Coalition formation is an important mechanism for cooperation in multiagent systems. In this paper we address the problem of coalition formation among self-interested agents in superadditive task-oriented domains. We assume that each agent has some “structure,” i.e., that it can be described by the values taken by a set of $m$ nonnegative attributes that represent the resources $w$ each agent is endowed with. By defining the coalitional value as a function $V$ of $w$, we prove a sufficient condition for the existence of a stable payment configuration—in the sense of the core—in terms of certain properties of $V$. We apply these ideas to a simple case that can be described by a linear program and show that it is possible to compute for it—in polynomial time—an optimal task allocation and a stable payment configuration.

Key words: agents, coalition, core.

1. INTRODUCTION

In recent years it has become clear that computer systems do not work in isolation. Currently, computer systems are increasingly acting as components in a distributed and complex environment of people and systems. In addition, the huge development and success of the Internet has emphasized this phenomenon. To carry out their tasks, these computer systems have to cooperate and coordinate their activities with other systems and people. Some examples include electronic commerce digital libraries, electricity networks, health institutions, military applications, and so on.

Coalition formation is an important mechanism for cooperation in Multiagent Systems (MAS) (Sandholm 1999; Shehory and Kraus 1999; Sandholm et al. 1999; Tsvetovat et al. 2000; Sen and Dutta 2000; Contreras and Wu 2000). The reason is that autonomous agents forming coalitions may improve their profits and abilities to satisfy their goals, sharing their resources, and distributing their tasks. For example, coalition formation mechanisms have been proposed for applications in electronic commerce (Tsvetovat et al. 2000; Yamamoto and Sycara 2001; Cuihong and Sycara 2002; Lerman and Sycara 2000) as a means to aggregate groups of customers coming together to procure goods at a volume discount and incentives for the creation of such groups. In other areas such as electricity networks (Contreras and Wu 2000) coalition formation has also been proposed to improve the efficiency of a system comprising several agents. In general, the desired goals of a coalition formation process include the following: (i) maximize coalition benefits or utility; (ii) divide the total utility among agents in a stable way, such that the agents in the coalition are not motivated to abandon it—stable payment configuration; and (iii) do it within a reasonable amount of time and with a reasonable amount of computational efforts.

Microeconomics studies interactions among self-interested partners, which is why a great number of cooperation mechanisms among self-interested agents use microeconomic techniques as their theoretical basis. Specifically, game theory studies mathematical models of cooperation and conflict in multipersonal situations (Aumann and Hart 1992). In these models, each of the participants tries to maximize its own utility, knowing that all the agents will adopt the same attitude. For this reason, game theory may be the right tool to design ideal self-interested interactions (Parsons, Gmytrasiewicz, and Wooldridge 2002).

A previous issue in this kind of problem is the computation of the utility obtained by each possible coalition. Since this computation is domain dependent, game theory does not...
address the issue and assumes that the utilities are given as data. However, in a concrete application, the generation of these data can become a very difficult task.

On the other hand, a coalition structure (i.e., a set of coalitions) must be selected for the problem. It is desirable to select the most profitable structure. However, it was shown in Sandholm et al. (1999) that any general algorithm for coalition formation guaranteeing a solution inside a neighborhood of the best one has to search in an exponential number of possible structures. Several paths have been taken to circumvent this intractability. Some of them are presented in Klusch and Shehory (1996b), Sandholm and Lesser (1997), and Shehory and Kraus (1998, 1999). For example, Sandholm et al. use computing costs as the term that modifies the utility of the rational agent, extending in this way the game theory framework, and presenting a coalition formation model for limited agents. Shehory et al. have proposed (Klusch and Shehory 1996b; Shehory and Kraus 1999) another anytime coalition formation algorithm. In this algorithm, the stability of the resulting utility distribution is defined in terms of the kernel and the computation has polynomial complexity. However, to achieve this, the number of agents in each coalition is limited, thus affecting the optimality of the result.

Another way to deal with this intractability is to consider only superadditive environments (Zlotkin and Rosenschein 1993, 1994; Ketchpel 1994, 1995; Shehory and Kraus 1995; Klusch and Shehory 1996a). This term is used in the game theory to name those environments where any two disjoint coalitions can do at least as well by joint effort as they can separately. This implies that the global coalition, that is, the coalition of all agents, is never worse that any other partition of the set of agents.

For each coalition structure, the benefits of every coalition must be distributed among its members, yielding a payment configuration. Game theory provides different concepts (core, kernel, Shapley value, etc.) (Kahan and Rapoport 1984) to describe the stability of coalitions in cooperative games with respect to the resultant payment configuration. Plainly speaking, a payment configuration belongs to the core if it leaves no coalition in a position to improve payoffs of all of its members by trying a different coalition. It is widely held (see, e.g., Kahan and Rapoport 1984, p. 60) that the core is the simplest to define and perhaps the most intuitively satisfying of these concepts. On the other hand, its existence is not guaranteed in the general case; for a given game the core can be empty.

From the point of view of computational MAS, these solutions can raise two kinds of problems: (i) it could be too complex to check the definition of these concepts for a given payoff division and (ii) many theories abstract from any underlying bargaining process, so they fail to explain how the players reach any solution or equilibrium. Exceptions are dynamic theories or transfer schemes for coalition formation, for example, Stearn’s transfer scheme for the kernel (Stearns 1968), Wu’s transfer scheme for the core (Wu 1977), and so on. However, these schemes may require an infinite number of iterations in order to converge and computational complexity may again be exponential.

Some prior computational works have addressed the problem of payoff division. For example, Rosenschein and Zlotkin in Zlotkin and Rosenschein (1994) give a mechanism for payoff division in concave superadditive task-oriented domains and prove that the expectations for calculated payments are the Shapley values for each agent. However, their results are only probabilistic, that is, they only hold for expected values, not for real ones. On the other hand, in Yamamoto and Sycara (2001) and Cuihong and Sycara (2002) the authors propose some buyer coalition schemes for e-marketplaces (a kind of reverse auction in Yamamoto and Sycara (2001), and a combinatorial auction in Cuihong and Sycara (2002)) and prove the stability of payoff division in terms of the core.

