Distributed Matching between Individuals and Activities with Additively Separable Preferences

Maxime Morge and Antoine Nongaillard

{maxime.morge,antoine.nongaillard}@univ-lille1.fr
CRIStAL, Univ. Lille

June 23, 2017

Abstract

We aim at providing a social network such that users form groups to practice together some activities. In this paper, we introduce a formal framework for coalition formation which is suitable for our usecase. We restrict ourselves to additively separable preferences in order to propose a distributed matching algorithm. We demonstrate that its outcome is Pareto-optimal. Our experiments shows we reach a better outcome than the classical local search techniques and that the distribution of our algorithm speeds up its runtime.

1 Introduction

Multiagent systems (MAS) is a relevant paradigm for the analysis, the design and the implementation of systems composed of autonomous entities in interaction. In order to design socio-technical systems for mediation or simulation, MAS allow to model the feedback loops between heterogeneous actors whose local decision-making leads to complex global phenomena. In this perspective, one of the challenges of the MAS community is to facilitate the elicitation of preferences.

Our work is part of a project which aims at understanding and modelling the dynamic of feedback occurring in a group of individuals interacting both within a virtual and a real social network. The usecase on which we are focusing here consists of forming some groups of seniors (60-70 years old) in order to share some activities. This scenario involves several thousand users. We aim at maximizing the satisfaction of users in order to improve social cohesion and avoid the isolation of seniors. We want to propose a social network such that the users form groups to practice together some activities. The purpose of this system is to suggest to each user some peers with whom practicing these activities.
In this paper, we introduce a formal framework for coalition formation which is suitable for our usecase. A set of individuals must be matched together in order to share one of the activities with respect to the preferences of individuals for their peers and for the activities. We restrict ourselves to additively separable preferences in order to propose an algorithm which returns a “good” matching where none activity is oversubscribed. We demonstrate the outcome of our algorithm is Pareto-optimal. By adopting the model of actor [3], we are able to distribute this algorithm. Our experiments show that the solution of our algorithm has a better quality than those obtained with the classical local search techniques. Moreover, its distribution speeds up its runtime (up to 3.5 times).

Section 2 compares our approach with the related works. We introduce our matching problem in Section 3. We propose a matching algorithm in Section 4. The distribution of this algorithm consists of agent behaviours described in Section 5. Section 6 exhibits our empirical results. Finally, we conclude with some directions for future work (cf Section 7).

2 Related works

Social choice theory aims at designing and analyzing collective decision-making processes which imply a set of agents selecting or classifying a subset of alternatives among the available ones. Contrary to Economy, Computer science is concerned in this field by the study of algorithms in order to propose operational methods. We focus here on a particular matching problem.

In the problem of hedonic coalition formation, which has been formalized in [5], each player is endowed with a single preference relation over all the coalitions which contain this player. Our problem is a specialization of the hedonic coalition problem. We can represent our problem as a hedonic game by creating one player for each activity which prefers not to be oversubscribed and one player for each individual with preferences aggregating the preferences over the activities and the preferences over its peers. We consider here these two preferences are independent, then we aggregate them in an utility function. Under this assumption, our setting has the useful structural properties of two-sided matching that distinguish it from a generic hedonic game as stated by our experiments. The fact that the activities play the role of focal points [13] and the succinct representation of individual’s preferences allow us to propose an algorithm to reach “good” matchings.

The problem of group activity selection has been proposed in [4]. In such a problem, each player participates in at most one activity and its preferences over activities depend on the number of participants in the activity. This is a generalization of anonymous hedonic games. Even if this problem has been extended [8] in order to take into account the relationships among the agents, the latter are encoded by a social network, i.e. an undirected graph where nodes correspond to agents and edges represent communication links between them. By contrast, in our problem, each individual is endowed with a preference relation over the activities and a preference relation over its peers.
The Hospital/Resident (HR) problem has been introduced in [7]. This problem is a specialization of the coalition formation game where a set of residents must be assigned to the hospitals in accordance with the preferences of the residents over the hospitals and the preferences of hospitals over the residents. The HR problem has many extensions [9]. To the best of our knowledge, none extension is suitable for our usecase.

How does an agent evaluate its preferences? We adopt here utility functions (cardinal preferences). For the sake of simplicity, we assume some additively separable preferences and that the evaluation of activities and individuals are comparable. Even if its expressiveness is limited, the representation of our preferences is linear with respect to the number of individuals.

What is the “best” solution for collective decision-making problem? In the literature, two kinds of rules derivate the social choice from the individual preferences: the first ones are based on the desirable properties of the solution (e.g. stability or Pareto-optimality), while the latter are based on the aggregation of the individual preferences (e.g. the utilitarian welfare). In this paper, we adopt the second approach since none concept from the first approach is suitable.

