Multi-Agent Dependence by Dependence Graphs

Jaime Simão Sichman
LTI / EP / USP
Av. Prof. Luciano Gualberto, trav. 3, 158
05508-900 São Paulo SP BRAZIL
+55 11 30915397
jaime.sichman@poli.usp.br

Rosaria Conte
IP / CNR
Viale Marx, 15
00137 Rome ITALY
+39 06 86090210
conte@ip.rm.cnr.it

ABSTRACT
In this paper, we present an abstract structure called *dependence graph,* an extension of the notion of dependence network, as proposed in [16]. While this latter can be applied to express a set of dependence relations of a single agent, this new structure can be applied to the multi-agent case. It can be used, therefore, for the study of emerging social structures, such as groups and collectives, and may form a knowledge base for managing complexity in both competitive and organisational or other cooperative contexts. We analyze several properties of this structure, relating them to some corresponding social phenomena regarding group formation and cohesiveness.

Categories and Subject Descriptors
I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *multiagent systems.*

General Terms: Theory

Keywords
Groups, teams, organizations and societies, group dynamics, formalisms for agents and MAS, self-organizing systems, emergent organization.

1. INTRODUCTION
This paper provides a formal model of multi-agent structures emerging from an aggregate of agents endowed with different goals and actions.

In [16], a notion of dependence network is proposed to represent the pattern of relationships holding between any given agent, on one hand, and one or more other agents on the other. In multi-agent systems, however, dependence relationships are *decentralised* structures, in which no agent involved is assigned a privileged role. As dependence networks are inadequate for representing decentralised structures, we propose instead a *graph* formalism to represent multi-agent dependence.

This new structure allows for the study of emerging social structures, such as groups and collectives, from simple aggregates of heterogeneous agents. Rather than situations in which tasks are already assigned and commitments made, we aim to model multi-agent interdependencies among different agents' goals and actions, and to build up a tool for predicting and simulating their emergence. This may form a knowledge base for managing multi-agent activity in both competitive and organisational or other cooperative contexts.

The paper is organized as follows. We discuss the importance of this work in the next section. In the third section, the original dependence theory proposed in [16] and some contributions [7] are summarized. In the fourth section, we present the formal description of dependence graphs. We use this notion in the fifth section, to illustrate how the dependence theory can be extended to include multi-agent dependence, with a special reference to group and collective phenomena. In the final section, some conclusions are drawn and ideas for future work are outlined.

2. MOTIVATION
The emergence and representation of social structures is a matter of growing concern in the (Multi) Agent Systems field [12] [13] [1]. This is so for different but interrelated reasons.

First, the emergence of groups, leadership and other social formations are receiving growing attention for designing and implementing robust open multi-agent systems [14]. Secondly, an efficient task distribution and execution is increasingly found to depend on dynamic, adaptive (self-) organised activity [2] [9] [8]. Thirdly, organisational practice is shown [4] to depend more on complex interrelationships among individuals and the environment rather than upon on an explicit hierarchical organisational design (cf. the PCANS model [11]). We are only too aware of the huge body of literature on theories, methods and techniques for exploring organisational structures, and for evaluating their performance (for a thorough analysis and a taxonomy, see [4]). Far from providing still another method or technique, we intend to propose a new perspective on the study of emerging organisational structure based upon systems of heterogeneous agents.

3. DEPENDENCE THEORY
The work presented in this paper proceeds from the assumption that heterogeneous agents endowed with goals, beliefs, able to perform actions and situated in a common world are involved in more or less complex and dynamic networks of relationships. In current agent systems, agents are often conceived of and designed as autonomous. However, they are not completely autonomous: agents may have goals that exceed or differ from their capacities to reach them. In particular, in teamwork, agents' autonomy is intrinsically limited [6].

More generally, socially situated agents may depend on one another to achieve their *own* goals. In terms of the dependence theory, an agent ag_i depends on some other agent ag_j with regard to one of its goal g_k, when:

AAMAS '02, July 15-19, 2002, Bologna, Italy.

1. ag_i is not autonomous with regard to g_k: it lacks at least one of the actions or resources necessary to achieve g_k, while

2. ag_j has the missing action/resource.

In the rest of this section, the dependence theory as presented in [16], on the basis of a pre-existing model developed by Castelfranchi et al. [5] is summarised. We also consider some extensions proposed in [7]. Only the notions relevant for the present exposition will be considered explicitly, namely those of *external description, dependence relations*, and *dependence networks*.

3.1 External Description

In order to be able to reason about the others, autonomous agents must have a data structure where the information about the others is stored, despite the possible different internal models they may have. This structure is called *external description*, and it is a private one, i.e., each agent has its own representation of the others. An external description is composed of several entries, each containing the beliefs that a certain *subject agent* has on a particular *object agent* that belongs to the agency.

An external description entry consists of the set of *goals* the object agent wants to achieve, the set of *actions* she is able to perform, the set of *resources* she controls and the set of *plans* she has. A plan consists of a sequence of actions with its associated resources needed to accomplish them. However, an agent may have a plan whose actions or resources do not necessarily belong to her own set of actions or resources. Therefore she may *depend on others* in order to carry on a certain plan, and achieve a certain goal.

As said before, the external description is an agency representation from within an agent's mind. One could ask what relationship holds between the agents' beliefs and the real matters. As one and the same instrument is intended to be employed for both types of representations (mental states and world states), in this paper an *objective representation* of the agency is adopted, namely that which corresponds to the mental state of a specific subject agent, assumed as the observer.

3.2 Dependence Relations

Let us suppose an agent ag_i who tries to achieve a goal g_k. In the original formulation presented in [16] [15], the notions of autonomy and dependence are strictly related to the set of plans $P(ag_q, g_k)$ that the subject agent ag_i uses in order to infer them[1]. For brevity, let us use respectively p_{qk} and P_{qk} as a shorthand notations for $p(ag_q, g_k)$ and $P(ag_q, g_k)$, with $p(ag_q, g_k) \in P(ag_q, g_k)$.