This paper is the continuation of a line of research (Belmonte 2002; Belmonte et al. 2002, 2004) about coalition formation in a certain class of superadditive task-oriented
domains (Rosenschein and Zlotkin 1994). The model proposed here guarantees optimum task allocation and a stable payoff division among the coalition members. Moreover, for this kind of domain, all the computations can be done in time polynomial with the number of agents.

The article is organized as follows: in Section 2 we present an example in order to motivate and understand the problem. In Section 3, after presenting some standard game-theoretical definitions, we define some new concepts to prove an abstract result regarding the existence of the core (Proposition 1). In Section 4, this abstract framework is applied to solving the problem stated in Section 2. Then we discuss the limitations of this approach in Section 5 and finish in Section 6 with some conclusions and future works.

2. A MOTIVATING PROBLEM

We start with a motivating example. Let us assume that there are a number of geographically dispersed hospitals in a certain region. Each hospital is responsible for the treatment of its district’s patients, and it has the necessary resources to treat different patient pathologies. However, each pathology treatment has a different cost in each hospital (due to different staff costs, hospitalization costs, infrastructure cost, etc.). Thus, it is possible to form a hospital coalition, in such a way that the coalition members transfer patients among themselves to minimize costs. Obviously, the costs of transferring patients must also be taken into account. These costs include physical transfer of patients, hospitalization, future patient posttreatment controls, etc.

Each pathology treatment is a different type of task. The estimated number of patients that have to be treated in each hospital is the initial task size (obviously, it is an integer number; however, if the number is large, we can approximate the problem by means of real magnitudes). The cost of a pathology treatment is the unitary execution cost. Let us assume that for each hospital different pathology treatments are independent, i.e., they make use of different resources; therefore, it can be assumed that each hospital has a maximum capacity for each pathology treatment and each task can be redistributed without taking into account the rest of the tasks. For this reason, we will assume in the following just one treatment, i.e., one task. Note that the costs of “physical” processes (i.e., transfer of patients) will be much larger than communication costs.

Let us abstract now from this concrete domain and let us assume that there are six agents, called Joe, Mary, Tim, Amy, Lisa, and John, who have to initially perform a certain amount of a generic task that we will call \(\text{task1}\). For each agent, its initial amount is less than its capacity; let us call the difference initial surplus. In Table 1 we show the initial amount of

| Table 1. Initial Amount of Task, Maximum Capacity, and Unitary Execution Cost of Each Agent for \(\text{task1}\) |
|-----------------|----------|------------|-------------|--------------|
| Task            | Unitary Cost | Max. Capacity | Initial Task | Initial Surplus |
| Joe \((a_1)\)  | 3769       | 624         | 450         | 174          |
| Mary \((a_2)\) | 5005       | 230         | 195         | 35           |
| Tim \((a_3)\)  | 4342       | 679         | 590         | 89           |
| Amy \((a_4)\)  | 8901       | 220         | 137         | 83           |
| Lisa \((a_5)\) | 6663       | 765         | 453         | 312          |
| John \((a_6)\) | 2981       | 645         | 320         | 325          |
task, the maximum capacity, the initial surplus, and the unitary execution cost of each agent. In addition, we assume that there exists a cost for transferring a task unit between each pair of agents (Table 2), and that the agents can communicate among themselves at a negligible cost.

Being aware of the possibility of improving their efficiency by redistributing some percentage of the task, the agents communicate and negotiate to reach a deal. A software architecture will be necessary to support communication and negotiation processes. In fact, we have developed such an architecture which is described elsewhere (Belmonte 2002; Belmonte et al. 2002).

There are several possibilities for our agents to redistribute the assigned task and save some costs, thus yielding a benefit. For example, they can try to apply a greedy algorithm (see, e.g., Shehory and Kraus 1998) to obtain a possible task allocation for task1 (Figure 1). Taking into account the unitary transfer benefits are shown in Table 3, first Amy would transfer 137 units to John, since the unitary benefit has a value of 5290, greater than any other in the table. Amy has transferred all of her task, so the best move is now to take 188 units from Lisa to John with a unitary benefit of 2842. Then Lisa transfers 174 units to Joe (unitary benefit of 2389), 89 units to Tim (unitary benefit of 2053), and her last 2 units to Mary (unitary benefit of 1162). Every remaining transfer has a negative value, hence we have finished. The flows are summarized in Figure 2 and the total benefit is 1859753. As we will show in Section 3, this value can be improved.

<table>
<thead>
<tr>
<th>From</th>
<th>Joe</th>
<th>Mary</th>
<th>Tim</th>
<th>Amy</th>
<th>Lisa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe(a1)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Mary(a2)</td>
<td>507</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Tim(a3)</td>
<td>373</td>
<td>629</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Amy(a4)</td>
<td>635</td>
<td>282</td>
<td>864</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Lisa(a5)</td>
<td>505</td>
<td>496</td>
<td>268</td>
<td>772</td>
<td>–</td>
</tr>
<tr>
<td>John(a6)</td>
<td>1058</td>
<td>558</td>
<td>1074</td>
<td>630</td>
<td>840</td>
</tr>
</tbody>
</table>

**TABLE 2. Costs for Transferring a Task Unit between Each Pair of Agents**

Figure 1. Greedy solution for the example.
Agents must agree on a task allocation, but they must also agree on the distribution of obtained benefits in a stable way. For example, if they agree on the redistribution just described but, say, Mary and John think that their payments are not big enough, they can try to make a private deal by transferring Mary’s task to John without considering the others’ concerns.

