Towards Building Agreement Spaces Using Consensus Networks*

M. Rebollo, A. Palomares, and C. Carrascosa

Universitat Politècnica de València,
Camino de Vera s/n 46022 Valencia (Spain)
{mrebollo, apalomares, carrasco}@dsic.upv.es

Abstract. If a set of agents tries to reach an agreement, there will exist an Agreement Space that models the maximum valid space according to the individual constraints of each agent, regarding the terms of the agreement. Previous approaches to Agreement Spaces model them by means of a Counsellor Agent that will help to agents trying to obtain an agreement knowing all the possible terms in that agreement, and monitoring such Agreement Space as it is being built. But, what happens when there is no Counsellor Agent? Is there any way of modeling such space in a distributed way by the different agents trying to achieve such agreement? This paper presents a possible solution to that question using a Consensus Network to build such Agreement Space, allowing to know if there are possibilities to achieve an agreement or not.

1 Introduction

Nowadays, there is a growing interest in the Multi-Agent Systems (MAS) area regarding concepts about cooperation such as *Consensus*, *Commitments*, *Contracts*, or *Agreements*. All these concepts are related, but it is necessary to clarify in some way such relation. A *Consensus* is the negotiation of a value in one term, including the final decision where all the negotiation parts agree in the value. A *Commitment* or, more concretely, a *Social Commitment* [3] is the relation between at least two agents, where one agent is committed to do something to other agents. An *Agreement* is the definition of a working context for two or more entities [2], and as such, it is a wider and more complex process that subsumes the previous concepts in a whole process that includes:

- The definition of a common ontology: it is necessary to assure that there exists a common ontology to allow an agreement.
- Deciding the terms the negotiation is going to take place about: such terms will be part of the common ontology.
- To reach an agreement, that can be formed by several related consensus in different terms. The result of such can include a set of social commitments.

* AT2012, 15-16 October 2012, Dubrovnik, Croatia.

When a set of agents try to reach an agreement, if such agreement is possible, there will exist an Agreement Space [2] modeling the maximum valid space according to the individual constraints of each agent, where each term of the possible agreement is reflected in a different dimension of such space.

The internal structure of many biological, social and economic systems displays complex topological properties such as clustering, a scale-free degree distribution and the presence of degree correlations, which are not reproduced by simple random graph models [5]. For instance, network properties of the global economic system have been documented in many studies, which have revealed scale-free weighted topology at various levels, increasing connectivity over time, and implications for economic development [4, 12].

The objective of this paper is to present a way of modeling such space in a distributed way by the different agents that are trying to achieve such agreement by means of a Consensus Network, allowing them to know if there are possibilities to achieve an agreement or not. Just to show the validity of the present approach, a market-based case of study is presented, with some interesting results when applying this proposal.

The rest of the paper is structured in the following way: the next section presents some related work, regarding agreements and consensus networks. After that, the modeling of Agreement Spaces by Consensus Networks is presented. The following section presents a case of study regarding right–based markets. Last, some conclusions are commented.

2 Related Work

Looking for a common space where agents can interact can be analyzed as a constraint satisfaction problem. Different strategies to solve this kind of problems have been widely studied in the last years. Two of these techniques are commented in the rest of this section.

2.1 DCOP

One of the most promising ones are Distributed Constraint Optimization Problems (DCOP) [7]. DCOPs are a model for representing multi-agent systems in which agents cooperate to optimize a global objective. A DCOP is a formalism that captures the rewards and costs of local interactions in a multiagent system (MAS) where each agent chooses a set of individual actions. In a DCOP each agent receives knowledge about all relations that involve its variable(s).

As [13] detail, there are two main types of complete algorithms to solve DCOPs: search and dynamic programming. Search algorithms, like ADOPT (or extensions such as IDB-ADOPT), require an exponential number of linear-size messages. Dynamic programming algorithms, like the distributed pseudo-tree optimization procedure (DPOP) and its extensions, only require a linear number of messages, but their complexity lies on the message size, which may be very large. On the other hand, as [11] details, there are some limitation in the way DCOP algorithms approaches the problem. Firstly, they assume that the environment is deterministic and fully-observable, meaning that agents have complete information about the utility of the outcomes of their possible decisions. Thus, agents' utilities do not change while the problem is being solved. Moreover, agents' actions are only applied once the problem is solved. Furthermore, the set of agents in the system is constant, not allowing the entrance or exit of the system.

