Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

# Towards a social responsible agents

W. Alshabi[1], M. Itmi[1], S. Ramaswamy[2], H. Abdulrab[1]

[1] LITIS Laboratory, INSA – Rouen, 76131 Mt. St.-Aignan Cedex, Rouen, France. [waled.alshabi, mhamed.itmi, habib.abdulrab]@insa-rouen.fr
[2] Computer Science Department, Univ. of ARKANSAS at Little Rock, AR 72204 USA. srini@acm.org

***Abstract*—Current research in autonomous Agents and Multi-agent systems (MAS) has reached a level of maturity sufficient enough for MAS to be applied as a technology for solving problems in an increasingly wide range of complex applications. Our aim in this paper is to define a simple, extendible, and formal framework for multi-agent cooperation, over which businesses may build their business frameworks for effecting cooperative business strategies using distributed multi-agent systems. It is a simple, fair and efficient model for orchestrating effecting cooperation between multiple agents.**

***Keywords*— Cooperation, Coordination, Multi-agent systems (MAS), Organization**

## I. INTRODUCTION

Cooperation and coordination are interconnected. For all successful coordinations, cooperation is essential [1, 2]. While many technologies today have worked on coordination strategies, most implementations are at best either ad-hoc or poorly adaptable and scalable. Similarly in the systems we have observed so far, cooperation is also best ad-hoc in the way it is developed. Hence in this paper, we attempt to define a simple and formal framework for multi-agent cooperation, over which businesses may build their framework for effecting cooperative business strategies using distributed multi-agent systems.

Cooperation is a key MAS concept [3-6]. Durfee and colleagues [7] have proposed four generic goals for agent cooperation:

- Increase the rate of task completion through parallelism;
- Increase the number of concurrent tasks by sharing resources (information, expertise, devices, etc);
- Increase the chances for task completion by duplication and possibility using different modes of realization;
- Decrease the interferences between tasks by avoiding the negative interactions.

However, cooperation in agent-based systems is at best unclear and at worst highly inconsistent [8]. Researchers like Galliers [9-10] and Conte [11] underlined the importance of adopting a common goal for agent of cooperation which they consider as an essential element of the social activity. We can characterize a MAS system by the type of implemented cooperation which can range from total cooperation to the total antagonism [12]. Completely cooperative agents can change their goals to meet the needs of other agents. Antagonistic agents, on the other hand, will not cooperate and, their respective goals may be blocked.

If there is no cooperation, entities (agents) will only realize business opportunities that they have a priori knowledge of, or their clients happened to find them. However, it is difficult to rapidly grow such a business in a competitive market space. The reasons are:

- Efforts in finding new business opportunities.
- Making sure their customers are always satisfied.
- Improving their business growth opportunities.

Cooperation helps business evolution/growth. Simple, fair and efficient cooperation techniques are fundamental to building efficient coordination mechanisms. In our work, we have focused on the development of a simple, fair and efficient model for effecting cooperation between multiple agents.

- It's simple, because it is quite intuitive in its approach and is computationally tractable such that it can be easily adapted and applied across multiple domains with different types of constraints w.r.t. issues such as bandwidth, etc.
- It is fair implies that each participating agent (if they can provide a needed service) gets a fair and equal bidding opportunity on incoming jobs.
- It is efficient because it is pragmatic in the way that cooperation is designed and orchestrated. By decoupling overarching business and policy expectations from the solution design, we provide a very efficient mechanism for not just implementing cooperation, but also to reason about issues that stifle cooperation (this is due to the DAB[1] [13], which will be explained subsequently).

In addition, within this framework, we also relegate the responsibilities for learning to the individual agents, hence allowing agents to evolve independent of one another. Thereby individual agents are not bound by predefined learning strategies and agents may choose to use any strategy as well as build locally driven heuristics for learning and for responding to a request for bids. Over time, since both their success factors and DAB are used in the bid evaluation process, agents representing entities that do not positively cooperate for the mutual benefit of the entire group will be evidently noticeable.

[1] Degree of Agent's Believability



978-1-4244-2794-9/09/$25.00 ©2009 IEEE

The technique proposed by this paper enforces very minimalistic global control policies, while at the same time allowing maximal control for bid decisions within each of the individual agents. This is also supported by our decision to decouple the dynamic execution hierarchy (in real-time) from the static business hierarchy (for organizational needs) within the proposed cooperation model.

This paper is organized as follows: The second section introduces our hierarchical model, the notion of the CPS[2] process as well as the different steps, which compose our CPSP[3]. The third section introduce the term Cooperation Indicator or CI for short, which has for function to quantify the cooperation. It works also as an individual-community balance which is required to ensure that the overall system, as well as the individual agents, is able to function in an effective manner. The fourth section gives an illustrative example which describes and details the proceeding of our choice mechanism. Different experiments as well as their results will be given in the fifth section. In the sixth section, we will have a discussion about the developed model. Finally, the seventh section concludes the paper.