### 3. A GENERAL FRAMEWORK

In this section we give some standard game theory definitions (Definitions 1–5) that we use in our model. Then we present some new concepts (Definitions 6–8) and prove a new result regarding the existence of the core (Proposition 1).

Let us consider a set of $n$ agents $N = \{1, 2, \ldots, n\}$ that can communicate and reach a binding agreement.

**Definition 1.** A coalition $S$ is a subset of the set of players $N$.

Coalition formation is often studied using the concept of characteristic function games. In such games, the value of each coalition $S$ is calculated by a characteristic function $v(S)$, in such a way that the coalition value is independent of nonmember actions.
**Definition 2.** The characteristic function with transferable utility or coalitional value, \( v(S) \), of a game with a set of players \( N \) is a scalar function \( v : 2^N \rightarrow \mathbb{R} \).

Thus, a coalition \( S \) is a group of agents that decide to cooperate and \( v(S) \) is the total utility that the members of \( S \) can reach by coordinating among themselves and acting together. Henceforth, we will use the notation \((N, v)\) to denote a game in which \( N \) is the set of agents and \( v \) is the characteristic function or coalitional value. We will assume that for every \( S, v(S) \geq 0 \).

Payoff division is the process of dividing the utility or benefits of a coalition among its members. Let us continue with some standard definitions.

**Definition 3.** A coalition structure \( CS = (T_1, \ldots, T_m) \) is a partition of the set of players into coalitions.

That is, each set \( T_k \) is not empty, \( T_i \cap T_j = \emptyset \) for all \( i, j = 1, \ldots, m, i \neq j \), and \( \bigcup T_k = N \). For example, in a game with three agents, there are seven possible coalitions: \( \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\} \), and 5 coalition structures: \( CS_1 = \{(1), (2), (3)\}, CS_2 = \{(1, 2), (3)\}, CS_3 = \{(1, 3), (2)\}, CS_4 = \{(2, 3), (1)\}, \) and \( CS_5 = \{(1, 2, 3)\} \).

**Definition 4.** A payment configuration, \( PC \), is a pair \((x, CS)\) where \( x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^+ \) is a payment vector and \( CS \) is a coalition structure.

That is, a \( PC \) is simply a concrete payment vector \( x \), paired with a concrete coalition structure \( CS \). The problem is to distribute the utility \( v(S) \) of every coalition \( S \) in a stable way, so the agents in \( S \) have no cause to abandon it. The core (Kahan and Rapoport 1984) is the strongest of these classical solution concepts and for an arbitrary game \((N, v)\) may include none, one, or many payoff vectors \( x \). Formally:

**Definition 5.** Let \( S \) be any subset of \( N \) and let \((x, CS)\) be any \( PS \) on \( N \). Let us denote \( x(S) = \sum_{i \in S} x_i \). The core of a characteristic function game \((N, v)\) is the set of all the \( PC \)s \((x, CS)\) such that for all subset \( S \subseteq N \), (i) \( v(S) \leq x(S) \) (group rationality) and (ii) \( v(N) = x(S) \) (global rationality).

Thus, the coalitional value \( v(S) \) of each coalition \( S \) will be lower than or at least equal to the sum of the payoffs obtained by the coalition members in compliance with the actual \( CS \). In addition, only \( CS \)s that maximize the profit—i.e., that fulfill the global rationality—will be stable in the sense of the core. The intuitive meaning of this definition is that, since they are granted at least the same total amount, no subgroup of agents has cause to secede from the coalition structure \( CS \) to form an independent coalition \( S \).

Now let us present some original definitions needed for our goals. First, we will assume that each agent has some “structure,” i.e., that it can be described by the values taken by a set of \( m \) nonnegative attributes that represent the resources each agent is endowed with. Then we will group all agents together in an array. Formally:

**Definition 6.** A \( w \)-agent is a pair \((i, a_i)\) where \( i \in \mathbb{N} \) and \( a_i \in \mathbb{R}^+ \), \( m \in \mathbb{N} \). A \( w \)-set of \( n \) agents is a set \( N_w = \{(1, a_1), (2, a_2), \ldots, (n, a_n)\} \) of \( w \)-agents. A coalition is a subset \( S \subseteq N_w \).

If \( S = \{ (i_1, a_{i_1}), \ldots, (i_s, a_{i_s}) \} \), we will call \( \bar{S} = \{i_1, \ldots, i_s\} \).

For each \( w \)-set \( N_w \), \( W(i, j) \) will denote the value of the \( i \)th attribute for the \( j \)th agent. Let us define \( k(i, j) = (i - 1) \times m + 1 + j, at(k) = (k - 1) \div m + 1, ag(k) = (k - 1) \mod m \).
m + 1. In this way, every \( w \)-set defines a vector \( w \in \mathbb{R}^{+mn} \), \( w_k = W(at(k), ag(k)) \), and every vector \( w \in \mathbb{R}^{+mn} \) defines a \( w \)-set with \( W(i, j) = w_{k(i,j)} \).

**Definition 7.** A \( w \)-family of games in characteristic form is a function that assigns a game in characteristic form \((N_w, v_w)\) to every \( w \in \mathbb{R}^{+mn} \).

Let us consider a \( w \)-family of games. For every \( w \in \mathbb{R}^{+mn} \) the utility of the global coalition will be, in general, a different value \( v_w(N_w) \) for each \( w \). Let us call \( v(w) = v_w(N_w) \).

Let us consider a \( w \in \mathbb{R}^{+mn} \). Given \( v(w) \), the values of \( v_w(S) \) are, in general, undefined for \( S \neq N_w \). However, let us assume that the inclusion of agents with no resources into a coalition \( S \) does not affect the coalitional value \( v_w(S) \). That means that in this game it is the same (i) to remove a set \( A \) of agents from \( S \) and (ii) to consider that the values of the attributes of the agents in \( A \) are all 0. Let us call such a case a simplifiable \( w \)-family. Formally:

**Definition 8.** A \( w \)-family of games \((N_w, v_w)\) in characteristic form is simplifiable when for every coalition \( S \subseteq N_w \), \( v_w(S) = V(w^S) \), where \( w^S_k = w_k \) if \( ag(k) \in S \), otherwise \( w^S_k = 0 \).