How to reach a matching which maximizes the utilitarian welfare? As stated later in this paper, this problem can be NP-hard. For this purpose, we can consider Distributed Constraint Optimization Problem (DCOP) algorithm or Local Search Techniques (LST). It has been shown in [6] that DCOP algorithms are not necessarily scalable for matching problems. We show here that LST are not suitable since the function to be optimized consists of many local optima. That is the reason why we adopt a multiagent approach, in particular a multi-level model as recommended by [10] where each “activity” agent represents a group of individuals.

3 IA Problem

We aim at providing a formal framework for coalition formation which is suitable for our usecase. We introduce here the individuals/activities (IA) problem. In such a problem, the individuals selects their favorite activities with the partners they prefer.

Definition 1 (IA Problem) An individuals/activities (IA) problem of size \((m,n)\), with \(m \geq 1\) and \(n \geq 1\), is a couple \(IA = (I,A)\) with \(m\) individuals and \(n\) activities, where:

- \(A = \{a_1, \ldots, a_n\}\) is a set of \(n\) activities. Each activity \(a_j\) has a positive integral capacity \(c_j\);
- \(I = \{1, \ldots, m\}\) is a set of \(m\) individuals. Each individual \(i\) is endowed with:

  1. a preference relation over the activities, i.e. a reflexive, complete and transitive preference ordering \(\succeq_i\) over the activities \(A \cup \{\emptyset\}\), including
the void activity (denoted \(\theta\)). The corresponding strict preference relation is denoted by \(\succ_i\).

2. a purely hedonic preference, i.e. a reflexive, complete and transitive preference ordering \(\succeq_i\) over the set of groups it belongs \(G(i) = \{G \subseteq I; i \in G\}\). The corresponding strict preference relation is denoted by \(\succ_i\).

Intuitively, the void activity corresponds to do nothing.

We aim at forming coalitions of individuals around the activities.

**Definition 2 (Coalition)** Let \(IA = \langle I, A \rangle\) be an IA problem. A coalition is a couple \(C = \langle a, G \rangle\) where \(a \in A \cup \{\theta\}\) and \(G \subseteq I\). The activity of a coalition \(C\) is \(a\) with a capacity \(1\) and the group \(G_{C}\). A non-empty coalition \(C\) is such that \(G_{C} \neq \emptyset\) and \(C\) is for \(i\) if \(i \in G_{C}\).

We expect the number of individuals to be considerably larger than the number of available activities \((m >> n)\).

**Definition 3 (Matching)** A matching \(M\) for the problem \(IA = \langle I, A \rangle\) is represented by the functions \(a_{M} : I \to A \cup \{\theta\}\) and \(g_{M} : I \to \mathcal{P}(I)\) such that:

\[
\begin{align*}
\forall i \in I, & \ a_{M}(i) \in A \cup \{\theta\} \quad (1) \\
\forall i \in I, & \ i \in g_{M}(i) \subseteq I \quad (2) \\
\forall i \in I, & \ a_{M}(i) = \theta \Rightarrow g_{M}(i) = \{i\} \quad (3) \\
\forall i \in I \forall j \in g_{M}(i), & \ a_{M}(j) = a_{M}(i) \quad (4) \\
\forall i, j \in I, & \ i \neq j \land a_{M}(i) = a_{M}(j) \neq \theta \Rightarrow g_{M}(i) = g_{M}(j) \quad (5)
\end{align*}
\]

The assignment of an individual is an activity, possibly the void activity (cf equation 1). Each individual is associated with the group it belongs (cf equation 2). All the individuals which are assigned to the void activity are alone (cf equation 3). All the individuals which are associated with each others have the same activity (cf equation 4) and reciprocally all the individuals which are assigned to the same activity, excepted the void activity, are associated with each other (cf equation 5).

In order to simplify the notations, we introduce the post function of a matching \(M\) which returns the set of individuals assigned to each activity:

\[
p_{M} : A \cup \{\theta\} \to \mathcal{P}(I) \quad p_{M}(a) = \{i \in I; a_{M}(i) = a\} \quad (6)
\]

The posts for an activity can be empty. If \(a_{M}(i) = \theta\), we say that \(i\) is not assigned. An activity \(a \in A\) is:

- oversubscribed if \(|p_{M}(a)| > c_{a}|\); \\
- full if \(|p_{M}(a)| = c_{a}|\); \\

\(^{1}\)the void activity has an infinite capacity.
A matching is said **sound** if none activity is oversubscribed.

A matching is a coalition structure, i.e. a partition of individuals.

**Property 1 (Partition)** Let \( M \) be a matching for the problem \( IA = \langle I, A \rangle \). We verify:

\[
\forall a \in A \cup \{\theta\}, \langle a, p_M(a) \rangle \text{ is a coalition} \tag{7}
\]

\[
\cup_{a \in A \cup \{\theta\}} p_M(a) = I \land \forall a_i, a_j \in A \cup \{\theta\} \ p_M(a_i) \cap p_M(a_j) = \emptyset \tag{8}
\]

**Proof 1** Straight forward from Def. 2 and Def. 3.

The coalition which contains \( i \) in the matching \( M \) is denoted \( C_M(i) \).