An agent ag_i is *autonomous*[2] for a given goal g_k, according to a set of plans P_{qk} if there is at least one plan p_{qk} in this set that achieves this goal and every action a_m appearing in this plan belongs to her own set of actions.

If an agent does not have all the actions to achieve a given goal, according to a set of plans, she may depend on others for this goal. An agent ag_i *depends* on another agent ag_j for a given goal g_k, according to a set of plans P_{qk} if she has g_k in her set of goals, she is not autonomous for g_k and there is a plan p_{qk} in P_{qk} that achieves g_k where at least one action used in this plan is in ag_j's set of actions.

An example of a basic dependence relation [15] could be:

$$dp_1: basic\_dep(ag_1, ag_2, g_1, p_{111} = a_1(),a_2(),a_4(), a_2)$$

which expresses that agent ag_1 depends on ag_2 to achieve goal g_1, because this latter may perform action a_2 needed in the plan $p_{111} = a_1(),a_2(),a_4()$ which achieves this goal.

Whenever two agents ag_1 and ag_2 depend on one another for their goals g_1 and g_2, there is a bilateral dependence relation between them. If their goals are the same (g_1=g_2), they have a *mutual dependence*; otherwise ($g_1 \neq g_2$), there is a *reciprocal dependence* between them.

3.3 Dependence Networks

Whenever an agent infers his basic dependence relations, he can internally represent them in a structure called *dependence network*.

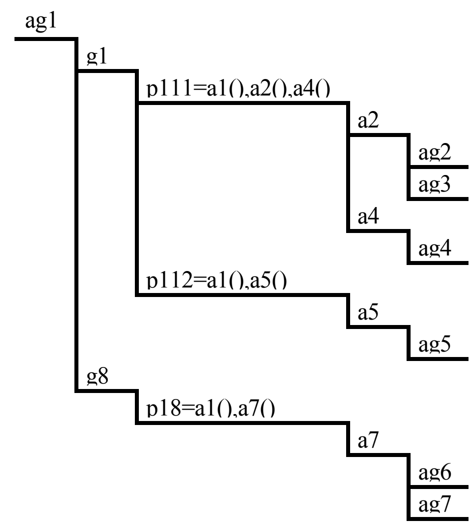

Figure 1. Dependence Network

As an example, let us consider the case where an agent ag_1 has two goals g_1 and g_8. For the first goal g_1, she has two alternative plans, $p_{111}= a_1(),a_2(),a_4()$ and $p_{112}= a_1(),a_5()$ to achieve this goal. On the other hand, for the second goal g_8, she has only one plan, $p_{18}= a_1(),a_7()$ that achieves this goal. Let us also suppose that ag_1 can perform actions a_1 and a_9, but she is unable to perform the actions a_2, a_4, a_5, and a_7, which can be performed respectively by the set of agents $\{ag_2, ag_3\}$, ag_4, ag_5 and by the set of agents $\{ag_6, ag_7\}$. In this scenario, the following basic dependence relations hold:

$$dp_1: basic\_dep(ag_1, ag_2, g_1, p_{111} = a_1(),a_2(),a_4(), a_2)$$
$$dp_2: basic\_dep(ag_1, ag_3, g_1, p_{111} = a_1(),a_2(),a_4(), a_2)$$
$$dp_3: basic\_dep(ag_1, ag_4, g_1, p_{111} = a_1(),a_2(),a_4(), a_4)$$
$$dp_4: basic\_dep(ag_1, ag_5, g_1, p_{112}= a_1(),a_5(), \quad a_5)$$
$$dp_5: basic\_dep(ag_1, ag_6, g_8, p_{18} = a_1(),a_7() \quad a_7)$$
$$dp_6: basic\_dep(ag_1, ag_7, g_8, p_{18} = a_1(),a_7(), \quad a_7)$$

The dependence network of ag_1 is presented in figure 1. In the general case, such a network can be much more complicated, as an agent can have several goals, with different plans to achieve them.

3.4 Some Extensions

One can notice that a basic dependence relation expresses a one-to-one relation. Some extensions were presented in [7], considering a

[1] The agent q whose plans are used to infer these notions is called source agent.

2 In the original theory, there were distinct definitions for action and resource autonomy and dependence. For simplicity only action-dependence will be considered, and we will call it dependence in the rest of the paper.

one-to-many and many-to-one dependence relations. In the rest of the paper, ag_i will denote a single agent, Ag_i will denote a set of agents and AG_i will denote a set of agents' sets.

3.4.1 OR-Dependence

An agent ag_i OR-depends on a set of agents Ag_j when she holds a disjunction set of dependence relations upon any member ag_k of Ag_j. Any member of the set Ag_j is sufficient but unnecessary for ag_i's goal. For example, in order to have information about how to fill a tax form, any financial expert will do. OR-dependence provides the dependent agent with a number of alternative ways to achieve her goal, among which she shall choose the most convenient. The number of alternatives amounts to the number of agents contained in the set Ag_j. One can notice that *OR-dependence mitigates social dependence*, if only because the probabilities that some agent willing to help is found increase.

Referring to the dependence network presented in figure 1, the notion of OR-dependence is related to the fifth level of the network, i.e., *the possible agents able to perform a single needed action in a particular plan that achieves a certain goal*. In this network, ag_1 OR-depends on the set $Ag_2 =\{ag_2, ag_3\}$, because she needs one of them to perform action a_2 to achieve goal g_1, according to plan p_{111}.

3.4.2 AND-Dependence

Sometimes, one and the same agent may depend on a bunch of others for achieving one of her goals. For example, a rogue may AND-depend on a handful of specialised fellows to organise and execute a robbery: a lookout, a skilled driver, etc. If we consider that ag_i's degree of dependence is a direct function of the costs of the actions that she needs to be performed, then, quite unlike the preceding link, *AND-dependence will be greater than ordinary dependence*, other things being equal.

