P_{hi}, P_{lo}\}$ (notational definitions in Appendix II) is computed and tabulated for comparative study of all *arbitration-negotiation* strategy combinations.

B. Performance Comparisons

The simulation results for negotiation speed and solution quality are tabulated in Tables I and II respectively. Since MA^3 does not guarantee an optimal agreement in general, it seems reasonable to interpret the agreement reached more qualitatively to categorize its acceptability level. So in the following discussion drawn on the tables, an assignment agreement is said to be *good enough*[6] if its total A-QoS value (2) is within 20% of the optimal; is *near optimal* if it is within 10%, and *almost optimal* if it is within 5%.

[5]From website http://www.magiclogic.com/assignment.html

[6] It seems appropriate to use 'good enough' as a qualitative reference for solutions with total A-QoS in $[0.8 \max\{S_{tot}\}, \max\{S_{tot}\}]$ since, applying *Pareto's 80/20 rule*, 80% of CLAP applications can tolerate a 20% deviation from their optimal solutions.

TABLE II

MA[3] SIMULATION RESULTS: QUALITY PERFORMANCE PROFILE

Strategy		Solution Quality						
A	N	ϵ_{wc} (%)	P_0	P_5	P_{10}	P_{15}	P_{20}	P_{wc}
MaxCon	Greedy	10.3596	0.3167	0.9250	0.9963	1.0000	1.0000	0.0012
Random	Greedy	10.3596	0.2928	0.8996	0.9926	1.0000	1.0000	0.0017
MaxCon	MinCon	10.3596	0.2729	0.8919	0.9925	1.0000	1.0000	0.0057
Random	Random	10.3596	0.2613	0.8886	0.9899	1.0000	1.0000	0.0019
Random	MinCon	10.3596	0.2585	0.8897	0.9926	1.0000	1.0000	0.0036
Greedy	Greedy	10.3596	0.2584	0.8940	0.9903	1.0000	1.0000	0.0021
MaxCon	Random	10.3596	0.2522	0.8772	0.9894	1.0000	1.0000	0.0024
Greedy	Random	10.3596	0.2506	0.8813	0.9877	1.0000	1.0000	0.0023
Greedy	MinCon	10.3596	0.2391	0.8777	0.9895	1.0000	1.0000	0.0038
MinCon	Greedy	10.3596	0.2168	0.8924	0.9926	1.0000	1.0000	0.0029
MinCon	MinCon	10.3596	0.1972	0.8829	0.9926	1.0000	1.0000	0.0039
MinCon	Random	10.3596	0.1845	0.8779	0.9899	1.0000	1.0000	0.0033
Random	MaxCon	10.3596	0.1816	0.8775	0.9913	1.0000	1.0000	0.0006
Greedy	MaxCon	10.3596	0.1754	0.9243	0.9953	1.0000	1.0000	0.0002
MinCon	MaxCon	10.3596	0.1702	0.9161	0.9913	1.0000	1.0000	0.0001
MaxCon	MaxCon	10.3596	0.1246	0.7805	0.9723	1.0000	1.0000	0.0029

TABLE I

MA[3] SIMULATION RESULTS: SPEED PERFORMANCE PROFILE

Strategy		Negotiation Speed			
A	N	n_{max}	P_{vhi}	P_{hi}	P_{lo}
Greedy	Greedy	16	0.0040	0.1162	0.0270
Greedy	MinCon	16	0.0037	0.1073	0.0414
MinCon	Greedy	16	0.0034	0.0953	0.0497
MinCon	MinCon	16	0.0032	0.0918	0.0550
Greedy	Random	17	0.0035	0.0980	0.0564
Random	Greedy	17	0.0028	0.0792	0.1092
MinCon	Random	18	0.0029	0.0777	0.0887
Random	MinCon	19	0.0025	0.0704	0.1826
Random	Random	20	0.0022	0.0585	0.2393
MaxCon	Greedy	21	0.0015	0.0345	0.2451
Greedy	MaxCon	24	0.0010	0.0200	0.4680
MaxCon	MinCon	26	0.0012	0.0209	0.4851
MinCon	MaxCon	27	0.0009	0.0157	0.6045
MaxCon	Random	27	0.0009	0.0149	0.5839
Random	MaxCon	35	0.0007	0.0127	0.6714
MaxCon	MaxCon	51	0.0002	0.0017	0.9641

