Consiglio Nazionale delle Ricerche

From Marriages to Coalitions: A Soft CSP Approach

S. Bistarelli, S. Foley, B. O'Sullivan, F. Santini

IIT TR-13/2008

Technical report

Ottobre 2008

Istituto di Informatica e Telematica

From Marriages to Coalitions: A Soft CSP Approach

Stefano Bistarelli[1,2], Simon Foley[3], Barry O'Sullivan[3,4], and Francesco Santini[2,5]

[1] Dipartimento di Scienze, Università "G. d'Annunzio" di Chieti-Pescara, Italy
bista@sci.unich.it
[2] Istituto di Informatica e Telematica (CNR), Pisa, Italy
[stefano.bistarelli,francesco.santini]@iit.cnr.it
[3] Department of Computer Science, University College Cork, Cork, Ireland
[s.foley, b.osullivan]@cs.ucc.ie
[4] Cork Constraint Computation Centre, University College Cork, Ireland
b.osullivan@4c.ucc.ie
[5] IMT - Scuola di Studi Avanzati, Lucca, Italy
f.santini@imtlucca.it

Abstract. In this work we represent the *Optimal Stable Marriage* problem as a *Soft Constraint Satisfaction Problem*. In addition, we extend this problem from couples of individuals to coalitions of generic agents, in order to define new coalition-formation principles and stability conditions. In the coalition case, we suppose the preference value as a trust score, since trust can describe a nodes belief in another nodes capabilities, honesty and reliability. Soft constraints represent a general and expressive framework that is able to deal with distinct concepts of optimality by only changing the related c-semiring structure, instead of using different ad-hoc algorithms. At last, we propose an implementation of the classical OSM problem by using *Integer Linear Programming* tools.

1 Introduction

The *Stable Marriage* (SM) problem [12, 18] and its many variants [15] have been widely studied in the literature, because of the inherent appeal of the problem and the important practical applications. A classical instance of the problem comprises a bipartite set of n men and n women, and each person has a preference list in which they rank all members of the opposite sex in a total strict order. Then, a match MT is simply a bijection between men and women. A man m_i and a woman w_j form a *blocking pair* for MT if m_i prefers w_j to his partner in MT and w_j prefers m_i to her partner in MT. A matching that admits no blocking pair is said to be *stable*, otherwise the matching is unstable. Even if the SM problem has its roots as a combinatorial problem, it has also been studied in Game Theory and Economics and in Operations Research [9].

However, in this paper we mainly concentrate on its optimization version, the *Optimal Stable Marriage* (OSM) problem [18, 15], which tries to find a match that is not only stable, but also "good" according to some criteria based on the preferences of all the individuals. Classical solutions deal instead only with men-optimal or women-optimal marriages, in which every man, or respectively woman, gets his best possible partner.

We propose soft constraints as a very expressive framework where it is possible to cast different kinds of optimization criteria by only modifying the c-semiring [4, 1] structure on which the corresponding *Soft Constraint Satisfaction Problem* (SCSP) [1] is based. In this sense, soft constraints prove to be a more general solving framework w.r.t. the other ad-hoc algorithms presented in literature for each different optimization problem [15]. In fact, we can deal also with problem extensions as incomplete preference lists and ties in the same list. Therefore, in this paper we build a bridge between the OSM and SCSP problems, as previously done between SM and (crisp) CSP problems [9, 21]. Since many variants of the OSM problems are NP-hard [15], in this way we can benefit from SCSP solving techniques.

Moreover, use *Integer Linear Programming* (*ILP*) as a general method to practically solve this classical problem. It is worth to notice that such ILP techniques are here applied to the OSM problem for the first time.

The second main result provided in the paper consists in extending the stable marriage definition from pairs of individuals to coalitions of agents. A coalition can be defined as a temporary alliance among agents, during which they cooperate in joint action for a common task [13]. Moreover, we use trust scores instead of plain preferences in order to evaluate the relationships among agents. Therefore, the notion of SM stability is translated to coalitions, and the problem is still solved by the optimization point of view: the final set of coalitions is stable and is the most trustworthy w.r.t. the used trust metric, represented by a c-semiring [5, 2, 20]. Even for this coalition extension we use soft constraints to naturally model the problem.

The classical SM problem (thus, the non-optimal version of the problem) has been already studied and solved by using crisp constraints in [9, 21]. In [9] the authors present two different encodings of an instance of SM as an instance of a *Constraint Satisfaction Problem* (CSP). Moreover, they show that *Arc Consistency* propagation achieves the same results as the classical *Extended Gale/Shapley* (EGS) algorithm, thus easily deriving the men/women-optimal solution [9]. Ad-hoc algorithms for the OSM problem (not using a constraint formalization) are instead presented in Sec. 3.

In Sec. 2 we summarize the background on soft constraints, while Sec. 3 does the same for the OSM problem. In Sec. 4 we represent the OSM problem with soft constraints and we solve it with ILP. Section 5 extends the OSM problem to coalitions, still representing the problem with soft constraints. At last, Sec. 6 draws the final conclusions.

