A Semiring-based framework for fair resources allocation

Paola Campli

University G.D'Annunzio - Pescara, Italy
campli@sci.unich.it

Abstract

In this paper a general framework (based on soft constraints) to model and solve the fair allocation problem is proposed. Our formal approach allows to model different allocation problems, ranging from goods and resources allocation to task and chore division. Soft constraints are employed to find a fair solution by respecting the agents's preferences; indeed these can be modeled in a natural fashion by using the Semiring-based framework for soft constraints. The fairness property is considered following an economical point of view, that is, taking into account the notions of *envy-freeness* (each player likes its allocation at least as much as those that the other players receive, so it does not envy anybody else) and *efficiency* (there is no other division better for everybody, or better for some players and not worse for the others).

1998 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases soft constraints fairness allocation resources

Digital Object Identifier 10.4230/LIPIcs.ICLP.2011.268

1 Introduction and problem description

The problem of "fair division", that is, fairly dividing resources or costs among a set of people, is an important issue in real life scenarios; it can refer to several situations, such as inheritance and divorce settlements, division of health resources, computer networking resources, voting power, intellectual property licenses, costs for environmental improvements, etc. In these cases, formal protocols for division are needed. Many variations of the basic problem exist, for example, the situation with divisible resources is quite different from the situation with indivisible objects; the items to be divided can be goods or sometimes "bads" like chores or other burdens; some problems can involve the division of money to compensate a "non fair share", or a payoff in exchange for performing a chore. Other aspects to consider are the number of objects with respect to the number of people. If goods are scarce, an *auction* is needed and the items are assigned to (usually) one winner; in this case fairness methods are studied in repeated auctions to guarantee that not always the same player will be the winner.

But most of the variation comes from the fact that there are many reasonable ways to formalize "fairness" including *max-min fairness*, *proportional fairness*, *envy-free* fairness, etc. which may or may not lead to the same optimal allocation; if we take into account a *global view* this means looking at the overall allocation in terms of social welfare, while a *local view* focus on the agents preferences.

In this paper we investigate on the allocation of indivisible objects which can be either goods or bads; thus, given a set of items and a set of people, each person states a weight for each object which, depending on the cases, can represent preferences or costs (such as time, money, resources etc.). According our model, the solution will be an *envy-free* allocation of objects to the agents, reminding that envy-freeness is a fairness property that guarantees

Technical Communications of the 27th International Conference on Logic Programming (ICLP'11).
Editors: John P. Gallagher, Michael Gelfond; pp. 268–273
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

that no agent would rather have one of the bundle allocated to any of the other agents, that is, each player prefers his bundle.

The paper is organized as follows: in Sec. 2.1 the various aspects of fair division problems are illustrated; in Sec. 2.2 we give a background on Soft Constraint Satisfaction Problems and semirings; in Sec. 4.1 we show how to use the SCSP framework to model allocation problems; in Sec. 4.2 the mapping between SCSP and allocation problems is provided; finally, Sec. 5 contains references to similar works in the field of fair division and allocation problems, a short summary and our future works.

2 Background and overview of the existing literature

In the indivisible resource area, most of the issues are based on the *Santa Claus problem* [4], in which the goal is to distribute n presents among k kids, in such a way that the least lucky kid is as happy as possible; a linear programming is used with the *Max-min fairness* objective function. Another variant is represented by the "housemate problem" [5], where goods are associated with bads. The problem consists in assigning different rooms to people sharing a house according the bid they submit, but also to determine a price to be paid by each roommate for his assigned room. Concerning Chore division, the problem of dividing an undesirable object has been investigated only for divisible goods, where cake cutting algorithms have been adapted in order to deal with chores instead of desirable goods. It is supposed that chores are infinitely divisible [3] and valuations over bundles are additive. Other works view the problem from a computational perspective and are based on approximation algorithms with the purpose of finding a solution closest to the optimum [6].

2.1 The problem of fair division

Fair division [2] is the problem of dividing one or several goods amongst two or more agents in a way that satisfies a suitable fairness criterion. Fair division has been studied in philosophy, political science, economics and mathematics for a long time, but is also relevant to computer science and multiagent systems (MAS), in which resource allocation is a central topic since the application or agents need resources to perform tasks. It is assumed that agents are autonomous. A solution needs to respect and balance their individual preferences; fairness definitions are required and once we have a well-defined fair division problem, we require an algorithm to solve it.

The elements of a Fair Division Problem are a set P of n players: $p_1, p_2, ..., p_n$ and a set of m objects O to be divided. The problem is to divide the set O into n shares ($o_1, o_2, ..., o_n$) so that each player gets a fair share of O. A *fair share* is any share that, in the opinion of the player receiving it, has a value that is at least $\frac{1}{n}$ of the total value of the set of goods O. It is crucial to understand that share value is subjective, and that each player may even have a different notion of how much the set to be divided is worth.