Each individual evaluates its preferences over the coalitions and so the matchings regarding the group it belongs and the activity it is assigned.

**Definition 4 (Rationality)** Let \( IA = \langle I, A \rangle \) be an IA problem.

- A coalition \( C \) for \( i \) is **individually rational** for \( i \) iff:

\[
(a_C \succeq_i \theta) \land G_C \succeq_i \{i\} \tag{9}
\]

- A matching \( M \) is **individually rational** iff:

\[
\forall i \in I, \ (a_M(i) \succeq_i \theta) \land g_M(i) \succeq_i \{i\} \tag{10}
\]

- Let \( C \) and \( C' \) be two coalitions which are individually rational for \( i \).

  - The individual \( i \) **prefers** \( C \) to \( C' \) (denoted \( C \succeq_i C' \)) iff:

\[
a_C \succeq_i a_{C'} \land G_C \succeq_i G_{C'} \tag{11}
\]

  - The individual \( i \) **strictly prefers** \( C \) to \( C' \) (denoted \( C \succ_i C' \)) iff:

\[
C \succeq_i C' \land (a_C \succeq_i a_{C'} \lor G_C \succ_i G_{C'}) \tag{12}
\]

- Let \( M \) and \( M' \) be two sound matching for \( IA \). The individual \( i \) **prefers** \( M \) to \( M' \) (denoted \( M \succeq_i M' \)) iff:

\[
C_M(i) \succeq_i C_{M'}(i) \tag{13}
\]

The strict preference relation over the matchings is denoted by \( \succ_i \). An individual chooses a coalition and so a matching such that it prefers its activity to be inactive and it prefers being with its partners to be alone (cf equation 9 and 10). It prefers a coalition to a second one if it prefers the activity and the group of the first one (cf equation 11). An individual prefers a matching to a second one if it prefers its coalition in the first one (cf equation 13). The preference relations over the sound matchings are reflexive, transitive and possibly partial.

A first desirable property to evaluate a matching is the core stability.
Definition 5 (Core stability) Let $M$ be a matching for the problem $IA = \langle I, A \rangle$. A non-empty coalition $C$ blocks the matching $M$ iff:

1. the activity is not oversubscribed:

$$|G_C| \leq c_C$$

2. all the individuals of the coalition prefer the latter to be assigned according to $M$:

$$\forall i \in G_C, C \succeq i C_M(i)$$

3. at least one individual of the coalition strictly prefers the latter to be assigned according to $M$:

$$\exists i \in G_C C \succ i C_M(i)$$

The matching $M$ is core stable if it is sound and there is no blocking coalition:

$$\forall C \subseteq (A \cup \{\theta\}) \times (2^I \setminus \emptyset) \text{ it is not the case } C \text{ blocks } M$$

Intuitively, the individuals in a blocking coalition would like to separate and form their own coalition, which makes the underlying partition unstable.

A matching is Nash stable whenever no individual has incentive to unilaterally change its coalition to another existing (possibly empty) one.

Definition 6 (Nash stability) Let $M$ be a matching for the problem $IA = \langle I, A \rangle$. The matching $M$ is Nash stable if it is sound, rational and:

$$\forall i \in I \forall a \in A, a \neq a_M(i) \Rightarrow p_M(a) = c_a \vee C_M(i) \succeq i (a, p_M(a) \cup \{i\})$$

A Nash stable matching is immune to individual movements since the activities are full or the coalition for any individual in the matching is at least as good as any coalition they can join.

As illustrated by the following example, even if the individual rationality of a matching is necessary condition to be either core stable or Nash stable, neither type of stability implies the other. Moreover, an IA problem does not necessarily have a core stable matching or a Nash stable one.

Example 1 (Stability) Let us consider the IA problem with 3 individuals ($1, 2$ and $3$). and one activity $a$ such that $a \succeq \theta$ with $i \in \{1, 2, 3\}$.

Let us first suppose that the capacity of $a$ is 2 and the social preferences are circular:

- $\{1, 2\} \succ_1 \{1\} \succ_1 \{1, 3\}$;
- $\{2, 3\} \succ_2 \{2\} \succ_2 \{1, 2\}$;
- $\{1, 3\} \succ_3 \{3\} \succ_3 \{2, 3\}$. 
This instance has no core stable matching since the matching $M_1$ (with $p_{M_1}(a) = \{1, 2\}$) is blocked by the coalition $\langle a, \{2, 3\} \rangle$, the matching $M_2$ (with $p_{M_2}(a) = \{2, 3\}$) is blocked by $\langle a, \{1, 3\} \rangle$ and the matching $M_3$ (with $p_{M_3}(a) = \{1, 3\}$) is blocked by $\langle a, \{1, 2\} \rangle$. Moreover, there is no Nash stable matching. In particular, $M_1$, $M_2$ and $M_3$ are not individually rational.