## II. A Hierarchical Model

Bond [14] describes the existence of two types of MAS architectures:

- Horizontal: This structure is useful in some contexts, for example, a situation where a group of agents having different (non-overlapping) capabilities and hence can work towards the goal without needing any conflict resolution. Here all the agents are on the same level with equal importance without a Master / slave relationship.
- Vertical: In a vertical architecture, the agents are structured in some hierarchical order. Agents at the same sub-level may share the characteristics of a horizontal structure. The 'horizontally structured' MAS model has several issues –a critical issue is that it quickly becomes too complex and unwieldy for practical applications, wherein agents in the MAS may share some common capabilities. Hence most current frameworks have adopted a hierarchical MAS model (vertical) by organizing the agents in some organizational structure.

In [15] we compared between three widely used models for agent cooperation in MAS: HOPES, HECODES [16] and MAGIC [17-18]. We highlighted also the shortcoming and the limitations of these models of our viewpoint.

We have developed a hierarchical MAS model [15, 19-20] focused on enabling effective cooperation. The choice of a hierarchical model was essentially to overcome the limitations of prior architectures and to avoid certain inconveniences that appear in classical multi-agent models. For instance, the number of links between agents and the quantity of information exchanged between agents that becomes complicated with the increasing number of agents involved in the interactions; and therefore turns into a serious handicap in such models. Additionally, other problems can also appear, such as coordination and control [21-24].

In our model, agents are autonomous entities having control of their own resources as well as bestowed with competences which allow them to cooperate, communicate and work with other such agents. Each agent is capable of providing specific solutions and of resolving local problems autonomously. Agents can either provide, or need the assistance of other agents, due to lack of information, resources, etc. to accomplish a particular task. This need for assistance by other agents can appear when an agent either doesn't have the necessary skills that allow the most effective realization of a specific task, or when an agent prefers a cooperative solution.

The skill of an agent is an important notion that characterizes the agents in our model. Each agent may have one or several specific skills at the same time. These skills may allow each agent to have specific roles.

Agents can also be divided into several groups. Each group of agents consists of several cooperative agents' members, called professional agents, and a superior member, called coordinator agent. The following sections describe, in a brief manner, the functionality of our CPS process model.

### A. *Hierarchical model functionality*

In this section, we give an overview about the model's functionality as well as its four phases. These different phases are explained in detailed manner in related article [19]. The CPS phases of our model include:

*Recognition:* The CPS process begins when an agent, or a group of agents, identify the need for cooperative action. This need for cooperative action may occur when an agent has to accomplish a goal for which it does not have the necessary capability, or when the agent has a preference towards a cooperative solution. The identification of the need for cooperative action is the recognition phase.

*Skills' Search:* In this phase, the agent that identifies the need for cooperative action solicits appropriate assistance. This solicitation is realized through a special process named skills' search process (SSP). The SSP searches for agents which have the necessary skill(s) to realize the task. The SSP is explained in detail in [19].

*Agents' Choice:* we will concentrate on this particular aspect in this paper. During this phase, the agent(s) which possess the necessary skills to realize the specific task, say T, will be identified as competent agents for accomplishing the needs of the customer. These agents will be contacted by an initiator agent, which attempts to negotiate with the set of competent agents to choose the best agent for accomplishing T. This process is often repetitive; since the chosen agent can repeat the same process independently for a set of tasks. This phase allows us to form the initial cooperation structure [25-26], termed a collective.

*Execution:* In this phase, the members of the collective realize the roles they have been negotiated to perform and provide a feedback that can be used to judge the quality of service provided in accomplishing the customer's request. The

[2] Cooperative Problem Solving.
[3] Cooperative Problem Solving Process.

initiator then is able to provide a performance evaluation of the ability of the agents in the collective to perform the job as negotiated before the actual task assignments.

## III. Our Choice Mechanism

In the prior section we mentioned the four phases that constitute the CPSP of our model. As the two first phases has been widely treated in a related article [9], this section concentrate on the choice mechanism.

The objective of this process is to allow for appropriate choices among agents that have the required skills to accomplish a necessary task (also referred to as operation). Before delving into the details of this process; let us recap and clarify the result of the search mechanism, discussed previously, which can be one of the following:

- No agent possesses the required skills; In this case the initiator agent doesn't have any possibilities of choice due to the non-existence of candidates to perform the required task.
- One agent possesses the required skills; In this case the initiator agent finds only one competent agent at the end of the search process. As a result, the initiator agent doesn't have a real choice to make, due to the fact that there is no more than one available candidate for the task.
- Several agents possess the required skills; In this case the initiator agent finds, at the end of the search process, several competent agents that have simultaneously the needed skills. For that reason, the initiator agent needs a negotiation as well as choice mechanism that allows for choosing amongst the available candidates.

The need of the choice mechanism emerges when we have several agents having the needed skill, these agents are the result of the skills' search process, and by consequence, the initiator needs to select among the competent agents. The selection will concern those who offer the optimal solutions. For this objective our choice mechanism will apply the negotiation in order to get the optimal solutions.

For example, when the initiator agent searches for a skill, all the competent agents (agents who possess the requested skill) will reply to the initiator by sending their bids. These agents will be considered as potential candidates. Consequently, their bids will be evaluated by the initiator according to several parameters.