The worst case agreement had a constant deviation of 10.3596%, and hence was (almost) *near optimal* or better than *good enough*. In fact, regardless of the strategy adopted, the probability of arriving at a *near optimal* agreement was a high score of over 97%. Besides, when the arbitration and negotiation agents randomly adopted one of the respective three strategies, the speed performance profile was in the mid-range and the quality performance profile was above the mid-range. Together, the findings suggest that where speed is not a major concern, better than good enough agreements can often be reached without knowing the strategies adopted by these agents.

The A-Greedy:N-Greedy strategy combination had the highest probabilities of reaching agreements at very high speed (P_{vhi}) and high speed (P_{hi}), and the lowest in the maximum number of rounds possible (n_{max}) and in the probability of reaching agreements at low speed (P_{lo}). The A-MaxCon:N-Greedy strategy combination had the highest probabilities of achieving optimal (P_0), almost optimal (P_5) and near optimal (P_{10}) agreements. Following, the former strategy combination produced the best speed performance profile with an above mid-range profile in solution quality, whereas the latter one produced the best quality performance profile with a mid-range profile in speed.

The A-MaxCon:N-MaxCon strategy combination produced the worst performance profiles in both speed and solution quality. In fact, with agents adopting the N-MaxCon strategy, the performance profiles are among the lowest in both speed and quality. This finding suggests that the negotiation agents should avoid the N-MaxCon strategy altogether.

With the negotiation agents adopting the N-Greedy or random strategy, the arbitration agent adopting the A-MaxCon strategy tended to lower the speed performance profile, but raise the quality performance profile. Conversely, the arbitration agent adopting the A-MinCon strategy tended to raise the speed performance profile, though not as high as adopting the greedy strategy, but lower the quality performance profile that was worse off than adopting the greedy one. This finding is significant in anticipation of autonomous negotiation agents preferring a greedy or unknown strategy, necessitating a decision on which strategy to use for the arbitration agent to influence speed or quality.

Finally, that the protocol is theoretically ϵ_{wc}-stable is supported with ϵ_{wc} found to be empirically small, approximately 10 %, implying that agents using the protocol can produce better than *good enough* global object allocations, regardless of the strategies adopted.

In summary, behaviorally, we observe that a Greedy agent is more focussed on the social goal as it seeks to optimize incremental gains in every round, whereas a MaxCon agent encourages more negotiation by inducing its exchange partner with more beliefs in a subsequent round. Where both the arbitration agent and the negotiation agents are focussed on the social goal, the result is highest speed performance profile. Where the arbitration agent (overseeing the negotiation process) encourages more negotiation, with the negotiation agents focussed on the social goal, the result is highest quality performance profile.

VI. Related Work

A. Distributed Constraint Reasoning

There are some efforts not cast in the context of assignment but appear to have addressed a similar problem in the context of *distributed constraint reasoning*, notably, work on distributed constraint optimization problem (DCOP) (e.g., [5], [6]). In principle, CLAP is a DCOP, as originally pointed out in [3, p. 262]. That CLAP is a DCOP not addressed by current DCOP techniques has been discussed in [7, p. 1580].