Let us now assume that the capacity of $a$ is 3, $a \geq_i \theta$ and $\theta \geq_i a$ with $i \in \{1, 2, 3\}$.

We consider that the individual $3$ is “undesired”, i.e. the coalitions with the individual $3$ are not individually rational for the others:

- $\{1, 2\} \succ_1 \{1\} \succ_1 \{1, 2, 3\} \succ_1 \{1, 3\}$;
- $\{1, 2\} \succ_2 \{2\} \succ_2 \{1, 2, 3\} \succ_2 \{2, 3\}$;
- $\{1, 2, 3\} \succ_3 \{2, 3\} \succ_3 \{1, 3\} \succ_3 \{3\}$.

The matching $M_1$ such that $p_{M_1}(a) = \{1, 2\}$ is core stable but it is not Nash stable. The matching $M_2$ such that $p_{M_2}(a) = \{3\}$ is Nash stable but it is not core stable.

Another desirable property to evaluate a matching is the Pareto-optimality.

**Definition 7 (Pareto-dominance/optimal)*** Let $M$ and $M'$ be two sound matchings for the problem $IA = \langle I, A \rangle$. $M'$ Pareto-dominates $M$ iff:

\begin{align*}
\forall i \in I, C_{M'}(i) & \succ_i C_M(i) \\
\exists i \in I, C_{M'}(i) & \succ_i C_M(i)
\end{align*}

A matching is Pareto-optimal if it is not Pareto-dominated.

A matching Pareto-dominates a second one if it is strictly better for at least one individual and not worst for the others. A matching is Pareto-optimal if there is no alternative in which all agents would be in an equivalent or better position.

The core stability is a sufficient condition for Pareto-optimality but it is not a necessary condition.

**Property 2 (Pareto-optimal)*** All core stable matching are Pareto-optimal.

**Proof 2 (Pareto-optimum)*** We prove by contradiction that a core stable matching is a Pareto-optimum. We assume that $M$ is a core stable matching which is not a Pareto-optimum. Then, there exists a matching $M'$ which Pareto-dominates $M$. Let us consider the coalition $C_{M'}(i)$ with $i$ satisfying the equation (20). We verify:

1. the equation (14) since $M'$ is sound;
2. the equation (15) since the equation (13);
3. the equation (16) since the equation (20).
Therefore, due to definition 3, we conclude that $C_{M'}(i)$ blocks $M$ which is in contradiction with our assumption.

In our previous example, when the capacity of $a$ is 2 and the social preferences are circular, all the matchings where the activity is full are Pareto-optimal. When the capacity of the activity is 3 and the individual 3 is undesired, the matching where the activity is full is also Pareto-optimal.

Even if the stability is a desirable property, there exists no necessarily such a solution. By contrast, the Pareto-optimality seems not to be discriminative. An alternative way to assess the quality of a matching consists of the concept of social welfare. For this purpose, we assume that the individuals have cardinal preferences. Moreover, in an IA problem, each individual evaluates its preferences with respect to $2^{m-1}$ groups. This preference representation takes exponential space. By contrast, the representation of additively separable preferences is linear with respect to the number of individuals.

**Definition 8 (Additively separable IA)** Let $IA = \langle I, A \rangle$ be a IA problem of size $(m, n)$. The problem is additively separable (ASIA) if each individual $i \in I$ is endowed with:

1. a valuation function $v_i : A \cup \{\theta\} \rightarrow [-1; 1]$ representing its preferences over the activities, possibly the void activity;
2. a valuation function $w_i : I \setminus \{i\} \rightarrow [-1; 1]$ representing its preferences over the potential partners.

The utility for an individual $i$ is the function $u_i : G(i) \times A \cup \{\theta\} \rightarrow [-1; 1]$ defined such that:

$$
\forall g \in G(i) \ \forall a \in A \cup \{\theta\}, \ u_i(g, a) = \frac{\sum_{j \in g, j \neq i} w_i(j) + v_i(a)}{2} \tag{21}
$$

We assume that the preferences over the individuals and the preferences over the activities are comparable. In particular, the utility for an individual to be alone only depends on its valuation of the activity. Moreover, we assume the more the individual has partners the better it is satisfied since we aim at improving the social cohesion and avoid the isolation.

In the rest of the paper, we do not consider that the utilities are defined to correspond to ordinal preferences as in [2] but we suppose a direct access to the utilities of agents since it will be the case in our practical application.

We adopt here the utilitarian approach inspired by Bentham. In other words, our goal is to maximize the sum of individual utilities.