2.2 Consensus Networks

The theoretical framework for solving consensus problems in dynamic networks of agents was formally introduced by Olfati-Saber and Murray [10] [9]. The interaction topology of the agents is represented using directed graphs and *consensus* means to reach an agreement regarding a certain quantity of interest that depends on the state of all agents in the network. This value represents the variable of interest in our problem *(agreement term)*, and might be for example a physical quantity, a control parameter or a price among others.

Let G be a directed graph of order n with the set of entities E as nodes and weighted adjacency matrix $A = [a_{ij}]$. Let (G, X) be the state of a network with value X and topology G, where $X = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$, where x_i is a real value associated with the node E_i. The value of a node might represent physical quantities measured in a distributed network of sensors (temperatures, voltages,...), or also quantities of interest in a network of buyer's and seller's in the market (prices, rights, quality,...). A network is a dynamic system if (G, X) evolves in time. A consensus algorithm is an interaction rule that specifies the information exchange between the agents and all of its neighbors on the network in order to reach the agreement. Consensus of complete network is reached if and only if $x_i = x_j \forall i, j$. Distributed solutions of consensus problems in which no node is connected to all nodes are especially interesting. The most commonly used consensus protocols are Average, Maximum and Minimum because the have broad applications in distributed decision-making multi-agent systems.

These authors have demonstrated that a convergent and distributed consensus algorithm in discrete-time can be written as follows:

$$x_i(k+1) = x_i(k) + \varepsilon \sum_{j \in N_i} (a_{ij}(x_j(k) - x_i(k))), \tag{1}$$

where N_i denotes the set formed by all nodes connected to the node i (neighbors of i). The collective dynamics of the network for this algorithm can be written as

$$X(k+1) = P_x(k) \tag{2}$$

where P is the Perron matrix of a graph with parameter ε, defined as $P = I - \varepsilon\,\mathcal{L}$, with I is the Identity matrix, ε is the step size, $\varepsilon > 0$, and $\mathcal{L}$ is the Laplacian matrix, $\mathcal{L} = D - A$, that is the difference of the degree matrix D and the adjacency matrix A of the graph. The algorithm converges to the average of the initial values of the state of each agent and allows computing the average for very large networks via local communication with their neighbors on a graph.

Solving consensus is a cooperative task because if a single agent decides not to cooperate and keep its state unchanged all others asymptotically agree with them. However, if there are multiple non cooperative agents then no consensus can be asymptotically reached. When an agent does not cooperate with the rest its links in the network might be disconnected in order all others will asymptotically agree. In this case it is impossible for all agents to reach an agreement but is possible to reach an agreement for the rest of the agents.

3 Modeling Agreement Spaces With Consensus Networks

In this section, some definitions regarding Agreement Spaces and what is an agreement process are presented, along with how can be used a consensus network to quickly determine if it will be possible to arrive to an agreement. That is, if there exist an agreement space between the agents involved.

3.1 Agreement Spaces

In order to deal with *Agreements* [2], some definitions are needed:

Definition 1 (Agreement): is defined as $Agr = (E, Cx)$, where: (i) $E = \{E_1, E_2, \ldots E_n\}$ is a set of entities participating in the agreement Agr, so that $\forall E_i, i = 1, \ldots n, \exists! O_i = \{o_{ij}\}, j = 1, \ldots m$, the ontology known by E_i; and (ii) $Cx = \{(cx_o, cx_o^I) | cx_o \in \bigcup_i O_i, cx_o^I \subseteq D_o\}$, where D_o is the domain of the ontology term cx_o. Thus, this context is formed by a set of ontological terms cx_o with its corresponding set of valid instances cx_o^I that have been agreed by E.

Definition 2 (Agreement Space of an agreement Agr –$AS(Agr)$–): That is, this space will be defined by the features the different entities E_i making the agreement are going to negotiate (Cx), each one of such features defining a dimension in this space ($\forall i : d_i \in dim(E_i, Agr)$).

In order to be possible an agreement Agr, for each entity E_i participating in it there will exist at least one other participating entity E_j so that $dim(E_i, Agr) \cap dim(E_j, Agr) \neq 0$.

