UNIVERSITÀ DEGLI STUDI DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Matematica Pura ed Applicata

DOTTORATO DI RICERCA IN MATEMATICA COMPUTAZIONALE
CICLO XIX

REASONING WITH PREFERENCES AND UNCERTAINTY

Coordinatore: Ch.mo Prof. Michelangelo Conforti

Supervisore: Ch.ma Prof.ssa Francesca Rossi

Dottorando: Maria Silvia Pini

31 dicembre 2006

Acknowledgements

First of all, I would like to thank my supervisor Professor Francesca Rossi for her continuous supervision and encouragement. Very frequent and intense discussions with her made it all possible. She provided guidance, support and freedom to purse my ideas.

Also, discussions with my collaborators Dr. Kristen Brent Venable, Professor Toby Walsh, Professor Stefano Bistarelli and Professor Jerome Lang have been a vital part of this study throughout these three years.

In addition, I would like to thank reviewers of this PhD thesis Professor Krzysztof R. Apt and Professor Andrea Lodi for their very positive judgments on this work and in particular Professor Krzysztof R. Apt for his interesting suggestions on further research areas.

Finally, many thanks to my husband and my parents for their support and encouragement during this period.

Contents

Introduzione

Molti problemi della vita reale presentano dei vincoli, cioè delle richieste che devono essere soddisfatte totalmente. A volte però risulta più naturale esprimere questi vincoli in maniera meno stringente tramite delle preferenze. Inoltre introducendo le preferenze riusciamo a trovare una soluzione comunque accettabile in alcuni problemi sovravincolati che non avrebbero alcuna soluzione in presenza di soli vincoli.

Oltre alle preferenze, un altro tipo di informazione presente in molti problemi reali è l'incertezza. Molti problemi sono infatti caratterizzati da eventi incerti che non possono essere controllati dall'utente. In alcuni casi l'utente può avere un'informazione di tipo probabilistico o possibilistico riguardo al verificarsi di questi eventi incerti, altre volte può non avere alcuna informazione.

Visto che le preferenze e l'incertezza sono due concetti chiave in molti problemi reali è importante saper modellare fedelmente questi due concetti. Lo scopo di questa tesi di dottorato è quello di definire e studiare dei formalismi che possano modellare problemi con molti tipi di preferenze e l'incertezza, studiare le proprietà di questi formalismi e sviluppare degli strumenti per risolverli, considerando anche il caso in cui le preferenze sono espresse da più agenti. Per raggiungere questo obiettivo abbiamo seguito diverse linee di lavoro.

Siamo partiti considerando un formalismo noto in letteratura per rappresentare le preferenze, cioè il formalismo dei vincoli soft [BMR97]. I vincoli soft sono vincoli classici a cui si associa o all'intero vincolo, oppure ad ogni assegnamento delle variabili, un certo elemento, che è solitamente interpretato come un livello di preferenza o di importanza. Questi livelli sono di solito ordinati e l'ordine riflette l'idea che alcuni livelli sono migliori di altri. Inoltre, tramite un opportuno operatore di combinazione, è possibile ottenere il livello di preferenza di una soluzione globale a partire dalle preferenze nel vincolo.

Dopo aver introdotto il formalismo dei vincoli soft per la rappresentazione delle preferenze, abbiamo presentato la teoria matematica utilizzata in questa tesi per rappresentare l'incertezza, cioè la teoria della possibilità [Zad78]. La teoria della possibilità è una teoria alternativa alla teoria della probabilità che permette di caratterizzare il verificarsi degli eventi incerti quando non si hanno eventi rispetto ai quali riferirsi.

A partire dal formalismo dei vincoli soft e dalla teoria della possibilità abbiamo poi definito un formalismo per modellare problemi con preferenze espresse da un singolo agente in presenza di incertezza. L'idea è quella di rimuovere l'incertezza, cioè la parte del problema che noi non possiamo controllare, come proposto in [DFP96a], e di definire dei nuovi vincoli solo sulla parte controllabile del problema, garantendo però che alcune proprietà desiderabili relative all'ordinamento delle soluzioni e alla robustezza delle soluzioni, cioè alla compatibilità delle soluzioni rispetto agli eventi incerti, vengano soddisfatte. Prima abbiamo considerato problemi con preferenze fuzzy e incertezza e poi abbiamo generalizzato il formalismo per rappresentare un qualunque tipo di preferenza dimostrando che le proprietà desiderate continuavano a valere. Inoltre abbiamo definito un risolutore basato su tecniche di branch and bound per trovare le soluzioni ottime di questi problemi secondo varie semantiche più o meno rischiose rispetto l'incertezza.

Abbiamo poi considerato il concetto della biplarità nell'ambito delle preferenze [BDKP02, BDKP06]. A questo proposito abbiamo definito un formalismo che permette di modellare problemi con preferenze bipolari, cioè con preferenze positive e negative, che rispecchia quello che avviene comunemente nella vita reale. Se devo prendere una decisione e ho due giudizi positivi su di essa allora tale decisione avrà una valutazione complessiva ancora più positiva, al contrario una decisione caratterizzata da due giudizi negativi avrà una valutazione complessiva ancora più negativa. Inoltre se su una stessa decisione si hanno sia dei giudizi positivi sia dei giudizi negativi è naturale che quella decisione abbia una valutazione globale che compensa i giudizi positivi con quelli negativi. Il formalismo che abbiamo definito per rappresentare le preferenze bipolari generalizza il formalismo dei vincoli soft, che modella solo le preferenze negative, permettendo di rappresentare anche le preferenze positive, l'indifferenza e la compensazione tra preferenze positive e negative. Abbiamo inoltre definito un risolutore basato su tecniche di branch and bound per trovare le soluzioni ottime in questi problemi con preferenze bipolari. Abbiamo poi considerato la presenza dell'incertezza anche in problemi bipolari e abbiamo definito una procedura per modellare e risolvere questi problemi che generalizza al caso di preferenze bipolari la procedura descritta sopra nel caso di preferenze fuzzy e incertezza.

Abbiamo poi preso in esame il contesto multiagente, dove più agenti esprimono contemporaneamente le loro preferenze, che possono essere parzialmente ordinate, su un insieme di alternative. In questo caso l'obiettivo è quello di aggregare le preferenze degli agenti per ottenere un ordinamento collettivo delle alternative (social welfare theory) oppure l'alternativa migliore (social choice theory). Abbiamo considerato dei risultati classici in letteratura nell'ambito dell'aggregazione di preferenze totalmente ordinate relativi a due proprietà desiderabili, la fairness e la non manipolabilità [Arr51, MS77, Gib73], e li abbiamo estesi al caso di preferenze parzialmente ordinate.

Quindi abbiamo considerato la presenza dell'incertezza anche nel contesto multiagente. In particolare abbiamo considerato l'aggregazione di preferenze nel caso in cui alcuni agenti decidano di non rivelare le loro preferenze su qualche coppia di alternative. A questo proposito abbiamo esaminato la complessità computazionale di determinare i vincitori necessari e possibili, cioè quelle alternative che sono sempre le più preferite dagli agenti indipendentemente da come l'incompletezza sarà risolta oppure in almeno un modo in cui questa incompletezza potrà essere risolta. Abbiamo dimostrato che è NP-hard calcolare questi vincitori, quindi abbiamo individuato delle condizioni sufficienti sulla regola di aggregazione delle preferenze che permettono di calcolare questi vincitori in tempo polinomiale. Abbiamo inoltre mostrato l'utilità di questi vincitori nell'ambito dell'elicitazione di preferenze. Abbiamo poi considerato una specifica regola di aggregazione delle preferenze (sequential majority voting) che effettua una sequenza di confronti di maggioranza tra coppie di alternative lungo un albero di voto e in cui il vincitore dipende dall'albero di voto scelto. Abbiamo provato che determinare i vincitori possibili e necessari in questo ambito, cioè quelle alternative che vincono in almeno un albero di voto o in tutti gli alberi di voto, è polinomiale, mentre calcolare i vincitori possibili diventa NP-hard se si richiede che l'albero di voto sia bilanciato. In questo caso quindi è difficile per chi decide l'albero di voto manipolare il risultato [BTT95]. Infine abbiamo dimostrato che questi risultati di complessità continuano a valere anche nel caso in cui gli agenti esprimono alcune delle loro preferenze in maniera incompleta.

Chapter 1

Introduction

The aim of this Ph.D. thesis is to define and study formalisms that can model problems with preferences and uncertainty, possibly defined by several agents, and to define tools to solve such problems.

1.1 Motivation

Preferences are ubiquitous in real life. In fact, most problems are over-constrained and would not be solvable if we insist that all their requirements are strictly met, hence it is more reasonable to express their requirements in a soft way, i.e., via preferences. Moreover, many problems are more naturally described via preferences rather than hard statements.

Preferences come in many kinds. In some cases it could be natural to express preferences in quantitative terms, while in other situations it could be better to use qualitative statements. Moreover, preferences can be unconditional or conditional. Finally, preferences can model priorities, rankings, different levels of importance, desires or rejection levels.

Preferences can help whenever the task involves decision making and/or knowledge representation [DFP02, DF05, DF06]. They are essential to treat reasoning about action and time, planning diagnosis and configuration [Jun02]. Preferences are the key to understand the non-crisp aspect of many human behaviors. For example, in mathematical decision theory, preferences (often expressed as utilities) are used to model people's economic behavior. In Artificial Intelligence (AI), preferences help to capture agents' goals. In databases, preferences help in reducing the amount of information returned in response to user queries. In philosophy, preferences are used to reason about values, desires, and duties. Thus, the representation and handling of preferences should be available and efficient in any sophisticated automated reasoning tool.

Preferences are gaining more and more attention in AI, in particular in the Constraint Programming (CP) area also in connection with Operations Research (OR). AI permits complex preference representations and thus allows to reason with and about preferences, providing a new perspective for formalizing preference information in qualitative and quantitative way, that is essential for many decision making problems [Jun02, DT99].

Preferences are one type of soft information present in real-life problems. Another important feature, which arises in many real world problems, is uncertainty. In fact, many problems are characterized by uncertain parameters which are not under the user's direct control, but that can be decided only by Nature. An example of an uncertain parameter is, in the context of satellite scheduling or weather prediction, the time when clouds will disappear, which can be decided only by Nature. Another example in which uncertainty occurs is a scheduling problem, which constrains the order of execution of various activities, where the duration of some activity is uncertain [DFP95]. In this case the goal is to define a schedule which is the most robust with respect to uncertainty.

Uncertainty can be represented in several ways. In some problems the user can be completely ignorant about the occurrence of the uncertain events, in others he can have additional information, which can be more or less precise regarding the occurrence of uncertain events [Zad78, DP88, Wal02, FLS96].

Preferences and uncertainty often coexist in real-word problems. Consider for example a scheduling problem with uncertain durations, which is over-constrained. It would be impossible to solve it if we insist that all its requirements are strictly met. Therefore, it is more reasonable to express (at least some of) its requirements as preferences rather than hard statements. Doing so, we obtain a problem defined by preferences and uncertainty, where the solutions are schedules with different levels of desirability. The goal is then to find solutions with the highest level of desirability which are also robust with respect to uncertain durations.

Since preferences and uncertainty are very often the core of real-life problems, it is important to model faithfully these two aspects, both for problems involving a single agent and for problems regarding multiple agents. While there are several formalisms to handle some notions of preferences and/or uncertainty, much work still needs to be done to handle them in a general and efficient way. The aim of this thesis is to give a contribution in this direction.

1.2 Objectives

There are many issues to be addressed about preferences and uncertainty. The main issue is preference and uncertainty specification and representation, i.e., which formalisms can be used to model the preferences of an agent and the uncertainty of the problem. In this respect, contributions have been brought from studying the axiomatic properties of preferences, as well as logics of preferences or their topological and algebraic structures [Jun02, BMR97, BFM$^+$96] and from defining formalisms for representing various kinds of uncertainty [FL93, DFP96b, DP98]. In a multi-agent scenario, instead, core issues are preference composition, merging and aggregation, as well as preference elicitation and learning priorities, conflict resolution and belief revision [Lan02, Lan04, DP93, Doy91].

The goal of this thesis is to define and study formalisms that can model problems with many kinds of preferences and/or uncertainty, to study properties of such formalisms, and to develop tools to solve such problems. Moreover, we intend to be able to deal also with scenarios where preferences are expressed by several agents and where preference aggregation is therefore needed to find the optimal solutions.

In order to achieve this objective, we start by defining formalisms for expressing preferences of a single agent in presence of uncertainty. We start considering problems where preferences are expressed in a quantitative non conditional way and where uncertainty is characterized by lack of data or imprecise knowledge. In some formalisms for dealing with preferences and uncertainty, uncertainty is expressed in terms of probability theory [Wal02, FLS96]. In this thesis we consider a different form of uncertainty, less precise than the probabilistic one, since we intend to model scenarios where probabilistic estimates are not available. We define a formalism for handling preferences and this kind of uncertainty, and we give algorithms for solving them. To achieve this goal, we exploit two formalisms: the semiring-based soft constraint formalism [BMR97] to deal with preferences, and possibility theory [Zad78] to reason with uncertainty.

Generally speaking, a soft constraint is just a classical constraint plus a way to associate, either to the entire constraint or to each assignment of variables, a certain element, which is usually interpreted as a level of preference or importance. Such levels are usually ordered and the order reflects the idea that some levels are better than others. Moreover, one has also to say, via suitable combination operators, how to obtain the level of preference of a global solution from the preferences in the constraint.

Many formalisms have been developed to describe one or more classes of soft constraints. For instance, consider fuzzy CSPs [Rut94], where the crisp constraints are extended with a level of preference represented by a real number between 0 and 1, or probabilistic CSPs [FL93], where the probability to be in the real problem is assigned to each constraint. Some other examples are partial CSPs [FW92] or valued CSPs [SFV95] where a preference is assigned to each constraint, in order to also solve over constrained problems. We choose to use one of the most general frameworks to deal with soft constraints [BMR95, BMR97]. The framework is based on a semiring structure that is equipped with the operations needed to combine the constraints present in the problem and to choose the best solutions. According to the choice of the semiring, this framework is able to model all the specific soft constraint notions mentioned above. The semiring-based soft constraint framework provides a structure capable of representing in a compact way problems with preferences.

For handling uncertainty we consider possibility theory, which is a mathematical theory for dealing with a certain type of uncertainty. This theory is an alternative to probability theory. It can be seen as an imprecise probability theory. Possibility theory has been introduced as an extension of the theory of fuzzy sets and fuzzy logic in [Zad78] and many contributions to its development have been presented, for example, in [DFP96a, DP98, DP88].

Another issue that we consider in this thesis to obtain is the representation of bipolarity. Bipolarity is an important focus of research in several domains, e.g. psychology [TK92, SFPM02, OST57, CGB97], multi-criteria decision making [GL05], and more recently in AI (argumentation [ABP05] and qualitative reasoning [BDKP02, BDKP06, DF05, DF06]). Preferences on a set of possible choices are often expressed in two forms: positive and negative statements. In fact, in many real-life situations agents express what they like and what they dislike, thus often preferences are bipolar. Starting from this observation, we define a formalism for handling quantitative (unconditional) preferences, which is able to represent positive and negative statements, and also to deal with uncertainty. Starting from an existing formalism for handling negative preferences, i.e., soft constraints [BMR97], we extend it to handle also positive preferences. The aim is to handle bipolar preferences in a way which is as similar as possible to what naturally happens in real-life scenarios. That is, combining two negative statements should be even worse, combining two positive statements should be even better, and combining a positive with a negative statement should be positive if the positive statement is stronger than the negative one, and negative otherwise. Moreover, we want to able to express indifference, i.e., a preference which is neither positive nor negative.

In many situations, we need to represent and reason about simultaneous preferences of several agents. To aggregate agents' preferences, which in general express a partial order over the possible outcomes, we can query each agent in turn and collect together the results. Hence, we can see preference aggregation in terms of voting, which is a topic widely studied in Operations Research [Vin82a, Vin82b, Bou92a, Bou92b, Bar82]. In this context, we study classical properties such as fairness and non-manipulability, and we consider classical results on fairness of social welfare functions as Arrow's impossibility theorem [Arr51, Kel78] and Sen's possibility theorem [Sen70], and results on non-manipulability of social choice functions as Gibbard-Satterthwaite's theorem [Gib73]. The main difference is that, in contrast to what is assumed in social welfare scenarios, our agents describe their preference using partial orders and not total orders, i.e., they can consider incomparable pairs of outcomes, which are too dissimilar to be compared. We study if results similar to the ones of social choice and social welfare settings still hold, by suitably adapting some of their assumptions to deal with incomparability.

Finally, we consider uncertainty in a multi-agent scenario. We consider a multi-agent setting where agents can hide some of their preferences [KL05]. In a preference ordering, the relationship between some pairs of outcomes may not be specified. For example, agents may have privacy concerns about revealing their complete preference ordering or, as in the context of preference elicitation, preferences have not been fully elicited [CS02b]. In this context it is interesting to determine the complexity of computing the outcomes which are always optimal, or optimal in at least one way in which incompleteness is resolved. Moreover, if this computation is difficult, it is useful to find cases where this computation is easy. Regarding this topic, we investigate such complexity results both in general and for specific preference aggregation systems, and we analyze the issue of manipulation in this context [BTT95].

1.3 Main results

In this thesis we have followed the research lines outlined in the previous section, and we have obtained the following main results.

We have started by considering a special case of quantitative preferences, i.e., fuzzy preferences, and we have considered an existing technique to integrate such preferences with uncertainty, which uses possibility theory [DFP96a]. We have shown that the integration provided by this technique is too tight since the resulting ordering over solutions does not allow one to discriminate between solutions which are highly preferred but assume unlikely events and solutions which are not preferred but robust with respect to uncertainty. Thus, while following the same basic idea of translating uncertainty into fuzzy constraints, we have proposed an algorithm which allows us to observe separately the preference and the robustness of the solutions. Moreover, we have defined suitable semantics for ordering the solutions in a more or less risky way with respect to uncertainty. Then, for finding optimal solutions according to the different semantics, we have developed a solver which exploits branch and bound techniques. Moreover, we have defined a more general formalism for handling different kinds of quantitative preference, proving that some desirable properties continue to hold. This has allowed us to handle the coexistence of preferences and uncertainty in a more general setting.