B. BDI Models

Among the agent architectures/models (see [8, Ch. 1]), the BDI model [9], [10] is one of the best known and studied model of practical reasoning. Based on a philosophical model of human practical reasoning, originally developed by M. Bratman [11], the basic model guides us to develop an agent to decide moment by moment which action to take in the furtherance of a goal. We adapt this model, motivated by its appropriateness in allowing us to conceptualize and metaphorically describe an agent's reasoning mechanism, moment by moment, in terms of the agent' mental attitudes B, D and I to solve CLAP. However, two aspects clearly differentiate our work from existing BDI models. In the first is our approach to modelling. Existing BDI models are developed without concisely formulating the problems they attempt to solve while in our work, the BDI model is developed with a clear formulation of the problem it addresses, namely, CLAP. In the second, each *moment* is not a moment of reasoning in reaction to changes in its environment, but a negotiation round of collaborative reasoning - in fact, existing BDI models give no architectural consideration to explicitly multiagent aspects of behaviour [12] that is essential for addressing CLAP.

C. Automated Negotiation

In the literature on general negotiation frameworks, agents that can negotiate with exact knowledge of each other's cost and utility functions, or such knowledge learnt in the initial step of interaction, have been proposed [13], [14]. There are agents that negotiate using the unified negotiation protocol in worth-, state-, and task-driven domains where agents look for mutually beneficial deals to perform task distribution [15], [16]. In negotiation via argumentation (NVA), the agents negotiate by sending each other proposals and counter-proposals. In [17], these proposals are accompanied by supporting arguments (explicit justifications) formulated as logical models. In [18], the distributed constraint satisfaction problem (DCSP) algorithm [19], [20] provides the computational model, extended with the supporting arguments (accompanying the proposals) formulated as local constraints. In [21], agents can conduct NVA in which an agent sends over its inference rules to its neighbour to demonstrate the soundness of its arguments. Finally, there are also negotiating agents that incorporate market driven techniques (e.g. [22]), auction mechanisms (e.g. [23], [24]) and other AI techniques (e.g. [25]).

The proposed MA^3 differs from existing work on negotiation in that it employs a new BDI negotiation model for CLAP (Section III) not specifically addressed in all existing work, which, to emphasize, deals with allocation of objects among agents *concurrently*. The important features of MA^3 include: (i) it can be easily adapted to respond *on-the-fly* (i.e., during the negotiation process) to online changes in individual A-QoS values, and (ii) it produces *anytime ready* solutions, rendering it the robustness to deal with a dynamic environment [3].

VII. Conclusions

This paper has visited the BDI negotiation model for CLAP in the standard framework of automated negotiation, conceptually separating it into a BDI assignment protocol and an adopted greedy strategy. Facilitated by this conceptual separation is a systematic and more extensive development of the model. On the one hand is a formal and more rigorous analysis of the protocol, establishing its salient properties, namely, *solution guarantee*, *simplicity*, and ϵ_{wc}*-stability*. On the other hand are new strategies developed based on a novel idea of *cooperative concession*, extending to a strategy set for the arbitration and negotiation agents. Extensive simulations of all possible arbitration-negotiation combinations of strategies running with the protocol, as embodied in MA^3, reveal several important findings (Section V-B) on the speed and quality performance profiles. Combining theoretical and empirical insights on ϵ_{wc}-stability, an important inference to draw is that by the BDI assignment protocol for CLAP, distributed agents can utilize local A-QoS information with BDI-driven communication. The outcome is better than good enough global allocations (Footnote 6), regardless of the strategies adopted.

Some future work includes (i) decentralizing the arbitration role to the negotiation agents to remove the centralized arbitration agent in MA^3 altogether, by adapting the idea of collaborative local mediation [7] to the new cooperative-concession framework proposed in this paper, (ii) extending the protocol to reach better agreements and (iii) the use of heuristics to speed up negotiation.

To conclude, the research on MA^3 for CLAP should provide a base reference for researchers interested in agent negotiation approaches for solving traditional combinatorial problems in general.