Definition 3 (Agreement Process): As it has been stated before, an agreement has associated a wide and complex process comprising several different phases:

1. Calculating the $ADU(Agr)$: that is, to calculate the common ontology shared by all the agents interested in the agreement. Associated to this $ADU(Agr)$ will exist the corresponding $ADS(Agr)$.
2. Reaching an agreement (defining the agreement context Cx): It comprises the negotiation process between two or more entities (belonging to E) to reach an agreement. In fact, it is decided in three levels:
   (a) Definition of the AS Dimensions: a decision must be taken about what are the concepts around which such agreement is going to be related, that is, $Cx \subseteq ADU(Agr)$. Or, if the concepts are considered as dimensions, decide the dimensions in which the agreement will be expressed, that

is, define the Agreement Space dimensions as a subset of the existing $ADS(Agr)$ dimensions.
   (b) Definition of the own AS: it is checked if it is possible to reach an agreement by calculating if there exists an AS, identifying both the concrete boundaries of such space and the entities involved, because at this level, entities may abandon the process if they disagree on the specific space boundaries.
   (c) Reaching the concrete agreement: the specific terms of such agreement (the values or intervals for the concepts in Cx) must be fixed, that is, the satisfying values or intervals for each one of the different dimensions comprising the $AS(Agr)$.
3. Agreement execution: In this phase each entity must fulfill the accomplished agreement executing the needed actions or calculus according to the context defined by such agreement. This execution could not even imply any kind of additional coordination.

In this way, the outcome of the first phase of the agreement will be the definition of this Agreement Space, fixing the satisfying values for each one of the different dimensions comprising this space. The second phase of the agreement will be to carry out the execution of the agreement taking into account that it has to be carried out inside the previously defined Agreement Space.

3.2 Agreement Process

This section presents the application of consensus networks to assure, in a distributed fashion, that there are possibilities to reach an agreement between the involved agents, that is, to assure that there is an Agreement Space. In this way, following the *Agreement Process* presented in Definition 3, this Consensus Network will be used to check that it is possible to carry out b and c levels of the second phase of such process. That is, in the phase where the agreement is reached, after the dimensions of the Agreement Space has been fixed, the Consensus Network will calculate that there exists an Agreement Space, being possible to reach an agreement calculating the specific terms of the different dimensions (values or intervals).

The process of Agreement Space's boundaries calculation may be defined in two different ways:

1. Unlimited: it continues till the boundaries of the AS are defined or it is not possible to define an AS between any two different entities. In this case, both 2.b and 2.c levels of the agreement process are carried out by the Consensus Network.
2. Limited: it is possible to define different limitations to the process:
   (a) A deadline.
   (b) A minimum number of participant entities, so during the process when too much entities abandon it, the process will end saying that it is not possible to define an Agreement Space.

(c) A maximum number of interactions or messages.
(d) A minimum length of an AS dimension.

In this way, the Consensus Network is only in charge of the level 2.b of the Agreement Process.

3.3 Consensus Network

Based on the basic model of consensus network of Section 2, a complete model useful to define Agreement Spaces has been defined.

Let (G, X) be the basic model, where G is a network and X the values of the nodes for the consensus variable x. Both, the network and the values, evolves with time, so we define the consensus problem as $(G(t), X(t))$: a consensus network with switching topology, meaning that the structure of the network change along with the agreement process. In the general case, this switching can be model through the adjacency matrix $A = [a_{ij}]$. In out case, this matrix is divided in two components: the first one refers to the existence of a link between agents i and j (as usual). The other component is a weight that represents the strength of the link between both agents. This weight can be use to model many different things depending on the application domain: centrality measures, importance, reputation, trust, and so on. Le $W = [\omega_{ij}]$ this weight matrix. Then,the component that can change with the time can be A, W or both.

$$x_i(k+1) = x_i(k) + \varepsilon \sum_{j \in N_i} ((\omega_{ij}(k) a_{ij}(k))(x_j(k) - x_i(k)), \tag{3}$$

In the construction of the Agreement Space, only some changes in the topology are allowed. Specifically, agents can leave the agreement process if it is out of the agent's bounds. No agent can be incorporate to an agreement in process. We do not consider the possibility of creating new groups that evolve independently of the original group either.[1]

Each agent has its own utility function to determine if it stays in the agreement process or if it leaves it. Then, a permanence function can be defined as $u_i(x)$ for each agent i.

$$a_{ij}(k+1) = \begin{cases} 1 \text{ if } u_i(x_i(k)) \geq 0 \\ 0 \text{ otherwise} \end{cases} \tag{4}$$

So the switching nature of the network can be captured and included in the model as a function of the current calculated value of the consensus dimension and the expected utility value for the agent. Including this utility function u_i, the resulting consensus problem can be modeled as $(G(t), X(t), U)$, where

- $G(t)$ is a network with switching topology,

[1] This case can be considered as two independent consensus problems and the same model can be applied to each isolated group, without considering any kind of transference of agents or information among them.

– $X(t) = \{x_1(t), \ldots, x_n(t)\}^T$ is the vector of values for the agreement dimension (consensus variable), and
– $U = \{u_1(x), \ldots, u_n(x)\}^T$ is a vector of the utility functions for each agent.