**Definition 9 (Social welfare)** Let $IA = \langle I, A \rangle$ be an ASIA problem of size $(m, n)$. The utilitarian welfare of a matching $M$:

$$
U_I(M) = \frac{1}{m} \sum_{i \in I} u_i(g_M(i), a_M(i)) \tag{22}
$$

The higher the welfare is, the better the matching is.
Example 2 (ASIA problem) Let us consider our previous example where the capacity of the activity is 3 and the individual 3 is undesired. We define the valuation functions such that \( v_1(a) = v_2(a) = v_3(a) = 0 \), \( w_1(2) = w_2(1) = \frac{1}{2} \), \( w_1(3) = w_2(3) = -1 \), \( w_3(1) = \frac{1}{2} \) and \( w_3(2) = 1 \). Therefore, the induced ordinal preferences are as previously. The matching where individuals 1 and 2 are assigned to the activity maximizes the utilitarian welfare.

4 The algorithm

In order to maximize the utilitarian welfare, we can consider Quadratic Programming (QP), i.e. an optimization method with a mathematical model which can be represented by a quadratic function. An ASIA problem can be modeled by: i) \( n \times m \) variables \( x_{ia} \in \{0, 1\} \) such that \( x_{ia} = 1 \) if individual \( i \) is assigned to activity \( a \) or \( x_{ia} = 0 \) otherwise; ii) \( m \) constraints \( \sum_{a \in A} x_{ia} \leq 1 \) representing the mutual exclusion of the assignments for the same individual; and iii) \( n \) constraints \( \sum_{i \in I} x_{ia} \leq c_a \) which warrants the soundness of the matching. If we consider the valuation functions, the objective function corresponding to the utilitarian welfare, which must be maximized, is:

\[
\sum_{a \in A} \sum_{i \in I} x_{ia} (v_i(a) + \frac{1}{m-1} \sum_{j \neq i} w_i(j) x_{ja})
\]

When this problem is in a standard form, i.e. minimizing an objective function like \( \frac{1}{2} x Q x^T + c^T x \) with a symmetric matrix \( Q \), the latter is not necessarily positive-definite and so the problem can be NP-hard [12].

This is the reason why we propose here an algorithm (cf algorithm 1) which computes a “good” matching for an ASIA problem instance. Initially, all the individuals are alone and assigned to the void activity. Each in turn, a free individual \( i \) considers the rational activity \( a \) it prefers. If no other individual is assigned to this activity, then \( i \) is assigned. Otherwise, the algorithm tries to improve the utilitarian welfare of the group, eventually by firing the individuals whose presence contributes the least to the social welfare of the group. If the assignment of \( i \) does not improve the social welfare of the group then \( i \) must concede, i.e. consider the next rational activity. The individuals which are replaced by \( i \) must concede. If the capacity of \( a \) is not reached the group may increase (line 21). An agent which is rejected by all its rational activities stay alone and it is definitely assigned to the void activity. An approximation algorithm consists of excluding only one individual at each step (line 20).

It can be noticed that our (approximation) algorithm always returns a sound matching.

Property 3 (Termination) Our (approximation) algorithm applied over an ASIA problem ends and returns a sound matching.

Proof 1 (Termination) Let \( IA = \langle I, A \rangle \) be an ASIA problem. We consider the following loop invariant:

\[
\sum_{i \in I} \text{concessions}(i) + |\text{Free}|
\] (23)
Algorithm 1: compute a matching for an ASIA problem

input: $IA = (I, A)$
output: a matching $M$

1 $Free = I$;
2 foreach $i \in I$ do
3     $concessions(i) = A.SortWith(v_i(\_) > v_i(\_)) > 0)$;
4     $a_M(i) = \emptyset$;
5     $g_M(i) = \{i\}$;
6 while $Free \neq \emptyset$ do
7     foreach $i \in Free$ do
8         if $concessions(i) = \emptyset$ then $Free \setminus \{i\}$;
9         else
10            $a = concessions(i).head$; // $a$ is the most preferred activity
11            $g = p_M(a)$;
12            $g' = g \cup \{i\}$;
13            if $g = \emptyset$ then
14                /* the posts of $a$ are empty and so $i$ is assigned */
15                $a_M(i) = a$;
16                $g_M(i) = \{i\}$;
17                $Free \setminus \{i\}$;
18            else
19                $u_{max} = -\infty$;
20                $bg = \emptyset$;
21                $SG = \{sg \subseteq g'; sg \neq \emptyset\}$;
22                /* eventually $SG = \{sg \subseteq g'; |sg| = |g'|-1\} */
23                if $c_a > |g|$ then $SG \cup g'$;
24                /* the group may increase */
25                foreach $sg \in SG$ do
26                    $u = \Sigma_{k \in sg} u_k(sg, a)$;
27                    if $u > u_{max}$ then
28                        $u_{max} = u$;
29                        $bg = sg$;
30                /* $bg$ is the best group */
31                foreach $j \in bg$ do $g_M(j) = bg$;
32                foreach $j \in g \setminus bg$ do
33                    /* $j$ is unassigned */
34                    $a_M(j) = \emptyset$;
35                    $g_M(j) = \{j\}$;
36                    $Free \cup \{j\}$;
37                    $concessions(j) = concessions(j).tail$;
38                if $i \in bg$ then
39                    /* $i$ is assigned */
40                    $a_M(i) = a$;
41                    $Free \setminus \{i\}$;
42                else
43                    /* $i$ is rejected */
44                    $concessions(i) = concessions(i).tail$;
45            end
46        end
47    end
48 end
49 return $M$
This invariant is positive. It strictly decreases after each loop since:

1. an individual, which is assigned, is removed from Free;
2. an individual, which is not assigned, concedes until it is definitely assigned to the void activity;
3. any individual, which is unassigned, concedes and at least one another individual is assigned (and so removed from Free).