There are three types of fair division schemes: the *Continuous Fair Division Schemes*, in which the set O is infinitely divisible (cake, land, etc.) and shares can be adjusted by arbitrarily small amounts; the *Discrete Fair Division Schemes* where the set O is made up of indivisible objects (cars, houses, etc) and the *Mixed Fair Division Schemes*. In this paper, since we are dealing with indivisible objects, we will focus on the discrete case.

2.2 Constraint Satisfaction Problems, Semirings and Soft Constraints

The classic definition of a Constraint Satisfaction Problem (CSP) is as follows [10]. A CSP P is a triple $P = \langle X, D, C\rangle$ where X is an n-tuple of variables $X = \langle x_1, x_2, \ldots, x_n\rangle$, D is a corresponding n-tuple of domains $D = \langle D_1, D_2, \ldots, D_n\rangle$ such that x_i can assume values within a determined domain D_i, C is a t-tuple of constraints $C = \langle C_1, C_2, \ldots, C_t\rangle$. A constraint C_j is a pair $\langle Rl_{S_j}, S_j\rangle$ where Rl_{S_j} is a relation on the variables in $S_i = scope(C_i)$. A solution to the CSP P is an n-tuple $A = \langle a_1, a_2, \ldots, a_n\rangle$ where $a_i \in D_i$ and each C_j is satisfied in that Rl_{S_j} holds on the projection of A onto the scope S_j . In a given task one may be required to find the set of all solutions, *sol(P)*, to determine if that set is non-empty or just to find any solution, if one exists. If the set of solutions is empty the CSP is unsatisfiable. A *c-semiring* [8, 9] 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 $\mathbf{0}, \mathbf{1} \in A$ (respectively the bottom and top elements of A) and with two operations $+$ and $\times$ that satisfy certain properties: $+$ is defined over (possibly infinite) sets of elements of A and is commutative, associative and idempotent; it is closed, $\mathbf{0}$ is its unit element and $\mathbf{1}$ is its absorbing element; $\times$ is closed, associative, commutative and distributes over $+$, $\mathbf{1}$ is its unit element and $\mathbf{0}$ is its absorbing element (for the exhaustive definition, please refer to [8]). The $+$ operation defines a partial order $\leq_S$ over A such that $a \leq_S b$ iff $a + b = b$; intuitively $a \leq_S b$ 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 min and $\mathbf{1}$ its max, $\langle A, \leq_S\rangle$ is a complete lattice and $+$ is its lub. A *soft constraint* [8, 9] 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.

Given the set $\mathcal{C}$, the combination function $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ is defined as $(c_1 \otimes c_2)\eta = c_1\eta \times c_2\eta$ [8, 9]. The $\otimes$ builds a new constraint 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. Given a constraint $c \in \mathcal{C}$ and a variable $v \in V$, the *projection* [8, 9, 11] of c over $V - \{v\}$, written $c \Downarrow_{(V \setminus \{v\})}$ is the constraint c' such that $c'\eta = \sum_{d \in D} c\eta[v := d]$. Informally, projecting means eliminating some variables from the support. A SCSP [9] is a tuple $P = \langle X, D, C, A\rangle$ where X is a set of variables, D is the domain of the variables and C is a set of constraints over X associating values form a c-semiring A. The *best level of consistency* notion defined as $blevel(P) = Sol(P) \Downarrow_\emptyset$, where $Sol(P) = \bigotimes C$ [9]. A problem P is α-consistent if $blevel(P) = \alpha$ [9]; P is instead simply “consistent” iff there exists $\alpha >_S \mathbf{0}$ such that P is α-consistent [9]. P is inconsistent if it is not consistent.

3 Goal of the research

Although there is wide literature on fair division within the fields of economics, game theory, political science, mathematics, operations research and computer science, it seems to lack a unified and general framework which allows to solve the different kinds of problems, each one with different objects (desirable or undesirable items), weights or preferences. Another issue encountered in previous works is that often it is not possible to find a solution and the problem remains unsolved; our approach might be applied in all these cases because the use of soft constraints allows to always find a solution; moreover we provide a general framework which can model several cases by choosing the appropriate semiring (see Sec. 2.2).