This choice of optimal solution can be realized after evaluating the different bids given by the agents. This evaluation can be realized through a special function defined for this purpose by the user according to the application's domain. Thus, the choice of optimal solution can be realized according to many parameters, for instance, it can be the agents who offer the best price or the best time to realize the needed skill.

As mentioned earlier, when any agent wishes to have some task performed (by one or more agents), the Initiator agent starts by initiating a search for competent agents. Then, the initiator will solicit proposals from these agents by issuing a call for proposals (*cfp*) act (see [FIPA00037]), which specifies the task, as well any conditions the Initiator is placing upon the execution of the task. Participants receiving the call for proposals are viewed as potential candidates and are able to generate N responses. The agents may refuse to propose or respond with proposals to perform the required tasks. The proposal of each agent should contain the agent's bid in addition to its Success Factor (detailed in the sequel paragraphs).

The Initiator will evaluate the received proposals and then selects the agent that have the most satisfactory (optimal if needed, dependent upon the design) proposal to perform the required task. For the realization of a specific task T, the evaluation of the different bids will be realized by using the following equation:

$$\left[ CI_{\text{expected}} \times \left( \frac{FV}{SF_{A_i}} \right) \times Bid_{A_i}^{\text{evaluation}} \times DAB_{A_i} \right]_{A_i} \qquad (1)$$

This evaluation, using (1), will be repeated automatically for each bid. Each parameter in (1) has an important role during the evaluation process:

- $CI_{expected}$: We write *CI* to represent the Cooperation Indicator, which has a fundamental role to define a new flexible model. The *CI* indicates the level of desired and expected cooperation. This provides the necessary framework to model both passive and active cooperation, depending on the user's need. The expected value of *CI* is given by the user as initial data, driven by appropriate business process needs, at initialization. The following figure demonstrates the notion of cooperation indicator (it can be any of the grey triangles).

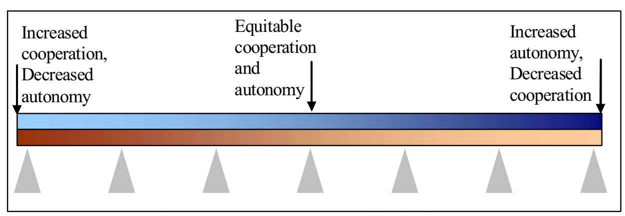


Figure 1 - Agents' Cooperation indicator.

- *FV* : the *FV* is used to represent the Fairness Value, which is an important parameter for the choice mechanism. *FV* can be regarded as the optimal value that each agent has to reach as its stated objective, again from an ideal perspective. FV is used to improve the different agents' success metric during a cooperation process. The *FV* is also a given set by the user as initial data, at the initialization.

- $SF_{Ai}$ : *SF* to signify a Success Factor. The notion of Success Factor was mentioned briefly earlier. We had pointed out that, agents having the required skills for a task may respond to the initiator agent's solicitation by sending their bids. These bids will be accompanied by a Success Factor for each agent. This value of *SF* defines the offers' acceptance average for each agent. For instance, if the *SF* of agent $A_i$ is equal to *0.4*, which means that, *4* offers of *10* that the agent has responded to, have been successful for this agent. Thus, for each agent $A_i$, the Success Factor called $SFA_i$, is calculated separately. This is not the case with the two first parameters (*CI & FV*), which are predefined for the entire system. Consequently, each agent in our model has the following values calculate with every transactional event:

- The Total number of Requested Bids for agent $A_i$, called $RB_{Ai}$.
- The Total number of Successful Bids received by agent Ai, called $SB_{Ai}$.

These two parameters will be essential to calculate the *SF* for each agent. For this purpose, we divide the number of Successful Bids for agent $A_i$ by the number of Requested Bids for this agent. Thus, $SF_{Ai} = SB_{Ai} / RB_{Ai}$, where the value of the *SF* for agent $A_i$ is situated between 0 and 1, $0 \leq SF_{Ai} \leq 1$. To evaluate the value of the $SFA_i$ resulting from the previous equation, we have to take in consideration two indispensable factors. The first factor is the Fairness Value (*FV*), this factor was the subject of the precedent point. The second factor is the Tolerance Value, called (*TV*). Both factors are defined for the entire system and set up during the initialisation. In the optimal case, the Success Factor for $A_i$ should be equal to the given Fairness value, $SF_{Ai} = FV_{initial}$. In less advantageous case, the Success Factor for agent $A_i$ is not equal to the given Fairness value, Nevertheless, the $SF_{Ai}$ may be situated within a tolerance zone provided by the Tolerance Value, $TV_{min} \leq SF_{Ai} \leq TV_{max}$. Therefore, all the agents having a *SF* that lies within the tolerance zone (defined by *TV*) will be considered as agents having a good degree of success within this system. In worst case, we find agents that have a *SF*, which is not in the appropriate ranges. Here we distinguish two positions; agents with an inferior or superior *SF*, outside the tolerance range defined by *FV* and *TV*. From a cooperation perspective, agents in an inferior position are worthy of being shown favour; than those that have had much success. Tuning of this measure, can be done based upon specific applicative needs.