The resulting matching is sound since the activities are never oversubscribed.

The outcome of our exact algorithm is a Pareto-optimum.

**Property 4 (Pareto-optimal)** Our algorithm applied over an ASIA problem returns a matching which is Pareto-optimal.

**Proof 2 (Pareto-optimal)** Let \( IA = (I,A) \) be an ASIA problem. We prove by contradiction that the outcome \( M \) of our algorithm is Pareto-optimal. We assume that \( M \) is Pareto-dominated by a sound matching \( M' \). Due to the equation 20 there is an individual \( i \) such that \( C_{M'}(i) \succ_i C_M(i) \). By the equation 13, \( C_{M'}(i) \succeq_j C_M(i) \) and:

- either \( g_{M'}(i) \succ_i g_M(i) \) and \( a_{M'}(i) = a_M(i) \). Since \( C_{M'}(i) \succeq_j C_M(i) \) and due to the equation 13, \( g_{M'}(i) \succeq_j g_M(i) \) for all the partners of \( i \). By the equation 27, \( \sum_{j \in g_{M'}(i)} u_j(g_{M'}(i),a_{M'}(i)) > \sum_{j \in g_M(i)} u_j(g_M(i),a_M(i)) \).
  This is a contradiction with our algorithm (line 27) which computes the posts with the best utilitarian welfare;

- or \( a_{M'}(i) \succ_i a_M(i) \) and so \( a_{M'}(i) \) precedes \( a_M(i) \) in concessions(\( i \)). According to the algorithm, \( i \) has been rejected or unassigned by \( a_M(i) \).
  Therefore, there is a individual \( j \in C_M(i) \) such that \( C_M(j) \succ_j C_{M'}(j) \).
  This is a contradiction with the equation 19.

**Example 3 (Algorithm)** Let us consider our previous example 2. For instance, if we consider the list of free individuals \((3,2,1)\), our algorithm operates the following assignments/unassignments:

1. the individual 3 is assigned since the posts of \( a \) is initially empty;
2. the individual 2 is assigned since the capacity of 3 is not reached and \( \{2,3\} \) is the best subgroup of \( \{2,3\} \);
3. the individual 1 replaces 3 since \( \{1,3\} \) is the best subgroup of \( \{1,2,3\} \);
4. the individual 3 concedes and so it is definitely assigned to the void activity.

The matching is a Pareto-optimum and maximizes the utilitarian welfare.

It is worth noticing that our algorithm does not always maximize the utilitarian welfare since the heuristic gives priority to the activities.
5 Agent behaviours

We consider the asynchronous message-passing model of actor for concurrent programming [3]. In this perspective, the primitives are agents and events. An agent represents an independent program that runs on its own processor. An event is the creation of an agent or the utterance/reception of a message. It is worth noticing that the system is distributed since the message transmission delay is arbitrary but not negligible. The underlying channels are assumed to be reliable (a message is delivered once and only once) and that the messages may arrive in different order from sending.

In order to propose a distributed solver based on this model, we distinguish 3 kinds of agents:

1. the “solver” agent which creates the other agents and records the assignments;

2. the “individual” agents which are endowed with the same behaviours but with different preferences;

3. the “activity” agents which are endowed with the same behaviours but managing different groups and capacities.

The behaviour of the “solver” agent consists of: i) creating other agents; ii) triggering the solving; iii) recording the assignments/unassignments; and iv) returning the matching when all the individual are assigned.

The behaviour of the “individual” agent (cf algorithm 2) builds the list of concessions, then it proposes itself to the rational activity it prefers. When the agent is assigned/unassigned, it informs the “solver” agent. For an unassignment, the “activity” agent waits for a confirmation before sending a new proposal such that a matching is not prematurely returned by the “solver” agent. If an “individual” agent becomes free, it concedes, i.e. sends a proposal to the next rational activity in the list of concessions, possibly until to be definitively assigned to the void activity.