The dynamic of the agreement is defined by Equations 3 and 4.

If the network contains one connected component (see Figure 1) then the convergence of the method is ensured. The result is an agreement about the value of the negotiated parameter. But it is possible that, if one or more agents leave the agreement process, the network is segregated in two (or possibly more) independent networks, each one of them eventually reaching their own agreement. For example, Figure 2 shows the effect of a node that leaves the agreement process in a very early stage.

4 Case of Study: Right-Based Markets

4.1 Agreement-Based Markets

Networks allow to model interactions when certain agents are not aware of each other by informational, regulatory or physical reasons. In particular this paper is focused in some kind of markets in which price, supply and demand should be previously balanced (agreement space) before negotiations begin. There exist markets in which some regulations are required and also the topology of the network induced by the internal relationships must be considered.

In order to verify that the proposed framework is adequate to generate the required initial data sets different repetitions are randomly generated using the previously specified probability distributions of agents, rights, optimal rights and prices. This framework of data generation is more similar to Montecarlo models.

Suppose rights price must be fixed in the market (a constraint) and some previous agreements, acceptable by the maximum number possible of agents, are required in order to establish this value. The average of the prices P_i is probably one the most acceptable agreement value for the price, and also can be calculated asymptotically using consensus algorithms. Dynamics of agents are defined by the following equations:

$$P_i^S(k+1) = P_i^S(k) + \varepsilon \sum_{j \in N_i} (B_j(P_j^B(k) - P_i^S(k) + C_i(k))), \tag{5}$$

$$P_i^B(k+1) = P_i^B(k) + \varepsilon \sum_{j \in N_i} (S_j(P_j^S(k) - P_i^B(k) + C_i(k))) \tag{6}$$

where k is the discrete time and the index S and B denotes seller and buyers respectively. The added term $C_i(k)$ is proportional to rights bought and sold by agents in each iteration, and is calculated as follows:

$$C_i(k) = \delta \cdot \frac{\sum_{j \in N_i} B_j}{\sum_{j \in N_i} S_j} \tag{7}$$

where $\delta > 0$.

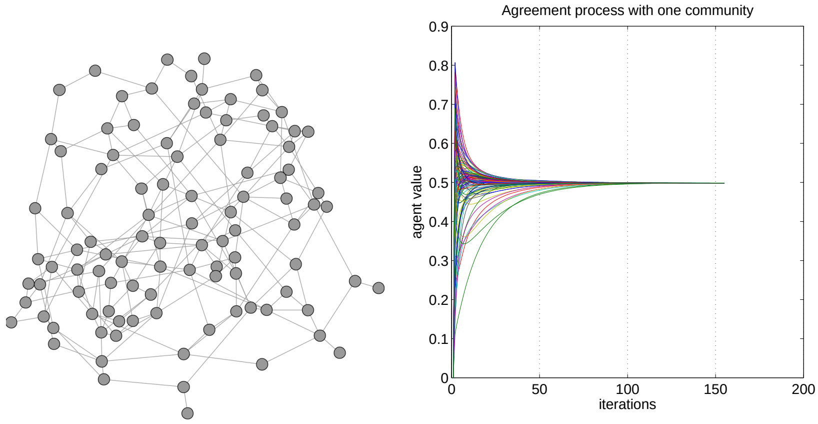

Fig. 1. Agreement process when the complete group is connected. All the agents reach a consensus about the value exchanging information with their neighbors. They do not know the structure of the network nor any other global information.

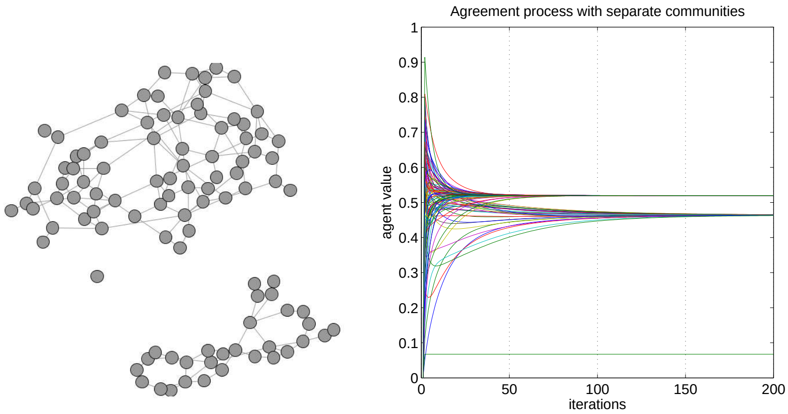

Fig. 2. Agreement process when the group is divided into 3 independent communities. Each group converges to a different value

Seller agents adapt iteratively prices in order to converge to the mean price of the connected buyer agents and simultaneously buyer agents adapt their prices in order to converge to the mean price of the connected seller agents. In this iterative process non cooperative agents are disconnected of the network in order all others agents to have the possibility of asymptotically agree. The algorithm converges and stops when prices converge to the consensus value the mean prices of buyers and sellers are approximately equal.