We have also defined a formalism to handle positive and negative preferences, which reflects the natural behaviour that the combination of positive and negative statements has in real-life scenarios. For doing so, we have first shown that the negative preferences are handled by the semiring-based formalism of soft constraints [BMR97]. Then, we have introduced a new algebraic structure for handling positive preferences, which has properties similar to semirings. Hence, we have defined a new mathematical structure for handling both positive and negative preferences by linking the positive and the negative structures in a suitable way, so that combination of positive preferences produces a better positive preference, the combination of negative preferences produces a worse negative preference, and the combination of positive and negative preferences produces a preference which is better than or equal to the negative preference and worse than or equal to the positive one. We have studied the properties of this formalism and we have defined a solver to solve such problems. Moreover, we have generalized this solver to handle also uncertainty.

We have then considered scenarios where several agents express their preferences via a partial order over the possible outcomes. We have seen each agent as voting if an outcome dominates another one. Thus, we have considered preference aggregation in terms of voting, analyzing some of the main results concerning fairness and non-manipulability [Arr51, Kel87, MS77, Gib73]. We have shown that they can be generalized to preference aggregation systems, where agents can express also incomparability between pair of outcomes.

We have finally considered scenarios where agents, for example for privacy reasons, decide to hide some of their preferences [KL05]. We have determined the computational complexity of computing optimal outcomes, where optimality has the meaning of being always the best outcome (regardless of how incompleteness is resolved), or in at least one possible complete world. We have shown that computing such outcomes is in general difficult, and we have determined cases where such a problem is tractable. Moreover, we have shown how the computation of such outcomes can be useful for deciding when preference elicitation is over, which is in general a difficult problem [CS02b]. Finally, we have investigated other tractability and intractability results for a specific voting rule, i.e, the sequential majority voting. Such a rule performs a sequence of pairwise comparisons between two candidates along a binary tree, and the winner depends on the chosen sequence. We have focused on candidates that that will win in some sequences or in all sequences and we have shown that in general it is easy to find them, while it is difficult if we insist that the tree is balanced. We have interpreted this difficulty in terms of difficulty for the chair to manipulate [BTT95].

1.4 Related research areas

The topics investigated in this thesis are connected to many research areas. The main ones are Artificial Intelligence, Operations Research and Economics areas.

In particular, the topics described in Chapters 2 and 3 can be seen in fields of Artificial Intelligence regarding knowledge representation and reasoning, constraint programming and constraint propagation. Whereas, the topics studied in Chapters 4 and 5 are typically investigated in Artificial Intelligence by researchers interested in multi-agent systems, and in Operations Research and Economics by researchers working on decision making and on voting theory.

1.5 Publications

The work presented in this thesis has been developed also via a collaboration with several colleagues and part of it has been published in the proceedings of international conferences, as we describe below.

The work on preferences and uncertainty has been developed in collaboration with Francesca Rossi and K. Brent Venable from the University of Padova. The following papers have been published on this subject:

- M. S. Pini, F. Rossi and K. B. Venable. Reasoning about fuzzy preferences and uncertainty. *In Proceedings of the 6th International Workshop on Soft Constraints and Preferences*, held in conjunction with the 10th International Conference on Principles and Practice of Constraint Programming (CP 2004), Toronto, Canada, October 2004.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *In Proceedings of the 10th Annual Workshop of ERCIM/CoLogNet on Constraint Solving and Constraint Logic Programming (CSCLP 2005)*, Uppsala, Sweden, June 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Possibility theory for reasoning about uncertain soft constraints. *In Proceedings of the 8th European Conference on Symbolic and*

Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005), Springer-Verlag LNAI 3571, pp. 800-811, Barcelona, Spain, July 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Possibilistic and probabilistic uncertainty in soft constraints problems. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling* held in conjunction of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005), Edinburgh, Scotland, July 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *Doctoral Paper in Proceedings of 11th International Conference of Principles and Practice of Constraint Programming (CP 2005)*, Springer-Verlag LNCS 3709, p. 865, Sitges, Spain, October 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *In Proceedings of International Conference on Intelligent Agents, Web Technology and Internet Commerce (IAWTIC 2005)*, IEEE Computer Society, pp. 583-589, Wien, Austria, November 2005.

The research on bipolar preferences is in collaboration with Francesca Rossi and K. Brent Venable from the University of Padova, Stefano Bistarelli from University of Pescara, and Henri Prade from IRIT (Toulouse), France. The following papers have been published on this subject:

- S. Bistarelli, M. S. Pini, F. Rossi and K. B. Venable. Positive and negative preferences. *In Proceedings of the 7th International Workshop on Preferences and Soft Constraints*, held in conjunction with the 11th International Conference on Principles and Practice of Constraint Programming (CP 2005), Sitges, Spain, October 2005.

- S. Bistarelli, M. S. Pini, F. Rossi and K. B. Venable. Modelling and solving bipolar preference problems. *In Proceedings of 11th Annual ERCIM Workshop on Constraint Solving and Constraint Logic Programming (CSCLP 2006)*, Lisbon, Portugal, June 2006.

- M. S. Pini, F. Rossi and K. B. Venable. Bipolar preference problems. *In Proceedings of the 17th European Conference on Artificial Intelligence (ECAI 2006)*, IOS Press, vol.141, pp. 705-706, Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in bipolar preference problems. *In Proceedings of the 8th International Workshop on Preferences and Soft Constraints*, held in conjunction with the 12th International Conference on Principles and Practice of Constraint Programming (CP 2006), Nantes, France, September 2006.

- M. S. Pini and F. Rossi. Reasoning on bipolar preference problems. *In Proceedings of the CP 2006 Doctoral Programme*, Nantes, France, September 2006.

The work on preference aggregation is a joint research with Francesca Rossi and K. Brent Venable from University of Padova, Toby Walsh from NICTA, Australia, and Jerome Lang from IRIT (Toulouse), France. Our joint publications are:

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Aggregating partially ordered preferences: possibility and impossibility results. *In Proceedings of 10th Conference on Theoretical Aspects of Rationality and Knowledge (TARK X)*, ACM Digital Library, National University of Singapore, pp. 193-206, Singapore, June 2005.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Strategic voting when aggregating partially ordered preferences. *In Proceedings of the 5th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2006)*, ACM Press, pp. 685-687, Hakodate, Japan, May 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Computing possible and necessary winners from incomplete partially-ordered preferences. *In Proceedings of the 17th European Conference on Artificial Intelligence (ECAI 2006), Best poster Award*, IOS Press, vol.141, pp. 767-768, Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Incompleteness and incomparability in preference aggregation. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, held in conjunction of the 17th European Conference on Artificial Intelligence (ECAI 2006), Riva del Garda, Italy, August 2006.

- J. Lang, M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Winner determination in sequential majority voting with incomplete preferences. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, held in conjunction of the 17th European Conference on Artificial Intelligence (ECAI 2006), Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Incompleteness and incomparability in preference aggregation: complexity results. *In Proceedings of the 8th International*

Workshop on Preferences and Soft Constraints, held in conjunction with the 12th International Conference on Principles and Practice of Constraint Programming (CP 2006), Nantes, France, September 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Preference aggregation and elicitation: tractability in the presence of incompleteness and incomparability. *In Proceedings of DIMACS/LAMSADE Workshop on Computer Science and Decision Theory II*, Paris, France, October 2006.

- J. Lang, M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Winner determination in sequential majority voting. *In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, to appear, Hyderabad, India, January 2007.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Incompleteness and incomparability in preference aggregation. *In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, to appear, Hyderabad, India, January 2007.

1.6 Structure of the thesis

The thesis is organized as follows:

- **Chapter 2**. We consider an existing technique to perform integration between fuzzy preferences and uncertainty and, while following the same basic idea, we propose various alternative semantics which allow us to observe both the preference level and the robustness with respect to uncertainty of the complete instantiations. Then we present a solver for this kind of problems that is based on branch and bound. Finally, we extend this approach to other classes of soft constraints proving that certain desirable properties still hold.

- **Chapter 3**. We focus on problems with both positive and negative preferences, that we call bipolar problems. We show that the soft constraints formalism models only negative preferences, and we define a new mathematical structure which allows to handle positive preferences as well. We address the issue of the compensation between positive and negative preferences, studying the properties of this operation. Then, we extend the notion of arc consistency to bipolar problems, and we show how branch and bound (with or without constraint propagation) can be easily adapted to solve such problems. Finally, we focus on bipolar problems with uncertainty, where some variables are uncontrollable, by extending existing techniques to handle bipolar problems and problems with uncertainty.

- **Chapter 4**. We consider ways of reasoning and aggregating preferences in order to choose outcomes that satisfy all the agents. We adapt the most popular aggregating criteria of social welfare and social choice theory [Kel87] to our context, studying their induced semantics and complexity. Finally, we push even further the bridge between social choice theory and aggregation of preferences obtained using AI representations, by considering the fairness [Kel87, MS77] and the non-manipulability [Gib73, Sat75] of the voting schemes we propose. In particular, we extend Muller-Satterthwaite's theorem [MS77], which is the Arrow's theorem [Arr51] in social choice theory, and Gibbard-Satterthwaite's theorem [Gib73, Sat75] to the situation in which the ordering of each agent is a partial order.

- **Chapter 5**. We consider how to combine the preferences of multiple agents despite the presence of incompleteness and incomparability in their preference orderings, by focusing on the problem of computing the possible and necessary winners, that is, those outcomes which can be or always are the most preferred for the agents. First we show that computing the sets of possible and necessary winners is in general a difficult problem as it is providing a good approximation of such sets. Then we identify sufficient conditions, related to general properties of the preference aggregation function, where such sets can be computed in polynomial time. Next, we show how possible and necessary winners can be used to focus preference elicitation. Then, we consider a specific voting rule which performs a sequence of pairwise comparisons between two candidates along a binary tree, where the winner depends on the chosen sequence. Also in this case there are candidates that will win in some sequences (called possible winners) or in all sequences (called Condorcet winners). While it is easy to find the possible and Condorcet winners, we prove that it is difficult if we insist that the tree is balanced. Finally we consider the situation where we lack complete informations about preferences, and we determine the computational complexity of computing possible and Condorcet winners in this extended case.

- **Chapter 6**. We summarize the results of the thesis, and we discuss directions for further work.

Chapter 2

Preferences and uncertainty

Preferences and uncertainty occur in any real-life problems. The theory of possibility is one non-probabilistic way of dealing with uncertainty, which allows for easy integration with fuzzy preferences. In this chapter we consider an existing technique to perform such an integration and, while following the same basic idea, we propose various alternative semantics which allow to observe both the preference and the robustness of a solution with respect to uncertainty. Then we present a solver for this kind of problems that allows a branch and bound approach. Finally, we extend this technique to other classes of soft constraints, proving that certain desirable properties still hold.

2.1 Motivations and chapter structure

Preferences and uncertainty occur in many real-life problems. We are concerned with the coexistence of such concepts in the same problem. In particular, we consider uncertainty that comes from lack of data or imprecise knowledge and scenarios where probabilistic estimates are not available.

The theory of possibility [DP88, Zad78] is one non-probabilistic way of dealing with uncertainty, which allows for easy integration with fuzzy preferences [DFP96a]. In fact, both possibilities and fuzzy preferences are values between 0 and 1 associated to events and express the level of plausibility that the event will occur, or its preference.

In our context, we will describe a real-life problem as a soft constraint problem, that is represented by a set of variables with finite domains and a set of soft constraints among subsets of the variables. A variable will be said to be uncertain if we cannot decide its value. In this case, we will associate a possibility degree to each value in its domain, which will tell how plausible it is that the variable will get that value.

Soft constraints allow to express preferences over the instantiations of the variables of the constraints. In particular, fuzzy preferences are values between 0 and 1, which are combined using the min operator, and are ordered in such a way that higher values denote better preferences.

In this chapter we consider an existing technique to integrate fuzzy preferences and uncertainty, which uses possibility theory [DFP96a]. This technique allows one to handle uncertainty within a fuzzy optimization engine. However, we claim that the integration provided by this technique is too tight since the resulting ordering over complete assignments does not allow one to discriminate between solutions which are highly preferred but assume unlikely events and solutions which are not preferred but robust with respect to uncertainty. This is due to the fact that a single value, which summarizes the contributions of both the uncertain variables and the fuzzy preferences, is associated to each solution.

While following the same basic idea of translating uncertainty into fuzzy constraints, we propose various alternative semantics which allow us to observe separately the preference level and the robustness of the complete instantiations. More precisely, each solution will be associated to a pair of values between 0 and 1: one value will refer to the preference level, while the other one will refer to the robustness of the solution with respect to the uncertain variables. In this way, given a solution and the pair of values associated to it, we can see how preferred it is according to the constraints, and also how robust it is.

The desired ordering over such pairs will then be used to order the solutions. Thus, by choosing different orderings, we can reason in a more or less risky way with respect to uncertainty, giving more or less importance to the preferences with respect to the robustness of the problem. In this way, we define a class of different semantics.

We then develop a solver, that can handle fuzzy problems and uncertainty expressed via possibility distributions, which is based on branch and bound techniques and which uses the various semantics for ordering the solutions.

Finally, we prove that the desired properties that the semantics have, hold also when other classes of soft constraints, not necessarily fuzzy, are used. This allows us to handle the coexistence of preferences and uncertainty in a more general setting.

The work presented in this chapter has appeared in the proceedings of the following conferences and workshops.

- M. S. Pini, F. Rossi and K. B. Venable. Reasoning about fuzzy preferences and uncertainty. *In Proceedings of the 6th International Workshop on Soft Constraints and Preferences*, held in conjunction with the 10th International Conference on Principles and Practice of Constraint Programming (CP 2004), Toronto, Canada, October 2004.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *In Proceedings of the 10th Joint Annual Workshop of ERCIM/CoLogNet on Constraint Solving and Constraint Logic Programming (CSCLP 2005)*, Uppsala, Sweden, June 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Possibility theory for reasoning about uncertain soft constraints. *In Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, Springer-Verlag LNAI 3571, pp. 800-811, Barcelona, Spain, July 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Possibilistic and probabilistic uncertainty in soft constraints problems. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling* held in conjunction of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005), Edinburgh, Scotland, July 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *Doctoral Paper in Proceedings of 11th International Conference of Principles and Practice of Constraint Programming (CP 2005)*, Springer-Verlag LNCS 3709, p. 865, Sitges, Spain, October 2005.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in soft constraints problems. *In Proceedings of International Conference on Intelligent Agents, Web Technology and Internet Commerce (IAWTIC 2005)*, IEEE Computer Society, pp. 583-589, Wien, Austria, November 2005.

The chapter is organized as follows.

- In Section 2.2 we present the background on which our work is based. First, we describe soft constraints by focusing on fuzzy constraints and we give the main notions of possibility theory. Next, we define uncertain fuzzy CSPs and we show an existing method (DFP) for handling uncertain fuzzy CSPs by integrating fuzzy preferences and uncertainty via possibility theory.

- In Section 2.3 we outline the fundamental aspects characterizing an uncertain fuzzy CSP, related to the satisfaction and the robustness, and we present some reasonable properties to require over the solution ordering.

- In Sections 2.4 and 2.5 we present a new method for handling fuzzy CSPs with uncertainty, that allows us to observe separately the satisfaction and the robustness of solutions with respect to uncertainty, and in Section 2.6, we check if the robustness satisfies the desired properties.

- In Section 2.7 we propose various semantics for ordering the solutions with respect to their satisfaction and robustness.
- In Section 2.8 and 2.9 we check if these semantics and the semantics produced by DFP satisfy the desired properties on solution ordering, and in Section 2.10 we compare the solution ordering that they induce.
- In Section 2.11 we show how a real-life problem can be modelled as an uncertain fuzzy CSP and solved by using our procedure.
- In Section 2.12 we present a solver for finding an optimal solution of an uncertain fuzzy CSP according to our semantics.
- In Section 2.13 we show how to generalize the procedure described in Sections 2.4 and 2.5, for integrating soft preferences not necessarily fuzzy and uncertainty, by preserving the desired properties that the semantics should have.
- In Section 2.14 and Section 2.15 we describe respectively related and future work.

2.2 Background

In this section we give an overview of the background on which our work is based. First, we present a formalism for representing soft preferences, i.e., the semiring-based soft constraints [BMR97]. Next, we describe the formalism we will use for representing uncertainty, i.e., possibility theory [Zad78]. Finally, we present an existing method for integrating fuzzy preferences and possibilistic uncertainty [DFP96a].

2.2.1 Soft constraints

In this section we will present *soft constraints*: a formalism which allows to handle different kinds of preferences. In the literature there are many formalizations of the concept of soft constraints [SFV95, Rut94]. Here we refer to the one described in [BMR97, BMR95], which however can be shown to generalize and express many of the others [BFM$^+$96].

In a few words, a soft constraint is just a classical constraint where each instantiation of its variables has an associated element (also called a preference) from a partially ordered set.

Combining constraints will then have to take into account such additional elements, and thus the formalism has also to provide suitable operations for combination ($\times$) and comparison (+) of tuples of preferences and constraints. This approach is based on the concept of c-semiring, which is just a set plus two operations.

Definition 1 (semirings and c-semirings) A semiring is a tuple $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ such that:

- A is a set and $\mathbf{0}, \mathbf{1} \in A$;
- $+$ is commutative, associative and $\mathbf{0}$ is its unit element;
- $\times$ is associative, distributes over $+$, $\mathbf{1}$ is its unit element and $\mathbf{0}$ is its absorbing element.

A c-semiring is a semiring $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ such that:

- $+$ is defined over possibly infinite sets of elements of A in the following way:
  - $\forall a \in A$, $\sum(\{a\}) = a$;
  - $\sum(\emptyset) = \mathbf{0}$ and $\sum(A) = \mathbf{1}$;
  - $\sum(\bigcup A_i, i \in S) = \sum(\{\sum(A_i), i \in S\})$ for all sets of indexes S (flattening property);
- $\times$ is commutative.

Let us consider the relation $\leq_S$ over A such that $a \leq_S b$ if and only if $a + b = b$. Then it is possible to prove that (see [BMR95]):

- $\leq_S$ is a partial order;
- $+$ 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, for all $a, b \in A$, $a + b = lub(a, b)$, where lub means least upper bound.

Moreover, if $\times$ is idempotent, then $\langle A, \leq_S \rangle$ is a complete distributive lattice and $\times$ is its greatest lower bound (glb). Informally, the relation $\leq_S$ gives us a way to compare (some of the) tuples of preferences and constraints. In fact, when we have $a \leq_S b$, we will say that *b is better than (or preferred to) a*.