2 Soft Constraints

A c-semiring [4, 1] S (or simply semiring in the following) is a tuple $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ where A is a set with two special elements ($\mathbf{0}, \mathbf{1} \in A$) and with two operations $+$ and $\times$ that satisfy certain properties: $+$ is defined over (possibly infinite) sets of elements of A and thus is commutative, associative, idempotent, it is closed and $\mathbf{0}$ is its unit element and $\mathbf{1}$ is its absorbing element; $\times$ is closed, associative, commutative, distributes over $+$, $\mathbf{1}$ is its unit element, and $\mathbf{0}$ is its absorbing element (for the exhaustive definition, please refer to [4]). The $+$ operation defines a partial order $\leq_S$ over A such that $a \leq_S b$ iff $a + b = b$; we say that $a \leq_S b$ if b represents a value *better* than a. Other properties related to the two operations are that $+$ and $\times$ are monotone on $\leq_S$, $\mathbf{0}$ is its minimum and $\mathbf{1}$ its maximum, $\langle A, \leq_S \rangle$ is a complete lattice and $+$ is its lub. Finally, if $\times$ is idempotent, then $+$ distributes over $\times$, $\langle A, \leq_S \rangle$ is a complete distributive lattice and $\times$ its glb.

Soft Constraints. A *soft constraint* [4, 1] may be seen as a constraint where each instantiation of its variables has an associated preference. Given $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ and an ordered set of variables V over a finite domain D, a soft constraint is a function which, given an assignment $\eta : V \to D$ of the variables, returns a value of the semiring. Using this notation $\mathcal{C} = \eta \to A$ is the set of all possible constraints that can be built starting from S, D and V.

Any function in $\mathcal{C}$ involves all the variables in V, but we impose that it depends on the assignment of only a finite subset of them. So, for instance, a binary constraint $c_{x,y}$ over variables x and y, is a function $c_{x,y} : V \to D \to A$, but it depends only on the assignment of variables $\{x, y\} \subseteq V$ (the *support* of the constraint, or *scope*). Note that $c\eta[v := d_1]$ means $c\eta'$ where η' is η modified with the assignment $v := d_1$. Note also that $c\eta$ is the application of a constraint function $c : V \to D \to A$ to a function $\eta : V \to D$; what we obtain, is a semiring value $c\eta = a$. $\bar{\mathbf{0}}$ and $\bar{\mathbf{1}}$ respectively represent the constraint functions associating $\mathbf{0}$ and $\mathbf{1}$ to all assignments of domain values; in general, the $\bar{a}$ function returns the semiring value a.

Given the set $\mathcal{C}$, the combination function $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ is defined as $(c_1 \otimes c_2)\eta = c_1\eta \times c_2\eta$ (see also [4, 1]). Informally, performing the $\otimes$ or between two constraints means building a new constraint whose support involves all the variables of the original ones, and which associates with each tuple of domain values for such variables a semiring element which is obtained by multiplying the elements associated by the original constraints to the appropriate sub-tuples. The partial order $\leq_S$ over $\mathcal{C}$ can be easily extended among constraints by defining $c_1 \sqsubseteq c_2 \iff c_1\eta \leq c_2\eta$. Consider the set $\mathcal{C}$ and the partial order $\sqsubseteq$. Then an entailment relation $\vdash \subseteq \wp(\mathcal{C}) \times \mathcal{C}$ is defined s.t. for each $C \in \wp(\mathcal{C})$ and $c \in \mathcal{C}$, we have $C \vdash c \iff \bigotimes C \sqsubseteq c$ (see also [1]).

Given a constraint $c \in \mathcal{C}$ and a variable $v \in V$, the *projection* [4, 1, 3] of c over $V - \{v\}$, written $c \Downarrow_{(V \setminus \{v\})}$ is the constraint c' s.t. $c'\eta = \sum_{d \in D} c\eta[v := d]$. Informally, projecting means eliminating some variables from the support.

A SCSP [1] defined as $P = \langle C, con \rangle$ (C is the set of constraints and $con \subseteq V$, i.e. a subset the problem variables). A problem P is α-consistent if $blevel(P) = \alpha$ [1]; P is instead simply "consistent" iff there exists $\alpha >_S \mathbf{0}$ such that P is α-consistent [1]. P is inconsistent if it is not consistent. The *best level of consistency* notion defined as $blevel(P) = Sol(P) \Downarrow_{\emptyset}$, where $Sol(P) = (\bigotimes C) \Downarrow_{con}$ [1].

3 The Optimal Stable Marriage Problem

An instance of the classical SM problem (SM) [8] comprises n men and n women, and each person has a preference list in which all members of the opposite sex are ranked in a total strict order. All men and women must be matched together in a couple such that no element x of couple a prefers an element y of different couple b that also prefers x (i.e. the stability condition of the pairing). If such (x, y) exists in the match, then it is defined as *blocking*; a match is stable if no blocking pairs exist.