In order to include the topology of the networks, the scale-free network models (SFN) have been considered [1] [6]. A SFN is a network whose degree distribution (fraction of nodes having k connections to other nodes) follows a power law: $P(k) = k^{-\gamma}$.

The scaling exponent γ is a constant whose value is typically in the range $2 < \gamma < 3$. A framework which enables generate scale-free networks of buyer and seller agents has been developed in Matlab. The purpose of this framework is not simulating real and complex markets but only analyze the possible application of consensus algorithms in these problems. The Lev Muchnik's package [8] has been used to generate the random scale-free graphs in the experiments.

The **number of agents** N (number of nodes in the network) and the **scaling exponent** γ are input parameters in the experiments. In some experiments the networks considered are "all connected" and then $\gamma = 0$.

The networks generated for this paper are undirected (edges have no orientation) and adjacency matrix is symmetric. Once this network has been generated the rights must be generated and distributed between the agents. The **Number of Rights** R is an input parameter of experiments:

$$R = \sum_{i=1,\dots,N} R_i \tag{8}$$

where R_i are the number of rights of agent i. Rights are randomly generated and distributed between the N agents. Uniform distribution of rights is used in the simulations.

In order to simulate the roles of the agents in the market (buyers, seller and agents not interested in the market) the "optimal rights" of an agent $\widetilde{R}_i$ are defined as the number of rights this agent would have. The comparison between $\widetilde{R}_i$ and R_i allows to establish the role of agent i in the market:

- If $\widetilde{R}_i = R_i$ then the agent i is not interested in the market.
- If $\widetilde{R}_i > R_i$ then the agent i would buy $(\widetilde{R}_i - R_i)$ rights (Buyer).
- If $\widetilde{R}_i < R_i$ then the agent i would sell $(R_i - \widetilde{R}_i)$ rights (Seller).

The **Number of Optimal Rights** $\widetilde{R}$ is an input parameter of experiments:

$$\widetilde{R} = \sum_{i=1,\dots,N} \widetilde{R}_i \tag{9}$$

The optimal rights $\widetilde{R}$ are also randomly generated and uniformly distributed among the N agents. Similar values of Rights and Optimal Rights represents markets where supply and demand is in equilibrium. In this paper only these "equilibrated markets" ($R \simeq \widetilde{R}$) are considered in the experiments. Once the agent roles have been established only buyers and sellers are considered in order to estimate the agreement space. Agents not interested in the market are deleted of the network. For this reason some buyers and sellers could be disconnected of the network when an agent is deleted. Disconnected agents are also deleted of the network iteratively until all agents are connected with others. The result is a bipartite, undirected graph, where buyers are connected with sellers.[2]

[2] The general case where agents of any type are connected each other has no effect on the behavior of the algorithm and barely affects to the convergence time of the method.

The next step is generate the initial prices each agent would pay (if is a buyer) P_i^B or receive (if is a seller) P_i^S in order to buy or sell rights in this market. These prices are randomly generated using normal distributions. The **mean and standard deviation of the buyer and seller prices distributions** are input parameters in experiments.

In our framework agents are allowed to leave the network when the consensus price is out of their acceptable boundaries. In order to simulate price boundaries an additional parameter is included in the model, the limit price for agent i (LP_i):

$$LP_i = P_i * (1 \pm F_i) \tag{10}$$

Where F_i values are randomly generated in the interval $[0, 0.5]$ using an uniform distribution and the sign (+ or -) depends if the agent is a buyer or a seller. Boundaries prices for agents are included in the interval $[P_i, LP_i]$.

Once initial data sets are generated then the consensus protocol (see Eq. 6) can be computed in order to obtain when the consensus price is possible. An additional restriction of the proposed simulation framework must be clarified: agents must necessarily adapt their prices in each iteration of the consensus algorithm because this algorithm only converges when all agents collaborate in the process.