- $Bid_{A_i}^{evaluation}$: As its name indicates, this parameter represents the agent's bid evaluation. The value of this parameter results after evaluating the bid given by the agent $A_i$. This evaluation can be realized through a special function defined for this purpose by the user according to the application's domain. This function should take in consideration different parameters essential and related to the application's domain specific needs. These parameters may or may not have the same importance in every application.

- $DAB_{Ai}$: Degree of Agent's Believability

Once the selected agent has completed its assigned task, it begins a completion process to the Initiator. As part of the completion process, the Initiator starts an evaluation process, used to arrive at Degree of Agent Believability (*DAB*), for the agent that begins the completion process. In this process each agent will be evaluated directly at the end of its task execution by its Initiator. The value of *DAB* for each agent will be saved in a matrix devised for this purpose. The access and the modification of *DAB* matrix are mandatory and locked for the Initiator agent until the process is completed.

Initially, the value of *DAB* for every agent is set to 1, which means $DAB_0^{A_i} = 1$, then the *DAB* value can be calculated using the following equation:

$$DAB_{A_i}^{new} = 1/2 \times \left( DAB_{A_i}^{last} + B_{A_i}^{initial} / B_{A_i}^{delivered} \right) \quad (2)$$

Where, $DAB_{A_i}^{last}$ is the last value of $DAB_{A_i}$, $B_{A_i}^{initial}$ is the evaluation of agent $A_i$ offer, $B_{A_i}^{delivered}$ is the evaluation of agent $A_i$ performance (for assigned and accepted task (job) offers).

We have to emphasize that the equation (1) is used to evaluate the different agents' proposals except when the Initiator has the requested skill. That authorizes the Initiator to participate fairly in the proposition process. In this case, the evaluation of Initiator proposal will be different from other agents' proposals evaluation. To evaluate its own proposal the Initiator has to apply the subsequent equation:

$$\left[ \left(1 - CI_{expected}\right) \times \left( \frac{FV}{SF_{A_i}} \right) \times Bid_{A_i}^{evaluation} \times DAB_{A_i} \right]_{Initiator} \quad (3)$$

The main distinction between the equation (1) and the equation (3) exist in the method of calculate the first parameter of the equation, which concerns the Cooperation Indicator. In the equation (1) the value of *CI* is equal to the value of *CI* expected (initial data) whereas, in the equation (3) we deduct the value of *CI* expected from 1. This operation has as objective to favours the cooperation with other agents.

The purpose of having two different behaviours, to evaluate the first party of the equation, is to permit our model to be flexible and to avoid the problem of rigidity that subsists in many other models. So in systems tuned to offer a higher degree of cooperation, priority is given to the best candidate agent.

## IV. EXAMPLE

In this section we will present a number of simple examples that illustrate the proposed idea and put into concrete form the principles of Agents' Choice Mechanism that have been described.

The first example is concerned with first case: when two of the agents' parameters ($DAB_{Ai}$ and $SF_{Ai}$) still have no values yet. These parameters, by consequence, will take the values predefined by default. The Second example deals with situation more developed than the first example: after a period of time, the prior agents' parameters ($DAB_{Ai}$ and $SF_{Ai}$) will have different values, and thus, we have to choose among them.

### A. Example statement

Let five agents $A_1$, $A_2$, $A_3$, $A_4$ and $A_5$ (the restricted number of agents is just for illustrative purposes). At the beginning, the multi-agent system is organized into a hierarchy and our agents are organized in the hierarchy as follows: agents $A_2$ and $A_3$ have $A_1$ as group superior (coordinator), $A_1$ and $A_4$ have $A_0$ as group superior. Simultaneously, the agent $A_0$ is also the main coordinator of the whole hierarchy (see figure 2).

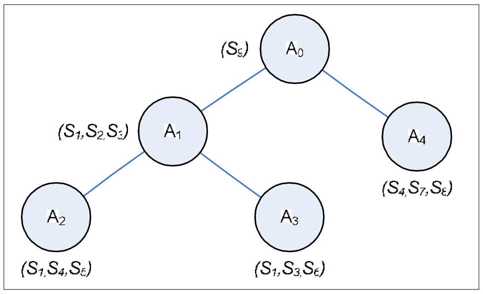


Figure 2 - Agents' hierarchy.

Evidently, each one of these agents may possess one or several skills at the same time. These skills allow the agent to realize a number of tasks (sometimes called operations), and consequently, permit each agent to hold a specific role within the group. Lets further assume that agent $A_1$ possess skills $S_1$, $S_2$ and $S_3$; agent $A_2$ possess skills $S_1$, $S_4$ and $S_5$; agent $A_3$ possess skills $S_1$, $S_3$ and $S_6$; agent $A_0$ possess skill $S_9$; and finally, agent $A_4$ possess skills $S_4$, $S_7$ and $S_8$. The distribution of these skills is shown in tableau 1. Nor the nature neither the functioning of these skills is the subject of our interest in this section.

| | *S1* | *S2* | *S3* | *S4* | *S5* | *S6* | *S7* | *S8* | *S9* |
|---|---|---|---|---|---|---|---|---|---|
| *A0* | | | | | | | | | ● |
| *A1* | ● | ● | ● | | | | | | |
| *A2* | ● | | | ● | ● | | | | |
| *A3* | ● | | ● | | | ● | | | |
| *A4* | | | | ● | | | ● | ● | |

Tableau 1 - Distribution of agents' skills.