Therefore, a simplifiable \( w \)-family of games is completely described by the function \( V(w) \). Now we will prove a sufficient condition for \( V(w) \) in order to ensure that the core of the game \((N_w, v_w)\) is not empty. In fact, we will explicitly construct a payment configuration and prove that it lies inside the core.

**Proposition 1.** Let \( V(w) \) describe a simplifiable \( w \)-family of games. Let us assume that \( V \) is homogeneous degree 1 and concave. Then, for almost every \( w \), the payment configuration given by \( x_i = \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right) \) for each \( i = 1, \ldots, n \) lies inside the core of the game \((N_w, v_w)\).

**Proof.** We must prove that (i) the payments are well defined; (ii) the payment configuration is global rational; (iii) it is also group rational. For (i), note that a concave function \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is differentiable almost everywhere. For (ii), homogeneity and Euler’s theorem guarantee that

\[
\sum_{1 \leq i \leq n} x_i = \sum_{1 \leq i \leq n} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right) = \sum_{1 \leq k \leq n \times n} w_k \left( \frac{\partial V}{\partial w_k} \right) = V(w) = v(N_w).
\]

For (iii), consider that for any coalition \( S \subseteq N_w \), by concavity of \( V \) and the definition of \( w^S \) it holds that

\[
v(N) - v(S) = V(w) - V(w^S) \geq \sum_{i \not\in S} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right).
\]

On the other hand,

\[
x(S) = \sum_{i \in S} x_i
\]

\[
= \sum_{i \in S} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right) (i = 1, \ldots, n)
\]

\[
= \sum_{1 \leq i \leq n} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right) - \sum_{i \not\in S} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right)
\]

\[
= v(N) - \sum_{i \not\in S} \sum_{ag(k) = i} w_k \left( \frac{\partial V}{\partial w_k} \right),
\]

so \( v(N) - v(S) \geq v(N) - x(S) \) and \( v(S) \leq x(S) \).
4. BACK TO THE PROBLEM

In this section, the problem in Section 2 will be described from an abstract point of view as a task-oriented domain where there are initial assignments of task and initial surpluses of capacity. It will be shown that, assuming negligible communication and coordination costs, the problem of computing the coalitional value of the global coalition is reduced to the resolution of a linear program (1)–(3) whose size is polynomial with the numbers of agents; therefore, it can be solved in polynomial time with the number of agents (Proposition 2). Then the problem will be cast into the formal framework described in Section 3 to show the existence and computability of a payment vector lying inside the core (Propositions 6 and 7).

Let us introduce some notation and terminology. Each agent $a_i$ must perform a certain initial amount $t_i^a$ of task units. Agent $a_i$ can perform a maximum of $k_i$ task units ($t_i^a \leq k_i$) at a unitary cost of $\gamma_i$. We will define the surplus capacity $h_i$ of $a_i$ by $h_i = k_i - t_i^a$ ($0 \leq h_i$) (see Table 1). The agents can communicate and change the amounts of task initially assigned to each one, in order to achieve a better global efficiency; for any $i, j, 1 \leq i, j \leq n, i \neq j$, let $t_{ij}$ be the number of task units transferred from $a_i$ to $a_j$, $t_{ij} \geq 0$. In this way, there are $n(n - 1)$ variables representing the transfers.

To allow this new deal, the agents must incur a transfer cost; namely, we will assume that the cost of transferring a task unit from $a_i$ to $a_j$ is $\gamma_{ij} > 0$. In this way, the unitary profit $\beta_{ij}$ attached to the transfer of a task unit between $a_i$ and $a_j$ is $\beta_{ij} = \gamma_i - \gamma_j - \gamma_{ij}$ and $B_{ij} = \beta_{ij} \times t_{ij}$ is the transfer profit. Note that $\beta_{ij}$ can be negative, when $\gamma_i - \gamma_j < \gamma_{ij}$. Unitary profits for our example are displayed in Table 3. The global profit will be $B = \sum_k \sum_{j \in S_k} \sum_{i \in S_k, i \neq j} \beta_{ij} \times t_{ij}$, where $S_k$ is a coalition of agents.

If we assume that the coalition formation costs (communication and coordination) are negligible compared to actual task transfer costs (physical transfer of materials), then the global coalition $N$ will allow for more transfers than any other coalition structure, without incurring additional costs and the former expression becomes $B = \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n, i \neq j} \beta_{ij} \times t_{ij}$. The coalitional value of $N$ will be defined as the optimum value $V$ of $B$, $V = \text{MAX} \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n, i \neq j} \beta_{ij} \times t_{ij}$ over all feasible sets of transfers.

A set of transfers is feasible if and only if the following conservation and capacity constraints hold for every agent $a_i$:

---

For every agent $a_i (1 \leq i \leq n)$, the task assigned after the transfers is not negative, i.e.,

$$0 \leq t_i^a - \sum_{1 \leq j \leq n, i \neq j} t_{ij} + \sum_{1 \leq j \leq n, i \neq j} t_{ji},$$

that is,

$$\sum_{1 \leq j \leq n, i \neq j} t_{ij} - \sum_{1 \leq j \leq n, i \neq j} t_{ji} \leq t_i^a.$$