The problem was first studied by Gale and Shapley [8]. They showed that there always exists at least a stable matching in any instance and they also proposed a $O(n^2)$-time algorithm to find one, i.e. the so-called *Gale-Shapley* (GS) algorithm. An extended version of the GS algorithm, i.e. the EGS algorithm [12], avoids some unnecessary steps by deleting from the preference lists certain (man, woman) pairs that cannot belong to a stable matching. Notice that, in the man-oriented version of the EGS algorithm, each man has the best partner (according to his ranking) that he could obtain, whilst each woman has the worst partner that she can accept. Similar considerations hold for the woman-oriented version of EGS, where men have the worst possible partner.

For this reason, the classical problem has been extended [8] in order to find a SM under a more equitable measure of optimality, thus obtaining an *Optimal SM* problem [18, 15, 14, 11]. For example, in [14] the authors maximize the total satisfaction of a SM by simply summing together the preferences of both men (e.g. $p_M(m_i, w_j)$) and women (e.g. $p_W(m_i, w_i)$) in the SM given by $MT = \{(m_i, w_j), \ldots, (m_k, w_z)\}$. This sum needs to be minimized since $p_M(m_i, w_j)$ represents the rank of w_j in m_i's list of preferences, where a low rank position stands for a higher preference, i.e. 1 belongs to the most preferred partner; similar considerations hold for the preferences of women, i.e. $p_W(m_i, w_j)$ which represents the rank of m_i in w_i's list of preferences. Therefore, we need to minimize this *egalitarian cost* [14]:

$$min\left(\sum_{(m_i,w_j)\in MT} p_M(m_i, w_j) + \sum_{(m_i,w_j)\in MT} p_W(m_i, w_j)\right) \tag{1}$$

This optimization problem was originally posed by Knuth [14]. Other optimization criteria are represented by minimizing the *regret cost* [11] as in (2), or by minimizing the *sex-equalness cost* [16] as in (3):

$$\min \max_{(m_i,w_j)\in MT} \max\{p_M(m_i, w_j) p_W(m_i, w_j)\} \tag{2}$$

$$min\left|\sum_{(m_i,w_j)\in MT} p_M(m_i, w_i) - \sum_{(m_i,w_j)\in MT} p_W(m_i, w_j)\right| \tag{3}$$

Even if the number of stable matchings for one instance grows exponentially in general [15], (1) and (2) have been already solved in polynomial time with ad-hoc algorithms (respectively in [14] and [11]), by exploiting a lattice structure that condense the information about all the matchings. On the contrary, (3) is an NP-hard problem for which only approximation algorithms have been given [16].

In the following, we consider the preference as a more general weight (taken from a proper semiring), instead of a plain position in the preference's list of an individual; thus, we suppose to have *weighted preference lists* [14]. A different but compatible (w.r.t. OSM) variant of the SM problem shows incomplete preference's lists, i.e. the *SMI* (SM with incomplete lists), if a person can exclude a partner whom she/he does not want to be matched with [15]. Another extension is represented by preference lists that allow ties, i.e. in which it is possible to express the same preference for more than one possible partner: the problem is usually named as SM with ties, i.e. *SMT* [15]. In this case, three stability notions can be proposed [15]: given any two couples (m_i, w_j) and (m_k, w_z) belonging to a match MT, then MT is *i) super* stable if $p_M(m_i, w_j) > p_M(m_i, w_z) \wedge p_W(m_i, w_j) > p_W(m_i, w_z)$, *ii) strongly* stable if $p_M(m_i, w_j) \geq p_M(m_i, w_z) \wedge p_W(m_i, w_j) > p_W(m_i, w_z)$ or *iii) weakly* stable if $p_M(m_i, w_j) \geq p_M(m_i, w_z) \wedge p_W(m_i, w_j) \geq p_W(m_i, w_z)$. Hence, if a match is super stable then it is strongly stable, and if it is strongly stable then it is weakly stable [15].

Allowing ties in preferences means that the (1), (2) and (3) problems above becomes hard even to approximate [15]. By joining together these two extensions, we obtain the *SMTI* problem: *SM with Ties and Incomplete lists* [15]. The preferences of men and women can respectively be represented with two matrices M and W, as in Fig. 1.

4 Representing the classical ST Problem with Soft Constraints

In order to define an encoding of an OSM instance I as a SCSP problem instance P (see Sec. 2), we introduce the set V of variables: $m_1, m_2, \ldots, m_n$ corresponding to men, and $w_1, w_2, \ldots, w_n$ corresponding to women. The domain D of m_i or w_j is $[1, n]$. For each i, j ($1 \leq i, j \leq n$), then $\eta : V \rightarrow D$ (as defined in Sec. 2) denotes the value of variable m_i and w_j respectively, i.e., the partner associated with the match. For example, $\eta(m_1) = 3$ means that m_1 is matched with w_3.