Definition 2 (soft constraints) Given a c-semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$, a finite set D (the domain of the variables), and an ordered set of variables V, a constraint is a pair $\langle def, con \rangle$ where $con \subseteq V$ and $def : D^{|con|} \rightarrow A$.

Therefore, a constraint specifies a set of variables (the ones in con), and assigns to each tuple of values in D of these variables an element of the semiring set A. This element can be interpreted in many ways: as a level of preference, or as a cost, or as a probability, etc. The correct way to interpret such elements determines the choice of the semiring operations.

Definition 3 (SCSP) A soft constraint satisfaction problem is a set of soft constraints C defined over a set of variables V.

Definition 4 (combination) Given two constraints $c_1 = \langle def_1, con_1 \rangle$ and $c_2 = \langle def_2, con_2 \rangle$, their *combination* $c_1 \otimes c_2$ is the constraint $\langle def, con \rangle$, where $con = con_1 \cup con_2$ and $def(t) = def_1(t \downarrow^{con}_{con_1}) \times def_2(t \downarrow^{con}_{con_2})$[1].

The combination operator $\otimes$ can be straightforwardly extended also to finite sets of constraints: when applied to a finite set of constraints C, we will write $\bigotimes C$.

In words, combining constraints means building a new constraint involving all the variables of the original ones, and which associates to 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 subtuples.

Definition 5 (projection) Given a constraint $c = \langle def, con \rangle$ and a subset I of V, the *projection* of c over I, written $c \Downarrow_I$, is the constraint $\langle def', con' \rangle$ where $con' = con \cap I$ and $def'(t') = \sum_{t / t \downarrow^{con}_{I \cap con} = t'} def(t)$.

Informally, projecting means eliminating some variables. This is done by associating to each tuple over the remaining variables a semiring element which is the sum of the elements associated by the original constraint to all the extensions of this tuple over the eliminated variables.

Definition 6 (local consistency) The degree of local consistency of a partial assignment, $d = (d_1, \ldots, d_k)$, is $\Pi_{\{c_i = \langle def_i, con_i \rangle | con_i \subseteq \{x_1, \ldots, x_k\}\}} \, def_i \, (d \downarrow_{con_i})$

Definition 7 (solution) A solution of a SCSP $\langle C, V \rangle$ is a complete instantiation, $(d_1, \ldots, d_n)$, of the variables in $V = \{x_1, \ldots x_n\}$.

[1] By $t \downarrow^X_Y$ we mean the projection of tuple t, which is defined over the set of variables X, over the subset of variables $Y \subseteq X$.

Definition 8 (solution preference) Given SCSP $\langle C, V \rangle$ and a solution s, the preference of s is $pref(s) = \Pi_{\{c_i = \langle def_i, con_i \rangle \in C\}} def_i(s \downarrow_{con_i})$,

Definition 9 (optimal solution) Given a SCSP P and a solution s, s is optimal if and only if $\nexists\ s'$, solution of P, such that $pref(s') >_S pref(s)$.

SCSPs can be solved by extending and adapting the techniques usually used for classical CSPs. For example, to find the best solution, we could employ a branch-and-bound search algorithm (instead of the classical backtracking). Also the so-called constraint propagation techniques, like arc-consistency [Mac77] and path-consistency, can be generalized to SCSPs [BMR95, BMR97].

Instances of semiring based SCSPs

We will now give an overview of the most common instances of the semiring-based framework [BMR97].

- **Classical SCSPs**. The semiring is

  $$S_{CSP} = \langle \{false, true\}, \vee, \wedge, false, true \rangle.$$

  The only two preferences that can be given are $true$, indicating that the tuple is allowed and $false$, indicating that the tuple is forbidden. Preferences are combined using logical and and compared using logical or. Optimization criterion: any assignment that has preference $true$ on all constraints is optimal.

- **Fuzzy SCSPs** [BMR97, DFP96a]. The semiring is

  $$S_{FCSP} = \langle [0, 1], max, min, 0, 1 \rangle.$$

  Preferences are values between 0 and 1. They are combined using min and compared using max. Optimization criterion: maximize minimal preference.

- **Weighted SCSPs** [BMR97]. The semiring is

  $$S_{WCSP} = \langle \Re^+, min, +, +\infty, 0 \rangle.$$

  Preferences are interpreted as costs from 0 to $+\infty$. Costs are combined with +, and compared with min. Optimization criterion: minimize sum of costs.

- **Probabilistic SCSPs** [FLS96]. The semiring is

$$S_{PCSP} = \langle [0,1], max, \times, 0, 1 \rangle.$$

  Preferences are interpreted as probabilities ranging from 0 to 1. As expected, they are combined using $\times$, and compared using max. Optimization criterion: maximize joint probability.

Example 1 Figure 2.1 shows an example of one of the instances presented above: the fuzzy CSP. Variables are within circles, and constraints are undirected links among the variables. Each constraint is defined by associating a preference level (in this case between 0 and 1) to each assignment of its variables to values in their domains. Figure 2.1 shows also two solutions, one of which, S_2, is optimal. □

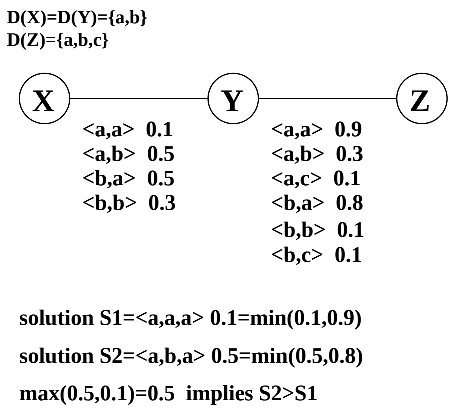

Figure 2.1: A Fuzzy CSP and two of its solutions, one of which is optimal (S_2).

2.2.2 Possibility theory

Possibility theory was introduced in [Zad78], in connection with the fuzzy set theory [Zad78, DP80, DP00], to allow reasoning to be carried out on imprecise or vague knowledge, making it possible to deal with uncertainties on this knowledge.

This theory and its developments constitute a method of formalizing non-probabilistic uncertainties on events, i.e., a way of assessing to what extent the occurrence of an event is possible and to what extent we are certain of its occurrence, without, however, knowing the evaluation of the probability of this occurrence. This can happen, for instance, when there is no similar event to be referred to.

Possibility theory, represents the uncertainty on the occurrence of an event in the form of possibility distributions. In what follows we will consider events represented by an uncontrollable variable taking a value from a particular subset.

A possibility distribution π_x associated to an uncontrollable variable x represents the set of more or less plausible, mutually exclusive values of x.

Definition 10 (possibility distribution) A possibility distribution π_x associated to a single valued variable x with domain D is a mapping from D to a totally ordered scale L (usually $[0, 1]$) such that $\forall d \in D$, $\pi_x(d) \in L$ and $\exists\ d \in D$ such that $\pi_x(d) = 1$, where 1 the top element of the scale L.

The following conventions hold: $\pi_x(d) = 0$ means $x = d$ is impossible; $\pi_x(d) = 1$ means $x = d$ is fully possible, unsurprising.

A possibility distribution is similar to a probability density. However, $\pi_x(d) = 1$ only means that $x = d$ is a plausible situation, which cannot be excluded. Thus, a degree of possibility can be viewed as an upper bound of a degree of probability. Possibility theory encodes incomplete knowledge while probability accounts for random and observed phenomena. In particular, the possibility distribution π_x can encode:

- *complete ignorance* about x: $\pi_x(d) = 1, \forall d \in D$. In this case all values $d \in D$ are plausible for x and so it is impossible to exclude any of them.

- *complete knowledge* about x: $\pi_x(\bar{d}) = 1, \exists \bar{d} \in D$ and $\pi_x(d) = 0, \forall d \in D$ s.t. $d \neq \bar{d}$. In this case only the value $\bar{d}$ is plausible for x.

Given a possibility distribution π_x associated to a variable x, the occurrence of the event $x \in E \subseteq D$ can be defined by the possibility and the necessity degrees.

Definition 11 (possibility degree) The possibility degree of an event "$x \in E$", denoted by $\Pi(x \in E)$ or simply by $\Pi(E)$, is $\Pi(x \in E) = sup_{d \in E} \pi_x(d)$.

The possibility degree of the event "$x \in E$' evaluates the extent to which "$x \in E$" is *possibly* true.

In particular,

- $\Pi(x \in E) = 1$ means that the event $x \in E$ is totally possible. However it could also not happen. Therefore in this case we are completely ignorant about its occurrence.

- $\Pi(x \in E) = 0$ means that the event $x \in E$ for sure will not happen.

Definition 12 (necessity degree) The necessity degree of "$x \in E$", denoted by $N(x \in E)$ or simply by $N(E)$, is $N(x \in E) = inf_{d \notin E} c(\pi_x(d))$, where c is the order reversing map in the interval $[0, 1]$ such that $\forall p \in [0, 1]$, $c(p) = 1 - p$ and E^C is the complement of E in D.

The necessity degree of the event "$x \in E$" evaluates the extent to which "$x \in E$" is *certainly* true.

In particular,

- $N(x \in E) = 1$ means that the event $x \in E$ is certain,
- $N(x \in E) = 0$ means that the event is not necessary at all, although it may happen. In fact, $N(x \in E) = 0$ if and only if $P(x \in E^C) = 1$.

The possibility and the necessity measures are related by the following formula $\Pi(E) = 1 - N(E^C)$. From this, follows $N(E) = 1 - \Pi(E^C)$.

In the following example we will compute the possibility and the necessity degrees that a variable, defined by a certain possibility distribution, belongs to a given set.

Example 2 Assume that x is an uncontrollable variable with domain $D = \{5, 6, 7, 8\}$, π_x is possibility distribution attached to x, defined by $\pi_x(5) = 0.9$, $\pi_x(6) = 0.4$, $\pi_x(7) = 0.7$, $\pi_x(8) = 0.5$ and $E = \{5, 6\}$ is a subset of D. Then the possibility degree of the event "$x \in E$" is $\Pi(E) = sup_{d \in E} \pi(d) = sup\{0.9, 0.4\} = 0.9$, whereas the necessity degree of the same event is $N(E) = inf_{d \notin E}\ c(\pi(d)) = inf\{c(\pi(7)), c(\pi(8))\} = inf\{c(0.7), c(0.5)\} = inf\{0.3, 0.5\} = 0.3$. Notice that if we compute $N(E)$ using the formula $N(E) = 1 - \Pi(E^C)$ we obtain the same result. In fact, $N(E) = 1 - \Pi(E^C) = 1 - sup_{d \in E^C} \pi(d) = 1 - sup\{0.7, 0.5\} = 1 - 0.7 = 0.3$. $\square$

2.2.3 Uncertainty in soft constraints

Whereas in usual soft constraint problems all the variables are assumed to be controllable, that is, their value can be decided according to the constraints which relate them to other variables, in many real-world problems uncertain parameters must be used. Such parameters are associated with variables which are not under the user's direct control and thus cannot be assigned. Only Nature will assign them.

Some real-life examples of uncertain soft constraint problems are:

- the problem of scheduling, where the duration of a task is uncertain, for example the task T_i has a duration of approximatively ten minutes [DFP95];

- the problem of deciding how many training sessions to perform in a tutorial, knowing that number of actual participants who will attend it is from 1 to 20 and more possibly from 10 to 15 [DFP96a];
- the problem of deciding how many buses a school has to rent for an excursion knowing that the number of the interested students is form 60 to 150 and more possibly between 90 to 120.

In [DFP96a] these problems are formalized as a set of variables, that can be controllable and uncontrollable and a set of fuzzy constraints linking these variables.

Example 3 Figure 2.2 shows an example of an uncertain fuzzy CSP. Each constraint is defined by associating a preference level (in this case between 0 and 1) to each assignment of its variables to values in their domains. In particular, the constraint $C_{xyz} = \langle \mu, \{x, y, z\} \rangle$ is defined on variables x, y and z by the preference function μ, whereas the constraint $C_{xw} = \langle \mu_1, \{x, w\} \rangle$ is defined on the variables x and w by the preference function μ_1. Variables are controllable (x and w) and uncontrollable (z). The values in the domain of the uncontrollable variable z are characterized by the possibility distribution π_Z. Constraints link controllable variables and controllable with uncontrollable variables. □

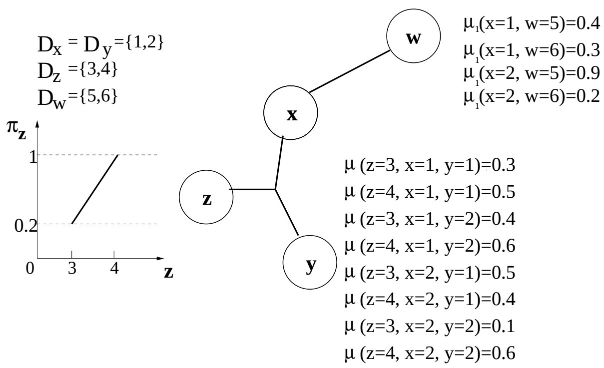

Figure 2.2: An uncertain fuzzy CSP.

In general, an uncertain soft CSP can be formally defined as follows.

Definition 13 (USCSP) An uncertain soft constraint satisfaction problem (USCSP) is a a tuple $\langle S, V_c, V_u, C \rangle$, where S is a c-semiring, V_c is a set of controllable variables, V_u is a set of uncontrollable variables, and C is a set of soft constraints involving any subset of variables of $V_c \cup V_u$.

While in a classical soft constraint problem we can decide how to assign the variables to make the assignment optimal, in the presence of uncertain parameters we must assign values to the controllable variables guessing what Nature will do with the uncontrollable variables. A solution in USCSP is an assignment to all its controllable variables. Depending on the assumptions made on the observability of the uncontrollable variables, different optimality criteria can be defined.

For example, an optimal solution for an USCSP can be defined as an assignment of values to the variables in V_c such that, *whatever* Nature will decide for the variables in V_u, the overall assignment will be optimal. This corresponds to assume that the values of the uncontrollable variables are never observable, i.e., that the values of the controllable variables are decided upon without observing the values of the uncontrollable variables. This is a pessimistic view, and, often, an assignment satisfying such a requirement does not exist.

In such a case, one can relax the optimality condition to that of having a preference above a certain threshold α in all scenarios. In this case solving the problem will consist of finding the assignments to variables in V_c which satisfy this property at the highest α.

Furthermore, one could be satisfied with finding an assignment of values to the variables in V_c such that, *for at least* one assignment decided by Nature for the variables in V_u, the overall assignment will be optimal. This definition follows an optimistic view. Other definitions can be between these two extremes.

Moreover, the uncontrollable variables can be equipped with additional information on the likelihood of their values. Such information can be given in several ways, depending on the amount and precision of knowledge we have. In this chapter for expressing such information we will consider possibility distributions. This information can be used to infer new soft constraints over the controllable variables, expressing the compatibility of the controllable parts of the problem with the uncertain parameters, and can be used to change the notion of optimal solution.

In this chapter we will consider the approach of guaranteeing a certain preference level α taking into account the additional information on the uncontrollable variables provided in the form of possibility distribution.

2.2.4 Unifying fuzzy preferences and uncertainty via possibility theory

Possibility theory [Zad78] can be used to code some information about the uncontrollable variables in uncertain soft constraint problems. The method presented in [DFP96a], which we call algorithm DFP (by the name of the authors), for managing uncertainty in fuzzy

CSPs, proposes to translate uncertain fuzzy CSPs into fuzzy CSPs and then to solve them as known in literature [DP80, BMR97]. More precisely, DFP takes in input an uncertain fuzzy CSP, say Q. Then Q is reduced to a fuzzy CSP Q', where every constraint C which links uncontrollable variables to controllable variables is replaced by a new fuzzy constraint C' only among these controllable variables. This happens in the following way.

Consider a fuzzy constraint C, represented by the fuzzy relation R, which relates a set of controllable variables $X = \{x_1, \ldots, x_n\}$, with domains $D_1, \ldots, D_n$, to a set of uncontrollable variables $Z = \{z_1, \ldots, z_k\}$ with domains $A_1, \ldots, A_k$. Assume the knowledge of the uncontrollable variables is modeled with the possibility distribution π_Z defined on $A_Z = A_1 \times \cdots \times A_k$. Assume the preferential information is instead represented by function $\mu_R : D_X \times A_Z \longrightarrow [0, 1]$, where $D_X = D_1 \times \cdots \times D_n$. Value $\mu_R(d, a)$ is the preference associated to the assignment to controllable and uncontrollable variables $(X = d, Z = a) = (x_1 = d_1, \ldots, x_n = d_n, z_1 = a_1, \ldots, z_k = a_k)$.

The constraint C is considered satisfied[2] by assignment $d = (d_1, \ldots, d_n) \in D_1 \times \cdots \times D_n$ if, *whatever the values of* $a = (a_1, \ldots, a_k)$, *these values are compatible*[3] *with* d, i.e., if the set of possible values for z is included in $T = \{a \in A_Z | \mu_R(d, a) > 0\}$. Given assignment $d \in D_X$, and $\mu_T(a) = \mu_R(d, a)$, the preference of d in the new constraint C' obtained from C removing uncontrollable variables is:

$$\mu'(d) = N(d\, satisfies\, C) = N(z \in T) = inf_{a \in A_Z} max(\mu_T(a), c(\pi_Z(a))) \tag{2.1}$$

where c is the order reversing map such that $c(p) = 1 - p$, $\forall p \in [0, 1]$. The value $\mu'(d)$, that is given by the necessity degree of the event "d satisfies C", represents the degree of satisfaction of C. It is characterized by the following property: $\mu'(d) \geq \alpha$ if and only if when $\pi_Z(a) > c(\alpha)$ then $\mu_R(d, a) \geq \alpha$, where a is the actual value of z.

Informally, the new preference level of the assignment d obtained reasoning with uncertainty, $\mu'(d)$, is greater or equal than α if and only if the assignments ($X = d, Z = a$), such that the possibility $\pi_Z(a)$ is strictly greater than $1 - \alpha$, had a preference $\mu_R(d, a)$ greater or equal than α in the original problem.

Notice that in 2.1, μ' is computed by applying the *max* operator between preferences and possibilities. This can be done, since the scales of the preferences and of the possibilities are equal, assuming the *commensurability* between preferences and possibilities.