---

For every agent $a_i (1 \leq i \leq n)$, the task assigned after the transfers is not greater than the total capacity $k_i = t_i^a + h_i$, i.e.,

$$t_i^a - \sum_{1 \leq j \leq n, i \neq j} t_{ij} + \sum_{1 \leq j \leq n, i \neq j} t_{ji} \leq k_i,$$

that is,

$$- \sum_{1 \leq j \leq n, i \neq j} t_{ij} + \sum_{1 \leq j \leq n, i \neq j} t_{ji} \leq h_i.$$
This value can be computed as the solution of the following linear program on \( n(n - 1) \) non-negative variables \( t_{ij} \) (\( 1 \leq i \leq n, \ i \neq j \)) and \( 2n \) constraints:

\[
\text{MAX} \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n, i \neq j} \beta_{ij} \times t_{ij}
\]

(1)

with every \( t_{ij} \geq 0 \) and subject to

\[
\sum_{1 \leq j \leq n, i \neq j} t_{ij} - \sum_{1 \leq j \leq n, i \neq j} t_{ji} \leq t_i^a \quad (i = 1, \ldots, n)
\]

(2)

and

\[
- \sum_{1 \leq j \leq n, i \neq j} t_{ij} + \sum_{1 \leq j \leq n, i \neq j} t_{ji} \leq h_i \quad (i = 1, \ldots, n).
\]

(3)

Let us apply this approach to our example (see Section 2 and Tables 1–3). There are \( 6 \times 5 = 30 \) variables \( t_{12}, \ldots, t_{65} \). The linear program is

\[
\text{MAX} \quad -1743t_{12} - 946t_{13} - 5767t_{14} - 3399t_{15} - 270t_{16} + 729t_{21} + 34t_{23} - 4178t_{24} - 2154t_{25} + 1466t_{26} + 200t_{31} - 1292t_{32} - 5423t_{34} - 2589t_{35} + 287t_{36} + 4497t_{41} + 3614t_{42} + 3695t_{43} + 1466t_{45} + 5290t_{46} + 2389t_{51} + 1162t_{52} + 2053t_{53} - 3010t_{54} + 2842t_{56} - 1846t_{61} - 2582t_{62} - 2435t_{63} - 6550t_{64} - 452t_{65}
\]

(4)

with every \( t_{ij} \geq 0 \) and subject to the following set of \( 2 \times 6 = 12 \) inequalities:

\[
t_{12} + t_{13} + t_{14} + t_{15} + t_{16} - t_{21} - t_{31} - t_{41} - t_{51} - t_{61} \leq 450
\]

\[
t_{21} + t_{23} + t_{24} + t_{25} + t_{26} - t_{12} - t_{32} - t_{42} - t_{52} - t_{62} \leq 195
\]

\[
t_{31} + t_{32} + t_{34} + t_{35} + t_{36} - t_{13} - t_{23} - t_{43} - t_{53} - t_{63} \leq 590
\]

\[
t_{41} + t_{42} + t_{43} + t_{45} + t_{46} - t_{14} - t_{24} - t_{34} - t_{54} - t_{64} \leq 137
\]

\[
t_{51} + t_{52} + t_{53} + t_{54} + t_{56} - t_{15} - t_{25} - t_{35} - t_{45} - t_{65} \leq 453
\]

\[
t_{61} + t_{62} + t_{63} + t_{64} + t_{65} - t_{16} - t_{26} - t_{36} - t_{46} - t_{56} \leq 320
\]

\[
-t_{12} - t_{13} - t_{14} - t_{15} - t_{16} + t_{21} + t_{31} + t_{41} + t_{51} + t_{61} \leq 174
\]

\[
-t_{21} - t_{23} - t_{24} - t_{25} - t_{26} + t_{12} + t_{32} + t_{42} + t_{52} + t_{62} \leq 35
\]

\[
-t_{31} - t_{32} - t_{34} - t_{35} - t_{36} + t_{13} + t_{23} + t_{43} + t_{53} + t_{63} \leq 89
\]

\[
-t_{41} - t_{42} - t_{43} - t_{45} - t_{46} + t_{14} + t_{24} + t_{34} + t_{54} + t_{64} \leq 83
\]

\[
-t_{51} - t_{52} - t_{53} - t_{54} - t_{56} + t_{15} + t_{25} + t_{35} + t_{45} + t_{65} \leq 312
\]

\[
-t_{61} - t_{62} - t_{63} - t_{64} - t_{65} + t_{16} + t_{26} + t_{36} + t_{46} + t_{56} \leq 325.
\]

(5)

The program (4) and (5) has a solution given by

\[
V = 1859761
\]

\[
t_{24} = -2; \quad t_{46} = 135; \quad t_{15} = -174; \quad t_{35} = -89; \quad t_{56} = 190; \quad \text{and for the rest} \quad t_{ij} = 0.
\]

(6)
Note that $V$ is greater than the value computed in Section 2. We can depict this solution in a more intuitive way (Figure 2). In the final assignment, Amy ($a_4$) and Lisa ($a_5$) have no task to perform. On the other hand, John ($a_6$), Tim ($a_3$), and Joe ($a_1$) have as much task as they can perform. Note that in an optimal assignment there can be no agents $a_i$, $a_j$ such that both $t_{ij}$ and $t_{ji}$ are positive. Moreover, in our problem there are no agents $a_i$, $a_j$, $a_k$ such that $t_{ij} > 0$ and $t_{ik} < 0$. In the optimal assignment (i.e., such that there are flows both from $a_i$ and to $a_j$), it can be proven that it always is the case when strict triangular inequality holds, i.e., for all $i$, $j$, $k$, $\gamma_{ik} < \gamma_{ij} + \gamma_{jk}$.

In general, programs arising from the problems defined in Section 2 always have a finite solution.

Proposition 2. Let us assume that for every agent $a_i (1 \leq i \leq n)$, $t_i$, $h_i \geq 0$. Then the linear program (1)–(3) always has a finite solution. Moreover, the solution can be computed in time polynomial with the number of agents.