At the initialization, the user will be required to provide the value of some essential parameters. These initial data given by the user is the Cooperation Indicator expected for the entire system (*CI*) and the Tolerance Value (*TV*). We supposed that the values given by the user at the initialization is equal to 80% for the *CI* and $\pm$ 10% for *TV*.

The value of *CI* given (by the user) indicates that the user prefer the cooperation among agents (prefer cooperative solution by 80%). To state the matter differently we can say that, four times of five the initiator agent will give advantage to other agents' offers instead of his own offers, and consequently supports the cooperative solution. Due to the existing of *TV* predefined by the user (*TV* =$\pm$ 10%), the *CI* for an agent Ai can be extended. Thus, we can consider the value of $CI_{A_i}$ is satisfied if it is situated between the value 0.72 and the value 0.88, thus $0.72 \le CI_{A_i} \le 0.88$.

### B. First example

In this example, we have the coming situation: the agent $A_2$ is asked to realize a task $T_3$, which the agent $A_2$ is incapable to realize. We say that an agent is capable to realize a task T if it has the appropriated skills. Thus, the realization of $T_3$ necessities the possession of the skill $S_3$. In the given example, we distinguish that the agent $A_2$ doesn't possess the required skill. Whereas, within this group the other two agents ($A_1$ and $A_3$) possess the required skill $S_3$. As a result, the agent $A_2$ (initiator) will solicit proposals from other agents that have the required and appropriated skills. This is effectuated by issuing a message that call for proposals (*cfp*) from other agents, as mentioned in section III.

Due to the fact that the initiator agent does not possess the appropriate skill (competence), the value of $FV_{initiator}$ is equal to 0, whereas the value of *FV* concerning the other agents is equal to $1/n$. Where $n$ is the number of the potential candidates and which is equal to 2 in this example.

The answer given by the agents $A_1$ and $A_3$, after evaluating their capabilities as well as their availability to realize $T_3$, is equal to 38 for $A_1$ and 41 for $A_3$. These values can represent the operation's cost, the operation's duration …etc. that depends on the application's domain.

In order to choose between these two competent agents, we should use the equation (1) in which the value of two parameters are already given by the user at the initialization, *CI* = 80 and we have calculated the value of another parameter, *FV* = 1/2. We still have to calculate three parameters: *DAB*, *SF* and Bid evaluation for agent $A_1$ and $A_3$ separately.

Being at the initialization, and because of the nonexistent of previous values:

- The DAB's value is equal to 1 for both A1 and A3; otherwise the DAB's value is calculated by the equation (2).
- In case of the existence of several parameters such as time, cost…etc. the agents' Bid evaluation depends on the importance of these parameters, which varies according to the application's domain. For instance, Bid evaluation = 3*Time +1/cost. In this example, the Bid evaluation is equal to 1 over the bids given by the competent agents; $Bid_{A_i}^{evaluation} = 1 / A_i^{offer}$ thus, bid evaluation for A1 =1/38 and 1/41 for A3.

The agents' SF is equal to 0.1 at the initialization, otherwise $SF_{Ai}=SB_{Ai}/RB_{Ai}$.

According to these values, for agent $A_2$ the offer of agent $A_1$ becomes:

$$\left[CI_{expected} \times FV / SF_{A1} \times Bid_{A1}^{evaluation} \times DAB_{A1}\right]_{A1}$$

$$\left[0.8 \times 0.5 / 0.1 \times 1/38 \times 1\right]_{A1} = 0.105$$

And the offer of agent $A_3$ becomes:

$$\left[CI_{expected} \times FV / SF_{A3} \times Bid_{A3}^{evaluation} \times DAB_{A3}\right]_{A3}$$

$$\left[0.8 \times 0.5 / 0.1 \times 1/41 \times 1\right]_{A3} = 0.098$$

From the two prior equations, we distinguish that the agent $A_2$ will favor the cooperation with A1 due to its results, which is superior to the result of $A_3$. Therefore, in the succeeding bids the value of SF as well as the value of *DAB* for agent $A_1$ will be different from the values given at the initialization.

Once the agent $A_1$ has finished the realization of the task $T_3$ and sends the results; the initiator ($A_2$) will revaluate the *DAB* concerning the agent $A_1$. The purpose of this revaluation is to insure the agent's engagement respect to its bid.

### C. Second example

At the previous example, we treated the case where the system is at the initialization. We attributed a default value for a number of parameters ($DAB=1$ and $SF=0.1$) because of the nonexistent of previous values.

In this example we will study the agent choice process after a period of time $T$. Thus, the two previous parameters would probably have values.