4.2 Experimental Design

The initial parameters of all experiments presented in this paper, except the scaling exponent γ, are the same those discussed previously (see Table 1). These experiments show that in general, the number of initial buyers and sellers, and the corresponding selling and buying rights are similar (supply and demand is in equilibrium) and also that initial prices are distributed around the initial mean values (parameters of the simulation). A more detailed analysis of these results shows that seller and buyer price distributions are not normal but this factor is not important in our experiments, the only one important is to be able to generate data sets adequate for our experiments.

Different conditions and parameters (see tables 2 and 3) are combined and analyzed in the proposed experiments:

- ε and δ (learning parameters) of consensus algorithm used as $\lambda = \delta * \varepsilon$.
- Epoch: number of iterations: 50, 100 and 200.
- The scaling exponent γ of the network distribution: $\gamma = 0.0$ (complete network) and $\gamma = 2.0$ (scale free network).
- β: agents are allowed to leave the network when $\beta = 1$.

Experiments Set 1 The results obtained in these experiments (1A and 1B) are summarized in Table 4. When $\beta = 0$ agents are not allowed to leave the network and therefore the number of consensus agents and rights are the initial ones (if a consensus is reached, it has to be by all the agents beginning the process), and the "consensus price" is the mean of all initial prices. When $\beta = 1$ agents are

Parameter	Value
agents N	1000
rights R	2000
optimal rights $\widetilde{R}$	2000
buyer price P^B_{mean}	10
buyer price σ^B_{mean}	2
seller price P^S_{mean}	10
seller price σ^S_{mean}	2

Table 1. Initial conditions

γ	0;2
$\varepsilon(10^{-2})$	0; 1; 2; 3
$\lambda = \delta * \varepsilon$	0;2;,4;6;8;10
Epochs	50; 100; 200
β	0; 1

Table 2. Experiment Parameters (exp. 1 & 2)

EXPERIMENT	#Runs	γ	β	λ
Init. Cond. (t = 0)	60	0	0	0
Exp. 1-A	450*2	0.0	(0,1)	(0; 2; 4; 6; 8; 10)
Exp. 1-B	90*2	0.0	1	(0, 10)
Exp. 2-A	450*2	2.0	(0,1)	(0; 2; 4; 6; 8; 10)
Exp. 2-B	90*2	2.0	1	(0, 10)

Table 3. Experiment design

allowed to leave the network if price (k) is out of their boundaries, and therefore the number of "consensus agents and rights" may be lower than the ones that initiated the process. Particular consensus value depends of initial simulation parameters.

When $\lambda = 0$, the additional term in the equation of consensus (see Eq. 7) is not considered, and simulations show that final supply and demand rights are unbalanced. The lowest value of rights limits the "possible agreement space". However when $\lambda > 0$ this additional term tends to balance supply and demand rights, and the obtained equilibrium value is higher than the lowest value obtained when $\lambda = 0$. Therefore the "possible agreement space" is better.

Experiments Set 2 The basic difference with experiment 1 is that the input scaling exponent $\gamma \neq 0$ and, therefore, scale free networks are considered. The results obtained in these experiments (2A and 2B) are summarized in Table 4. The proposed framework excludes explicitly not connected agents before consensus algorithm starts, and for this reason the initial date set (number of agents with the roles of seller or buyer and their corresponding rights) is minor than previous experiment (all connected network). These results are very similar to previously obtained when $\gamma = 0$ (all connected networks) except for the slightly higher variances see Table 5). However a detailed analyses of this experiments show than the simultaneous combination of scale free networks ($\gamma > 0$) with the

Experiment		P^B_{mean}	P^S_{mean}	R^B_{mean}	R^B_{mean}	B	S
Init.	MEAN	10.0	12.0	691	692	392	392
Cond.	STD	0.1	0.1	17	21	11	10
1-A	MEAN	10.9	11.1	686	688	390	391
$\beta = 0$	STD	0.4	0.4	20	22	9	10
1-A	MEAN	10.7	10.7	524	543	298	309
$\beta = 1$	STD	0.2	0.3	31	39	17	22
1-B	MEAN	11.1	11.2	492	577	278	328
$\lambda = 0$	STD	0.1	0.1	44	34	23	16
1-B	MEAN	10.6	10.7	528	538	298	306
$\lambda = 10$	STD	0.2	0.2	32	41	18	21
2-B	MEAN	10.6	11.1	443	489	251	277
$\lambda = 0$	STD	0.3	0.2	42	63	24	35
2-B	MEAN	10.4	10.8	476	480	272	273
$\lambda = 10$	STD	0.7	0.4	39	42	21	23

Table 4. Experiment configuration for the different cases

Variance (%)	R^B Variance (%)	R^S Variance (%)
Exp 2-A	6.1%	7.5%
Exp 2-B	8.2%	8.7%

Table 5. Variances

possibility agents can leave the network when the price is out of their boundaries ($\beta = 1$) sometimes does not converge.