If the uncontrollable variables $z_1, \ldots, z_k \in Z$ are logically independent from each other, the knowledge about each z_j is completely described by the possibility distributions π_{z_j},

[2]Here "satisfi ed" means "at least partially satisfi ed".

[3]A value a is compatible with d if $\mu_R(d, a) > 0$.

and the joint possibility π_Z is defined as follows: $\forall a = (a_1, \dots, a_k) \in A_1 \times \cdots \times A_k$, $\pi_Z(a) = min_{\{j=1,\dots,k\}} \pi_{Z_j}(a_j)$.

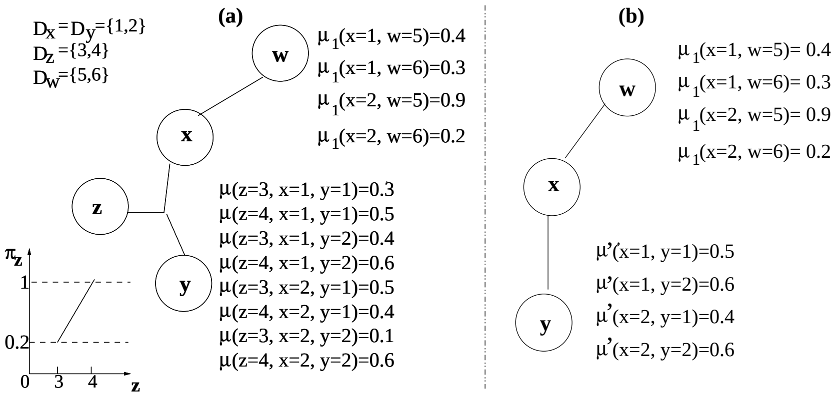

Figure 2.3: An example of application of algorithm DFP.

Example 4 The result of applying algorithm DFP to an uncertain fuzzy CSP is shown in Figure 2.3. Part (a) shows the same uncertain FCSP Q shown in Figure 2.2. Recall that the variables x, y and w are controllable, while z is uncontrollable. Part (b) shows the FCSP Q' defined on variables x, y and w obtained by applying algorithm DFP. □

2.3 Desired features and properties of an USCSP

In this section we outline the fundamental aspects and the desired properties which we believe are crucial in the characterization of an USCSP. In the next section we will present a formalism which will be shown to fulfill all the requirements described here.

For stating formally the desired properties we need to give some definitions. Consider an uncertain soft CSP, $Q = \langle S, V_c, V_u, C \rangle$, where $S = \langle A, +, \times, 0, 1 \rangle$[4] is a c-semiring and $\leq_S$ is the ordered induced by the operator $+$ of S. Consider a solution s of Q, i.e., a complete assignment to the controllable variables. We define the overall preference of a solution as follows.

Definition 14 (overall preference) The overall preference of s, given an assignment a to the uncontrollable variables, is $pref(s, a) = \Pi_{\{\langle \mu, con \rangle \in C\}} \mu((s, a) \downarrow_{con})$.

In words, $pref(s, a)$ is the global preference in the original problem of choosing s for the controllable variables in the scenario that assigns a to the uncontrollables. This preference

[4] when $+$ (respectively $\times$) is applied to a two-element set we will use symbol $+$ (respectively $\times$) in infi x notation, while in general we will use the the symbol $\sum$ (respectively $\prod$) in prefi x notation.

is obtained by combining all the preferences associated to the projections of the tuple (s, a) on all the constraints in C. If Q is a fuzzy uncertain CSP, then $pref(s, a)$ is the minimum of these preferences, i.e., $pref(s, a) = min_{\{\langle \mu, con \rangle \in C\}} \mu((s, a) \downarrow_{con})$.

Given a uncertain SCSP $Q = \langle S, V_c, V_u, C \rangle$, we consider problem $Q_{control} = \langle S, V_c, C_{control} \rangle$, where $C_{control} = C_f \cup C_p$ contains all the constraints of Q defined only on controllable variables, that we call C_f, and all the constraints of Q obtained projecting on the controllable variables the constraints involving also uncontrollable variables, that we call C_p. This can be seen as applying the usual variable elimination technique to uncontrollable variables [Dec03]. We define the satisfaction degree of a solution as follows.

Definition 15 (satisfaction degree) The satisfaction degree of s is $sat(s) = \Pi_{\{<\mu, con> \in C_{control}\}} \mu(s \downarrow_{con})$.

In words, $sat(s)$ is the preference obtained by combining all the preferences associated to the projections of s on all the constraints of $Q_{control}$. If S is the fuzzy c-semiring then $sat(s)$ is the minimum of such preferences, i.e., $sat(s) = min_{\{<\mu, con> \in C_{control}\}} \mu(s \downarrow_{con})$. $sat(s)$ measures how good s is in terms of preferences when the effects of Nature are projected out. In fact, by considering the projections of the constraints involving also variables in V_u, only the best possible choices for the uncontrollable variables are taken into account, allowing to focus only on the preferential aspect of the problem.

Another kind of satisfaction degree, that we call controllable satisfaction degree, can be obtained by simply forgetting all the constraints of the problem involving at least an uncontrollable variable. In particular, given a solution s of an USCSP Q, we can consider the satisfaction degree restricted to the constraints of Q involving only controllable variables, i.e., C_f. More formally,

Definition 16 (controllable satisfaction degree) The controllable satisfaction degree of s is $sat_c(s) = \Pi_{\{<\mu, con> \in C_f\}} \mu(s \downarrow_{con})$.

When we deal with uncertain SCSPs, we have to consider another interesting aspect that characterizes a solution, that is its robustness with respect to the uncertainty. More formally,

Definition 17 (robustness) The robustness of s, say $rob(s)$, is a value indicating the degree of compatibility of s with uncertain events. In particular, the higher $rob(s)$, the more assignments to uncontrollable variables will yield in Q preferences higher than a given threshold when s is chosen.

Let us call C_{fu} the set of constraints $< \mu, con >$ in C involving controllable and uncontrollable variables, i.e., such that $con \cap V_u \neq \emptyset$ and $con \cap V_c \neq \emptyset$. In general, the robustness depends on the preferences associated to subtuples of (s, a) on the constraints in C_{fu} and on the possibility distributions, $\pi(a)$, defined on the assignments a to the uncontrollables.

In order to characterize robustness we require that it satisfies the following two properties, that refer to the ordering $\leq_S$. We recall that this ordering is the one induced by the additive operator of the c-semiring S.

Property 1 *Given solutions s and s' of an UFCSP $\langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, if for every constraint $< \mu, con > \in C_{fu}$, $\mu((s, a) \downarrow_{con}) \leq_S \mu((s', a) \downarrow_{con})$ for every assignment a to uncontrollables, then it should be that $rob(s) \leq_S rob(s')$.*

In words, if the preferences associated to subtuples of s on the constraints involving controllable and uncontrollables are always greater or equal to that associated to subtuples of s', then it is reasonable that the robustness of s is greater or equal than the robustness of s'.

Property 2 *Given a solution s of an UFCSP $\langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, assume that the uncertainty on the variables in V_u is described by a possibility distribution π_1. Assume also to lower the possibility of every event a to $\pi_2(a) \leq \pi_1(a)$. Then it should be that $rob_{\pi_1}(s) \leq_S rob_{\pi_2}(s)$.*

In other words, if we lower the possibility of any value of uncontrollables, then the solution s should have an higher value of robustness.

We will now describe some properties which we believe should be satisfied by a preferential ordering over the solutions of an USCSP.

Property 3 *Given two solutions s and s' of an UFCSP $\langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, if $pref(s, a) >_S pref(s', a)$ $\forall a$ assignment to V_u, then it should be that $s >_S s'$.*

In other words, a solution should dominate any other solution which has a lower overall preference in every possible scenario a.

Property 4 *Given two solutions s and s' of an UFCSP $\langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, if $rob(s) = rob(s')$ and $sat(s) >_S sat(s')$, it should be that $s >_S s'$.*

Property 5 *Given two solutions s and s' of an UFCSP $\langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, such $sat(s) = sat(s')$, and $rob(s) >_S rob(s')$, then it should be that $s >_S s'$.*

In words, two solutions which are as good with respect to one aspect (robustness or satisfaction degree) and differ on the other should be ordered.

2.4 Algorithm SP

In this section we describe a new algorithm to handle uncertain fuzzy CSPs, that is partially based on algorithm DFP described in Section 2.2.4. We call it algorithm SP (from *separation* and *projection*).

It starts from an uncertain fuzzy CSP $Q = \langle S_{FCSP}, V_c, V_u, C = C_f \cup C_{fu} \rangle$, where C_f is the set of constraints of Q defined only on controllable variables and C_{fu} is the set of constraints of Q defined on both controllable and uncontrollable variables. Then, it obtains a fuzzy CSP $Q' = \langle S_{FCSP}, V_c, C' = C_{control} \cup C_u \rangle$, where:

- $C_{control} = C_f \cup C_p$, where C_p is the set of constraints obtained by projecting the constraints C_{fu} on their controllable variables,
- C_u is the set of constraints, defined only on controllable variables, obtained from the constraints C_{fu} applying the method described in Section 2.2.4.

We will now illustrate with an example how SP works.

Example 5 Let us consider the fuzzy uncertain CSP, say Q, in Figure 2.2, which is shown again in Figure 2.4 (a). Figure 2.2 (b) shows the corresponding fuzzy CSP Q' obtained by SP from Q. Notice that Q' is defined only on the controllable variables of Q, namely x, y and w. The set of constraints $C_{control}$ consists of the constraint $\langle \mu_1, \{x, w\} \rangle$ (which is a constraint of Q defined only on controllable variables) and the constraint $\langle \mu_P, \{x, y\} \rangle$, obtained projecting constraint $\langle \mu, \{x, y, z\} \rangle$ of Q on the controllable variables x and y. The set of constraints C_u, instead, contains only the constraint $\langle \mu', \{x, y\} \rangle$, obtained applying the procedure described in Section 2.2.4 to constraint $\langle \mu, \{x, y, z\} \rangle$ in Q. □

Algorithm SP can be applied also to uncertain fuzzy CSPs with constraints involving only uncontrollable variables by performing, before applying SP, the procedure described below.

Let us assume to have a constraint C_i involving only a set $\{z_1, \ldots, z_k\}$ of uncontrollable variables with possibility distributions respectively $\pi_1, \ldots, \pi_k$. In order to apply SP we must obtain a UFCSP, which is equivalent to the given one, but where every constraint contains at least a controllable variable. We propose to project C_i on its uncontrollable variables and to propagate [Dec03, BMR97] these projection constraints and the existing constraints relating some of these uncontrollable variables with a controllable one.

However, performing this procedure, some uncontrollable variable could be not related to any other variable. In order to avoid this fact, we propose to add induced constraints

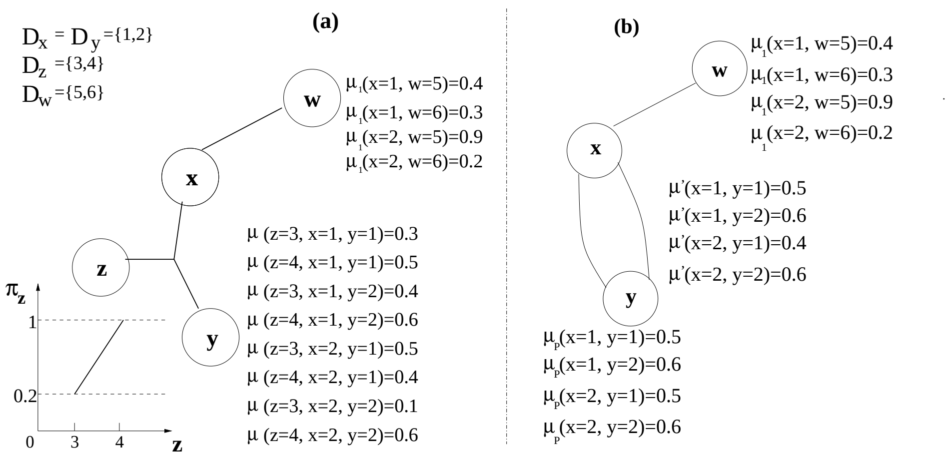

Figure 2.4: How SP works.

[BMR97] between every uncontrollable variable, that is not directly related with a controllable variable, and a controllable variable.

More precisely, let us consider an uncertain FCSP Q where there are constraints involving only uncontrollable variables. Let us call X, the set of controllable variables, Z, the set of uncontrollable variables, $Z_X \subseteq Z$, the set of uncontrollable variables, that are connected directly to a controllable variable and Z_U, the set of uncontrollable variables which are not connected directly with any controllable variable. Algorithm 1 shows how to add induced constraints between every variable in Z_U and a controllable variable in X.

Algorithm 1 takes in input an uncertain FCSP $Q = \langle S_{FCSP}, V_c = X, V_u = Z_X \cup Z_U, C\rangle$, where C may contain constraints not involving controllable variables and it returns an uncertain FCSP $Q^* = \langle S_{FCSP}, V_c = X, V_u = Z_X \cup Z_U, C^*\rangle$, where C^* doesn't contain constraints without controllable variables. At first the algorithm initializes C^* with C and it defines two sets, say $OldBorder$ and $NewBorder$, for containing during the algorithm respectively the variables related by a constraint with a controllable variable and those ones of Z_U which have been connected to a controllable one during algorithm. $OldBorder$ is initialized with the set Z_X and $NewBorder$ with the empty set. Then, for every variable z_x in $OldBorder$ related to a controllable variable, say x_z, we consider every variable $z_u \in Z_U$ connected to z_x and we add the induced constraint between x_z and z_u, by using the function InducedConstraint(x_z, z_u). Such a constraint is defined on the variables x_z and z_u and the preference function of every assignment to these variables is obtained by performing the maximum of the minimum of the preferences associated by the original constraints to the appropriate subtuples. After that, variable z_u is added to $NewBorder$ and z_x is removed from $OldBorder$. When $OldBorder$ is empty but $NewBorder$ is not empty, then

Algorithm 1: Adding deduced constraints relating controllable and uncontrollable variables

Input: $Q = \langle S_{FCSP}, V_c = X, V_u = Z_X \cup Z_U, C \rangle$: UFCSP;
Output: $Q^* = \langle S_{FCSP}, V_c = X, V_u = Z_X \cup Z_U, C^* \rangle$: UFCSP;
$C^* \leftarrow C$;
$OldBorder \leftarrow Z_X$;
$NewBorder \leftarrow \emptyset$;
while $OldBorder \neq \emptyset$ **do**
    foreach $z_x \in OldBorder$ **do**
        foreach $z_u \in Z_U$ *connected to* z_x **do**
            $C^* = C^*$ + InducedConstraint(x_z, z_u) ;
            $NewBorder \leftarrow NewBorder \cup \{z_u\}$;
        $OldBorder \leftarrow OldBorder \setminus \{z_x\}$;
        if $OldBorder$ = $\emptyset$ **then**
            $OldBorder \leftarrow NewBorder$;
            $NewBorder \leftarrow \emptyset$;
return $Q^* = \langle S_{FCSP}, V_c = X, V_u = Z_X \cup Z_U, C^* \rangle$

$OldBorder$ is set to $NewBorder$ and $NewBorder$ to the empty set. When both $OldBorder$ and $NewBorder$ are the empty set, algorithm terminates.

Example 6 An example of how SP preprocessing works is shown in Figure 2.5. Figure 2.5 (a) shows an uncertain FCSP Q with a constraint between two uncontrollable variables z_1 and z_2 respectively with domain Dz_1 and Dz_2. Figure 2.5 (b) shows the UFCSP Q^* obtained from Q by adding the projections constraints on z_1 and z_2, i.e., $C_4 = \langle \mu_{p1}, \{z_1\} \rangle$ and $C_5 = \langle \mu_{p2}, \{z_2\} \rangle$ and by adding the induced constraint $C_3 = \langle \mu, \{x, z_2\} \rangle$, between the controllable variable x and z_2. In C_3 the preference of every assignment $(x = \bar{a}, z_2 = \bar{c})$ is computed as follows: $\mu(x = \bar{a}, z_2 = \bar{c}) = max_{\{\bar{a}_i \in D_{z_2}\}} min(\mu_1(x = \bar{a}, z_1 = \bar{a}_i), \mu_2(z_1 = \bar{a}_i, z_2 = \bar{c}))$. The next step to perform, for having a UFCSP with only constraints containing at least a controllable variable, is to propagate the constraints C_1, C_3, C_4 and C_5 of the UFCSP Q^* and to remove C_2. The resulting UFCSP, that is shown in Part (c), is $Q^{**} = \langle S_{FCSP}, V_c = \{x\}, V_u = \{z_1, z_2\}, C^{**} = C'_1 \cup C'_3, \rangle$, where C'_1 and C'_3 are the constraints obtained respectively from C_1 and C_3 after propagation. In this particular example C'_1 coincides with C_1 and C'_3 with C_3. □

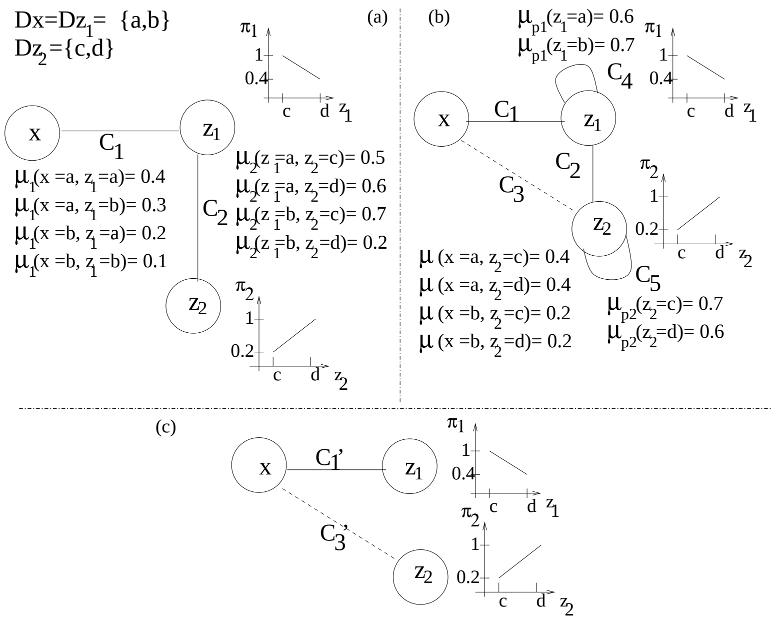

Figure 2.5: SP preprocessing.