Referring to the dependence network presented in figure 1, the notion of AND-dependence is related to the fourth level of the network, i.e., *the needed actions to perform a particular plan that achieves a certain goal*. Since different agents may perform a needed action, as was explained in the last subsection, the notion of AND-dependence must be built on OR-dependence.

In the network presented in figure 1, ag_1 AND-depends on the set of agents' sets $AG_3 =\{\{ag_2, ag_3\}, \{ag_4\}\}$, because he needs both actions a_2 (which can be performed either by ag_2 or by ag_3) and a_4 (which can be performed by ag_4) in order to execute p_{111} to achieve goal g_1.

3.4.3 CO-Dependence

In this case, a set of agents Ag_j depend on ag_i each for its own goal. The lesser the actions that ag_i can perform simultaneously, the more ag_i will be contended for by Ag_j's members.

4. DEPENDENCE GRAPHS

As one may observe, a dependence network contains all basic dependence relations of a *single agent*. Sometimes, it is useful to represent in a single structure several dependence networks, relating a *set of agents*. In order to do so, we introduce the notion of *dependence graphs*.

4.1 Definition

Mathematically, a graph G is an ordered triple $(V(G), E(G), \psi_G)$ consisting of a nonempty set of $V(G)$ of vertices (or nodes), a set $E(G)$, disjoint from $V(G)$, of edges (or arcs) and an incidence function ψ_G that associates with each edge of G an unordered pair of (not necessarily distinct) vertices of G [3]. The degree $d_G(v)$ of a vertex v in G is the number of edges incident with v, each loop counting as two edges. A graph H is a subgraph of G (written $H \subseteq G$) if $V(H) \subseteq V(G)$, $E(H) \subseteq E(G)$ and ψ_H is the restriction of ψ_G to $E(H)$. Let v_0 and v_n be vertices in a graph. A *path* from v_0 to v_n of length m is an alternating sequence of $m+1$ vertices and m edges beginning with vertex v_0 and ending with vertex v_n $path(v_0, v_n)=(v_0, e_1, v_1, e_2, ..., e_m, v_n)$ in which edge e_i is incident on vertices v_{i-1} and v_i, $for\ i = 1,2,...,m$. A *simple path* from v_i to v_j is a path from v_i to v_j with no repeated vertices. A cycle (or circuit) $cycle(v_i)$ is a path of nonzero length from v_i to v_i with no repeated edges. A *simple cycle* is a cycle from v_i to v_i in which, except for the beginning and ending vertices that are both equal to v_i, there are no repeated vertices. A graph is *connected* if there is a path between any of its vertices.

A *tree* is a connected acyclic graph. A *rooted tree* $T(v)$ is a tree in which a particular vertex v is designated the root. If $path(v_0, v_n)=(v_0, e_1, v_1, e_2, ..., e_m, v_n)$ is a simple path in a rooted tree $T(v_0)$, then v_{i-1} is the *parent* of v_i, v_i is the *child* of v_{i-1}, and if v_i has no children, v_i is a terminal vertex (or leaf), for $i = 0, ..., n$. The *depth* of the root node is 0, and the depth of any node is the depth of its parent node plus one. The depth of the rooted tree $T(v)$ is the maximum depth of its nodes.

When the incidence function ψ_g associates with the edge of G an ordered pair of vertices, the graph is called a *directed graph* (or digraph). In this case, we can propose some new definitions for *directed paths* and *directed cycles*. If G is a digraph and ψ_G associates an edge e_1 of G to an ordered pair of vertices (v_0, v_1) of G, then v_0 is the *tail* of e_1, and v_1 is the *head* of e_1. The *indegree* $d_G^-(v)$ of a vertex v in G is the number of edges with head v, and the *outdegree* $d_G^+(v)$ of a vertex v in G is the number of edges with tail v.

A *bipartite graph* is one whose vertices can be partitioned into two subsets X and Y, so that each edge has one end in X and one end in Y. This definition can be extended to n subsets, and the graph is called a *n-partite graph*.

Using these definitions, we can define a *dependence graph*. We will represent the basic notions of our model, agents, goals, plans and actions, as vertices of the graph. Moreover, as a pair of vertices will be linked by a single edge, we will represent a path in a simpler manner, without making an explicit reference to the edges, i.e., we will use $path(v_0, v_n)=(v_0, v_1, ..., v_n)$ instead of $(v_0, e_1, v_1, e_2, ..., e_m, v_n)$.

Formally, a dependence graph $DPG=(V(DPG), E(DPG), \psi_{DPG})$ is a 4-partite directed graph with the following characteristics:

1. the set $V(DPG)=V_{ag}(DPG) \cup V_g(DPG) \cup V_p(DPG) \cup V_a(DPG)$ is the union of the following disjoint sets:

   1.1. $V_{ag}(DPG)=\{ag_1, ag_2, ..., ag_n\}$ is the set of agents;

   1.2 $V_g(DPG)=\{g_1, g_2, ..., g_n\}$ is the set of the possible goals these agents may want to achieve;

   1.3. $V_p(DPG)=\{p_1, p_2, ..., p_n\}$ is the set of plans the agents may use to achieve their goals;

   1.4. $V_a(DPG)=\{a_1, a_2 ..., a_n\}$ is the set of actions that can be performed by these agents.