This time, the agent $A_2$ is asked to realize the task $T_3$. Contrarily to the previous example, the agent $A_2$ has the appropriated skill $S_3$, and so it is capable to realize the required task. Within its group the other two agents ($A_1$ and $A_3$) possess also the required skill. The agent $A_2$ (*initiator*) will solicit again proposals from these agents. The offers solicitation this time is justified by the need to choose the best offer and to cooperate with other agents which may be essential for their survive [27, 28].

Evidently, the initial data given by the user previously (80% for the $CI$ and $\pm$ 10% for $TV$) will be used also in this example. As the initiator agent $A_2$ possess the required skill, its bid will be evaluated by the equation (3). Whereas, for other agents the evaluation will be realized using the equation (1).

Let the proposals given by agents are equal to 38 for $A_1$, 29 for $A_2$ and 41 for $A_3$. We will consider the following values at the period $T$: The agents' $SF$ is equal to 0.5 for $A_1$, 0.333 for $A_2$ and 0.166 for $A_3$. The agents' $DAB$ is equal to 1 for $A_1$, 0.75 for $A_2$ and 1 for $A_3$. From the algorithm given previously we get: $FV_{initiator}$=1. As the number of potential candidates is equal to 3 thus, FV for $A_1$ and $A_3$=1/3-1.

According to these values, for agent $A_2$ the offer of agent $A_1$ becomes:

$$\left[CI_{expected} \times FV / SF_{A1} \times Bid_{A1}^{evaluation} \times DAB_{A1}\right]_{A1}$$

$$\left[0.8 \times 0.5 / 0.5 \times 1/38 \times 1\right]_{A1} = 0.021$$

And the offer of agent $A_3$ becomes:

$$\left[CI_{expected} \times FV / SF_{A3} \times Bid_{A3}^{evaluation} \times DAB_{A3}\right]_{A3}$$

$$\left[0.8 \times 0.5 / 0.166 \times 1/41 \times 1\right]_{A3} = 0.058$$

And its own offer becomes:

$$\left[(1-CI_{expected}) \times FV / SF_{Ai} \times Bid_{Ai}^{evaluation} \times DAB_{Ai}\right]_{Initiator}$$

$$\left[0.2 \times 1/0.333 \times 1/29 \times 0.75\right]_{A3} = 0.015$$

From the prior equations, we distinguish that the agent $A_2$ will favor the cooperation with $A_3$ due to its results, which is superior to its own result as well as the result of $A_3$.

We can distinguish that the evaluation of the initiator's offer is realized by using the equation (3), whereas we used the equation (1) for the evaluation of other agents' offers. The first parameter in this equation (1) is $1\text{-}CI_{expected} = 0.2$, that disfavored the initiator's offer. This last agent was also disfavored because of its $DAB$ value. Agents $A_1$ and $A_3$ have a $DAB$ equal to $1$, which signify that they fully respected their engagement (time, cost, etc.).

Although, the agents $A_1$ and $A_3$ have presented the same offers as in the first example, the agent $A_2$ have preferred the cooperation with $A_3$ instead of $A_1$. This choice can be justified by the low value of $SF_{A3}$ comparing to the other candidates. In a general manner, such a choice aims to improve the agents $SF$.

## V. Experiments & Results

The previous sections aimed essentially to introduce the idea of the cooperation indicator which allows quantifying the cooperation. They also illustrated the principal of the choice mechanism which allows the initiator agent to choose among the different competent agents, or more precisely, among the potential candidates while respecting the cooperation indicator expected.

In this section, we will present some of realized experiments as well as the obtained results:

### A. First experiment

We start, firstly, by sending a hundred of requests to be realised to a specific agent. Consequently, this last will be considered as the initiator agent for these given requests. Therefore, the initiator agent should start the cooperative problem solving process in order to find the competent agents and then choose among them the most appropriated agent for the requested task realisation, according to the choice mechanism detailed previously (section III).

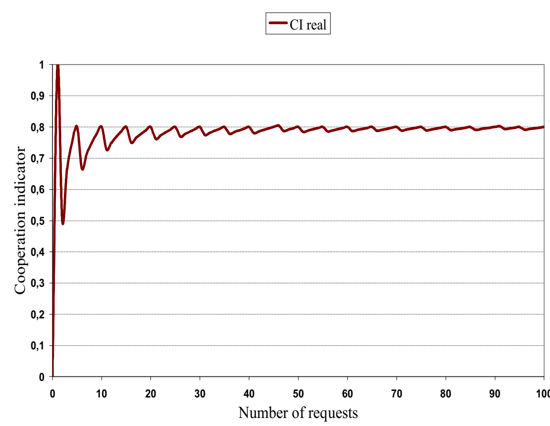


Figure 3 – Obtained results for CI expected = 80% (100 requests case).

We can distinguish from the earlier figure that the convergence towards the objective, which is equal to 80% of cooperation in this case, becomes progressively evident until almost be reached.

The previous experiment was repeated while increasing the number of requests sent to the imitator agent until reaching 300. The following figure demonstrates the obtained results that confirm the rapid progress until reaching the predefined objective (80%).