2.5 Satisfaction degree and robustness

In this section we show how to compute the preference of a solution of an UFCSP. We define this preference via two values and we show that they correspond to the satisfaction degree and to the robustness value presented in Section 2.3.

We recall that, given a UFCSP $Q = \langle S_{FCSP}, V_c, V_u, C = C_f \cup C_{fu} \rangle$, algorithm SP obtains a FCSP $Q' = \langle S_{FCSP}, V_c, C = C_f \cup C_p \cup C_u \rangle$. We associate to every solution s of Q i.e., to every complete assignment to V_c, a preference that is given by the values $F_P(s) = min(F(s), P(s))$ and $U(s)$, where $F(s)$, $P(s)$ and $U(s)$ are respectively the minimum preference over the constraints in C_f, C_p and C_u.

Example 7 In Figure 2.6 (a) we show the FCSP Q' obtained by SP from the UFCSP Q in Figure 2.4 (a). Figure 2.6 (b) shows all the solutions of Q', i.e. all the complete assignments to controllable variables, associated with a preference given by μ_t, which is defined by the values F_P and U. □

The values F_P and U associated to every solution correspond to the satisfaction degree and to the robustness as defined in Section 2.3.

More precisely, given a solution s, the value $F_P(s)$ represents the *satisfaction degree* of s.

In fact, $F_P(s) = min_{\{<\mu,con>\in C_{control}\}}\mu(s \downarrow_{con})$ coincides with the definition of $sat(s)$ given in Section 2.3. Thus, it represents the satisfaction degree of s in terms of preferences,

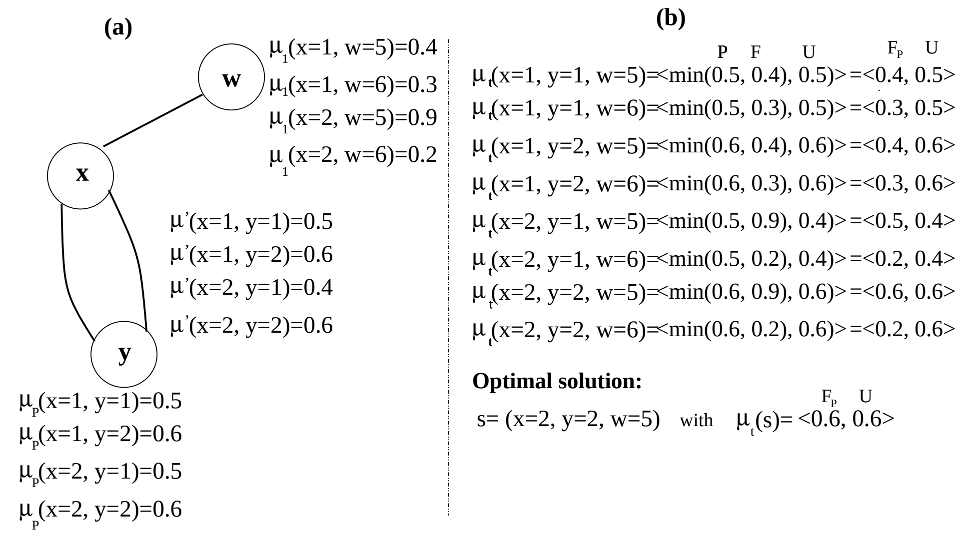

Figure 2.6: Solutions of a FCSP Q' obtained by SP.

when the uncontrollable variables are eliminated by projecting the constraints involving them on the controllable variables.

Given a solution s, $U(s)$ matches the definition of *robustness* given in Section 2.3.

In fact, $U(s) = min_{\{<\mu',con'>\in C_u\}}\mu'(s \downarrow_{con'})$, that is, $U(s)$ is the "partial" preference obtained by s on the constraints C_u, which are generated by the process eliminating the uncontrollable variables. Let us recall that such a procedure replaces every constraint $\langle \mu, con\rangle$ in the USCSP such that $con \cap V_u = Z \neq \emptyset$ and $con \cap V_c = X \neq \emptyset$, with a new constraint $\langle \mu', con'\rangle$ such that $con' = con \cap V_c$ and $\mu'(s \downarrow_{con'}) = inf_{a\in A_z} max(\mu(s \downarrow_{con'}, a), c(\pi_Z(a)))$ where $a \in A_z$ (A_z is the domain of Z) is an assignment to the uncontrollable variables in Z.

From this, it can be clearly seen that $U(s)$ depends on the preferences of s on constraints involving uncontrollable variables and on the possibility distributions defined on such variables.

Moreover, as indicated in Section 2.2.4, if $\mu'(s \downarrow_{con'}) = \alpha$, then $\mu(s \downarrow_{con'}, a) \geq \alpha$, $\forall a$ such that $\pi(a) > 1 - \alpha$. This means that if $U(s) = \beta$ then on each constraint involving uncontrollable variables we are sure that choosing s will give a preference of at least β when Nature chooses values with possibility higher than $1 - \beta$. Thus the higher the robustness the higher preference is guaranteed in a larger number of scenarios.

Notice that this is a generalization of a more intuitive measure of robustness which considers the minimum preference which can be obtained in any possible case. In particular, if we consider the case in which there is no additional information on the uncertain events, then $U(s)$ is exactly the minimum of preferences obtained by s on the constraints involving controllable and uncontrollable variables. In detail, if we are in the case of complete ignorance, then the possibilities of every assignment a to the uncontrollable variables are equal to 1, i.e. $\pi_Z(a) = 1\ \forall a$, then $U(s) = min_{\{<\mu', con'> \in C_u\}} \mu'(s\downarrow_{con'})$ and for each constraint $\mu'(s\downarrow_{con'}) = inf_{a \in A_z} max(\mu(s\downarrow_{con'}, a), c(\pi_Z(a))) = inf_{a \in A_z} max(\mu(s\downarrow_{con'}, a), c(1))$ $= inf_{a \in A_z} \mu(s\downarrow_{con'}, a)$, since $c(1) = 0$.

The general definition of robustness presented here generalizes the intuitive notion taking into account the possibility distribution associated to the uncontrollable variables. Neglecting such additional information can lead to unreasonable judgments, as illustrated by the following example. In particular, using the intuitive definition of robustness we could consider as bad a situation that behaves well in almost all cases and that is bad only in one (very unlikely) case. For example, let us consider the constraint $\langle \mu, \{x, z\} \rangle$ linking the controllable variable, x, with the uncontrollable variable, z defined by preference function μ as follows: $\mu(x = d, z = a_1) = 0.9$, $\mu(x = d, z = a_2) = 0.9$, $\mu(x = d, z = a_3) = 0.9$, $\mu(x = d, z = a_4) = 0.2$. Let us assume that assignments $z = a_1$, $z = a_2$ and $z = a_3$ have possibility equal to 1 and that assignment $z = a_4$ has possibility 0.1. In this case, according to the intuitive notion, solution $x = d$ has a robustness $= inf(0.9, 0.9, 0.9, 0.2) = 0.2$, even if it behaves badly only in one case. Instead, considering the more refined notion gives a robustness $U(d) = inf_{\{a_1,a_2,a_3,a_4\}}(max(0.9, 0), max(0.9, 0), max(0.9, 0), max(0.2, 0.9)) = 0.9$, which states that solution $x = d$ behaves well in the most possible cases.

2.6 Desired properties on robustness

We will now show that U satisfies Properties 1 and 2 presented in Section 2.3.

Proposition 1 *Consider two uncertain Fuzzy CSPs:* $Q_1 = \langle S_{FCSP}, V_c, V_u, C_1 = C_{f_1} \cup C_{fu_1} \rangle$ *and* $Q_2 = \langle S_{FCSP}, V_c, V_u, C_2 = C_{f_2} \cup C_{fu_2} \rangle$*, where* C_1 *and* C_2 *differ only by the preference functions of constraints involving variables in* V_u*, i.e.,* $C_{f_1} = C_{f_2}$*,* $C_{fu_1} = \bigcup_i \langle \mu_1^i, con^i \rangle$ *and* $C_{fu_2} = \bigcup_i \langle \mu_2^i, con^i \rangle$*. In particular, for every such constraint,* $c^i = \langle \mu^i, con^i \rangle$*, such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$*, with possibility distribution* π_{Z^i}*, let* $\mu_1^i(d, a) \leq \mu_2^i(d, a)$*, for all* a *assignments to* Z^i *and for all* d *assignments to* X^i*. Then, given solution* s *of* Q_1 *and* Q_2*, such that* $s\downarrow_{X^i} = d$*,* $U_1(s) \leq U_2(s)$*.*

Proof: We recall that, for every constraint $c^i = \langle \mu^i, con^i \rangle$ in the statement of Proposition 1, $\mu_1'^i(d) = inf_{a \in A_{z^i}} max(\mu_1^i(d, a), c(\pi_{Z^i}(a)))$ and $\mu_2'^i(d) = inf_{a \in A_{z^i}} max(\mu_2^i(d, a), c(\pi_{Z^i}(a)))$, where A_{z^i} is the Cartesian product of the domains of the variables in Z^i. Since $\mu_1^i(d, a) \leq \mu_2^i(d, a)$, $\forall a, d$, then $max(\mu_1^i(d, a), c(\pi_{Z^i}(a))) \leq max(\mu_2^i(d, a), c(\pi_{Z^i}(a)))$, $\forall a, d$. Therefore, $inf_{a \in A_{z^i}}\ max(\mu_1^i(d, a), c(\pi_{Z^i}(a))) \leq max(\mu_1^i(d, a), c(\pi_{Z^i}(a))) \leq max\ (\mu_2^i(d, a),$

$c(\pi_{Z^i}(a)))$, $\forall a$, $\forall d$. This allows to conclude that, since $s \downarrow_{X^i}= d$, $\mu_1'^i(s \downarrow_{X^i}) = inf_{a\in A_{z^i}}$ $max\ (\mu_1(s \downarrow_{X^i}, a), c(\pi_{Z^i}(a))) \leq inf_{a\in A_{z^i}}\ max(\mu_2(s \downarrow_{X^i}, a), c(\pi_{Z^i}(a))) = \mu_2'^i(s \downarrow_{X^i})$. The fact that $U_1(s) = min_i\mu_1'^i(s \downarrow_{X_i})$ and $U_2(s) = min_i\mu_2'(s \downarrow_{X_i})$ allows us to conclude. □

Proposition 2 *Consider two uncertain Fuzzy CSPs:* $Q_1 = \langle S_{FCSP}, V_c, V_u, C\rangle$ *and* $Q_2 = \langle S_{FCSP}, V_c, V_u', C\rangle$*, where* V_u *and* V_u' *are the same set of uncontrollable variables described, however, by different possibility distributions. In particular, for every constraint,* $c^i = \langle \mu^i, con^i\rangle$*, such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$*, let* $\pi^1_{Z^i}(a) \geq \pi^2_{Z^i}(a)$*, for all* a *assignments to* Z^i*. Then, given solution* s *of* Q_1 *and* Q_2*, such that* $s \downarrow_{X^i}= d$*,* $U_1(s) \leq U_2(s)$*.*

Proof: As in the proof of Proposition 1, we have that, for every constraint, $\mu_1'^i(d) = inf_{a\in A_{z^i}}$ $max(\mu^i(d, a), c(\pi^1_{Z^i}(a)))$ and $\mu_2'^i(d) = inf_{a\in A_{z^i}}\ max(\mu^i(d, a), c(\pi^2_{Z^i}(a)))$. Moreover, since c is an order-reversing map, if $\pi^1_{Z^i}(a) \geq \pi^2_{Z^i}(a)$ then $c(\pi^1_{Z^i}(a)) \leq c(\pi^2_{Z^i}(a))$, $\forall a$. Thus, $max(\mu^i(d, a), c(\pi^1_{Z^i}(a))) \leq max(\mu^i(d, a), c(\pi^2_{Z^i}(a)))$, $\forall a$. From here we can conclude as above. □

2.7 Semantics

Once each solution is associated with two values, the satisfaction degree F_P and the robustness U, then there can be several ways to order the solutions. We will now propose various approaches which differ on the attitude toward risk they implement. In the following we will present some semantics that we believe to be reasonable.

Definition 18 (semantics) Given an uncertain FCSP Q, consider a solution s with corresponding satisfaction degree $F_P(s)$ and robustness $U(s)$. Each semantics associates to s the ordered pair $\langle a_s, b_s\rangle$ as follows:

- **Risky** (R), **Diplomatic** (D): $\langle a_s, b_s\rangle$=$\langle F_P(s), U(s)\rangle$;
- **Safe** (S): $\langle a_s, b_s\rangle$=$\langle U(s), F_P(s)\rangle$;
- **Risky1** (R1): $\langle a_s, b_s\rangle$=$\langle min(F_P(s), U(s)), F_P(s)\rangle$;
- **Safe1** (S1): $\langle a_s, b_s\rangle$=$\langle min(F_P(s), U(s)), U(s)\rangle$.

Given two solutions s and s', let $\langle a_s, b_s\rangle$ and $\langle a_{s'}, b_{s'}\rangle$ represent the pairs associated to the solutions by each semantics in turn. The **Risky, Safe, Risky1, Safe1** semantics work as follows:

- if $a_1 > a_2$ then $\langle a_1, b_1 \rangle >_J \langle a_2, b_2 \rangle$ (and the opposite for $a_2 > a_1$)
- if $a_1 = a_2$ then
  - if $b_1 > b_2$ then $\langle a_1, b_1 \rangle >_J \langle a_2, b_2 \rangle$ (and the opposite for $b_2 > b_1$)
  - if $b_1 = b_2$ then $\langle a_1, b_1 \rangle = \langle a_2, b_2 \rangle$;

where $J = R, S, R1, S1$.

The **Diplomatic** semantics works as follows:

- if $a_1 \leq a_2$ and $b_1 \leq b_2$ then $\langle a_1, b_1 \rangle \leq_D \langle a_2, b_2 \rangle$ (and the opposite for $a_2 \leq a_1$ and $b_2 \leq b_1$);
- if $a_1 = a_2$ and $b_1 = b_2$ then $\langle a_1, b_1 \rangle = \langle a_2, b_2 \rangle$;
- else $\langle a_1, b_1 \rangle \bowtie \langle a_2, b_2 \rangle$ ($\bowtie$ means incomparable).

As it can be seen by Definition 18 all semantics, except Diplomatic, can be regarded as a Lex ordering on pairs $\langle a_s, b_s \rangle$ with the first component as the most important feature. Diplomatic, instead, is a Pareto ordering on the pairs.

The first semantics we propose, which we call *Risky*, considers F_P as the most important feature. Informally, the idea is to give more relevance to the satisfaction degree that can be reached in the best case considering less important a high risk of being inconsistent. Hence we are risky, since we disregard almost completely the uncertain part of the problem.

The second semantics, called *Safe*, represents the opposite attitude with the respect to the previous one, since it considers $U(s)$ as the most important feature. Informally, the idea is to give more importance to the robustness level that can be reached considering less important having a high preference. In particular, in this case we consider a solution better than another one if its robustness is higher, i.e., if it guarantees an higher number of scenarios with an higher preference. This semantics considers the satisfaction degree of a solution only for ordering solutions having the same robustness. This can be useful for reasoning with uncertain problems when we are mainly interested in the part of the problem that we cannot control. In this case we want to find the most robust solution independently from its satisfaction degree. If the chosen solution has a very bad satisfaction degree we could modify that solution if we want, since we can decide the controllable part.

The *Risky1* semantics tries to overcome the myopic attitude of *Risky*, which concentrates only on the satisfaction degree (except when there is a tie), by first considering the ordering generated by the minimum of $F_P(s)$ and $U(s)$. This allows to avoid considering as good, solutions which will give a low overall preference in most of the possible scenarios.

Similarly the *Safe1* semantics mitigates the relevance given to robustness in *Safe*. In fact, not considering the minimum value between F_P and U before focusing on U, as *Safe* does, can lead to consider as optimal solutions which have a poor overall preference despite guaranteeing high preference on the constraints involving uncontrollables.

The last semantics proposed, called *Diplomatic*, aims at giving the same importance to the two aspects of a solution: satisfaction degree and robustness. As mentioned above, the Pareto ordering on pairs $\langle a_s, b_s \rangle$ is adopted. The idea is that a pair is to be preferred to another only if it wins both on preference and robustness, leaving incomparable all the pairs that have one component higher and the other lower. Contrarily to the Diplomatic semantics, the other semantics produce a total order over the solutions.

All semantics differ with respect to the attitude toward risk they implement and with respect to the point of view from which they consider the problem. In particular, in *Risky* and *Safe* for each solution, the two aspects, i.e., how well the solution performs respectively on the controllable part of the problem and on the constraints involving uncontrollables, are kept separated. The relation between how a solution satisfies the two aspects is ignored. This can be seen as a myopic attitude which focuses on the satisfaction of the controllable part in *Risky*, and on the compatibility with uncertainty in *Safe*. It should be noticed, however that, given a solution s, while on the robustness, $U(s)$, the preference obtained by s on the constraints involving only controllables has absolutely no impact, on the satisfaction degree, $F_P(s)$, the compatibility of the solution with uncertain events is taken into account through the projection constraints. In this sense, *Risky* can be regarded as less myopic than *Safe*.

A more global view of the problem characterizes instead the *Risky1*, *Safe1* and *Diplomatic* semantics. In fact, in the first two, considering the minimum of $F_P(s)$ and $U(s)$ allows to order the solutions first with respect to their predominant aspect, that is the one on which they have a worst performance. *Diplomatic* resolves by using incomparability situations in which each solution beats the other one only in one of the aspects. This corresponds to consider both aspects separate but with the same importance.

Example 8 Figure 2.6 (b) shows a solution of the FCSP in Figure 2.6 (a) which is optimal according to all the semantics described in Definition 18. □

Let us now consider an example that explains the differences between the various semantics.

Example 9 Let us consider two solutions of an UFCSP, s_1 and s_2, such that $F_P(s_1) = 0.3$,

$U(s_1) = 0.5$, $F_P(s_2) = 0.5$ and $U(s_2) = 0.3$. According to the semantics defined above we have the following orderings:

- $s_1 <_{R,R1} s_2$;
- $s_1 >_{S,S1} s_2$;
- $s_1 \bowtie_D s_2$.