2. the set $E(DPG)$ is a set of edges;

3. the function ψ_{DPG}: $E(DPG) \rightarrow V(DPG) \times V(DPG)$ is defined as follows:

   3.1. $\psi_{DPG}(e) = (ag_i, g_j)$ associates an edge e with an ordered pair of vertices (ag_i, g_j), with $ag_i \in V_{ag}(DPG)$ and

$g_j \in V_g(DPG)$ and represents the fact that ag_i has the goal g_j;

3.2. $\psi_{DPG}(e) = (g_i, p_j)$ associates an edge e with an ordered pair of vertices (g_i, p_j), with $g_i \in V_g(DPG)$ and $p_j \in V_p(DPG)$ and represents the fact that goal g_i *is* achieved by plan p_j;

3.3. $\psi_{DPG}(e) = (p_i, a_j)$ associates an edge e with an ordered pair of vertices (p_i, a_j), with $p_i \in V_p(DPG)$ and $a_j \in V_a(DPG)$ and represents the fact that plan p_i needs the action a_j and agent ag_k can not perform this action[3];

3.4. $\psi_{DPG}(e) = (a_i, ag_j)$ associates an edge e with an ordered pair of vertices (a_i, ag_j), with $a_i \in V_a(DPG)$ and $ag_j \in V_{ag}(DPG)$ and represents the fact that action a_i can be performed by agent ag_j.

4.2 A Single Agent Example

As an example, the dependence network shown in figure 1 is presented as a dependence graph in figure 2. In a single agent case, like shown in figure 2, the dependence graph results in a tree.

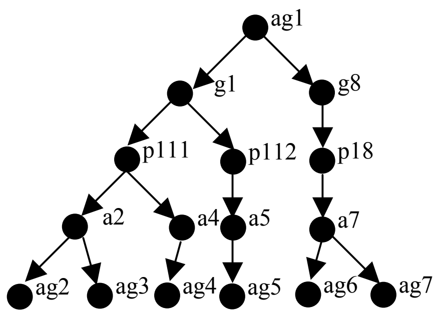

Figure 2. Dependence Graph for a Single Agent

One may notice the following interesting points:

1. a basic dependence relation *basic_dep*$(ag_i, ag_j, g_k, p_{qk}, a_m)$ is represented in the dependence graph by a simple path $path(ag_i,ag_j)=(ag_i, g_k, p_{qk}, a_m, ag_j)$ of length 4;

2. an OR-dependence of an agent ag_i upon the agent set Ag_j is represented in the dependence graph by a subtree $T(ag_i)$ of depth 4, containing one single vertex in each depth level 0, 1, 2 and 3 and at least two leaves in level 4, representing agents belonging to Ag_j;

3. an AND-dependence of an agent ag_i upon the set of agents' sets AG_j is represented in the dependence graph by a subtree $T(ag_i)$ of depth 4, containing one single vertex in each depth level 0, 1, and 2, and at least two vertices in level 3, representing different actions which can be performed by elements of AG_j.

4.3 A Multi-Agent Example

In the scenario presented in figure 2, the dependence graph represents only the goals of agent ag_1. However, as was stated in the introduction of this section, the interest of dependence graphs is that they can represent goals and dependence relations of *several* agents. In order to illustrate this concept, let us consider again the example shown in figures 1 and 2. Let us put forward the following additional hypothesis:

1. agent ag_2 has a goal g_2 and a certain plan $p_{22}= a_2(),a_6()$ to achieve this goal, she can perform action a_2 but not a_6. Another agent ag_6 can perform this last action, as well as action a_7, but she needs action a_1 to achieve her goal g_6, according to her plan $p_{66}= a_1(),a_6()$;

2. agent ag_4 has a goal g_4 and a certain plan $p_{44}= a_4(),a_7()$ to achieve this goal, she can perform action a_4 but not a_7. As well as ag_6, another agent ag_7 can perform this last action, but she needs action a_9 to achieve her goal g_7, according to her plan $p_{77}= a_7(),a_9()$;

In this scenario, the following additional basic dependence relations hold:

dp_7: *basic_dep*$(ag_2, ag_6, g_2, p_{22}= a_2(),a_6(), a_6)$
dp_8: *basic_dep*$(ag_4, ag_6, g_4, p_{44}= a_4(),a_7(), a_7)$
dp_9: *basic_dep*$(ag_4, ag_7, g_4, p_{44}= a_4(),a_7(), a_7)$
dp_{10}: *basic_dep*$(ag_6, ag_1, g_6, p_{66}= a_1(),a_6(), a_1)$
dp_{11}: *basic_dep*$(ag_7, ag_1, g_7, p_{77}= a_7(),a_9(), a_9)$

The dependence graph for this scenario is shown in figure 3.

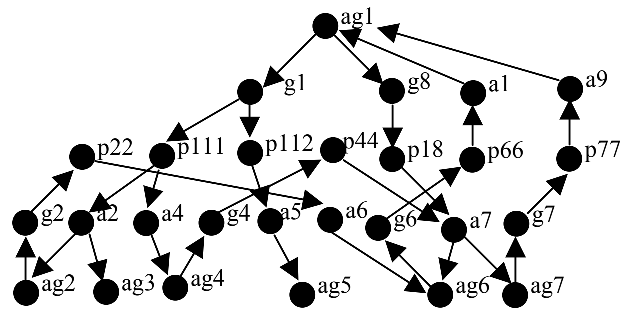

Figure 3. A Multi-Agent Dependence Graph

4.4 Reduced Dependence Graphs

As one may notice, the dependence graph is quite a complex structure with lots of paths. In some situations, like the ones that will be treated in the rest of the paper, this structure can be simplified.

Let us consider the case where (i) each agent has a *single goal* to achieve and (ii) each agent has a *single plan* to achieve this goal. In this case, we do not need to represent in the graph neither the vertices representing goals nor the ones representing plans, and the set of vertices will be reduced to the union of sets of agents and actions vertices $V(DPG) = V_{ag}(DPG) \cup V_a(DPG)$. Consequently, the function $\psi_{DPG}(e)$ will also be simplified, containing either elements like $\psi_{DPG}(e) = (ag_i, a_j)$ or $\psi_{DPG}(e) = (a_i, ag_j)$.