We need three different set of soft constraints to describe a OSM problem, according to each of the relationships we need to represent:

1. *Preference* constraints. These unary constraints represent the preferences of men and women: for each of the values in the variable domain, i.e. for each possible partner, they associate the relative preference. For example, $c_{m_i}(m_i = j) = a$ represents the fact that the man m_i has a degree of preference value a for the woman w_j (when the variable m_i is instantiated to j); on the other hand, $c_{w_j}(w_j = i) = b$ means that the same woman (w_j) has a preference for the same man (m_i) equal to b; a and b are elements of the chosen semiring set. We need $2n$ unary constraints: one for each man and woman.

2. *Marriage* constraints. This set constrains the marriage relationships: if m_i is married with w_j (i.e. $\eta(m_i) = j$), then w_j must be married with m_i (i.e. $\eta(w_j) = i$). Formally, it can be defined by $c_m(m_i, w_j) = \mathbf{0}$ if $\eta(m_i) = h \wedge \eta(w_j) = k \wedge (h \neq j \vee k \neq i)$. We need n^2 marriage constraints, one for each possible man-woman couple.
3. *Stability* constraints. This set of 4-ary constraints avoids the presence of blocking couples in the set of matches: $c_s(m_i, m_k, w_j, w_z) = \mathbf{0}$ if m_i and w_j are married (i.e. $\eta(m_i) = j$ and $\eta(w_j) = i$) and if there exists a different matched couple (m_k, w_z) (i.e. $k \neq i, z \neq j$ and $\eta(m_k) = z$ and $\eta(w_z) = k$) such that $c_{m_i}(m_i = j) <_S c_{m_i}(m_i = z) \wedge c_{w_z}(w_z = k) <_S c_{w_z}(w_z = i)$, where S represents the chosen semiring (see Sec. 2). In this case we use the $\leq_S$ since we are looking for a weakly stable marriage (see Sec. 3), otherwise we should define the stability constraints by using $<_S$ for super and strong stabilities (see Sec. 3). Therefore, we need n^4 stability constraints of this kind.

Given this encoding, the set of consistent solutions of P is equivalent to the set of solutions of I (i.e. a OSM problem instance). Therefore, unsatisfying the marriage or stability constraints makes P inconsistent (see Sec. 2). By using this formalization it is now possible to easily maximize the global satisfaction of all the couples, and thus finding a solution for the OSM problem. In practice it is possible to obtain the best possible solution of the considered SCSP problem by exploiting the properties of the chosen semiring operators, i.e. + and ×.

For example, we could consider the preference as a cost, and the cost of the complete match could be obtained by summing together the costs of all the found (non-blocking) pairs. In this case, and if we want to minimize the cost of the n marriages, we can use the *Weighted* semiring [1, 4], i.e. $\langle \mathbb{R}^+, min, \hat{+}, +\infty, 0\rangle$ ($\hat{+}$ is the arithmetic sum). Therefore, what we solve is exactly the (1) problem in Sec. 3.

Otherwise, we can use the *Fuzzy* semiring $\langle [0, 1], max, min, 0, 1\rangle$ [1, 4] to maximize the "happiness" of the least sympathetic couple overall: the fuzzy values in the interval $[0, 1]$ represent an "happiness degree" of the people relationships and are aggregated with *min*, but preferred with *max*. Again, what we solve with this semiring is exactly the (2) problem in Sec. 3, if we consider the ordering of the preferences as inverted (i.e. a high preference is better than a lower one); this is the reason why we use max – min instead of min – max.

As a last example on the expressiveness of our framework, we can use the *Probabilistic* semiring $\langle [0..1], max, \hat{\times}, 0, 1\rangle$ [1, 4] ($\hat{\times}$ is the arithmetic multiplication) in order to maximize the probability that the obtained couples will not split.

Moreover, we can represent the SMI extension reported in Sec. 3 by simply declaring a preference constraint with value corresponding to $\mathbf{0}$: $c_{m_i}(m_i = j) = \mathbf{0}$ if m_i has not expressed a preference for w_j. Further on, by having the same value in the same preference list, i.e. $c_{m_i}(m_i = j) = a$ and $c_{m_i}(m_i = z) = a$, we can represent also the SMT problem defined in Sec. 3. In Sec. 4.1 we consider and solve the most general problem among those presented in Sec. 3, i.e. the *Optimal SMTI* (*OSMTI*).

Notice that such semiring structures allows us to consider also the preferences of men and women as partially ordered (see Sec. 2), which is clearly more generic and expressive w.r.t. the total ordering of the classical problem: Bob could love/like Alice and Chandra more than Drew, but he could not relate the first two girls with each other.

4.1 Solving the OSM Problem in ILP

In this section we solve the soft constraint formalization of the OSMTI problem given with preference, marriage and stability constraints. To achieve this goal, we represent and solve it in ILP by using AMPL [7]. AMPL is a modeling language for mathematical programming with a very general and expressive syntax. It covers a variety of types and operations for the definition of indexing sets, as well as a range of logical expressions. The solution can be obtained with different solvers which can interface to AMPL; for our example we use the *CPLEX* solver (developed by ILOG) for mathematical programming. The soft constraints can be represented with AMPL statements.