If we consider two solutions, s_3 and s_4, such that $F_P(s_3) = 0.5$, $U(s_3) = 0.3$, $F_P(s_4) = 0.6$ and $U(s_4) = 0.2$, then

- $s_3 <_R s_4$;
- $s_3 >_{S,S1,R1} s_4$;
- $s_3 \bowtie_D s_4$.

□

2.8 Desired properties on the solution ordering

We will now check if the semantics presented in Definition 18 satisfy the desired properties described in Section 2.3 on the solution ordering. In particular, we will show that one of these semantics, i.e., the Risky semantics, satisfies all these properties.

The following proposition states that Property 3 is satisfied only by Risky and Risky1.

Proposition 3 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C \rangle$. Given two solutions s and s' of Q, i.e., assignments to V_c, if $\forall a_i$ assignments to V_u in Q, $pref(s, a_i) > pref(s', a_i)$, then $s >_J s'$, where $J = R, R1$. Instead, it could happen $s \not>_J s'$ for $J = S, S1, D$.*

Proof:

- *Risky1.* From UFCSP Q we can obtain an equivalent UFCSP $QP = \langle S_{FCSP}, \{V^c\}, \{V^u\}, C_1 \cup C_2 \cup C_3 \rangle$ where: V^c is a controllable variable and V^u is an uncontrollable variable, representing respectively all the variables in V_c and V_u, having as domains the corresponding Cartesian products. The uncontrollable variable V^u is described by a possibility distribution, π, which is the joint possibility (see Section 2.2.4) of all the possibility distributions of the uncontrollable variables in V_u. Constraints $C_1 = \langle \mu_1, V^c \rangle$ and $C_2 = \langle \mu_2, \{V^c, V^u\} \rangle$ are, respectively, defined as the combination of all the constraints in C connecting variables in V_c and as the combination of all the constraints in C connecting variables in V_c to variables in V_u. Constraint $C_3 = \langle \mu_3, V^c \rangle$ is defined as the combination of all the constraints obtained from constraints in C_2 by projecting them over the controllable variables in V_c (i.e., $C_3 = C_2 \downarrow_{V_c}$). Thus, given assignment $V^c = s$ in QP, which corresponds to an assignment to all the variables in V_c, its preference on constraint C_1 is $\mu_1(s) = F(s)$, on C_3 is $\mu_3(s) = P(s)$ and on $C_1 \bigotimes C_3$ is $min(\mu_1(s), \mu_3(s)) = min(F(s), P(s)) = F_P(s)$. Given assignment $(V^c = s, V^u = a_i)$, instead, which corresponds to a complete assignment to variables in V_c and V_u, its preference, $\mu_2(s, a_i)$, is obtained performing the minimum of the preferences associated to all the subtuples of (s, a_i) by the constraints in C involving at least one variable in V_u and one in V_c. Using this new notation we have that $\forall s, a_i$ assignments to V^c and V^u, $pref(s, a_i) = min(\mu_1(s), \mu_2(s, a_i)) = min(F(s), \mu_2(s, a_i))$.

We want to show that if $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$, then $s >_{R1} s'$, i.e., $min(F_P(s), U(s)) > min(F_P(s'), U(s'))$ or $(min(F_P(s), U(s)) = min(F_P(s'), U(s'))$ and $F_P(s) > F_P(s'))$. First, we show that if $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$, then $F_P(s) > F_P(s')$. Then, for proving $s >_{R1} s'$, it is sufficient to prove that $min(F_P(s), U(s)) \geq min(F_P(s'), U(s'))$.

<u>*First part.*</u> If $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$ assignment to V^u, then $F_P(s) > F_P(s')$.

In fact, if $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$, then this holds also for a_{i*} such that $P(s') = \mu_2(s, a_{i*})$. Then we have $min(F(s), \mu_2(s, a_{i*})) > min(F(s'), \mu_2(s', a_{i*})) = min(F(s'), P(s')) = F_P(s')$. Since $P(s) \geq \mu_2(s, a_{i*}))$, then $F_P(s) = min(F(s), P(s)) \geq min(F(s), \mu_2(s, a_{i*})) > F_P(s')$, and so $F_P(s) > F_P(s')$.

Notice that from the result above follows that: if $F_P(s) \leq F_P(s')$, then $pref(s, a_i) \leq pref(s', a_i)$, $\exists a_i$.

<u>*Second part.*</u> If $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$ assignment to V^u, then we have that $min(F_P(s), U(s)) \geq min(F_P(s'), U(s'))$.

The proof is given by contradiction. That is, we will show that if $min(F_P(s), U(s)) < min(F_P(s'), U(s'))$, then there is an assignment $a_{\bar{1}}$ such that $pref(s, a_{\bar{1}}) \leq pref(s', a_{\bar{1}})$.

1. Assume $min(F_P(s), U(s)) = F_P(s)$ and $min(F_P(s'), U(s')) = F_P(s')$. Since we are assuming $min(F_P(s), U(s)) < min(F_P(s'), U(s'))$, then it must be $F_P(s) < F_P(s')$. Then we can conclude by the first part of the proof.

2. Assume $min(F_P(s), U(s)) = F_P(s)$ and $min(F_P(s'), U(s')) = U(s')$. Then, $U(s') \leq F_P(s')$. Since we are assuming that $F_P(s) < U(s')$, then, this implies that $F_P(s) < F_P(s')$. Hence we can conclude as in the previous step.

3. Assume $min(F_P(s), U(s)) = U(s)$ and $min(F_P(s'), U(s')) = U(s')$. Moreover, and without loss of generality[5], let us consider the case in which $U(s) < F_P(s)$, $U(s') < F_P(s')$, and thus, $U(s) < U(s') < F_P(s')$.

   If we consider the uncertain FSCP QP, then $U(s) = inf_{a_i}\ (max(\mu_2(s,\ a_i),$ $c(\pi(a_i))))$ and $U(s') = inf_{a_i}\ (max(\mu_2(s', a_i),\ c(\pi(a_i))))$. For the sake of notation we will indicate $max(\mu_2(s, a_i),\ c(\pi(a_i)))$ (respectively $max(\mu_2(s',\ a_i),$ $c(\pi(a_i)))$ with m_i (respectively m'_i). Let $a_{\bar{1}}$ and $a_{\bar{\jmath}}$ be the values for V^u such that $U(s) = m_{\bar{1}}$ and $U(s') = m'_{\bar{\jmath}}$ (i.e., $m'_{\bar{\jmath}} = max(\mu_2(s', a_{\bar{\jmath}}), c(\pi(a_{\bar{\jmath}})))$). Then $U(s) = m_{\bar{1}} \geq \mu_2(s, a_{\bar{1}})$. Thus, since $F_P(s) > U(s)$, then $F(s) > F_P(s) > U(s) = max(\mu_2(s, a_{\bar{1}}),\ c(\pi(a_{\bar{1}}))) \geq \mu_2(s, a_{\bar{1}})$. This allows us to conclude that

$$pref(s, a_{\bar{1}}) = min(F(s), \mu_2(s, a_{\bar{1}})) = \mu_2(s, a_{\bar{1}}).$$

   We will now show that, for the assignment $a_{\bar{1}}$, we have $pref(s, a_{\bar{1}}) < pref(s', a_{\bar{1}})$. In order to do that we will consider all the possible cases from which $m_{\bar{1}}$ and $m'_{\bar{1}}$ can derive, where $m'_{\bar{1}} = max(\mu_2(s', a_{\bar{1}}), c(\pi(a_{\bar{1}})))$.

   First of all, since $m_{\bar{1}} = U(s) < U(s') = m'_{\bar{\jmath}}$ and since $m'_{\bar{\jmath}} \leq m'_{\bar{1}}$, then it must be $m_{\bar{1}} < m'_{\bar{1}}$. The cases to be considered are the following:

   – $m_{\bar{1}} = m'_{\bar{1}} = c(\pi(a_{\bar{1}}))$. This can never occur since it contradicts $m_{\bar{1}} < m'_{\bar{1}}$.
   – $m_{\bar{1}} = \mu_2(s, a_{\bar{1}})$ and $m'_{\bar{1}} = \mu_2(s', a_{\bar{1}})$. Thus, $\mu_2(s, a_{\bar{1}}) = m_{\bar{1}} < m'_{\bar{1}} = \mu_2(s', a_{\bar{1}})$.
     * If $m'_{\bar{1}} = \mu_2(s', a_{\bar{1}}) < F(s')$, then $pref(s', a_{\bar{1}}) = \mu_2(s', a_{\bar{1}})$. Since we know that $pref(s, a_{\bar{1}}) = \mu_2(s, a_{\bar{1}})$, then $pref(s, a_{\bar{1}}) < pref(s', a_{\bar{1}})$.
     * If $m'_{\bar{1}} = \mu_2(s', a_{\bar{1}}) \geq F(s')$, then $pref(s', a_{\bar{1}}) = F(s')$ and $pref(s, a_{\bar{1}}) = \mu_2(s, a_{\bar{1}}) = m_{\bar{1}} = U(s) < m'_{\bar{\jmath}} = U(s') < F_P(s') \leq F(s') = pref(s', a_{\bar{1}})$. Again $pref(s,\ a_{\bar{1}}) < pref(s',\ a_{\bar{1}})$.
   – $m_{\bar{1}} = \mu_2(s, a_{\bar{1}})$ and $m'_{\bar{1}} = c(\pi(a_{\bar{1}}))$. This case can never occur since it would give the following contradiction: $c(\pi(a_{\bar{1}})) \leq m_{\bar{1}} = \mu_2(s, a_{\bar{1}}) < m'_{\bar{1}} = c(\pi(a_{\bar{1}}))$.

[5]We consider only the case with strict inequalities since if $U(s) = F_P(s)$, $U(s') = F_P(s')$ we are in Case 1, if $U(s) = F_P(s)$, $U(s') < F_P(s')$ we are in Case 2, and if $U(s) < F_P(s)$, $U(s') = F_P(s')$ we are in Case 4.

- $m_{\bar{\imath}} = c(\pi(a_{\bar{\imath}}))$ and $m'_{\bar{\imath}} = \mu_2(s', a_{\bar{\imath}})$. Thus, $pref(s, a_{\bar{\imath}}) = min(F(s), \mu_2(s, a_{\bar{\imath}})) \leq min(F(s), c(\pi(a_{\bar{\imath}}))) = min(F(s), U(s)) =$ (since $P(s) \geq U(s)$) $min(F(s), P(s), U(s)) = min(F_P(s), U(s)) < min(F_P(s'), U(s')) \leq min(F_P(s'), \mu_2(s', a_{\bar{\imath}})) \leq min(F(s'), \mu_2(s', a_{\bar{\imath}})) = pref(s', a_{\bar{\imath}}))$.

4. Assume $min(F_P(s), U(s)) = U(s)$ and $min(F_P(s'), U(s')) = F_P(s')$.

   Again, we consider only the case with strict inequalities ($U(s) < F_P(s)$, $F_P(s') < U(s')$) since all the others can be treated as one of the previous cases.

   Since $U(s) < F_P(s')$, then we have $U(s) < F_P(s') < U(s')$. Let $m_{\bar{\imath}} = U(s)$ and $m'_{\bar{\jmath}} = U(s')$ as in Case 3. Since $U(s) < F_P(s) \leq F(s)$, then, as before, $pref(s, a_{\bar{\imath}}) = \mu_2(s, a_{\bar{\imath}})$.

   We will show that $pref(s, a_{\bar{\imath}}) < pref(s', a_{\bar{\imath}})$. As in the previous case, in order to do so, we consider all the possible cases from which $m_{\bar{\imath}}$ and $m'_{\bar{\imath}}$ can derive. First, notice that from $U(s) < U(s')$ we get $m_{\bar{\imath}} < m'_{\bar{\jmath}}$. The cases to be considered are the following:

   - $m_{\bar{\imath}} = m'_{\bar{\imath}} = c(\pi(a_{\bar{\imath}}))$. We conclude as in the corresponding step of Case 3.
   - $m_{\bar{\imath}} = \mu_2(s, a_{\bar{\imath}})$ and $m'_{\bar{\imath}} = \mu_2(s', a_{\bar{\imath}})$. Then $pref(s, a_{\bar{\imath}}) = \mu_2(s, a_{\bar{\imath}}) = m_{\bar{\imath}} = U(s) < F_P(s') < U(s') = m'_{\bar{\jmath}} \leq m'_{\bar{\imath}} = \mu_2(s', a_{\bar{\imath}})$. Hence, $pref(s', a_{\bar{\imath}}) = min(F(s'), \mu_2(s', a_{\bar{\imath}})) \geq min(F_P(s'), \mu_2(s', a_{\bar{\imath}})) = F_P(s') > pref(s, a_{\bar{\imath}})$.
   - $m_{\bar{\imath}} = \mu_2(s, a_{\bar{\imath}})$ and $m'_{\bar{\imath}} = c(\pi(a_{\bar{\imath}}))$. We conclude like in the corresponding step of Case 3.
   - If $m_{\bar{\imath}} = c(\pi(a_{\bar{\imath}}))$ and $m'_{\bar{\imath}} = \mu_2(s', a_{\bar{\imath}})$, then $pref(s, a_{\bar{\imath}}) = \mu_2(s, a_{\bar{\imath}}) \leq m_{\bar{\imath}} = U(s) < F_P(s') < U(s') \leq m'_{\bar{\imath}} = \mu_2(s', a_{\bar{\imath}})$. Hence $pref(s', a_{\bar{\imath}}) = min(F(s'), \mu_2(s', a_{\bar{\imath}})) \geq min(F_P(s'), \mu_2(s', a_{\bar{\imath}})) = F_P(s') > pref(s, a_{\bar{\imath}})$.

- *Risky.* We can conclude that $s >_R s'$ for Risky semantics, by using the first part of the proof for Risky1 semantics.

- *Safe and Diplomatic.* For these semantics it can happen that $s \not> s'$.

  In fact, let us consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C \rangle$ where $V_c = x$, $V_u = z$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.4$, $\mu_2(s, a_2) = 0.5$, $\mu_2(s', a_1) = 0.8$, $\mu_2(s', a_2) = 0.9$, $\mu_1(s) = 0.3$ and $\mu_1(s') = 0.2$. Then the overall preferences are:

$pref(s, a_1) = 0.3$, $pref(s, a_2) = 0.3$, $pref(s', a_1) = 0.2$, $pref(s', a_2) = 0.2$, i.e., $pref(s, a_i) > pref(s', a_i), \forall a_i$, $i = 1, 2$, hence s and s' satisfy the hypothesis. The robustness values for s' and s are $U(s) = 0.4$, $U(s') = 0.8$ and the satisfaction degrees are $F_P(s) = 0.3$ and $F_P(s') = 0.2$. Therefore, $s <_S s'$ for Safe semantics, and $s \bowtie_D s'$ for Diplomatic semantics.

- *Safe1*. For this semantics it can happen that $s \not> s'$.

  In fact, let us consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$ where $V_c = x$, $V_u = z$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.5$, $\mu_2(s, a_2) = 0.2$, $\mu_2(s', a_1) = 0.4$, $\mu_2(s', a_2) = 0.1$, $\mu_1(s) = 0.9$ and $\mu_1(s') = 0.9$. Then the overall preferences are: $pref(s, a_1) = 0.5$, $pref(s, a_2) = 0.2$, $pref(s', a_1) = 0.4$, $pref(s', a_2) = 0.1$, i.e., $pref(s, a_i) > pref(s', a_i), \forall a_i$, $i = 1, 2$, hence s and s' satisfy the hypothesis. The robustness values for s' and s are $U(s) = 0.3$, $U(s') = 0.3$ and the satisfaction degrees are $F_P(s) = 0.5$ and $F_P(s') = 0.4$. Since $min(F_P(s), U(s)) = 0.3 = min(F_P(s'), U(s'))$ and $U(s) = U(s')$, then $s =_{S1} s'$ for Safe1 semantics.

□

Property 4 is satisfied by Risky, Safe, Diplomatic, and Risky1, but it is not satisfied by Safe1 as shown in the following proposition.

Proposition 4 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$. Given two solutions s and s' of Q, if $U(s) = U(s')$ and $F_P(s) > F_P(s')$, then $s >_J s'$, where $J = R, S, D, R1$. Instead, it could happen that $s \not>_{S1} s'$.*

Proof:

- *Risky*, *Safe* and *Diplomatic* satisfy this property by definition.

  Also *Risky1* satisfies this property. $s >_{R1} s'$ means that $min(F_P(s), U(s)) > min(F_P(s'), U(s'))$, or that $min(F_P(s), U(s)) = min(F_P(s'), U(s'))$ and $F_P(s) > F_P(s')$. Since $F_P(s) > F_P(s')$ and $U(s) = U(s')$ then $min(F_P(s), U(s)) \geq min(F_P(s'), U(s'))$. If we have $min(F_P(s), U(s)) > min(F_P(s'), U(s'))$, then we conclude immediately. If $min(F_P(s), U(s)) = min(F_P(s'), U(s'))$, we conclude by observing that $F_P(s) > F_P(s')$.

- *Safe1*. In this case it can happen $s \not>_{S1} s'$. Let us recall that $s >_{S1} s'$ means that $min(F_P(s), U(s)) > min(F_P(s'), U(s'))$, or that $min(F_P(s), U(s)) = min(F_P(s'), U(s'))$ and $U(s) > U(s')$. Assume, for example, that s and s' are such that $P(s) = 0.9$, $F(s) = 0.8$ and $U(s) = 0.5$ and $P(s') = 0.9$, $F(s') = 0.7$ and $U(s') = U(s) = 0.5$. Then, s and s' satisfy the hypothesis since $F_P(s) = min(P(s) = 0.9, F(s) = 0.8) = 0.8 > F_P(s') = min(P(s') = 0.9, F(s') = 0.7,) = 0.7$. However, $s =_{S1} s'$, since $min(F_P(s) = 0.8, U(s) = 0.5) = 0.5$ is equal to $min(F(s') = 0.7, U(s') = 0.5)$ and $U(s') = U(s) = 0.5$.

□

The next proposition shows that Property 5 is satisfied by Risky, Safe, Diplomatic and Safe1, but that it is not satisfied by Risky1.

Proposition 5 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$. Given two solutions s and s' of Q, if $F_P(s) = F_P(s')$ and $U(s) > U(s')$, then $s >_J s'$, where $J = R, S, D, S1$. Instead, it could happen that $s \not>_{R1} s'$.*

Proof:

- *Risky*, *Safe* and *Diplomatic* satisfy this property by definition.