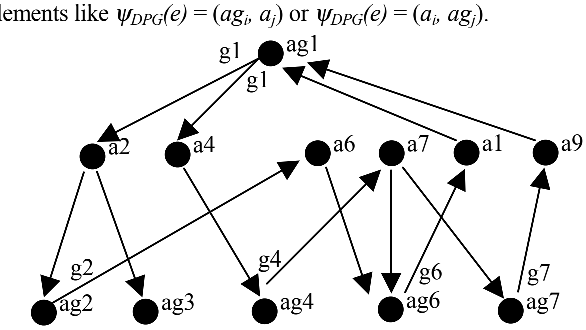

Figure 4. Reduced Dependence Graph

As it may be important for the further discussion to identify the agents' goals, we will use them as *labels* of the directed edges linking agents to actions. We will call this representation a *reduced dependence graph*. An example of such a graph is presented in

[3] Agent ag_k is the origin of the path to which this edge belongs, as illustrated in the sequence.

figure 4, which corresponds to the scenario presented in the last subsection.

5. MULTI-AGENT DEPENDENCE

The above concepts and the whole set of edges in a dependence graph can be used to describe more or less structured and complex multi-agent systems. In particular, different levels of complexity and internal cohesiveness/fragility of a multi-agent system can be shown to emerge from some features of the dependence graph described above.

As will be shown throughout this section, rather than a none-or-all notion, multi-agent dependence indicates a phenomenon of growing complexity, from loose group-dependence to a more structured and more cohesive collective dependence.

In the remaining of the paper, for simplicity, we consider a set Ag_i of non-autonomous agents with regard to their goals: each of them needs one or more actions, different from one another, to achieve their goals. Any member ag_j of Ag_i tries to achieve *a single goal,* and has *one single plan* that achieves this goal. Therefore, we will use reduced dependence graph to describe them.

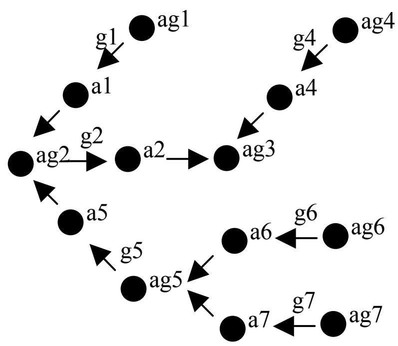

Figure 5. An Acyclic Dependence Graph

Figure 5 shown an acyclic dependence graph. This graph depicts a loose social relation: no partnership can emerge. Let us now see under which conditions the dependence graph may favour possible multi-agent partnerships.

5.1 AMONG-Dependence

Let us look now at figure 6. The graph contains two cycles, *cycle*(ag_1)={ag_1, a_1, ag_2, a_2, ag_7, a_7, ag_1} and *cycle*(ag_4)={ag_4, a_4, ag_3, a_3, ag_4} of different lengths: *cycle*(ag_4) has length 4, while *cycle*(ag_1) has length 6. The first one represents a not so complex situation: two agents depend on one another for their different goal (reciprocal dependence) [6]. Conversely, *cycle*(ag_1) represents a more complex situation: it contains more than two agents, where each may receive help from someone and may provide help to another. Sociologists [17] would say that in *cycle*(ag_1) a "generalised" form of exchange, requiring a rather complex negotiation process, might occur. We will call AMONG-dependence the dependence relationship holding in *cycle*(ag_1).

More precisely, a set of agents Ag_i can be said to AMONG-dependent when the following three clauses hold:

Dependence clause - for each ag_j belonging to Ag_i there is at least another agent ag_k upon which ag_j depends for her goal;

Utility clause - there is at least another agent ag_l belonging to Ag_i which depends upon ag_j for his goal;

Generalised reciprocity clause – $ag_k \neq ag_l$.

Considering the dependence graph, an AMONG-Dependence will hold if there is at least one subgraph $G \subseteq DPG$ which contains a cycle whose length is greater than 4.

Obviously, the larger the set of AMONG-depending agents, the more complex the structure and the more difficult the agreement and the coordination[4].

In figure 6, a simple and ideal case of AMONG-dependence is offered, represented by *cycle*(ag_1)={ag_1, a_1, ag_2, a_2, ag_7, a_7, ag_1}. Under certain conditions, which we are going to examine, an AMONG-dependence may prove fragile, and either split or collapse.

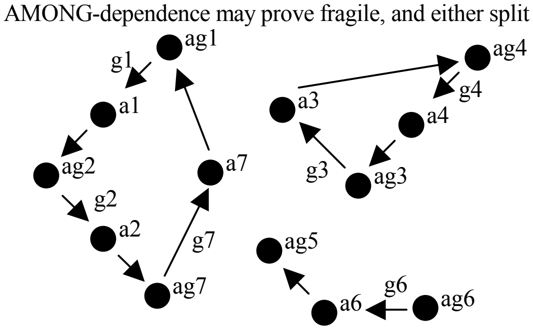

Figure 6. AMONG-Dependence

5.1.1 AMONG-Dependence and OR-Dependence

Let us consider what happens if one agent ag_j, belonging to a set of agents Ag_i which satisfies the AMONG-dependence clauses, OR-depends on a subset $Ag_k \subseteq Ag_i$, as shown in figure 7. Here, we have one agent (ag_1) OR-depending on two agents (ag_2 and ag_6) for action a_1. An interesting phenomenon emerges:

If one agent ag_j, belonging to a set of agents Ag_i which satisfies the AMONG-dependence clauses, OR-depends upon a subset Ag_k of Ag_i, there are two or more cycles of different lengths starting at agent ag_j which may satisfy the AMONG-dependence clauses.

Considering the dependence graph *DPG*, in this situation there is at least one node $a_i \in V_a(DPG)$ (a_1 in figure 7) whose outdegree is greater that its indegree, i.e., with $d_G^+(a_i) > d_G^-(a_i)$.