4.3 Results

Some simulations have been chosen to show the behavior of the network to achieve a decision. The algorithm converges and stops when the mean prices of buyers and sellers are approximately equal.

- The simplest case (Figure 3, row 1) is a complete graph where all the agents are forced to stay and reach a consensus value. This experiment shows that the number of exchanged rights remains constant (because no agents leave the system), but there is a difference between the desired rights and the offered rights. The modification of the basic algorithm solves this problem by introducing a second factor, represented by the $C_i(k)$ term and analyzed in the λ parameter.
- In the experiments 1A and 1B (Figure 3, row 2), besides the adjustment between the involved rights, agents are allowed to leave the system if the terms of the negotiation (the consensus value for the price of the rights) is out of its bounds. In this case, the number of rights decreases while the consensus evolves. Nevertheless, the offer and the demand are biased (see how the lower lines matches in the upper graphic).
- Finally, in the experiments 2A and 2B (Figure 3, row 3) a scale free network is used instead of the full connected graph. Furthermore, the initial price

distribution is different too. In this case, the network does not converge to one unique value. The topology just affect to the speed of the convergence. But when an agent leaves the consensus it can split the network apart. This fact prevents the hole group to reach a common consensus value, because each isolated group converges to their own consensus value and they never receive the values agreed by the other group. The result is that, instead a unique consensus value, there are as many as isolated groups. This can be seen in Figure 3, where the prices do not converge. Anyway, the values of the prices stabilizes. In this case, instead of a final agreed value, a range is found. This fact allows to identify which agents are interested in the agreement and the range of prices. After this process, some other techniques can be used to find an agreement. Then, the consensus network can be used at least to identify the subset of participants interested in reaching and agreement and the boundaries of such an agreement.

5 Conclusions

This paper shows the application of consensus networks to model agreement spaces for virtual organizations in a distributed and self-organized fashion. The results obtained show that this approach seems to be valid and it has been applied to a rights-based markets (such as water, electrical or stocks) in order to check its viability in real-world problems.

Consensus networks are used to check the existence of an Agreement Space among a network of agents. If such space exists, then an agreement is possible and the involved agents can obtain the concrete terms of such agreement using different approaches that are out of the scope of this paper.

The topology of the networks has been introduced in the consensus and its influence has been tested.

When scale-free networks are combined with the possibility of agents to leave the system provokes that the convergence is not guaranteed, as some experiments have shown. This will be object of a more deep study in the future.

An experimental framework has been created in order to simulate the behavior of the consensus algorithm varying fundamental parameters such as the number of agents, the topology of the network, some learning adaptive parameters and the effects that the kind of Agreement Space boundaries calculation process has in the performance of the system.

As additional future works, we want to apply this approach to a real environment, where topological, dynamics and multidimensional factors have to be taken into account.

Acknowledgements

This work is supported by TIN2009-13839-C03-01 and PROMETEO/2008/051 projects of the Spanish government, CONSOLIDER-INGENIO 2010 under grant CSD2007-00022, and PAID-06-11-2084.