We consider an instantiation of the (1) problem in Sec. 3, and therefore the adopted semiring is $\langle \mathbb{R}^+, min, \hat{+}, +\infty, 0\rangle$, even if, as said before, we can solve also other criteria by changing the semiring. The two matrices M and W in Fig. 1 respectively represent the preference values of $n = 6$ men ($MEN = \{m_1, m_2, m_3, m_4, m_5, m_6\}$) and $n = 6$ women ($WOMEN = \{w_1, w_2, w_3, w_4, w_5, w_6\}$) taken from the Weighted semiring set. Notice that both M and W are displayed Fig. 1 with men on rows and women on columns, in order to improve the readibility during a comparison of the two matrices. The lists of preferences of men are represented by the rows of M, and the preferences of women are instead the columns of W.

```
set MEN := m1 m2 m3 m4 m5 m6 ;
set WOMEN := w1 w2 w3 w4 w5 w6 ;

param M:                                param W:
      w1  w2  w3  w4  w5  w6 :=              w1  w2  w3  w4  w5  w6 :=
m1    1   4   Inf 5   5   3            m1    1   4   6   2   4   2
m2    3   4   6   1   5   2            m2    5   1   4   5   2   6
m3    1   Inf 4   2   3   5            m3    4   5   2   2   Inf 3
m4    6   1   3   4   2   1            m4    4   2   1   4   5   5
m5    3   1   2   4   5   6            m5    2   6   5   Inf 6   1
m6    3   3   1   6   5   4 ;          m6    3   Inf 3   6   3   4 ;
```

Fig. 1. The data file of our example in AMPL: the sets of *MEN* and *WOMEN* and their respective preference lists (M and W).

Since we want to deal with incomplete lists, the preference value corresponds to the bottom element of the semiring (in Weighted semiring, it is ∞) if that preference has not been expressed; *Inf* in Fig. 1 is a shortcut for a very big value that we can safely consider as the infinite value (e.g. 10000). For example, in Fig. 1 $M[m_1, w_3] = \infty$ means that m_1 has no preference for w_3. Moreover, we can deal with ties at the same time, e.g. $M[m_4, w_2] = M[m_4, w_6] = 1$ in Fig. 1.

Notice that this problem could have no solution in general due to the fact that the preference lists are incomplete and we want to find a perfect match (n pairs). Moreover, since we have ties and we require a weakly stable matching, the problem is NP-hard [15].

```
option solver cplex;

### PARAMETERS ###
set MEN;
set WOMEN;
param M {i in MEN, j in WOMEN};
param W {k in MEN, z in WOMEN};

### VARIABLES ###
var Marriage {i in MEN, j in WOMEN} binary;

### OBJECTIVE ###
minimize EgalitarianCost:  sum {i in MEN, j in WOMEN}
     (( Marriage[i,j] * M[i,j] ) +
     ( Marriage[i,j] * W[i,j] )) ;

### CONSTRAINTS ###
subject to MenMarriages {i in MEN}:
     sum {j in WOMEN} Marriage[i,j] = 1 ;
subject to WomenMarriages  {j in WOMEN}:
     sum {i in MEN} Marriage[i,j] = 1 ;
subject to Stability {i in MEN, k in MEN, j in WOMEN, z in WOMEN:
     ( M[i,z] < M[i,j] ) and
     ( W[i,z] < W[k,z] )}:
     Marriage[i,j] + Marriage[k,z] <= 1;
```

Fig. 2. The file storing the model for our example in AMPL.

A Formalization in ILP. With AMPL we need to create two files storing the data of the problem (Fig. 1) and its model (Fig. 2). The *Marriage* variable in Fig. 2 corresponds to the couples representing the best stable marriage, while the *EgalitarianCost* is exactly computed as for the problem (1) in Sec. 3 and the goal is to minimize it. Notice that by changing the mathematical operators of the *OBJECTIVE* in Fig. 2, it is possible to solve also problems (2) and (3) of Sec. 3. The *MenMarriages* and *WomenMarriages* constraints respectively state that each man and each woman must have a partner, that is we require a perfect match. At last, the *Stability* constraint prevents blocking pairs.

The three marriages that can be obtained with this formalization are respectively $SM_1 = \{(m_1, w_1), (m_2, w_2), (m_3, w_4), (m_4, w_6), (m_5, w_5), (m_6, w_3)\}$, $SM_2 =$

$\{(m_1, w_1), (m_2, w_2), (m_3, w_4), (m_4, w_3), (m_5, w_6), (m_6, w_5)\}$ and, at last, $SM_3 = \{(m_1, w_1), (m_2, w_2), (m_3, w_4), (m_4, w_5), (m_5, w_6), (m_6, w_3)\}$. The egalitarian costs for these three matches are respectively $ec(SM_1) = 32$, $ec(SM_2) = 30$ and $ec(SM_3) = 29$, which is also the result of the program in Fig. 2 since it corresponds to the lowest possible cost.