The behavior of the “activity” agent is described by the deterministic finite state automaton in Figure 1. When a proposal is received, it is accepted if the current group is empty (in the state Available). Otherwise, the agent handles the proposals one by one (in the state Casting) by identifying the subgroup which maximizes the utilitarian welfare. If the capacity is reached, the subgroups of size $c_a$ or less are considered. Otherwise, the number of posts may increase. If the proposer is not in the best subgroup then the proposal is rejected. Otherwise, the proposal is accepted when the fired member has confirmed the unassignment (in the state Firing). When the proposal is processed, the “activity” agent is ready to evaluate the next proposals, eventually those which have been stashed.
Algorithm 2: The behaviour of an “individual” agent

\textbf{input} : $i \in I$

1. \textbf{switch }ReceivedMessage\textbf{ do}

2. \hspace{5mm} case $\text{Inform}(\text{adr})$ \textbf{do}

3. \hspace{10mm} concessions = $\text{adr}.\text{SortWith}(v_i(\_)>v_i(\_)>0)$;

4. \hspace{10mm} if concessions $\neq \emptyset$ then

5. \hspace{15mm} concessions.head $\Rightarrow$ Propose($i$)\textit{ /i proposes itself to its most preferred activity}

6. \hspace{10mm} else

7. \hspace{15mm} $\text{parent!Allocated}(i, \theta)$

8. \hspace{10mm} \textbf{case }Accept\textbf{ do}

9. \hspace{15mm} $\text{parent!Allocated}(i, \text{concessions.head});$ \hspace{5mm} //\textit{i is assigned}

10. \hspace{10mm} \textbf{case }Reject\textbf{ do}

11. \hspace{15mm} concessions = $\text{concessions.tail};$ \hspace{5mm} //\textit{i is not assigned}

12. \hspace{15mm} if concessions $= \emptyset$ then $\text{parent!Allocated}(i, \theta)$ ;

13. \hspace{10mm} else concessions.head $\Rightarrow$ Propose($i$) ;

14. \hspace{10mm} \textbf{case }Eject\textbf{ do}

15. \hspace{15mm} $\text{parent!Disallocated}(i, \text{concessions.head}) ;$ \hspace{5mm} //\textit{i notifies the “solver” agent about the unassignment}

16. \hspace{15mm} /* The unassignment has been recorded */

17. \hspace{15mm} concessions.head$\Rightarrow$Confirm(concessions.head);

18. \hspace{15mm} concessions = $\text{concessions.tail};$

19. \hspace{15mm} if concessions $= \emptyset$ then $\text{parent!Allocated}(i, \theta)$ ;

20. \hspace{15mm} else concessions.head $\Rightarrow$ Propose($i$) ;

21. \hspace{10mm} \textbf{case }Query($g, a$) \textbf{ do}

22. \hspace{15mm} $\text{sender!Reply}(g, a, u_i(g, a));$ \hspace{5mm} //\textit{i informs a about its preferences}

Figure 1: The behaviour of an “activity” agent
6 Experiments

Our experiments aim at evaluating the quality of the matching reached by our algorithm and the speedup due to its distribution.

We have implemented our prototype with the Scala programming language\(^2\) and the Akka toolkit\(^3\). The latter, which is based on the actor model\(^3\), allows us to fill the gap between the specification and its implementation. In order to tackle a large number of individuals, we consider the approximation algorithm.

In order to evaluate the advantage of the structural properties of two-sided matching that distinguish an ASIA problem from a generic hedonic game, we translate any ASIA instance into a hedonic game and we compute a contractually individually stable (CIS) coalition as suggested by\(^1\). For this purpose, we associate: i) one player for each activity such that all the coalitions which are compliant with its capacity are equally preferred; and ii) one player for each individual whose preferences are deduced from the utility function of the latter. Even if the resulting preferences are not additively separable, we can represent them using rational lists for coalitions (RLC)\(^1\). We consider some ASIA instances with 2 activities and \(m\) individuals (\(2 \leq m \leq 20\)). For sake of simplicity, all the activities have the same capacity (\(c = m/2\)). We (pseudo)-randomly generated 100 instances for each \(m\). Figure 2 compares the utilitarian welfare of the matching reached by our algorithm to the one reached by the algorithm proposed in\(^1\) and their runtime. Please note the logarithmic scale used for the time. The outcome of our algorithm benefits of the structural properties of our problem and the runtime of the CIS algorithm is due to the generation of RLC which take exponential space. Even if our algorithm does not always maximize the utilitarian welfare, we observe that it is closed to the best utilitarian welfare which is computed by an exhaustive search until 12 individuals.

We have implemented a local search algorithm\(^11\) to be compared with our algorithm. This hill-climbing algorithm, which starts with a sound and random matching (with a random number of inactive individuals), iteratively tries to improve the utilitarian welfare. Two matchings are neighbours if they are identical with the exception for one individual which moves to another activity, eventually the void one. If this new activity is full, then all the swaps of individuals with the members of that activity are considered.

We consider some ASIA instances with \(n\) activities and \(m\) individuals. For sake of simplicity, all the activities have the same capacity (\(c = m/n\)). For each set of parameters (\(n\) and \(m\)), we (pseudo)-randomly generated 100 instances.

Firstly, we compare the utilitarian welfare of the matching reached by our algorithm with the one reached by local search. Figure 3a presents the mean utilitarian welfare obtained for each set of parameters (with \(2 \leq n \leq 10\) and \(2 \times n \leq m \leq 100\)). Our algorithm overreaches the local search. Actually, the utilitarian welfare for a ASIA problem is a function with many local optima.