  Also *Safe1* satisfies this property. If $F_P(s) = F_P(s')$ and $U(s) > U(s')$ then $min(F_P(s), U(s)) \geq min(F_P(s'), U(s'))$. If $min(F_P(s), U(s)) > min(F_P(s'), U(s'))$, then we conclude immediately. If $min\ (F_P(s), U(s)) = min(F_P(s'), U(s'))$ we conclude by observing that $U(s) > U(s')$.

- In *Risky1* it can happen that $s \not>_{R1} s'$. Consider for example solutions s and s' such that $P(s) = 0.9$, $F(s) = 0.5$ and $U(s) = 0.8$ and $P(s') = 0.8$, $F(s') = 0.5$ and $U(s') = 0.7$. We have that $U(s) > U(s')$ and $F_P(s) = min(P(s) = 0.9, F(s) = 0.5) = 0.5 = F_P(s') = min(P(s') = 0.8, F(s') = 0.5)$. However, since $min(F_P(s) = 0.5, U(s) = 0.8) = 0.5 = min(F_P(s') = 0.5, U(s') = 0.7)$ and $F_P(s) = F_P(s') = 0.5$, then $s =_{R1} s'$.

□

Summarizing, Risky satisfies all the desired properties on solution ordering (i.e., Properties 3, 4 and 5), Risky1 satisfies Properties 3 and 4, Safe and Diplomatic satisfy 4 and 5, Safe1 satisfies only Property 5.

2.9 Desired properties in algorithm DFP

Let us now briefly reconsider the DFP algorithm [DFP96a] (see Section 2.2.4) and the ordering it produces on solutions in terms of the desired properties.

Since in DFP the robustness is computed like in our approach, then Properties 1 and 2 on the robustness continue to hold. Using our notation, according to DFP, the preference of a solution s of an UFCSP is a single value equal to $min(F(s), U(s))$. Thus, given two solutions s and s' of an UFCSP, $s >_{DFP} s'$ if and only if $min(F(s), U(s)) > min(F(s), U(s))$. We will show that the solution ordering produced by DFP doesn't satisfy any desired properties regarding the solution ordering. Before showing this, we give a result which will be useful in the proof of the following propositions.

Theorem 1 *Consider an uncertain FCSP* $Q = \langle S_{FCSP}, V_c, V_u, C = C_f \cup C_{fu}\rangle$ *where* $C_{fu} = \bigcup_i \langle \mu^i, con^i\rangle$*, such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$*, with possibility distribution* π_{Z^i} *and domain* A_{Z_i}*. For every solution* s *of* Q*, i.e., for every assignment to* X^i*, we have* $U(s) \leq P(s)$*, where* $P(s) = min_i P_i(s)$ *and* $P_i(s) = sup_{a \in A_{Z^i}} \mu^i(s, a)$*.*

Proof: We recall that $U(s) = min_i \mu'^i(s)$, where for every constraint $c^i = \langle \mu^i, con^i\rangle$, $\mu'^i(s) = inf_{a \in A_{z^i}} max(\mu^i(s, a), c(\pi_{Z^i}(a)))$. By the definition of $\mu'^i(s)$, $\mu'^i(s) \leq max(\mu^i(s, a), c(\pi_{Z^i}(a)))$, $\forall a$, and so this holds also for a such that $\pi_{Z^i}(a) = 1$. Let us call this a as $\bar{a}$. For such $\bar{a}$ we have $max(\mu^i(s, \bar{a}), c(\pi_{Z^i}(\bar{a}))) = max(\mu^i(s, \bar{a}), c(1)) = max(\mu^i(s, \bar{a}), 0) = \mu^i(s, \bar{a})$. Therefore we have $\mu'^i(s) \leq \mu^i(s, \bar{a}) \leq P_i(s)$, by the definition of $P_i(s)$. The fact that $U(s) = min_i \mu'^i(s)$ and that $P(s) = min_i P_i(s)$ allows us to conclude. □

Proposition 6 *Consider an uncertain FCSP* $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$*. Given two solutions* s *and* s' *of* Q*, i.e., assignments to* V_c*, if* $\forall a$ *assignments to* V_u*,* $pref(s, a) > pref(s', a)$*, then it could happen that* $s \not>_{DFP} s'$*.*

Proof: For showing this we can use the same example considered in the proof of Proposition 3 for Safe1 semantics. □

Proposition 7 *Consider an uncertain Fuzzy CSP* $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$*. Given two solutions* s *and* s' *of* Q*, if* $U(s) = U(s')$ *and* $F_P(s) > F_P(s')$*, then it could happen that* $s \not>_{DFP} s'$*.*

Proof: Let us consider any pair of solutions s and s' such that $F_P(s) > F_P(s') > U(s') = U(s)$. Since, by Theorem 1, $\forall s$, $min(F_P(s), U(s)) = min(F(s), U(s))$, and since

$min(F_P(s), U(s)) = min(F_P(s'), U(s'))$, then $s =_{DFP} s'$. $\Box$

Proposition 8 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$. Given two solutions s and s' of Q, if $F_P(s) = F_P(s')$ and $U(s) > U(s')$, then it could happen that $s \not>_{DFP} s'$.*

Proof: Let us consider for example the solutions s and s' satisfying the hypothesis, such that $P(s) = 0.9$, $F(s) = 0.3$ and $U(s) = 0.4$ and $P(s') = 0.3$, $F(s') = 0.3$ and $U(s') = 0.3$. Then $min(F(s) = 0.3, U(s) = 0.4) = 0.3 = min(F(s') = 0.3, U(s') = 0.3) = 0.3$ and so $s =_{DFP} s'$. $\Box$

Notice that many desired properties don't hold in DFP approach because, by using the *min* operator, one forgets about all the other elements, which are higher than the minimum. This is usually called the "drowning effect" [DP93].

2.10 Comparing the semantics

In this section we will compare the semantics we have considered in terms of the ordering they produce over the solutions and in terms of the properties they satisfy.

Table 2.1 shows how a pair of solutions, which is ordered in a given way by DFP, is ordered by the other semantics.

DFP	Risky	Safe	Dipl.	Risky1	Safe1
$=$	$<,>,=$	$<,>,=$	$<,>,=,\bowtie$	$<,>,=$	$<,>,=$
$>$	$<,>$	$<,>$	$>,\bowtie$	$>$	$>$

Table 2.1: The solution ordering produced by the DFP semantics compared to that of Risky, Safe, Diplomatic, Risky1 and Safe1.

The first row of Table 2.1 indicates that if a pair of solutions is equally preferred by DFP, it can be equally preferred or ordered in any way for Risky, Safe, Risky1 and Safe1, and it can also be incomparable for Diplomatic.

Example 10 Consider two solutions, s_1 and s_2 respectively with satisfaction degree and robustness $F_P(s_1) = 0.5$, $U(s_1) = 0.7$ and $F_P(s_2) = 0.7$, $U(s_2) = 0.5$, then

- $s_1 =_{DFP} s_2$;

- $s_1 <_{R,R1} s_2$;
- $s_1 >_{S,S1} s_2$;
- $s_1 \bowtie s_2$.

Consider, instead, two solutions s_3 and s_4 such $F_P(s_3) = 0.2$, $U(s_3) = 0.2$ and $F_P(s_4) = 0.5$, and $U(s_4) = 0.2$, then

- $s_4 =_{DFP,S1} s_3$;
- $s_4 >_{R,R1,S,D} s_3$.

□

The second row of Table 2.1 states that, if a a pair is ordered in some way by DFP, then it can be ordered in the same way by all the semantics, or in the opposite way in Risky and Safe, or it can be incomparable in Diplomatic.

Example 11 Consider six solutions, s_1, s_2, s_3, s_4, s_5 and s_6 such that:

- $F_P(s_1) = 0.4$, $U(s_1) = 0.3$;
- $F_P(s_2) = 0.5$, $U(s_2) = 0.4$;
- $F_P(s_3) = 0.5$, $U(s_2) = 0.3$;
- $F_P(s_4) = 0.4$, $U(s_4) = 0.4$;
- $F_P(s_5) = 0.3$, $U(s_5) = 0.8$;
- $F_P(s_6) = 0.4$, $U(s_6) = 0.7$.

Then:

- $s_1 <_{DFP,R,S,D,R1,S1} s_2$;
- $s_3 <_{DFP} s_4$ and $s_3 >_R s_4$;
- $s_5 <_{DFP} s_6$ and $s_5 >_S s_6$.

□

Notice that two solutions which are strictly ordered in DFP cannot be equally preferred with respect to Safe or Risky. In fact, two solutions are equally preferred for Safe and Risky (and Diplomatic) only if they have the same satisfaction degree and the same robustness, and thus the same minimum.

A pair ordered by DFP can either maintain its ordering or become incomparable according to Diplomatic (as shown in Table 2.1) as proved in the following proposition.

Proposition 9 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$. Given two solutions of Q, s_1 and s_2, if $s_1 >_{DFP} s_2$ then either $s_1 >_D s_2$ or $s_1 \bowtie_D s_2$.*

Proof: Assume, for the sake of contradiction, that $s_1 <_D s_2$. Thus it must be that either $F_P(s_1) \leq F_P(s_2)$ and $U_P(s_1) < U(s_2)$, or $F_P(s_1) < F_P(s_2)$ and $U_P(s_1) \leq U(s_2)$. In both cases, $min(F_P(s_1), U(s_1)) \leq min(F_P(s_2), U(s_2))$, which is in contradiction with $s_1 >_{DFP} s_2$. □

Risky1 and Safe1 are semantics which refine the ordering given by DFP, i.e., they can order tuples that are considered equal for DFP, but they never reverse the DFP ordering.

Proposition 10 *Consider an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$. Given two solutions of Q, s_1 and s_2, if $s_1 >_{DFP} s_2$ then $s_1 >_{R1,S1} s_2$.*

Proof: Since $s_1 >_{DFP} s_2$, $min(F_P(s_1), U(s_1)) > min(F_P(s_2), U(s_2))$ and thus, $s_1 >_{R1,S1} s_2$. □

Notice that from Proposition 10 it derives that the set of optimal solutions according to DFP is a superset of the set of optimal solutions of Risky1 and Safe1.

Table 2.2 summarizes which properties hold in the various semantics.

	DFP	Risky	Safe	Dipl.	Safe1	Risky1
P1	X	X	X	X	X	X
P2	X	X	X	X	X	X
P3		X				X
P4		X	X	X		X
P5		X	X	X	X	

Table 2.2: Properties satisfied in the various semantics. The presence of X in a cell (Pi, S) denotes that semantics S satisfies Property Pi.

By looking at Table 2.2 we can make the following remarks.

- Considering the first two rows, we see that the properties pertaining the definition of robustness are satisfied by all the semantics. Indeed all semantics use the same value, $U(s)$, as a measure of robustness. We recall that Property 1 states that an increase in the preferences on constraints involving uncontrollable variables results in an increase of robustness, assuming that possibility distribution is kept fixed. The same result can be obtained by lowering the possibilities while maintaining the preferences fixed (Property 2). Propositions 1 and 2 show that the definition of U, given in [DFP96a] and adopted here, satisfies these properties. However, it should be noticed that only in some semantics such changes in the preference values (respectively in the possibilities) can have a direct impact on the final preference associated to a solution s. In particular it does in $Safe$, $Risky$, $Diplomatic$ and $Safe1$, since the final preference of a solution s is a pair which contains $U(s)$. In DFP and $Risky1$, the effect of an increase of the above-mentioned preferences may be drowned by the min.

- Property 3 measures the coherence of the ordering produced by the semantics with the original one defined on the uncertain FCSP. Notice that not necessarily the absence of such coherence, as for $Safe$, $Diplomatic$, $Safe1$ and DFP should be considered as a drawback. In fact, while, at first sight, it may seem desirable to prefer an assignment to controllables which outperforms another one in every circumstance, this is not so obvious when the performance measure is the min of the preference over all the constraints. In particular, it may be reasonable to sacrifice this property in order to allow a higher discriminating power among the two fundamental aspects which are the satisfaction in term of preferences and the robustness with respect to uncertainty.

- The last two properties are satisfied when, given two solutions that have the same robustness then their ordering is determined by the satisfaction degree (Property 4) and, given two solutions that have the same satisfaction then their ordering is determined by the robustness (Property 5). Both of these properties are satisfied by $Risky$, $Safe$ and $Diplomatic$, since, such semantics consider the two features separately and independently. The other two semantics, $Risky1$ and $Safe1$, which first consider the min of the two values, allow to discriminate only with respect to the feature which appears as the second element of the pair. This can be explained in terms of a trade off between the discrimination power of the semantics and its coherence with the original ordering of the UFCSP.

2.11 An example

In this section we will present an example with some a real-life meaning that can be modelled by an uncertain FCSP and solved by applying Algorithm SP (see Section 2.4) and by choosing one of the semantics presented in Section 2.7.

Let us consider the work of a conference chair that must organize a tutorial. The tutorial must involve some hours of lectures, some of exercises and some of training. The chair must find an optimal partition of the hours for the various parts which minimizes costs and takes into account the requirements of all the involved teachers and students, and several other constraints. For example, he must establish the number of training hours, knowing that they require very expensive rooms to book and that they can contain only a certain number of students for hour, but without knowing the definite number of the students that will attend them.

In the following we will present in detail the constraints and the requirements that the chair has to consider. We will show that the whole problem can be modelled by an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C = C_f \cup C_{fu} \rangle$. Notice that we model the constraints as fuzzy constraints since the chair wants to find the solution that maximizes the minimum preference of all the people involved in the tutorial.

- The tutorial must involve some hours of lectures, some hours of exercises and some hours of training. Then V_c, the set of the controllable variables that the conference chair can decide, contains the variables x, which represents the number of lectures hours, y which stands for the exercises hours and w, which represents the number of training hours.

- A requirement of the conference is that the various courses (lectures, exercises, training) must last 10, 20 or 30 hours. Hence the domains of the variables x, y and w are $D_x = D_y = D_w = \{10, 20, 30\}$.

- Tutorial hours must be performed in rooms that can contain comfortably only a certain number of people. However, the conference chair doesn't know the definite number of students that will attend them. Therefore V_u, the set of uncontrollable variables, contains z, that is the number of students that can attend training hours.

- The conference chair has received 90 student registrations, hence he knows that at most there will 90 students, and so that their definite number can be between 0 and 30, between 30 and 60 and between 60 and 90. He believes that it's more possible that students will be between 30 and 60. In fact, he thinks that not all of them will refuse to come, since they have paid the registration, but maybe not all of them will come, since there are other interesting conferences in that period. This fact can be modelled giving to the uncontrollable variable z three values in its domain that are few (between 0 and 30), $average$ (between 30 and 60) and $many$ (between 60 and 90) and associating them with the following possibilities: $\pi(few) = 0.4$, $\pi(average) = 1$, $\pi(many) = 0.3$.

- C_f, the set of constraints defined only on controllable variables, is composed by the following fuzzy constraints.

  - The lecture professor prefers to teach for many hours, since he wants to earn much money. This can be modelled by the fuzzy constraint $c_1 =< \mu_1, \{x\} >$, where $\mu_1(10) = 0.2$, $\mu_1(20) = 0.9$ and $\mu_1(30) = 1$.
  - The exercise professor prefers to teach for few hours, since he is very busy in that period. This can be described by the fuzzy constraint $c_2 =< \mu_2, \{y\} >$, where $\mu_2(10) = 0.9$, $\mu_2(20) = 0.4$ and $\mu_2(30) = 0.1$.
  - The training hours must be done in laboratory rooms, that are expensive, hence the conference chair, that wants to reduce costs, prefers to reduce these hours: $c_3 =< \mu_3, \{w\} >$, where $\mu_3(10) = 0.6$, $\mu_3(20) = 0.5$ and $\mu_3(30) = 0.3$.
  - A requirement of the conference is that the tutorial lasts at most 50 hours. Let x', y' and w' the values of the variables x, y and z, then this can be represented by the fuzzy constraint $c_4 =< \mu_4, \{x, y, w\} >$, where $\mu_4(x', y', z') = 1$ if $x' + y' + z' \leq 50$ and $\mu_4(x', y', z') = 0$ if $x' + y' + z' > 50$.
  - The students prefer to attend many hours of lessons for learning better the various subjects. This can be modelled by fuzzy constraint $c_5 =< \mu_5, \{x, y, w\} >$, where $\mu_5(x', y', z') = 0.4$ if $x' + y' + z' = 30$, $\mu_5(x', y', z') = 0.5$ if $x' + y' + z' = 40$ and $\mu_5(x', y', z') = 1$ if $x' + y' + z' = 50$.
  - Moreover, the students prefer to attend many hours of exercises for learning better the theory. We can model this requirement with the fuzzy constraint $c_6 =< \mu_6, \{y\} >$, where $\mu_6(10) = 0.7$, $\mu_6(20) = 0.8$ and $\mu_6(30) = 1$.

- C_{fu}, the set of constraints involving both controllable and uncontrollable variables, is composed by the following fuzzy constraint.

– The laboratory rooms can contain for every hour comfortably 30 $(= few)$ people. We can model this requirement with the constraint $c_7 = < \mu_7, \{w, z\} >$, where $\mu_7(10, few) = 1$, $\mu_7(10, average) = 0.5$, $\mu_7(10, many) = 0.1$, $\mu_7(20, few) = 0.1$, $\mu_7(20, average) = 0.9$, $\mu_7(20, many) = 0$, $\mu_7(30, few) = 0.1$, $\mu_7(30, average) = 0.2$ and $\mu_7(30, many) = 0.8$.

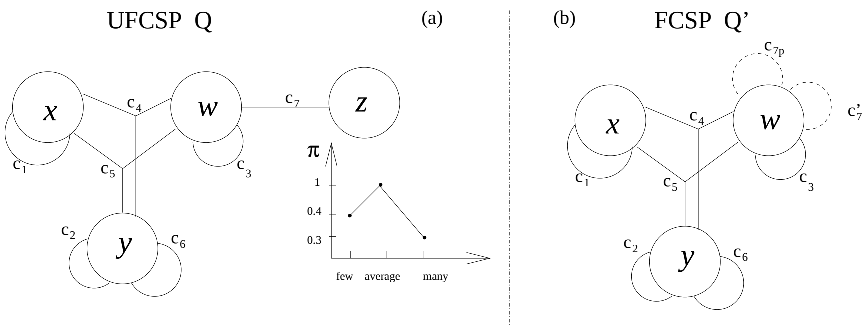

Figure 2.7: An uncertain FCSP $Q = \langle S_{FCSP}, V_c = \{x, y, w\}, V_u = \{z\}, C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7\}\rangle$ and the corresponding FCSP $Q' = \langle S_{FCSP}, V_c = \{x, y, w\}, C = \{c_1, c_2, c_3, c_4, c_5, c_6, c'_7, c_{7p}\}\rangle$ obtained by applying SP.