5 Multi-Agent Systems and the Stable Marriage of Coalitions

Cooperating groups, referred to as coalitions, have been thoroughly investigated in Artificial Intelligence and Games Theory and has proved to be a useful strategy in both real-world economic scenarios and multi-agent systems [13].

Coalitions in general are task-directed and short-lived, but however last longer than team organization [13] (for example) and in some cases they have a long lifetime once created [10]. Given the population of entities E, the problem of coalition formation consists in selecting the appropriate partition of E, $P = \{C_1, \ldots, C_n\}$ ($|P| = |A|$ if each entity forms a coalition on its own), s.t. $\forall C_i \in P$, $C_i \subseteq E$ and $C_i \cap C_j = \emptyset$, if $i \neq j$. P maximizes the utility (utility against costs) that each coalition can achieve in the environment. Therefore, agents group together because an utility can be gained by working in groups, but this growth is somewhat limited by the costs associated with forming and maintaining such a structure.

Cooperation involves a degree of risk arising from the uncertainties of interacting with autonomous self-interested agents. Trust [17] describes a nodes belief in another nodes capabilities, honesty and reliability based on its own direct experiences. Therefore trust metrics have been already adopted to perceive this risk, by estimating how likely other agents are to fulfill their cooperative commitments [10, 6]. Since trust is usually associated with a specific scope [17], we suppose that this scope concerns the task that the coalition must face after its formation; for example, in electronic marketplaces the agents in the same coalition agree with a specific discount for each transaction executed [6, 19]. Clearly, an entity can also trust itself in achieving the task, and can form a singleton coalition.

In the individually oriented approach an agent prefers to be in the same coalition with the agent with whom it has the best relationship [6]. In the socially oriented classification the agent instead prefers the coalition in which it has most summative trust [6]. Alternatively, in this Section we would like to rephrase the classical notion of stability in SM problems (presented in Sec. 3) as coalition formation criteria. Moreover, instead of a preference (as in Sec. 3), we need to consider a trust relationship between two entities, which, inherently expresses a preference in some sense. To do so, in Def. 1 we formalize how to compute the trustworthiness of a whole coalition:

Definition 1 *Given a coalition C of agents defined by the set $\{x_1, \ldots, x_n\}$ and a trust function t defined on ordered couples (i.e. $t(x_i, x_y)$ is the trust score that x_i has collected on x_j), the trustworthiness of C (i.e. $T(C)$) is defined as the composition (i.e. $\circ$) of the*

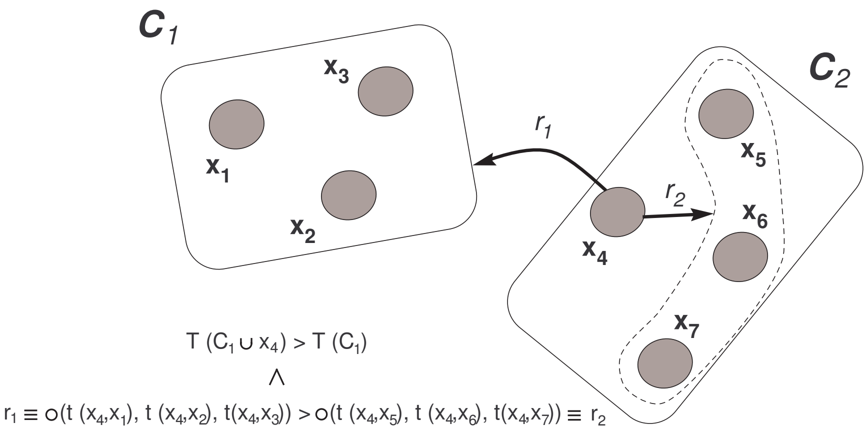

Fig. 3. A graphical intuition of two blocking coalitions.

1-to-1 trust relationships, i.e. $\forall x_i, x_j \in C. \circ t(x_i, x_j)$ *(notice that i can be equal to j, modeling the trust in itself).*

The $\circ$ function has already been defined in [5]; it models the composition of the 1-to-1 trust relationships. It can be used to consider also subjective ratings [17] (i.e. personal points of view on the composition), even if in this paper we will consider objective ratings [17] in order to easily represent and compute trust with a mathematical operator. For instance, some practical instantiations of the $\circ$ function can be the *arithmetic mean* or the *max* operator: $\forall x_i, x_j \in C. \, avg\, t(x_i, x_j)$ or $\forall x_i, x_j \in C. \max t(x_i, x_j)$. Notice that the $\circ$ operation is not only a plain "addition" of the single trust values, but it must take into account also the "added value" (or "subtracted value") derived from the combination effect.