Secondly, we compare the runtime of the centralized version of our approximation algorithm with its decentralized version. Figure 3b shows the mean

\(^2\)http://www.scala-lang.org
\(^3\)http://akka.io
Figure 2: Utilitarian welfare and runtime of our algorithm and CIS algorithm

Figure 3: Utilitarian welfare of matching reached by our algorithm/hill-climbing (at left) and decentralized/centralized algorithm runtime (at right)
runtimes for each set of parameters (with $2 \leq n \leq 10$ and $2 \times n \leq m \leq 400$). While the centralized algorithm is faster when the number of individuals is low ($\sim 40$), its runtime quickly grows with the number of individuals (18 ms for 100 individuals and 10 activities) while the runtime of the distributed version is less (9 ms in the latter case). Moreover, the runtime of the hill-climbing algorithm is too high to be represented in the figure (1,660 ms for 100 individuals and 10 activities). We can expect a higher runtime if we adopt a local search method such as the simulated annealing without any warranty about the optimality of the outcome.

In summary, the outcome reached by our algorithm seems to be good. Moreover, the distribution of our algorithm allows to speedup (up to 3.5 times) its runtime.

7 Conclusions

We have introduced here the generic problem of individuals/activities where some individuals must be assigned to the activities they enjoy with their favorite partners. Even if the stability is a desirable property, there exists no necessarily such a solution. By contrast, the Pareto-optimality seems not to be discriminative. This is the reason why we have adopted additively separable preferences in order to evaluate the quality of a solution with the help of the utilitarian welfare. Moreover, the representation of additively separable preferences is linear with respect to the number of individuals. Maximizing the utilitarian welfare can be a NP-hard problem. For this purpose, we have proposed a heuristic where the individuals propose themselves to the activities they prefer and eventually concede. The individuals which contribute the least to the utilitarian welfare of the group are unassigned and they concede. We have shown that this algorithm always returns a sound match which is a Pareto-optimum. By adopting the actor model, We have distributed this algorithm. The difficulty lies in: (i) the detection of the halting (as messages can arrive in a different order from the sending), an “activity” agent must wait for the confirmation of an unassignment before assigning a new individual) ; (ii) the synchronization (as the acceptance of an individual depends on the group, an “activity” agent must deal with the proposals one-by-one). Our experiments show that our algorithm benefits of the the structural properties of two-sided matching and it overreaches local search techniques. Finally, the distribution of our algorithm speeds up its execution (up to 3.5 times).

In future works, we aim at evaluating our matching engine with the real-world data from our project. In order to propose fair outcomes, we are considering modifying the local decision of our algorithm for maximizing the egalitarian welfare.
8 Acknowledgement

This work is part of the PartENS research project supported by the Nord-Pas de Calais region (researcher/citizen research projects).

References

[1] Coralio Ballester. NP-completeness in hedonic games. *Games and Economic Behavior*, 49(1):1–30, 2004.

[2] Craig Boutilier, Ioannis Caragiannis, Simi Haber, Tyler Lu, Ariel D. Procaccia, and Or Sheffet. Optimal social choice functions: A utilitarian view. *AIJ*, 227:190–213, 2015.

[3] William Douglas Clinger. *Foundations of actor semantics*. PhD thesis, Massachusetts Institute of Technology, 1981.

[4] Andreas Darmann, Edith Elkind, Sascha Kurz, Jérôme Lang, Joachim Schauer, and Gerhard Woeginger. Group activity selection problem. In *Proceedings of the 8th International Conference on Internet and Network Economics*, pages 156–169, Liverpool, UK, 2012. Springer Berlin Heidelberg.

[5] Jacques Dreze and Joseph Greenberg. Hedonic coalitions: Optimality and stability. *Econometrica*, 48:9871003, 1980.

[6] Patricia Everaere, Maxime Morge, and Gauthier Picard. Minimal Concession Strategy for Reaching Fair, Optimal and Stable Marriages. In *Proc. of AAMAS*, pages 1319–1320, 2013.

[7] David Gale and Lloyd Stowell Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69:9–14, 1962.

[8] Ayumi Igarashi, Dominik Peters, and Edith Elkind. Group activity selection on social networks. In *Proc. of AAAI*, pages 565–571, 2017.

[9] David F. Manlove. *Algorithmics of Matching Under Preferences*. World Scientific, 2014.

[10] Antoine Nongaillard and Sébastien Picault. Multilevel Agent-Based Modelling for Assignment or Matching Problems. In *Proc. of ECAI*, 2016.

[11] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*, chapter Chapter 4: Informed Search and Exploration, pages 94–136. Pearson Education, 2003. 2nd edition.

[12] Sartaj Sahni. Computationally related problems. *SIAM Journal on Computing*, 3(4):262–279, 1974.

[13] Thomas C Schelling. *The strategy of conflict*. Harvard university press, 1980.