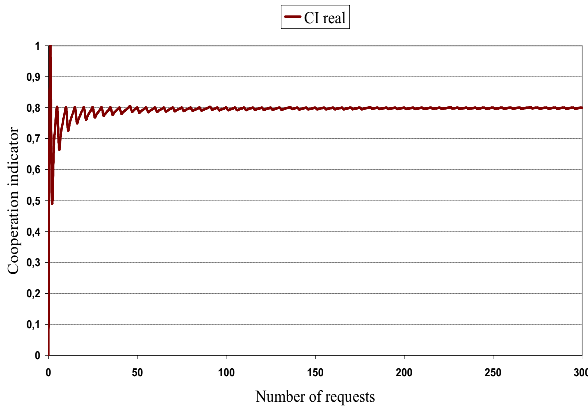


Figure 4 - Obtained results for CI expected = 80% (300 requests case).

### B. *Second experiment*

In order to demonstrate the flexibility of our framework and its capability to answer the two extremities of multi-agent systems, explained in detailed manner previously, we have realized a new experiment.

In this experiment, the value of cooperation indicator expected by the user is changed to be equal to 20% instead of 80% in the previous experiments.

Contrarily to the previous experiments, the new value of CI aims essentially to indicate the secondary importance of cooperation among agents.

Moreover, we keep increasing the number of requests sent to the imitator agent until reaching 500. The coming figure illustrate the obtained results for this experiment.

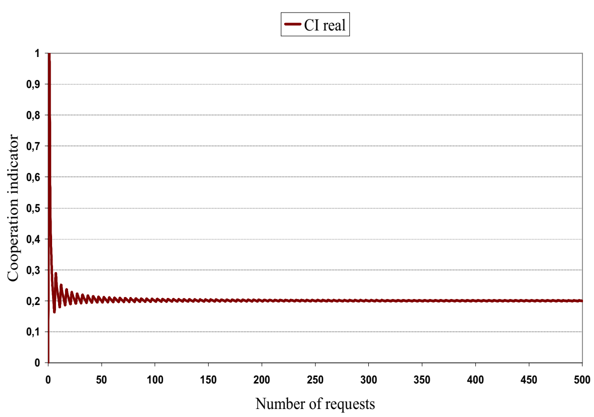


Figure 5 - Obtained results for CI expected = 20% (500 requests case).

## VI. Discussion

The prior mechanism is a flexible mechanism, which is capable to treat passive and active cooperation. This mechanism allows the user to determine the percentage of expected cooperation. This can be realized by the *CI* given by the user at the initialization. In the figure below, the discontinuous arrow represents the *CI* expected and given by the user, which is equal to 0.5.

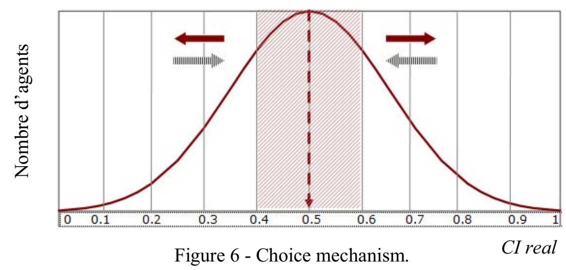


Figure 6 - Choice mechanism.

Our model allows also the user to precise the *tolerance value expected*. All the agents situated within the *tolerance zone* (defined by *TV*) will be considered as satisfying from the user's point of view. Whereas the value of *TV* given earlier by the user is equal to 20%, that means 10% less than 0.5 (the *CI*'s value) and 10% more than 0.5. This *TV* is represented in the area situated between the values 0.4 and 0.6 in the previous figure.

The curved line in figure 3 represents the desired performance of the agents' cooperation. Thus, we aim to maintain the maximum number of agents within the *tolerance zone* (zone situated between 0.4 and 0.6). All the agents having a performance situated out of the *tolerance zone* will be oriented towards this zone.

One of the advantages of our mechanism is to give an equal opportunity for the different agents. Consequently, it favours the cooperation among agents. This leads to improve the global performance.

## VII. Conclusions

In this paper, we have presented a hybrid hierarchical model for cooperative problem solving, which describes all aspects of the cooperation process, from recognition of the potential for cooperation through to execution. A measure of cooperation, the Cooperation Index (CI) has been developed and it indicates the level of desired and expected cooperation among the different agents in the MAS. This provides the necessary framework to model both passive and active cooperation, depending on the user's needs. An illustrative example as well as a number of realized experiments and obtained results was given all through this paper.

## VIII. References


[1] H. S. Nwana, L. C. Lee, and N. R. Jennings, "Co-ordination in software agent systems," The British Telecom Technical Journal, vol. 14(4), pp. 79-88, October 1996.

[2] J. Ferber, "Coopération réactive et émergence," in Intellectica, 1994, pp. 19-52.

[3] S. Cammarata, D. McArhur, and R. Steeb, "Strategies of Cooperation in Distributed Problem Solving," in Proceedings of the Eighth International Joint Conference on Artificial Intelligence (IJCAI'83), Karlsruhe, Germany, 1983, pp. 767-770.

[4] Y. Demazeau and J.-P. Müller, editors. Decentralized AI 2 - Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90). Elsevier Science Publishers, 1991.