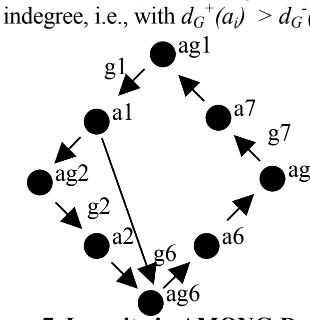

Figure 7. Inequity in AMONG-Dependence

When this property holds, there will be at least one agent that is more useful (more required) than dependent (ag_6 in figure 7). Consequently, the dependence graph will have two or more cycles, of different complexity.

In figure 7, for example, these are the two cycles starting at vertex ag_1, $cycle_1(ag_1)$={ag_1, a_1, ag_6, a_6, ag_7, a_7, ag_1} and

[4]AMONG-dependence could be quantified as a function of the number of agents belonging to the set of agents involved (and the number of steps that should be realised in order for all agents to receive and give help). However, other variables should be taken into account as will be seen later on in the paper.

$cycle_2(ag_1)=\{ag_1, a_1, ag_2, a_2, ag_6, a_6, ag_7, a_7, ag_1\}$ and both satisfy the AMONG-dependence clauses. The OR-depending agent (ag_1) will have a chance to decide. Its choice will determine whether a sub-group will be formed at the expense of a (subset of) agent(s) (in our case, at ag_2's expense). OR-dependence is an obstacle for AMONG-dependence to lead to group formation. Since OR-dependence reduces the degree of dependence, an OR-depending agent involved in an AMONG-depending network will always be less depending than useful. This introduces an unbalance, or inequity, in the network at the benefit of the OR-depending agent. As will be shown below, unbalance or inequity endangers AMONG-dependence and obstacles group formation.

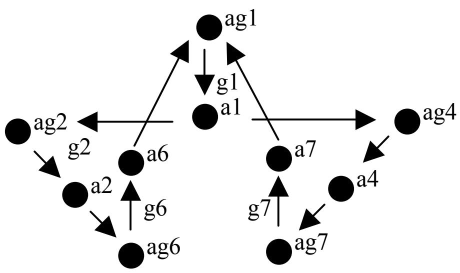

Figure 8. Incompatibility in AMONG-Dependence

A special case occurs when the OR-depending agent is useful to the same number of agents that she OR-depends upon, shown in figure 8:

If the number of agents in the subset Ag_l depending upon ag_j is equal to the number of agents of the subset Ag_k ag_j OR-depend upon, there are alternative incompatible cycles starting at agent ag_j in all of which AMONG-dependence clauses apply.

Considering the dependence graph *DPG*, in this situation there is at least one node $a_j \in V_a(DPG)$ with $d_G^+(a_j) > d_G^-(a_j)$ (a_1 in figure 8), one node $ag_i \in V_{ag}(DPG)$ with $d_G^+(ag_i) < d_G^-(ag_i)$ (ag_1 in figure 8), and there is an edge $e \in E(DPG)$ with $\psi_{DPG}(e) = (ag_i, a_j)$.

In other words, an agent may OR-depend upon as many agents as those depending on her. In figure 8, for example, there are two incompatible cycles starting at vertex ag_1, $cycle_1(ag_1)=\{ag_1, a_1, ag_2, a_2, ag_6, a_6, ag_1\}$ and $cycle_2(ag_1)=\{ag_1, a_1, ag_4, a_4, ag_7, a_7, ag_1\}$, between which the OR-depending agent, contended for by the others, will have the power to choose. In either case, OR-dependence disrupts or obstacles AMONG-dependence and consequently, group formation.

5.1.2 AMONG-Dependence and AND-Dependence

Let us consider what happens if we introduce an AND-dependence link, like the situation illustrated in figure 9.

In this case, group exchange can occur only if one of the agents accepts to give more than she receives (in figure 9, ag_6). Again, if at least one agent in the set depends more than it is useful for (inequity condition), the AMONG-dependence is seriously endangered and a fragile agreement is likely to emerge.

If the set Ag_i satisfies the AMONG-dependence conditions, but at least one agent ag_j in Ag_i AND-depends on a set of sets of agents AG_k and $|AG_k| > |Ag_l|$, Ag_l is the subset of agents depending on ag_j, AMONG-dependence is endangered.

Considering the dependence graph *DPG*, in this situation there is at least one two nodes ag_i, $ag_j \in V_{ag}(DPG)$ (ag_1 and ag_6 in figure 9) with $d_G^+(ag_i) < d_G^-(ag_i)$ and $d_G^+(ag_j) > d_G^-(ag_j)$.

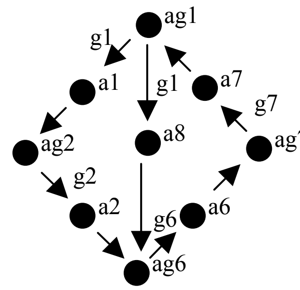

Figure 9. Fragile AMONG-Dependence

Nonetheless, sometimes AND-dependence may strengthen AMONG-dependence, as one may notice in figure 10. Here, once more, one can notice two cycles starting at vertex ag_1. However, they are not only compatible but actually inter-dependent: neither $cycle_1(ag_1)=\{ag_1, a_1, ag_2, a_2, ag_6, a_6, ag_1\}$ nor $cycle_2(ag_1)=\{ag_1, a_8, ag_4, a_4, ag_7, a_7, ag_1\}$ can independently form a group. They can only work together.

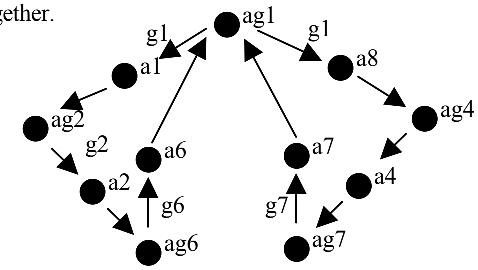

Figure 10. Strong AMONG-Dependence

Given a set of agents Ag_i which satisfies the AMONG-dependence clauses, if there is at least one agent ag_j belonging to Ag_i which AND-depends upon a set of sets of agents AG_k and $|AG_k| = |Ag_l|$, Ag_l is the subset of agents depending on ag_l, there are two or more compatible cycles, each satisfying the AMONG-dependence conditions.