As proposed also in Sec. 4 for the classical problem, by changing the semiring structure we can represent different trust metrics [5, 20]. Therefore, the optimization of the set of coalitions can follow different principles, as, for example, minimizing a general cost of the aggregation or maximizing "consistency" evaluation of the included entities, i.e. how much their interests are alike. In order to extend the stability condition of the classical problem, blocking coalitions are defined in Def. 2:

Definition 2 *Two coalitions* C_u *and* C_v *are defined as* blocking *if, an individual* $x_k \in C_v$ *exists such that,* $\forall x_i \in C_u, x_j \in C_v$ *with* $j \neq k$, $\circ_{x_i \in C_u} t(x_k, x_i) > \circ_{x_j \in C_v} t(x_k, x_j)$ *and* $T(C_u \cup x_k) > T(C_u)$ *at the same time.*

Clearly, a set $\{C_1, C_2, \ldots, C_n\}$ of coalitions is *stable* if no blocking coalitions exist in the partitioning of the agents. An example of two blocking coalitions is sketched in Fig. 3: if x_4 prefers the coalition C_1 (i.e. relationship r_1 in Fig. 3) to the elements in its coalitions C_2 (i.e. r_2 in Fig. 3), i.e. $\circ(t(x_4, x_1), t(x_4, x_2), t(x_4, x_3)) >$

$\circ(t(x_4, x_5), t(x_4, x_6), t(x_4, x_7))$, and C_1 increases its trust value by having x_4 inside itself, i.e. $T(C_1 \cup x_4) > T(C_1)$, then C_1 and C_2 are two blocking coalitions and the partitioning $\{C_1, C_2\}$ is not stable and thus, it is not a feasible solution of our problem.

We therefore require the stability condition to be satisfied, but at the same time we want also to optimize the trustworthiness of the partitioning given by aggregating together all the trustworthiness scores of the obtained coalitions.

5.1 A Formalization of the Problem

As accomplished in Sec. 4 for the classical problem, in this Section we define the soft constraints needed to represent the coalition-extension problem. As an example, we adopt the *Fuzzy* semiring $\langle [0, 1], max, min, 0, 1\rangle$ in order to maximize the minimum trustworthiness of all the obtained coalitions (as proposed also in [5, 2]). The following definition takes the general $\circ$ operator (presented in Sec. 5) as one of its parameters: it can be considered in some sense as a "lower level" operator with respect to the other two semiring operators (i.e. + and $\times$).

The variables V of this problem are represented by the maximum number of possible coalitions: $\{co_1, co_2, \ldots, co_n\}$ if we have to partition a set $\{x_1, x_2, \ldots, x_n\}$ of n elements. The domain D for each of the variables is the powerset of the element identifiers, i.e. $\mathcal{P}\{1, 2, \ldots, n\}$; for instance, if $\eta(co_1) = \{1, 3, 5\}$ it means the the coalition co_1 groups the elements x_1, x_2, x_5 together ($\eta : V \to D$ is the variable assignment function shown in Sec. 1). Clearly, $\eta(co_i) = \emptyset$ if the framework finds less than n coalitions.

1. *Trust* constraints. As an example from this class of constraint, the soft constraint $c_t(co_i = \{1, 3, 5\}) = a$ quantifies the trustworthiness of the coalition formed by $\{x_1, x_3, x_5\}$ into the semiring value represented by a. According to Def. 1, this value is obtained by using the $\circ$ operator and composing all the 1-to-1 trust relationships inside the coalition. In this way we can find the best set of coalitions according to the semiring operators. This kind of constraints resembles the preference constraints given in Sec. 4.
2. *Partition* constraints. This set of constraints is similar to the *Marriage* constraints proposed in Sec. 4. It is used to enforce that an element belongs only to one single coalition. For this goal we can use a binary crisp constraint between any two coalition, as $c_p(co_i, co_j) = \mathbf{0}$ if $\eta(co_i) \cap \eta(co_j) \neq \emptyset$, and $c_p(co_i, co_j) = \mathbf{1}$ otherwise (with $i \neq j$). Moreover, we need to add one crisp constraint more, in order to check that all the elements are assigned to one coalition at least: $c_p(co_1, co_2, \ldots, co_n) = \mathbf{0}$ if $|\eta(co_1) \cup \eta(co_2) \cup \cdots \cup \eta(co_n)| \neq n$, and $c_p(co_1, co_2, \ldots, co_n) = \mathbf{1}$ if $|\eta(co_1) \cup \eta(co_2) \cup \cdots \cup \eta(co_n)| = n$.
3. *Stability* constraints. These crisp constraints model the stability condition extended to coalitions, as proposed in Def. 2. We have several ternary constraints for this goal: $c_s(co_v, co_u, x_k) = \mathbf{0}$ if $k \in \eta(co_v)$ (i.e. x_k belongs to the co_v coalition), $\circ_{i \in \eta(co_u)} t(x_k, x_i) > \circ_{j \in \eta(co_v)} t(x_k, x_j)$ and $c_t(\eta(co_u) \cup k) > c_t(co_u)$. Otherwise, $c_s(co_v, co_u, x_k) = \mathbf{1}$.