Proof. First, note that if for every agent $a_i (1 \leq i \leq n)$, $t_i$, $h_i \geq 0$, then the region $S$ defined by (2) and (3) is nonempty, since always $0 \in S$. On the other hand, let us show that the objective function $B$ is bounded in $S$. Let $T$ be the total amount of task to be performed, i.e., $T = \sum_{1 \leq i \leq n} t_i$. Let $\Gamma$ be the maximum difference in efficiency between any pair of agents, i.e., $\Gamma = \max_{1 \leq i \leq n} \gamma_i - \min_{1 \leq i \leq n} \gamma_i$. It is obvious that the value $T \times \Gamma$ is better than any benefit obtained by a set of transfers satisfying the constraints (2) and (3). Therefore, there is an upper bound for the values of $B$ in $S$, so $B$ reaches a maximum in $S$ and the linear program (1)–(3) has a finite solution.

On the other hand, it is well known that there are algorithms (see, e.g., Schrijver 1986) that solve any linear program in (worst-case) time polynomial in terms of the number of variables and constraints; hence, for the program (1)–(3), in terms of the number of agents.

In this way, we have shown a feasible method for computing the co-coalitional value of the global coalition. We will show now a feasible method for computing a payment vector lying inside the core. The tools are those given in Section 3; in order to use them, we must cast our problem into that formal framework. Let us start with some notational conventions.

In the following reasoning we will represent the program (1)–(3) in the compact notation

$$\begin{align*}
\text{MAX } c\mathbf{x} & \quad \text{subject to } A\mathbf{x} \leq \mathbf{b} \quad \text{with } \mathbf{x} \geq 0, \\
\text{MIN } b\mathbf{y} & \quad \text{subject to } A^t\mathbf{y} \geq \mathbf{c} \quad \text{with } \mathbf{y} \geq 0,
\end{align*}$$

whose dual program is

$$\begin{align*}
\text{MIN } b\mathbf{y} & \quad \text{subject to } A^t\mathbf{y} \geq \mathbf{c} \quad \text{with } \mathbf{y} \geq 0,
\end{align*}$$

where $A$ is a $2n \times n(n - 1)$ matrix, $b$ is a $2n$-dimensional vector, and $c$, $x$ are $n(n - 1)$ dimensional vectors. The mapping between $x$ and transfer variables $t_{ij}$ is given by $x_1 = t_{12}$, $x_2 = t_{13}$, $\ldots$, $x_{n \times (n-1)} = t_{n-n-1}$, or explicitly $x_i = t_{pq}$ where $i = (n - 1) \times (p - 1) + q$, if $p > q$, $i = (n - 1) \times (p - 1) + q - 1$, if $p < q$; and analogously for $c_i$. On the other hand, $b_i = t_i^a$ if $i \leq n$, otherwise $b_i = h_{i-n}$. Finally, the mapping between $A$ and the coefficients in (2) and (3) is given as follows: let $p = (j - 1) \div (n - 1) + 1$, $q = (j - 1) \mod (n - 1) + 1$. Then, for the upper half of $A$, i.e., for $i \leq n$,

$$A_{ij} = \begin{cases} 1 & \text{if } i = p \\ -1 & \text{if } i < p \text{ and } i = q \\ 0 & \text{otherwise} \end{cases}$$
and for the lower half

\[
A_{n+i,j} = \begin{cases} 
-1 & \text{if } i = p \\
1 & \text{if } i < p \text{ and } i = q \\
0 & \text{otherwise}.
\end{cases}
\]

Notice that \( A \) is always the same for every set of \( n \) agents and \( c \) is independent of agent descriptions given by \( t_i^a, h_i \). In this way, only \( b \) depends on agent resources \( t_a^i, h_i \).

We can now cast our problem into the formal framework described in Section 3. Each agent \( a_i \) can be described by two resources, namely its initial task \( t^a_i \) and its surplus \( h_i \). It is clear, then, that the resource vector \( w \) has \( 2n \) components and \( w_i = t^a_i, w_{i+n} = h_i (i \leq n) \). We want to compute a payment vector inside the core. Proposition 1 provides a method; however, we must show that necessary conditions hold. Some preliminary results are needed. We will use the following result, as stated by Greenberg (1997), ultimately based on results of Mills (1956).

Let us arrange \( A, b, c \) in a matrix

\[
M = \begin{pmatrix} A & b \\ c & 0 \end{pmatrix}.
\]

Assume that the program (7) defined by \( M \) has an optimal solution. \( \Delta M \) is an admissible change when the program defined by \( M + \Delta M \) has an optimal solution. \( \Delta M \) is an admissible direction if there exists \( \theta^* > 0 \) such that \( \theta^* \delta M \) is an admissible change.

**Proposition 3** (Greenberg 1997, pp. 3–6). Let \( V \) be the optimum value for the linear program \( P \) given by (7). Let \( G \) be the set of solutions to the dual program given by (8). Then, when the associate direction is admissible,

\[
\frac{\partial V}{\partial b^+_i} = \min_{y \in G} y_i; \quad \frac{\partial V}{\partial b^-_i} = \max_{y \in G} y_i.
\]

We will also make use of the following result:

**Proposition 4** (Gal 1995, p. 180). Let \( V(\lambda) \) be the optimum value for the linear program given by \( A, b + F \lambda, c \). Let \( K \) be the set of all vector parameters \( \lambda \) such that there exists a finite optimal solution of the problem given by \( A, b + F \lambda, c \). Then the function \( V(\lambda) \) is concave over \( K \).

Now we can prove the following lemma:

**Proposition 5.** Let \( w_i = t^a_i > 0, w_{n+i} = h_i > 0 \). Then \( V(w) \) is (i) homogeneous of degree 1 and (ii) concave.

**Proof.** For (i), consider the dual program (8). The minimum found for (8) will be the maximum for the primal one (7). Assume \( w' = \alpha w \), i.e., all initial tasks and surpluses are multiplied by \( \alpha \). Then, \( V(w') \) is the optimum for a problem

\[
\text{MIN } \alpha b y \quad \text{subject to } A_{ij}' \leq c.
\]

Then the values of \( A, c \) are not modified; hence the feasible region of the dual program (9) is the same as the feasible region of (8). On the other hand, the objective function is that of
(8) multiplied by \(\alpha\). It follows that the minimum value for (9) is that of (8) multiplied by \(\alpha\), i.e., \(V(w') = \alpha \cdot V(w)\).