Considering the dependence graph *DPG*, in this situation for all nodes $ag_i \in V_{ag}(DPG)$ we have $d_G^+(ag_i) = d_G^-(ag_i)$. When this condition is not satisfied, the AMONG-depending graph is bound to either split or shrink, or group agreement is less likely to occur.

5.2 GROUP-Dependence

AMONG-dependence holding in a set of agents may give rise to GROUP-dependence when the equity clause is satisfied:

Equity clause *- For all agents ag_j belonging to the set Ag_i of AMONG-depending agents, the subset of agents Ag_k upon which ag_j depends is equal to the subset Ag_l of agents depending on ag_j.*

In this case, AMONG-dependence leads to a global network, a potential group structure, since no one can achieve its goal independent of the others' achievements and actions. This will be called *GROUP*-dependence.

Below, we will distinguish two conceptually different types of group dependence, which can co-exist in real matters.

5.2.1 Decentralized GROUP-Dependence

Consider the following clause:

Egalitarian clause *- For all agents ag_j belonging to the set Ag_i of AMONG-depending agents, ag_j depends upon the same number of agents, and is useful for the same number of agents.*

When both the equity and the egalitarian clauses hold, a decentralised structure emerges: none plays a leading role, and all share an equal condition. All the agents involved may find an agreement leading them to receive what they need and provide what they are expected to deliver. If, and only if, all agents respect the agreement, the group-dependence will actually lead to an effective group partnership. Of course, a complex social process is necessary for the agreement both to be established and to be respected by all members. An example of a decentralised GROUP-dependence can be found in figure 6, if we consider exclusively the set of agents $Ag_i=\{ag_1, ag_2, ag_7\}$.

Considering the dependence graph *DPG*, in this situation for all nodes $ag_i \in V_{ag}(DPG)$ we have $d_G^{+}(ag_i) = d_G^{-}(ag_i) = n$.

5.2.2 Centralized GROUP-Dependence

The fundamental dependence relationship that occurs in group-exchange is *reciprocal* dependence. Agents depend on one another each to achieve their own goals. Both reciprocal and group dependence, in fact, occur quite frequently in competitive contexts, like markets.

An intermediate phenomenon, bridging the gap between group and collective dependence, group exchange and teamwork, is centralised group dependence. Consider the dependence graph presented in figure 11. Here, one agent ag_1 AND-depends on all others to achieve her goal g_1, and all others depend on ag_1 to achieve each its own goal. There is a complete intersection among the set of agents ag_1 AND-depends upon and the set of agents depending upon ag_1.

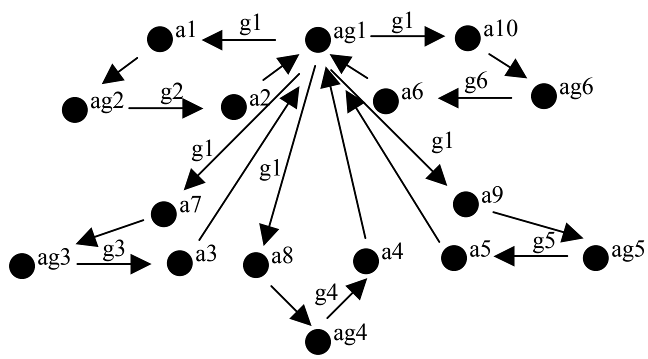

Figure 11. Centralized GROUP-Dependence

Centralised GROUP-dependence is a special case of AMONG-dependence in which the egalitarian clause does not apply (although the equity clause does), but there is at least one agent which plays a leading role. The dependence graph is centralised thanks to a multi-agent plan owned by one of the agents. All others are complementary for this one to execute her plan. This agent will "hire" all others to achieve this plan. In formal terms, centralised GROUP-dependence occurs in a set of AMONG-depending agents Ag_i when all the following conditions hold:

1. the equity clause is satisfied;

2. there is at least one agent ag_j belonging to Ag_i which AND-depends on a set of sets of agents AG_k, where each element of $AG_k = \{\{ag_p\}, \{ag_q\}, \ldots, \{ag_v\}\}$ is a singleton whose element $ag_x \in Ag_i$, $x = p, q, \ldots v$ and $|AG_k| = |Ag_i|-1$;

3. any agent ag_k belonging to Ag_i depends upon ag_j.

In centralised group-dependence a group is likely to be formed and led by one agent which represents the head of the network.

Obviously, if one single agent OR-depends on all others, no group will emerge, but only two agents will form a partnership in exchange (the equity clause is not satisfied). Being the most useful and the least depending agent, the leader of the network would choose her partner for exchange.

5.3 COLLECTIVE-Dependence

Collective dependence is to group dependence what mutual dependence is to reciprocal dependence. The fundamental relationship of dependence is here *mutual* dependence [6]. Rather than a clear-cut distinction, however, group and collective dependence are situated on a continuum. AMONG-dependence graphs vary, among other factors (number of links, decentralised/non-decentralised), according to whether the agents involved share the goals with regard to which they depend on one another, or else pursue different goals. A COLLECTIVE-dependence holds in a set of AMONG-depending agents when each agent depends on all others to achieve a shared goal. This situation is represented in figure 12.

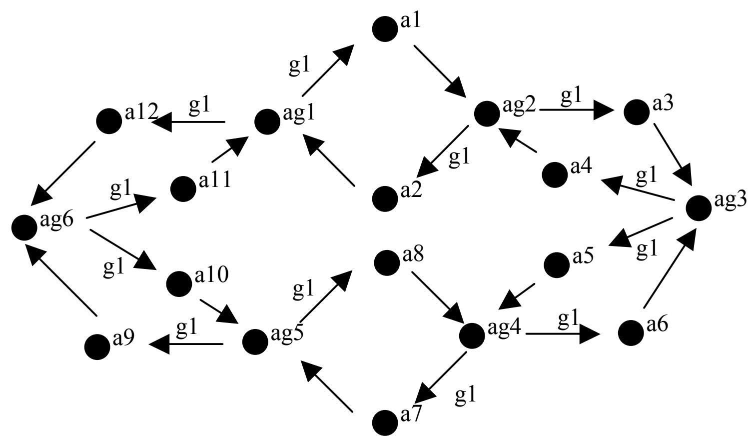

Figure 12. COLLECTIVE-Dependence

Collective dependence therefore leads to a rather cohesive group executing a multi-agent plan, i.e. teamwork. All agents are complementary to achieve one and the same goal.