[5] E. H. Durfee, V. R. Lesser, and D. D. Corkill. Cooperative Distributed Problem Solving. Vol. 4, page 83-147,Addison Wesley,1989.

[6] P. Leroux, "Conception et réalisation d'un système coopératif d'apprentissage. Etude d'une double coopération : maître / ordinateur et ordinateur / groupe d'apprenants." vol. Thèse de Doctorat Le Mans, France: Université du Maine, 1995.
[7] E. H. Durfee and V. Lesser. Negotiating task decomposition and allocation using partial global planning. In L. Gasser and M. Huhns, editors, Distributed Artificial Intelligence Volume II, pages 229-244. Morgan Kaufmann: San Mateo, CA, 1989.
[8] J. E. Doran, S. Franklin, N. R. Jennings and T. J. Norman, “On Cooperation in Multi-Agent Systems”, The Knowledge Engineering Review, 12(3): 309-314, 1997.
[9] J. R. Galliers, A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict. PhD thesis, Open University, UK, 1988.
[10] J. R. Galliers, Modeling autonomous belief revision in dialogue. In Yves Demazeau et Pierre Müller (éd.) Decentralized Artifi-cial Intelligence 2: Proceedings of the Second European Workshop on Autonomous Agents in a Multi-Agents World, Elsevier Science Pub./North Holland, 1991.
[11] R. Conte, M. Miceli, and C. Castelfranchi. Limits and levels of cooperation. In Y. Demazeau and J.-P. Müller, editors, Decentralized AI 2 – Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90), pages 147-160. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1991.
[12] B. Moulin and B. Chaib-draa, “An overview of distributed artificial intelligence” In G. M. P. O'Hare and N. R. Jennings, editors, Foundations of Distributed AI, pages 3-54. John Wiley & Sons: Chichester, England, 1996.
[13] W. Alshabi, M. Itmi, S. Ramaswamy, and H. Abdulrab, "Un mécanisme de coopération dans une organisation hiérarchique," in 7ème Conférence Internationale de MOdélisation et SIMulation - MOSIM’08 - « Modélisation, Optimisation et Simulation des Systèmes : Communication, Coopération et Coordination.». Paris- France, 2008, pp. 1506-1512.
[14] A. H. Bond, “Distributed Decision Making in Organisation” IEEE Transactions on SMC Conference, November 1990.
[15] Alshabi W., Ramaswamy S., Itmi M., and Abdulrab H., "Coordination, Cooperation and Conflict Resolution in Multi-Agent Systems," in Intnl Joint Conferences on Computer, Information, and Systems Sciences, and Engineering, (IEEE CISSE'06), 2006, p. 6p.
[16] D. Bell and J. Grimson, “Distributed Database Systems”, Addison-Wesley, 1992.
[17] N. Bensaid and P. Mathieu, “A Framework for Cooperation in Hierarchical Multi-Agent Systems”, Mathematical Modeling and Scientific Computing. Vol 8. 1997.
[18] N. Bensaid and P. Mathieu, “An Autonomous Agent System to Simulate a Set of Robots Exploring a Labyrinth”, Proceedings of the 11th Intl. FLAIRS Conference, FLAIRS'98. AAAI Press. 1998.
[19] W. Al-shabi, S. Ramaswamy, M. Itmi, and H. Abdulrab, "A cooperative problem-solving process in hierarchical organization " in Summer Computer Simulation Conference, (SCSC'07), July 15 - 18, 2007, p. 6. San Diego, CA, USA.
[20] W. Alshabi, M. Itmi, S. Ramaswamy, and H. Abdulrab, "Towards a social responsible agents in hybrid organization," in *International Simulation Multi-conference (ISMc'08) : Grand Challenges in Modeling & Simulation (GCMS’08)*, Edinburgh, Scotland, 2008.
[21] N. E. Bensaid and P. Mathieu, "MAGIQUE Un Modèle d'Architecture Multi-Agents Hiérarchique", 1997.
[22] N.E. Bensaid and P. Mathieu, "A hybrid and hierarchical multi-agent architecture model," in Proceedings of the Second International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM’97), London- United Kingdom, 21st-23rd April 1997, pp. 145–155.
[23] N. E. Bensaid and P. Mathieu, "A Hybrid Archtecture for Hierarchical Agents," in *Proceedings of the International Conference on Computationnal Intelligence and Multimedia Applications (ICCIMA'97)*, Gold-Coast, Australia, 1997, pp. 91-95.
[24] J.-P. Briot and Y. Demazeau. In *Principes et architecture des systèmes multi-agents*, Hermès Sience Publications ed, J.-P. Briot and Y. Demazeau, Eds.: Lavoisier, 2001.
[25] Li Minqiang Li Changhong, Kou Jisong, Cooperation Structure of Multi-agent and Algorithms. In proceedings of the 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS’02), 2002.
[26] M. Luck M. d'Inverno, and M. Wooldridge. Cooperation Structures. In Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97), Nagoya, Japan, August 1997.
[27] J. Ferber, Les systèmes multi-agents : vers une intelligence collective, InterEditions ed. Paris, 1995.
[28] J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. London: Addison Wesley Professional, 1999.