Appendix I
Basic Graph-Theoretic Terminology & Results

1) A (finite state) graph denoted G is said to be k-colorable if each of its states (or vertices) can be assigned one of k colours in such a way that no two adjacent vertices (i.e., vertices connected by an edge) are assigned the same colour.
2) The chromatic number of a graph G, denoted $\chi(G)$, is the smallest k for which the graph is k-colorable. For a k'-colorable graph G, $2 \leq \chi(G) \leq k'$.
3) A graph is *bipartite* if the vertices can be partitioned into two sets, V-0 and V-1, so that the only edges of the graph are between the vertices in V-0 and the vertices in V-1.

Two elementary graph-theoretic results are as follows.

Lemma 1: A graph is 2-colorable if and only if it has no cycles of odd length.

Lemma 2: $\chi(G) = 2$ if and only if G is a bipartite graph.

Appendix II
Simulations: Legend

1) n : number of negotiation rounds.
2) n_{max} : maximum n, obtained when algorithm terminates, and is the largest of all n obtained from each of the $N!$ different initial assignments simulated for a $N \times N$ problem instance.
3) α : total A-QoS value S_{tot} (2).
4) α_{opt} : optimal α.
5) α_{wc} : worst-case α, obtained when algorithm terminates, and is the worst of all α's computed based on the simulation of $N!$ different initial assignments generated for a $N \times N$ problem instance.
6) ϵ : error, given by $\left(|\frac{\alpha-\alpha_{opt}}{\alpha_{opt}}| \times 100\%\right)$ (Note: This is equivalent to (14)).
7) ϵ_{wc} : worst-case ϵ, where $\alpha = \alpha_{wc}$.
8) P_x : probability that an initial assignment can lead to a solution with total A-QoS α within $x\%$ of optimal, i.e., $\alpha \in [\frac{(100-x)}{100}\alpha_{opt}, \alpha_{opt}]$.
9) P_{wc} : probability that an initial assignment can lead to a worst-case solution, i.e., with $\alpha = \alpha_{wc}$.

10) P_{vhi} : probability that the algorithm runs at very high speed, i.e., the number of negotiation rounds it can take to reach a solution is not more than the greatest integer $\leq 0.3N$.
11) P_{hi} : probability that the algorithm runs at high speed, i.e., the number of negotiation rounds it can take to reach a solution is not more than the greatest integer $\leq 0.5N$.
12) P_{lo} : probability that the algorithm runs at low speed, i.e., the number of negotiation rounds it can take to reach a solution exceeds N.

All the probabilities of interest defined above are computed using formula $\left(\frac{\beta}{\gamma}\right)$, where integer β is the number of initial assignments satisfying the associated conditions *upon termination of algorithm*, and integer γ is the total number of different initial assignments input for simulation. $\gamma = N!$.

REFERENCES

[1] R. E. Burkard and E. Cela, "Linear assignment problems and extensions," in *Handbook of Combinatorial Optimization, Vol.4*, P. M. Pardalos and D. Z. Du, Eds. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, pp. 75–149.

[2] J. S. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. Cambridge, M.A, USA: The MIT Press, 1994.

[3] K. T. Seow and K. Y. How, "Collaborative assignment: A multiagent negotiation approach using BDI concepts," in *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'02)*. Palazzo Re Enzo, Bologna, Italy: ACM Press, July 2002, pp. 256–263.

[4] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. -, pp. 325–340, 1987.

[5] R. Mailler and V. Lesser, "Solving distributed constraint optimization problems using cooperative mediation," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'04)*. New York City, New York, USA: IEEE Computer Society, July 2004, pp. 438–445.

[6] P. J. Modi, W.-M. Shen, M. Tambe, and M. Yokoo, "An asynchronous complete method for distributed constraint optimization," in *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'03)*. Melbourne, Victoria, Australia: ACM Press, July 2003, pp. 161–168.

[7] K. T. Seow and K. M. Sim, "Decentralized assignment reasoning using collaborative local mediation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 11, pp. 1576–1580, November 2006.