4 Preliminary results accomplished

4.1 The SCSP framework for allocations problems

In this section we define a quantitative framework to model fair division problems, where each assignment of objects to people have an associated preference/weight and, consequently, modeling this kind of problems as Soft CSPs (see Sec. 2.2) leads to an allocation of goods to people that optimize the criteria defined by the chosen semiring. For instance, the *Weighted* semiring $\langle \mathbb{R}^+, min, \hat{+}, 0, 1\rangle$, where $\hat{+}$ is the arithmetic plus ($\mathbf{0} = \infty$ and $\mathbf{1} = 0$), can be used to model the undesirable objects case (such as chore division) by expressing the (e.g. money) cost for performing a chore; the optimum solution in this scenario corresponds to an allocation with minimum total cost. The *Fuzzy* semiring $\langle [0..1], max, min, 0, 1\rangle$ is well suited for modeling the players's preferences with respect to each good; the solution we obtain with this semiring corresponds in choosing the highest among the minimum preferences. The *Probabilistic* semiring $\langle [0..1], max, \hat{\times}, 0, 1\rangle$ can be used when preferences are unknown, thus, weights corresponds to probabilities; we can express for instance, that person p_1 prefers object o_3 with probability 0.4. The arithmetic $\hat{\times}$ is used to compose the probability values together (since we assume that preferences and thus probabilities are independent). By using the *Boolean* semiring $\langle \{true, false\}, \vee, \wedge, false, true\rangle$ we can solve the non weighted allocation problems, that is, each person states only the goods he/she desires (or the tasks he is able to perform).

4.2 Mapping Allocation Problems to SCSPs

In this section we show a mapping from the allocation problem to SCSPs. An allocation problem is formed by a set of m indivisible objects (or items) $O = \{o_1, o_2, \ldots, o_m\}$ and a set of people (or players) $P = \{p_1, \ldots, p_n\}$. Each player has their own preferences or costs regarding the allocation of goods/tasks to be selected. The problem consists in partitioning the set of objects in n subsets (or bundles) in a way that each person receives a (non-empty) bundle that satisfies a suitable fairness criterion. In order to model this problem with a SCSP, we define a variable for each object $o_i \in O$, i.e. $V = \{o_1, o_2, \ldots, o_m\}$ and the domain of each variable is the set of people in P: $D = \{p_1, \ldots, p_n\}$, meaning that an object can be assigned to a person in the set P; for example $o_1 = p_2$ means that player p_2 receives object o_1. A soft constraint associates a semiring value for each assignment of the variables in its scope, which represent the preference of the player for a given item; if no weights are considered , the corresponding variable assignment is *not admitted* or *admitted* and the values **0** or **1** of the boolean semiring set are respectively returned.

▶ **Example 1.** As a simple example, suppose we must assign 3 objects (o_1, o_2, o_3) to 2 players (p_1, p_2). The corresponding SCSP, by using (for instance) a Fuzzy semiring, has three variables: o_1, o_2 and o_3, each with the domain $D = \{p_1, p_2\}$; we define the following unary constraints: $C_{o1} := \{(p_1, 0.7); (p_2, 0.2); \}$ meaning that object 1 can be assigned either to person 1 (who has a preference of 0.7 for this objects) or person 2 (with preference 0.2); $C_{o2} := \{(p_1, 0.3); (p_2, 0.1); \}$ that is, object 2 can be assigned either to person 1 or 2 with preferences 0.3 and 0.1 respectively; $C_{o3} := \{(p_2, 0.7)\}$ meaning that object 3 can only be assigned to person 2 who desires the object with preference 0.7;

the solution is illustrated in the table below:

o1	o2	o3		$Sol(P)$
p1	p1	p1	not allowed	
p1	p1	p2	$0.7 \times 0.3 \times 0.7$	**0.3**
p1	p2	p1	not allowed	
p1	p2	p2	$0.7 \times 0.1 \times 0.7$	0.1
p2	p1	p1	not allowed	
p2	p2	p1	not allowed	
p2	p2	p2	$0.2 \times 0.1 \times 0.7$	0.1
p2	p1	p2	$0.2 \times 0.3 \times 0.7$	0.2

The (unique) optimal solution of this problem is $o_1 = p_1, o_2 = p_1 o_3 = p_2$ (with preference 0.3).