6 Conclusions

In this paper we have presented a general soft-constraint based framework where to represent and solve the Optimal Stable Marriage (OSM) problem [14] and its variants: with incomplete preference lists ad also ties inside the same list. The optimization criteria depend on the chosen semiring (e.g. *Weighted* or *Fuzzy*) which can be used to solved problems already proposed in literature, as, for example, to minimize the egalitarian cost (see Sec. 3 and Sec. 4). Therefore, it is possible to solve all these different optimization problems with the same general framework, and we do not need an ad-hoc algorithm for each distinct case (e.g. [14, 11, 16]). One of the aims of this paper is to relate the OSM and the SCSP problems as done also for the classical SM and (crisp) CSP problems [9, 21]. Since many variants of the OSM problem are NP-hard [15], representing and solving the problem as a SCSP can be a valuable strategy [9].

Notice also that *Integer Linear Programming*, i.e. the tool adopted to find a solution for the related soft constraint problem, is here applied to this kind of problems for the first time.

Moreover, in this paper we have extended the OSM problem to achieve stable coalitions of agents/individuals by using trust metrics as way to express preferences. Thus, we extend the stability conditions from agent-to-agent to agent-to-coalition (of agents); in this case the marriage is between an agent and a group of agents. What we obtain is a partition of the set of agents into trusted coalitions, such that no agent or coalition is interested in breaking the current relationships and consequently changing the partitioning. As a future work, we would like to use ILP to solve also the problem extension to coalition formation, which has been modeled in Sec. 5.1.

In the future, we could try to extend the results of this paper by modeling the formation and the consequent behaviour of the other organizational paradigms presented in [13], e.g. Holoarchies, Federations or Teams. To do so, we need to represent the different grouping relationships among the entities with soft constraints.

We would like also to further explore the strong links between OSM and Games Theory, for example by developing even more sophisticated notions of stability.

References

1. S. Bistarelli. *Semirings for Soft Constraint Solving and Programming*, volume 2962 of *LNCS*. Springer, 2004.
2. S. Bistarelli, F. Martinelli, and F. Santini. Trust evaluation in ad-hoc networks. In *to appear in Autonomic and Trusted Computing (ATC-08)*. Springer, 2008.
3. S. Bistarelli, U. Montanari, and F. Rossi. Soft concurrent constraint programming. *ACM Trans. Comput. Logic*, 7(3):563–589, 2006.
4. S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based Constraint Solving and Optimization. *Journal of the ACM*, 44(2):201–236, March 1997.

5. S. Bistarelli and F. Santini. Propagating multitrust within trust networks. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 1990–1994, New York, NY, USA, 2008. ACM.
6. S. Breban and J. Vassileva. A coalition formation mechanism based on inter-agent trust relationships. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 306–307, New York, NY, USA, 2002. ACM.
7. R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Brooks/Cole Publishing Company, 2002.
8. D. Gale and L. S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.
9. I. P. Gent, R. W. Irving, D. Manlove, P. Prosser, and B. M. Smith. A constraint programming approach to the stable marriage problem. In *Conference on Principles and Practice of Constraint Programming*, pages 225–239, London, UK, 2001. Springer-Verlag.
10. N. Griffiths and M. Luck. Coalition formation through motivation and trust. In *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 17–24, New York, NY, USA, 2003. ACM.
11. D. Gusfield. Three fast algorithms for four problems in stable marriage. *SIAM J. Comput.*, 16(1):111–128, 1987.
12. D. Gusfield and R. W. Irving. *The stable marriage problem: structure and algorithms*. MIT Press, Cambridge, MA, USA, 1989.
13. B. Horling and V. Lesser. A survey of multi-agent organizational paradigms. *Knowl. Eng. Rev.*, 19(4):281–316, 2004.
14. R. W. Irving, P. L., and D. Gusfield. An efficient algorithm for the "optimal" stable marriage. *J. ACM*, 34(3):532–543, 1987.
15. K. Iwama and S. Miyazaki. A survey of the stable marriage problem and its variants. In *International Conference on Informatics Education and Research for Knowledge-Circulating Society (icks 2008)*, pages 131–136. IEEE Computer Society, 2008.
16. K. Iwama, S. Miyazaki, and H. Yanagisawa. Approximation algorithms for the sex-equal stable marriage problem. In *Algorithms and Data Structures, 10th International Workshop, WADS*, volume 4619 of *LNCS*, pages 201–213. Springer, 2007.
17. A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decis. Support Syst.*, 43(2):618–644, 2007.
18. D. E. Knuth. *Stable marriage and its relation to other combinatorial problems: An introduction to the mathematical analysis of algorithms*. American Mathematical Society, 1997.
19. K. Lerman and O.Shehory. Coalition formation for large-scale electronic markets. In *ICMAS*, pages 167–174. IEEE Computer Society, 2000.
20. G. Theodorakopoulos and J. S. Baras. Trust evaluation in ad-hoc networks. In *WiSe '04: Proceedings of the 3rd ACM workshop on Wireless security*, pages 1–10, New York, NY, USA, 2004. ACM.
21. C. Unsworth and P. Prosser. Specialised constraints for stable matching problems. In *CP*, volume 3709 of *Lecture Notes in Computer Science*, page 869. Springer, 2005.