For (ii), consider that \(w = b\). Then, by Proposition 4, \(V(w)\) is concave over \(K = \mathbb{R}^{+2n}\). 

We can now prove the desired results.

**Proposition 6.** Let \(w\) be the vector of initial tasks and surpluses. Assume that for each \(k (1 \leq k \leq 2n)\), \(\frac{\partial V}{\partial w_i} = \frac{\partial V}{\partial w_j}\). Then the payment vector given by

\[
x_i = w_i \frac{\partial V}{\partial w_i} + w_{n+i} \frac{\partial V}{\partial w_{n+i}} \quad (i = 1, \ldots, n)
\]

(10)

lies inside the core.

**Proof.** Note that \(V\) is concave, therefore the existence of partial derivatives ensures differentiability. Then, the desired conclusion follows from Propositions 1 and 5.

**Proposition 7.** The payment vector given by equation (10) is computable in time polynomial in terms of the number \(n\) of agents.

**Proof.** By Proposition 3, given the set of solutions to the dual program, the directional partial derivatives are also given. But these solutions can be obtained in time polynomial with the number of agents, as stated in Proposition 2.

Let us apply this method to our example. We have computed in (6) the optimal task transfers. But, at the same time, the algorithm gave us the corresponding “shadow prices” or Lagrange multipliers for the constraints, i.e., the optimal solutions of the dual program (we only show nonzero values):

<table>
<thead>
<tr>
<th>Dual Variable</th>
<th>Value</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>(y_4)</td>
<td>3614</td>
<td>(t'^4)</td>
</tr>
<tr>
<td>(y_5)</td>
<td>1166</td>
<td>(t'^5)</td>
</tr>
<tr>
<td>(y_7)</td>
<td>1223</td>
<td>(h_1)</td>
</tr>
<tr>
<td>(y_9)</td>
<td>887</td>
<td>(h_3)</td>
</tr>
<tr>
<td>(y_{12})</td>
<td>1676</td>
<td>(h_6)</td>
</tr>
</tbody>
</table>

(11)

If we look at Figure 2, the intuitive meaning of these values becomes apparent. For example, consider the dual variable \(y_{10}\). It corresponds to the resource \(h_4\), i.e., the capacity surplus of \(a_4\). Since this surplus has no relevance at all in the solution (\(a_4\) is a task dispatcher, not a receiver), \(\frac{\partial V}{\partial h_4} = y_{10} = 0\). On the other hand, consider the dual variable \(y_4\), corresponding to \(t'^4\). A small variation \(\epsilon\) in \(t'^4\) implies the same variation \(\epsilon\) in the transfer \(t_4\); and, since the unitary transfer profit \(b_{42}\) (see Table 3) is 3614, it implies a variation 3614\(\epsilon\) in the optimal value; this is consistent with the value computed for \(y_4\). In the same way, with a little effort we could explain every value in (11) in terms of flows in Figure 2 and unitary profits in Table 3.
Now, by applying Proposition 6, we can compute the payment vector $x$:

\begin{align*}
    x_1 &= y_1 \times t_1^a + y_7 \times h_1 = 212802 \\
    x_2 &= y_2 \times t_2^a + y_8 \times h_2 = 0 \\
    x_3 &= y_3 \times t_3^a + y_9 \times h_3 = 78943 \\
    x_4 &= y_4 \times t_4^a + y_{10} \times h_4 = 495118 \\
    x_5 &= y_5 \times t_5^a + y_{11} \times h_5 = 528198 \\
    x_6 &= y_6 \times t_6^a + y_{12} \times h_6 = 544700.
\end{align*}

It is easy to check global rationality, i.e., that $V = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$. More cumbersome, but equally easy, is to check group rationality. For example, the coalition $\{a_1, a_2\}$ has a coalitional value of 126846 (i.e., $a_1$ and $a_2$ could get this global benefit by redistributing their task); on the other hand, $x_1 + x_2 = 212802 > 126846$.

Proposition 1 (and also Proposition 3) ensures that differentiability holds almost everywhere; in this case, where the resource vector $b$ is such that the linear program is not degenerate. Let us consider now a degenerate case. For example, let us assume two agents, $t_1 = h_2 = T, h_1 = t_2 = 0, \beta_{12} > 0, \beta_{12} < 0$. It is obvious that the optimal solution is given by $t_{12} = T$ and the global profit is $V = \beta_{12}T$. If we compute the partial directional derivatives, we obtain

\[
\frac{\partial V}{\partial t_1^+} = \frac{\partial V}{\partial h_2^-} = 0; \quad \frac{\partial V}{\partial t_1^+} = \frac{\partial V}{\partial h_1^+} = \beta_{12}^i; \quad \frac{\partial V}{\partial t_2^+} = \frac{\partial V}{\partial t_2^-} = 0.
\]

$V$ is not differentiable at $(T, 0, 0, T)$ and Proposition 6 is not applicable. However, note that—in this case—a suitable choice of directional partial derivatives yields a payment configuration inside the core. If for instance, $x_1 = \frac{\partial V}{\partial t_1^-} t_1 + \frac{\partial V}{\partial h_1^-} h_1 = 0$ and $x_2 = \frac{\partial V}{\partial t_2^+} t_2 + \frac{\partial V}{\partial h_2^+} h_2 = \beta_{12} T$, then $(x_1, x_2)$ satisfies the conditions of global and group rationality, i.e., $x_1 + x_2 = V, x_1 \geq 0, x_2 \geq 0$.