In figure 12, a dependence graph where each agent depends on all others is shown. Such a network is inherent to the team and is based upon the multi-agent task that the team is supposed to accomplish. Full collective dependence occurs when the complementary agents share the goal for which they are needed.

In such a case, each member in the set of AMONG-depending agents depends on all others to achieve a common goal. A band of rogues offers a typical example of COLLECTIVE-dependence.

One may notice that COLLECTIVE-dependence need not be decentralised as in figure 12, if there is at least one agent ag_j (or a subset of agents Ag_j) member of (or contained in) Ag_i upon which all others depend upon in order to a leading role. A typical example is an orchestra, which is led by one member, the director.

6. CONCLUSIONS AND FURTHER WORK

In this paper, we have presented an abstract structure called dependence graph, which can be used to represent a set of dependence relations of a multi-agent system. A similar approach using graphs to represent cooperation structures is presented in [10], but our approach is richer, since in the former one could detect if a cooperation was possible to achieve a single agent goal.

To sum up, a bunch of agents searching partners for bargain are plunged in a more or less complex dependence graph, where groups and collectives arise. Rather than a clear-cut distinction, the difference between groups and collectives consist of different levels of complexity and cohesiveness of the underlying dependence graph. Full collective dependence occurs when the complementary agents share the goal for which they are needed. In such a case, each member in the set of complementary agents depends on all others to achieve the shared goal.

In the future, we intend to take the following steps: (i) to work out a formal model of the quantity of dependence; (ii) to incorporate the notions here presented in a computational system in order to check the efficiency of the model; and (iii) to run simulations to test its predictive power with regard to the emergence of groups and collectives.

7. ACKNOWLEDGMENTS

This paper has been realized with contributions from the 5th Framework European Projects ALFEBIITE and FIRMA. The authors would like to thank Claudia Linhares Salles, from UFC, Fortaleza, Brazil, for the key advice of using n-partite graphs in order to construct the model. Jaime Simão Sichman is also grateful to CNPq, Brazil, which has partially financed this research, grant 301041/95-4.

8. REFERENCES

[1] Balch, T., Z. Khan, and M. Veloso. Automatically Tracking and Analyzing the Behavior of Live Insect Colonies. In J. Mueller, E. Andre, S. Sen, C. Frasson (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press, 521-29.

[2] Barber, K.S. and C.E. Martin. Dynamic Reorganisation of Decision-Making Groups. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press, 513-20.

[3] Bondy, J.A. and Murty, U.S.R. *Graph Theory and Applications*. McMillan Press Ltd., London, UK, 1977.

[4] Carley, K.M., and D. Krackhardt. A Typology for C2 Measure. In *Proceedings of the 1999 International Symposium on Command and Control Research and Technology*, Newport, RI, June.

[5] Castelfranchi, C., A. Cesta, and M. Miceli. Dependence Relations in Multi-Agent Systems. In Y. Demazeau and E. Werner (Eds.) *Decentralized AI - 3*, North-Holland: Elsevier, 1992.

[6] Conte, R. and J.S. Sichman. DEPNET: How to Benefit from Social Dependence. *Journal of Mathematical Sociology*, 20(2-3), 161-177, 1995.

[7] Conte, R., V. Veneziano, and C. Castelfranchi. The Computer Simulation of Partneraship Formation. *Computational and Mathematical Organization Theory*, 4(4), 293-315, 1998.

[8] Fatima, S.S. and M. Wooldridge. Adaptive Task and Resource Allocation in Multi-Agent Systems. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press Horling et al., 2001.

[9] Horling, B, B. Benyo, and V. Lesser. Using Self-Diagnosis to Adapt Organizational Structures. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press Horling et al., 2001.

[10] d'Inverno, M., M. Luck and M. Wooldridge. Cooperation Structures. In *Proceedings of 15th. Int. Joint Conference on AI*, Nagoya, Japan, 600-605, 1997.

[11] Krackhardt, D. and K.M. Carley. A PCANS Model of Structure in Organizations. In *Proceedings of the 1998 International Symposium on Command and Control Research and Technology*, June, Monterray, CA.

[12] Parunak, Van Dyke H. and S. Brueckner. Entropy and Self-Organization in Multi-Agent Systems. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press 2001.

[13] Parunak, Van Dyke H. and J. Odell. Representing Social Structures in UML. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press 2001.

[14] Schillo, M., H-J. Buerckert, K. Fischer, and M. Klutsch. Towards a Definition of Robustness for Market-Style Open Multi-Agent Systems. In J. Mueller et al., (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Canada, May 28 – June 01, 2001. New York: ACM Press.

[15] Sichman, J.S. and Y. Demazeau. On Social Reasoning in Multi-Agent Systems. *Revista Ibero-Americana de Inteligência Artificial*, 13, 68-84. Madrid: AEPIA, 2001.

[16] Sichman, J.S., R. Conte, C. Castelfranchi, and Y. Demazeau. A Social Reasoning Mechanism Based on Dependence Networks. In A G. Cohn (Ed.), *Proceedings of the 11th. European Conference on Artificial Intelligence*, Baffins Lane, England: John Wiley & Sons, 1994.

[17] Yamagishi, T. and K.S. Cook. Generalized exchange and social dilemmas. *Social Psychology Quarterly*, 560, 235-24, 1993