[8] G. Weiss, Ed., *Multiagent System : A Modern Approach to Distributed Artificial Intelligence*. The MIT Press, London, U.K, 1999.

[9] M. E. Bratman, D. J. Israel, and M. E. Pollack, "Plans and resource-bounded practical reasoning," *Computational Intelligence*, vol. 4, no. 4, pp. 349–355, 1988.

[10] M. P. Georgeff and A. L. Lansky, "Reactive reasoning and planning," in *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI'87)*, Seattle, WA, USA, July 1987, pp. 677–682.

[11] M. E. Bratman, *Intentions, Plans and Practical Reason*. Harvard University Press, Cambridge, M.A, 1987.

[12] M. P. Georgeff, B. Pell, M. E. Pollack, M. Tambe, and M. Wooldridge, "The Belief-Desire-Intention model of agency," in *Intelligent Agents V: Agent Theories, Architectures, and Languages (ATAL): Proceedings of the 5th International Workshop, ATAL'98*, J. P. Müller and A. S. R. Munindar Singh, Eds. Paris, France: Springer Verlag, London, UK, July 1998, pp. 1–10.

[13] S. Kraus, "Beliefs, time and incomplete information in multiple encounter negotiations among autonomous agents," *Annals of Mathematics and Artificial Intelligence*, vol. 20, no. 1-4, pp. 111–159, 1997.

[14] S. Kraus, J. Wilkenfeld, and G. Zlotkin, "Multiagent negotiation under time constraints," *Artificial Intelligence*, vol. 75, no. 2, pp. 297–345, 1995.

[15] J. S. Rosenschein and G. Zlotkin, "Designing conventions for automated negotiation," *AI Magazine*, vol. 15, no. 3, pp. 29–46, 1994.

[16] G. Zlotkin and J. S. Rosenschein, "Mechanism design for automated negotiation, and its application to task oriented domains," *Artificial Intelligence*, vol. 86, no. 2, pp. 195–244, 1996.

[17] S. Kraus, K. Sycara, and A. Evenchik, "Reaching agreements through argumentation : a logical model and implementation," *Artificial Intelligence*, vol. 104, no. 1-2, pp. 1–69, 1998.

[18] H. Jung, M. Tambe, and S. Kulkarni, "Argumentation as distributed constraint satisfaction : Applications and results," in *Proceedings of the 5th International Conference on Autonomous Agents*. Montreal, Quebec, Canada: ACM Press, July 2001, pp. 324–331.

[19] M. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara, "The distributed constraint satisfaction problem : Formalization and algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, no. 5, pp. 673–685, September/October 1998.

[20] M. Yokoo and K. Hirayama, "Distributed constraint satisfaction algorithm for complex local problems," in *Proceedings of the Third International Conference on Multiagent Systems (ICMAS-98)*, 1998, pp. 372–379.

[21] C. Sierra, N. R. Jennings, P. Noriega, and S. Parsons, "A framework for argumentation-based negotiation," in *Proceedings of The Fourth International Workshop on Agent Theories, Architectures and Languages*, Rhode Island, USA, 1997, pp. 167–182.

[22] A. Chavez and P. Maes, "Kasbah : An agent marketplace for buying and selling goods," in *Proceedings of the First International Conference on Practical Application of Intelligent Agents and Multi-Agent Technology*, London, UK, April 1996, pp. 75–90.

[23] C. Boutilier, M. Goldszmidt, and B. Sabata, "Sequential auctions for the allocation of resources with complementarities," in *Proceedings of the 8th International Joint Conference on Artificial Inteligence*, 1999, pp. 527–534.

[24] K. M. Sim and E. Wong, "Toward market-driven agents for electronic auction," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 31, no. 6, pp. 474–484, November 2001.

[25] R. Krovi, A. C. Graesser, and W. E. Pracht, "Agent behaviors in virtual negotiation environments," *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 29, no. 1, pp. 15–25, February 1999.