5. DISCUSSION

For the problem stated in Section 4, we have shown a method to (i) compute an optimal allocation of task, i.e., the greatest utility for the agents; and (ii) compute a distribution of the obtained utility that lies inside the core, i.e., such that the agents have no reason to abandon the coalition. Moreover, the method is effective, i.e., polynomial in terms of the number of agents. However, it must be admitted that many simplifying—although natural—assumptions have been made when defining the problem. A reasonable concern is then to question if the assumptions are unrealistic, and what the consequences of withdrawing them are.

First at all, we have assumed that the costs of “physical” processes will be much larger than coordination and communication costs. In this case, the agents can avoid the process of finding an optimal coalition structure, since they know that the alternative of grouping together is at least as profitable as any other, i.e., they assume superadditivity. Note that the problem of finding the optimal coalition structure is, in general, intractable, as shown by Sandholm et al. (1999). In our opinion, this assumption can be justified in cases where agents are connected via Internet and can communicate at a negligible cost. It would be interesting to study the threshold value of communication costs for which the assumption of superadditivity becomes unrealistic.
On the other hand, we have stated the problem of finding the optimal allocation of task as one of linear programming. This amounts to the feasibility of the computations needed; should the problem be, for example, one of integer programming, the computations would become intractable. Nevertheless, it is possible to pose or approximate many problems by means of linear-programming techniques.

However, the crux of the matter is the feasible computation of a payment configuration that lies inside the core. Our Proposition 1 sets a sufficient condition for a configuration to lie inside the core. This condition comprises two features: (a) homogeneity of degree 1 and (b) concavity. Concerning (b), it is common in real environments that additional inputs yield progressively smaller increments of benefits, so implying a concave \( V(w) \). Thus, the restriction will be usually given by (a); and, if we are able to find a vector \( w \) such that \( V(w) \) is homogeneous of degree 1, then it is very likely that we can compute a payment configuration inside the core.

Let us sketch how the results of Section 3 and the modelizations shown in Section 4 could be applied to more complex cases. As a first example, let us assume that agent \( a_i \) can perform a maximum of \( k_i' \) task units at a unitary cost of \( \gamma_i' \) and a maximum of \( k_i'' \) additional task units at a unitary cost of \( \gamma_i'' \). We can “split” \( a_i \) into two agents \( a_i' \) and \( a_i'' \) with capacities \( k_i' \) and \( k_i'' \), and costs \( \gamma_i' \) and \( \gamma_i'' \), respectively. Transfer costs between both \( a_i' \) and \( a_i'' \) will be \( \infty \). In this way, we have reduced the problem to the one solved in Section 4. Notice that each agent’s attributes (that determine its remuneration) are now its initial task and surplus below and above \( k_i' \).

As another example, let us consider a more realistic problem: we have assumed that different kinds of tasks are independent, therefore that each kind of task can be redistributed without taking into account the other kinds of task. For this reason, only one task has been considered. What happens if it is not the case, i.e., if different tasks \( \{t_1, \ldots, t_p\} \) make use of the same resources \( \{r_1, \ldots, r_m\} \) of an agent? Then we must consider at once all tasks and all transfers; we have \( n \times (n - 1) \times p \) variables (transfers) and \( n \times p + n \times r \) constraints: (i) for each agent, each kind of task must be nonnegative and (ii) for each agent and each resource, the consumption of the resource must not exceed its available amount. A linear program can be written that expresses these constraints and computes the optimal joint distribution of tasks. Note that each agent’s attributes are now its initial tasks and its initial surpluses in each resource \( \{r_1, \ldots, r_m\} \).

On the other hand, if communication and coordination costs are not negligible, all these techniques become useless. In general, it would be necessary to compute an exponential number of coalitional values and consider an exponential number of coalition structures.

Finally, two issues must be considered when defining a coalition formation procedure. First, the robustness of the model in view of the failure of one or more system agents must be analyzed, due to the fact that a service provided by a MAS must be maintained, even with a bad quality, when faced with the failure of processors, communication networks, or agents. On the other hand, since agents have some kind of execution autonomy and are self-interested, they can deceive or mislead each other when they reveal their information if they believe that they will obtain more profits doing so. So it is necessary to study the effects of deception and manipulation on the model. These analyses have been performed by means of simulations and are described elsewhere (Belmonte et al. 2004, 2006).

6. CONCLUSIONS

Coalition formation is an important mechanism for cooperation in multiagent systems. Self-interested autonomous agents forming coalitions may improve their benefits and abilities
to satisfy their goals. Game theory techniques are an appropriate technical basis for this cooperation mechanism, as they can be proven to have certain desirable properties including, for instance, that agreements which are acceptable to all participants can be guaranteed to be reached.

We have restricted our attention to superadditive task-oriented domains, where the global coalition is always the most profitable. We have defined the concept of \( w \)-family of games, i.e., of games in which the agents can be described by the values taken by a set of \( m \) nonnegative attributes that represent the resources \( w \) each agent is endowed with. When the function \( V(w) \) that gives the global benefit is homogeneous of degree 1 and concave, we have proved that a certain payment configuration—defined in terms of the partial derivatives of \( V(w) \)—lies inside the core.

When the computation of the optimal transfer of task can be stated as a linear-programming problem in which the costs are minimized subject to the capacity constraints of each agent, we have proved, by applying the former results, that there exists a procedure that guarantees in polynomial time optimum task allocation and a stable—in the sense given by the core—payoff division among the coalition members.

As pointed out in Section 5, further research is needed to extend our analysis to other more realistic cases. In particular, we are planning to relax several assumptions that do no hold in some real-world domains where coalitions are necessary (Blankenburg, Klusch, and Shehory 2003; Klusch and Gerber 2002), for example, complete information or bounded time. However, this will inevitably require that the optimality of the result be compromised. In addition, we are trying to impose new constraints, such as limiting the quantity of task transferred among agents, in such a way that a broader range of problems could be modeled.

REFERENCES