Depending on the cases, the solution provided might not be fair if we only use the previous method. For example, with different preferences, the solution (p_2, p_2, p_2) could be returned, which is certainly unfair, since player 1 does not get any object and envies player 2. For this reason, we need to specify additional constraints in order to solve the allocation problem and guarantee the *envy-freeness* property of fairness. Let x_{ij} be a boolean variable which is equal to 1 if item j is assigned to player i and 0 otherwise and let $u_i(B_i)$ be the value for person i of the set of objects (B_i) assigned to him; this value represents the valuation of the bundle (that is, the subset of items) for each person; an issue encountered in this case is that requesting an input to the agents for every possible combination of goods is NP-hard, in fact for m object there are 2^m valuations for each of the n players. In order to reduce the size of the problem, we can automatically calculate the value of the bundle, by specifying in the problem if the valuations are *additive* (thus, the value is obtained by summing the weights of the single objects in the bundle), *super-additive* (the value of the bundle is greater than the values of the single objects), *sub-additive* (the value of the bundle is lower than the values of the single objects) or *maximal* (the value of the bundle corresponds to the maximum weight among the objects in the bundle); in this way we can compute the value of the entire bundle $u_i(B_i)$; the type of valuation depends on the kind of goods; for example if the items considered are *complementary* (e.g. printers and ink cartridges) the valuation of the bundle might be super-additive, or conversely, if the goods are *substitute* (e.g. petroleum and natural gas), the valuation might be sub-additive. The defined constraints are the following: [1]

1. Each object must be assigned to at most one person $\forall j \quad \sum_i x_{ij} = 1$;
2. Each person must receive at least one item: $\forall i \quad \sum_j x_{ij} \geq 1$;
3. *Envy-freeness constraint.* Each person does not prefer the set of objects assigned to the other players: $\forall i \quad u_i(Bi) \geq u_i(Bj) \quad$ for each $j \neq i$;

Moreover, since we are assuming that the number of objects is greater (or equal) than the number of people, our solution is also efficient, as shown in [12], which proves that when $m \geq n$ envy-freeness implies efficiency.

5 Open issues and expected achievements

We investigated on the use of the semiring-based framework for soft constraints in order to model and solve the fair allocation of objects problem. According the chosen semiring, we can easily represent the different set of preferences, their combination and the various kind of objects. In the future we plan to use the framework for the Stable Marriage Problem, which can be casted in a particular fair allocation problem involving the same number of objects and people.

[1] constraints 1 and 2 are based on those used in the Santa Claus Problem's paper [4]

References

1 B. Ivona and D. Varsha, Allocating indivisible goods, SIGecom Exch., volume 5, number 3, 2005, pages 11–18, doi: http://doi.acm.org/10.1145/1120680.1120683, ACM, NY, USA

2 U. Endriss, Lecture Notes on Fair Division,2009, ILLC, University of Amsterdam

3 Elisha Peterson and Francis Edward Su, Four-Person Envy-Free Chore Division, Mathematics Magazine, 2002

4 B. Nikhil and S. Maxim, The Santa Claus problem, STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing, 2006, isbn 1-59593-134-1, pages 31–40, Seattle, WA, USA, doi: http://doi.acm.org/10.1145/1132516.1132522, NY, USA

5 R.F. Potthoff, Use of Linear Programming to Find an Envy-Free Solution Closest to the BramsKilgour Gap Solution for the Housemates Problem, Group Decision and Negotiation, vol 11, pages 405–414, 2002

6 F. Uriel, On allocations that maximize fairness, SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms, 2008, pages 287–293, San Francisco, California, Society for Industrial and Applied Mathematics, PA, USA

7 I. P. Gent, Principles and Practice of Constraint Programming - CP 2009, 15th International Conference, CP 2009, Lisbon, Portugal, September 20-24, 2009, Proceedings, CP, Springer, Lecture Notes in Computer Science, vol: 5732, 2009, isbn: 978-3-642-04243-0, ee: http://dx.doi.org/10.1007/978-3-642-04244-7, DBLP, http://dblp.uni-trier.de

8 S. Bistarelli and U. Montanari and F. Rossi, Semiring-based Constraint Solving and Optimization, Journal of the ACM, vol 44, 2, pages 201-236, 1997

9 S. Bistarelli, Semirings for Soft Constraint Solving and Programming, Springer, LNCS, vol 2962, 2004, isbn 3-540-21181-0, DBLP, http://dblp.uni-trier.de

10 F. Rossi and P. Van Beek and T. Walsh, Handbook of Constraint Programming, 2006, isbn 0444527265, Elsevier Science Inc., NY, USA

11 S. Bistarelli and U. Montanari and F. Rossi, Soft concurrent constraint programming, ACM Trans. Comput. Logic, vol 7, num 3, 2006, issn 1529-3785, pages 563–589, doi: http://doi.acm.org/10.1145/1149114.1149118, ACM Press,NY, USA

12 A. Ahmet and D. Gabrielle and G. David, Fair Allocation of Indivisible Goods and Criteria of Justice,Econometrica, vol 59, num 4, 1991, pag 1023–1039, http://www.jstor.org/stable/2938172, ISSN 00129682, The Econometric Society, Copyright © 1991 The Econometric Society