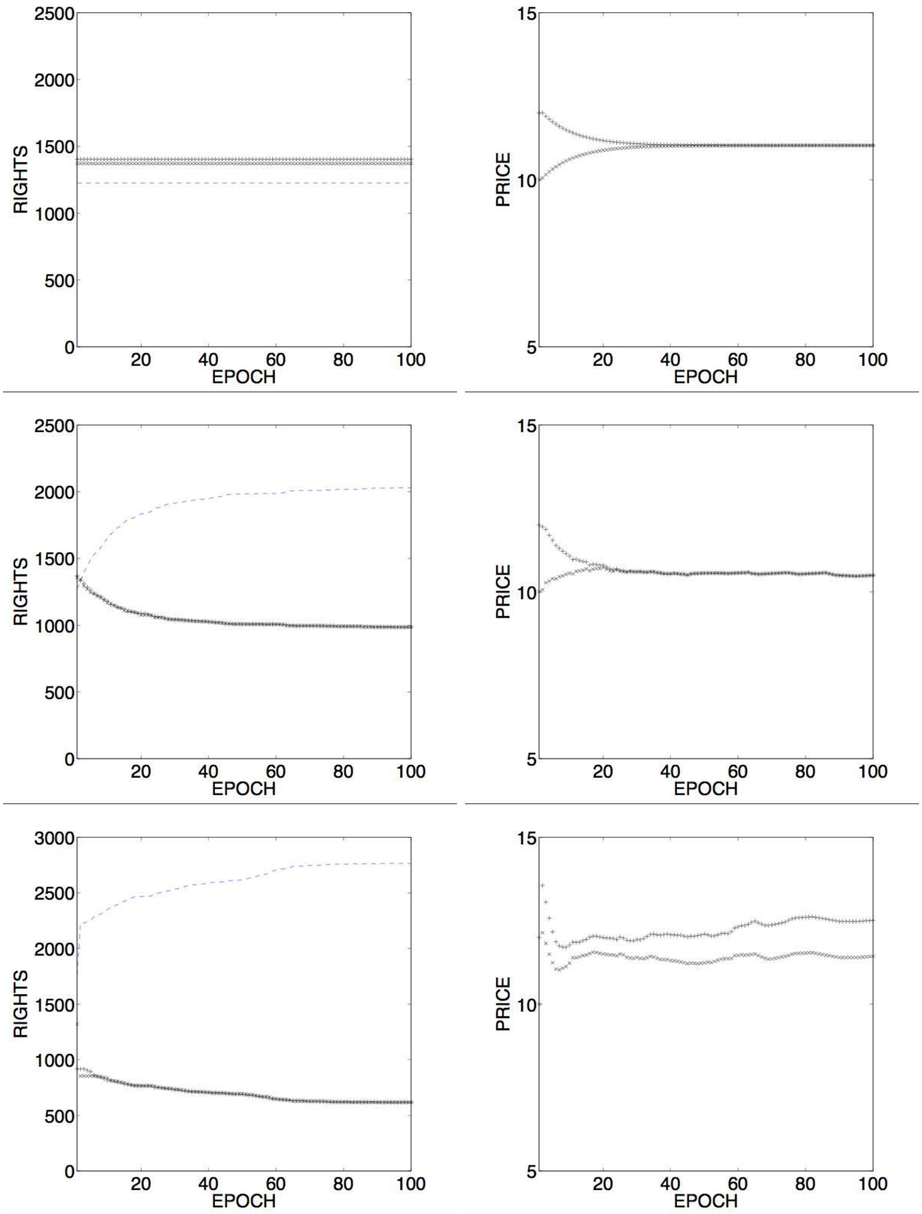

Fig. 3. Consensus evolution. First column shows the evolution of the interchanged rights. Second column shows the evolution of the average price (for sellers and buyers). In rows, top-down, full connected graph, fixed topology and unbiased rights // full connected, switching topology, biased rights // scale free topology ($\gamma = 2.0$) switching topology, biased rights. This last one is a sample of not convergence of the consensus algorithm

References

1. R. A. Albert-Laszlo Barabosi. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.
2. C. Carrascosa and M. Rebollo. Agreement spaces for counselor agents (short paper). In *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*. Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10-15, 2009, Budapest, Hungary, 2009.
3. C. Castelfranchi. Commitments: From individual intentions to groups and organizations. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 41–48, 1995.
4. D. Garlaschelli, S. Battiston, M. Castri, V. D. Servedio, and G. Caldarelli. The scale-free topology of market investments. *Physica A: Statistical Mechanics and its Applications*, 350(2-4):491 – 499, 2005.
5. D. Garlaschelli and M. I. Loffredo. Structure and evolution of the world trade network. *Physica A: Statistical Mechanics and its Applications*, 355(1):138 – 144, 2005.
6. J. C. D. L. Li, D. Alderson and W. Willinger. Towards a theory of scale-free graphs: Definitions, properties, and implications. *Internet Mathematics*, 4(2):431–523, 2006.
7. 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 - Volume 1*, AAMAS '04, pages 438–445, Washington, DC, USA, 2004. IEEE Computer Society.
8. L. Muchnik, R. Itzhack, S. Solomon, and Y. Louzoun. Self-emergence of knowledge trees: Extraction of the wikipedia hierarchies. *Phys. Rev. E*, 76(1):016106, Jul 2007.
9. R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.
10. R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. on Automatic Control*, 49(9):1520–1533, 2004.
11. M. Pujol-Gonzalez. Multi-agent coordination: Dcops and beyond. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pages 2838–2839, 2011.
12. M. A. Serrano and M. Boguñá. Topology of the world trade web. *Phys. Rev. E*, 68:015101, Jul 2003.
13. M. Vinyals, J. Rodriguez-Aguilar, and J. Cerquides. Constructing a unifying theory of dynamic programming dcop algorithms via the generalized distributive law. *Autonomous Agents and Multi-Agent Systems*, 22:439–464, 2011. 10.1007/s10458-010-9132-7.