Applying SP to the uncertain FCSP shown in Figure 2.7 (a) we obtain the FCSP $Q' = \langle S_{FCSP}, V_c, C'\rangle$ shown in Figure 2.7 (b) with $C' = C_f \cup C_p \cup C_u$, where:

- C_f, the set of constraints of Q defined only on V_c, is $\{c_1, c_2, c_3, c_4, c_5, c_6\}$;
- C_p, the set of constraints obtained by projecting the constraints involving variables in V_u on their controllable variables, is composed only by $c_{7p} = < \mu_{7p}, \{w\} >$, where $\mu_{7p}(10) = 1$, $\mu_{7p}(20) = 0.9$ and $\mu_{7p}(30) = 0.8$;
- C_u, the set of constraints defined on V_c, obtained applying the procedure for removing uncontrollable variables described in Section 2.2.4, is composed by only $c'_7 =< \mu'_7, \{w\} >$, where $\mu'_7(10) = 0.5$, $\mu'_7(20) = 0.6$ and $\mu'_7(30) = 0.2$.

Now we can compute, for each complete assignment $s = (x = x', y = y', w = w')$ to V_c that doesn't violate any constraints, the satisfaction level $F_P(s)$ and the robustness value $U(s)$. We recall that $F_P(s) = min(F(s), P(s))$ and $F(s)$, $P(s)$, $U(s)$, are respectively the minimum preference over the constraints in C_f, C_p and C_u of Q'. Hence we have the following solutions:

- $s_1 = (10, 10, 10)$ with $F_P(s_1) = 0.2$ and $U(s_1) = 0.5$,

- $s_2 = (10, 10, 20)$ with $F_P(s_2) = 0.2$ and $U(s_2) = 0.6$,
- $s_3 = (10, 20, 10)$ with $F_P(s_3) = 0.2$ and $U(s_3) = 0.5$,
- $s_4 = (20, 10, 10)$ with $F_P(s_4) = 0.5$ and $U(s_4) = 0.5$,
- $s_5 = (10, 20, 20)$ with $F_P(s_5) = 0.2$ and $U(s_5) = 0.6$,
- $s_6 = (20, 20, 10)$ with $F_P(s_6) = 0.4$ and $U(s_6) = 0.5$,
- $s_7 = (20, 10, 20)$ with $F_P(s_7) = 0.5$ and $U(s_7) = 0.6$,
- $s_8 = (10, 30, 10)$ with $F_P(s_8) = 0.1$ and $U(s_8) = 0.5$,
- $s_9 = (30, 10, 10)$ with $F_P(s_9) = 0.6$ and $U(s_9) = 0.5$,
- $s_{10} = (10, 10, 30)$ with $F_P(s_{10})) = 0.2$ and $U(s_{10}) = 0.2$.

The optimal solutions for the semantics presented in Section 2.7 are:

- $s_9 = (30, 10, 10)$ for Risky and Risky1 semantics, that associate to each solution s respectively the pair $\langle F_P(s), U(s) \rangle$ and $\langle min(F_P(s), U(s)), F_P(s) \rangle$;
- $s_7 = (20, 10, 20)$ for Safe and Safe1, that associate to each solution s respectively the pair $\langle U(s), F_P(s) \rangle$ and $\langle min(F_P(s), U(s)), U(s) \rangle$;
- $s_7 = (20,\ 10,\ 20)$ and $s_9 = (30,\ 10,\ 10)$ for Diplomatic.

This result shows that if the conference chair is Risky or Risky1, for saving money, he proposes an optimal subdivision of the hours that doesn't cost very much, but that is risky. In fact, if there will be an average or a big number of students, he will not be able to allow to every student to participate to the training hours even if they have paid for them. Whereas if he is Safe or Safe1 he prefers to guarantee the training hours to an average number of students, and so he will pay laboratory rooms for more hours, even if this increases the cost to pay. If he is Diplomatic, he would like to find a solution that is not expensive and that guarantees tutorial hours to almost all students that will come. Hence he considers incomparable and so equally optimal the solutions decided for Risky and Safe semantics.

Notice that in general Safe and Safe1 (respectively Risky and Risky1) don't give the same ordering over solutions. Consider for example the solutions $s_2 = (10, 10, 20)$ and $s_6 = (20, 20, 10)$. We have $s_2 >_S s_6$ for Safe, since $U(s_2) = 0.6 > U(s_6) = 0.5$, whereas $s_2 <_{S1} s_6$ for Safe1, since $min(F_P(s_2) = 0.2, U(s_2) = 0.6) = 0.2 < min(F_P(s_6) = 0.6, U(s_6) =$

$0.5) = 0.5$. Safe states that $s_2 > s_6$, since it prefers the most robust solution independently from its satisfaction degree, i.e., independently by the preference of the controllable part, whereas Safe1 states the opposite since s_2 has a poor overall preference (on both controllable and uncontrollable variables) with respect to s_6, despite guaranteeing high preference on the constraints involving uncontrollable variables.

Summarizing, the various semantics propose different optimal solutions depending by different attitudes to the risk with respect to uncertainty. This attitude is very different from the one of DFP that considers equally optimal solutions $s_4 = (20, 10, 10)$, $s_7 = (20, 10, 20)$ and $s_9 = (30, 10, 10)$ even if, as we have just shown, they have very different meanings.

2.12 A solver for UFCSPs

We present a solver for finding an optimal solution of an uncertain FCSP according to the semantics defined in Section 2.7. Since the solver is based on a branch and bound approach, first we briefly describe how standard Branch and Bound (that we call BB) for FCSPs works.

FCSPs are NP-hard, since they are a particular case of soft constraints problems, which are already known to be difficult problems [BMR97]. However, they can be solved via a branch and bound technique, possibly augmented via soft constraint propagation, which may lower the preferences and thus allow for the computation of better bounds [BMR97]. Following BB [Dec03], whenever a solution is found, its preference, if higher that those found before, is kept as a lower bound, L, for the optimal preference in the maximization task. Moreover, for each partial solution t an upper bound, $ub(t)$, is computed by overestimating the best preference of a solution extending t. If $ub(t) \leq L$, i.e., if the preference of the best solution in the subtree below t is worse than the preference of the best solution found so far, then the subtree below t is pruned.

We propose to use an algorithm similar to BB [Dec03] used for fuzzy preferences, but that it is based on our framework. More precisely, we will associate to every solution s not a single value, like in standard BB, but an ordered pair of values, that is composed by the satisfaction level $F_P(s)$ and the robustness $U(s)$ and then we will use different semantics for comparing solutions. Hence, also the upper bound $ub(t)$ of a partial solution t will be given by a pair of values, that we call $ubF_P(t)$ and $ubU(t)$, that are respectively overestimations of the best level of satisfaction and of the best value of robustness of a solution extending t.

Since we consider fuzzy preferences, these overestimations are computed as follows: $ubF_P(t) = min(F_{P_i}(t), F_{P_{ni}}(t))$ and $ubU(t) = min(U_i(t), U_{ni}(t))$, where $F_{P_i}(t)$ and $ubU_i(t)$ are respectively the satisfaction level and the robustness value of the part of the problem instantiated with t, that we know, while $F_{P_{ni}}(t)$ and $ubU_{ni}(t)$ are overestimations respectively of the satisfaction level and of the robustness value of the non-instantiated part of problem, that we have to compute.

In some semantics, given the partial solution t, we can prune the subtree below t, without computing the overestimations $F_{P_{ni}}(t)$ and $ubU_{ni}(t)$ mentioned above, but considering only the satisfaction level and the robustness value of the part of the problem instantiated with t, i.e., by using only $F_{P_i}(t)$ and $U_i(t)$. For example, assume that the best solution found so far, s, is associated with values F_{P_*} and U_*. If we consider Risky semantics, then for pruning the subtree below t it is sufficient that $F_{P_i}(t) < F_{P_*}$, since this implies that $ubF_P(t) \leq F_{P_i}(t) < F_{P_*}$, and so that solutions in the subtree below t are worse for Risky semantics than the best solution found so far. Analogously, if we reason with Safe semantics, we can prune the subtree below t only knowing that $U_i(t) < U_*$.

In order to state in compact way the conditions that allow to prune the subtree below a partial assignment t in the various semantics, we use the ordered pair $\langle a(t), b(t)\rangle$ for representing the values associated to t. This pair $\langle a(t), b(t)\rangle$ is $\langle F_P(t), U(t)\rangle$ in Risky and Diplomatic, $\langle U(t), F_P(t)\rangle$ in Safe, $\langle min(F_P(t), U(t)),\ F_P(t)\rangle$ in Risky1 and $\langle min(F_P(t), U(t)),\ U(t)\rangle$ in Safe1. Moreover, we use index $_i$ (i.e., $\langle a_i(t), b_i(t)\rangle$), when we refer to the part of the problem instantiated with t, and index $_{ni}$ (i.e., $\langle a_{ni}(t), b_{ni}(t)\rangle$), for referring to non-instantiated part.

More formally, assume that the best solution found so far is associated with the pair $\langle lb_a, lb_b\rangle$, and that the partial assignment t is defined by the pair $\langle a_i(t), b_i(t)\rangle$. Then, in our branch and bound algorithm (Algorithm 3), we prune the subtree below t, if one of the following conditions holds:

1. $a_i(t) < lb_a$;
2. $a_i(t) \geq lb_a$ and $a_{ni}(t) < lb_a$;
3. $a_i(t) \geq lb_a$, $a_{ni}(t) = lb_a$ and $b_i(t) < lb_b$;
4. $a_i(t) \geq lb_a$, $a_{ni}(t) = lb_a$, $b_i(t) = lb_b$ and $b_{ni}(t) \leq lb_b$.

Our branch and bound algorithm is similar to the standard one except that at every step, instead of consider a single value for every partial assignment, it considers a pair of values and it uses the various semantics for ordering these pairs. For a tight comparison between our algorithm and standard BB, first we present standard BB and then we underline the main novelties of our algorithm. The standard BB is described in Algorithm 2. It takes as input an assignment t and a valuation lb, that is the the preference of the best solution found so far, and it returns a valuation. At first it sets the value v to the upper bound $ub(t)$, that is an overestimation of the best solution that can be found in the subtree under t. Hence if this overestimation is worse than or equal to lb, then the algorithm doesn't consider subtree below t and it returns the worse element of the fuzzy c-semiring, i.e., 0. Instead, if $v > lb$, then if the cardinality of t is equal to n, i.e., if t is complete assignment, then it returns v. If this does not happen it considers a non-instantiated variable k, and, for every value a in its domain, it computes the new lower bound lb. This value is obtained by performing the maximum between the previous value of lb and the valuation returned by a recursive call of the algorithm, that takes as parameters the assignment that extends t with the value a, that is assigned to the variable k, and the present value of lb. At the end it returns the value lb.

Algorithm 2: Branch and Bound Algorithm, BB

```
Input: t: assignment; lb: valuation;
Output: valuation;
v ← ub(t);
if v > lb then
    if |t| = n then
        return v;
    Let k be the future variable;
    foreach a ∈ d_k do
        lb = max(lb, BB(t ∪ {(k, a)}, lb));
    return lb;
return 0;
```

We propose to adapt Algorithm 2 to our framework, thus obtaining Algorithm 3, that takes in input another parameter, i.e., S, that is the considered semantics, and it associates to every assignment, not only a single value, but a pair of values. Therefore the values lb, $ub(t)$ and v are pairs of values in Algorithm 3. Moreover, instead of checking if $v > lb$ is true, the algorithm checks if none of the pruning conditions described previously in this section is satisfied. Moreover, since now we have pairs of values and not single values to compare, we replace the operator max with a new operator, that we call $best_S$, that compares two pairs and returns the best one according to the semantics S. This algorithm could be used with all our semantics.

Notice that, in Diplomatic semantics, the pruning conditions defined previously lead always to optimal solutions that are optimal solutions in Risky or in Safe semantics. Since the main feature of Diplomatic semantics is to produce also optimal solutions that neither Risky

Algorithm 3: Our Branch and Bound Algorithm

Input: t: assignment; $\langle lb_a, lb_b \rangle$: pair of valuations; S: semantics

Output: pair of valuations;

$\langle v_a, v_b \rangle \leftarrow \langle min(a_i(t), a_{ni}(t)), min(b_i(t), b_{ni}(t)) \rangle$;

if ***(($a_i(t) < lb_a$) and ($b_i(t) < lb_b$)) or***
(($a_i(t) < lb_a$) and ($b_i(t) \geq lb_b$) and ($b_{ni}(t) < lb_b$)) or
(($a_i(t) \geq lb_a$) and ($a_{ni}(t) < lb_a$) and ($b_i(t) < lb_b$)) or
(($a_i(t) \geq lb_a$) and ($a_{ni}(t) < lb_a$) and ($b_i(t) \geq lb_b$) and ($b_{ni}(t) \leq lb_b$)) **then**

  return $\langle 0, 0 \rangle$;

else

  if $|t| = n$ **then**

    return $\langle v_a, v_b \rangle$;

  Let i be the future variable;

  foreach $a \in d_k$ **do**

    $\langle lb_a, lb_b \rangle = best_S(\langle lb_a, lb_b \rangle, BB(t \cup \{(k, a)\}, \langle lb_a, lb_b \rangle, S)$;

  return $\langle lb_a, lb_b \rangle$;

nor Safe semantics give, then we state new pruning conditions that produce this particular kind of Diplomatic optimal solutions. These conditions are more tight and so they allow us for less pruning. Given a partial assignment t with values $< a_i(t), b_i(t) >$, for obtaining Diplomatic optimal solutions that are neither Risky nor Safe optimal solutions, then we can adapt the branch and bound algorithm allowing to prune the subtree below t only if one of the following conditions holds:

1. $a_i(t) \leq lb_a$ and $b_i(t) \leq lb_b$;

2. $a_i(t) \leq lb_a$, $b_i(t) > lb_b$ and $b_{ni}(t) \leq lb_b$;

3. $a_i(t) > lb_a$, $a_{ni}(t) \leq lb_a$ and $b_i(t) \leq lb_b$;

4. $a_i(t) > lb_a$, $a_{ni}(t) \leq lb_a$, $b_i(t) > lb_b$ and $b_{ni}(t) \leq lb_b$.

In order to find this kind of Diplomatic optimal solutions, we have defined another algorithm (Algorithm 4), that is equal to Algorithm 3 except that the semantics considered here is only Diplomatic semantics and the pruning conditions in the first IF are those that we have mentioned just before.

Algorithm 4: Special Diplomatic Branch and Bound Algorithm

Input: t: assignment; $\langle lb_a, lb_b \rangle$: pair of valuations; D: Diplomatic semantics

Output: pair of valuations;

$\langle v_a, v_b \rangle \leftarrow \langle min(a_i(t), a_{ni}(t)), min(b_i(t), b_{ni}(t)) \rangle$;

if *(($a_i(t) \leq lb_a$)* **and** *($b_i(t) \leq lb_b$))* **or**

(($a_i(t) \leq lb_a$) **and** *($b_i(t) > lb_b$)* **and** *($b_{ni}(t) \leq lb_b$;))* **or**

(($a_i(t) > lb_a$) **and** *($a_{ni}(t) \leq lb_a$)* **and** *($b_i(t) \leq lb_b$))* **or**

(($a_i(t) > lb_a$) **and** *($a_{ni}(t) \leq lb_a$)* **and** *($b_i(t) > lb_b$)* **and** *($b_{ni}(t) \leq lb_b$))* **then**

  return $\langle 0, 0 \rangle$;

else

  if $|t| = n$ **then**

    return $\langle v_a, v_b \rangle$;

  Let i be the future variable;

  foreach $a \in d_k$ **do**

    $\langle lb_a, lb_b \rangle = best_S(\langle lb_a, lb_b \rangle, BB(t \cup \{(k, a)\}, \langle lb_a, lb_b \rangle, S)$;

  return $\langle lb_a, lb_b \rangle$;

2.13 A generalized approach for USCSPs

In Sections 2.4 and 2.5 we have defined a method for handling Fuzzy CSPs with uncertainty. In this section we propose a method for generalizing this procedure in order to deal with other classes of Soft CSPs. First, we generalize algorithm SP for generic soft CSPs with uncertainty by extending the notion of robustness in this context, and then we show that it satisfies the desired properties presented in Section 2.3. Next, we show how to compute the preference of a solution in this general framework and we give more general semantics for ordering the solutions. After that, we instantiate the general framework with two well known SCSPs, to which we add uncertainty, that are, Probabilistic CSPs, where preferences are interpreted as probabilities and the goal is to maximize their product and Weighted CSPs, where preferences represent costs and the goal is to minimize their sum. Finally, we check if the solution ordering produced by the new more general semantics continues to satisfy the desired properties on the solution ordering presented in Section 2.3.

2.13.1 Algorithm G-SP for USCSPs

In this section we show how to generalize to generic uncertain soft CSPs the algorithm SP described in Section 2.2.4.

We recall that SP translates fuzzy constraints linking controllable and uncontrollable variables into fuzzy constraints involving only their controllable variables. In particular, given an uncertain FCSP $Q = \langle S_{FCSP}, V_c, V_u, C\rangle$, where S_{FCSP} is the fuzzy c-semiring $\langle [0,1], max, min, 0, 1\rangle$, SP translates every constraint $\langle \mu, con\rangle$, where $con \cap V_c = X$ and $con \cap V_u = Z$, into a new constraint $\langle \mu', con'\rangle$, where $con' = X$ and μ' is such that, $\forall d$ assignment to X, $\mu'(d) = inf_{a \in A_Z}(\mu(d,a) + c(\pi_Z(a)))$, where c is the order reversing map in $[0,1]$, such that $c(p) = 1 - p$, $\forall p \in [0,1]$, A_Z is the cartesian product of the domains of variables in Z and π_Z is the possibility distribution on Z. The preference function μ' is characterized by a property stating that, given an assignment d to X, $\mu'(d) \geq \alpha$ if and only if $\forall a$ with $\pi_Z(a) > c(\alpha)$, $\mu(d,a) \geq \alpha$.

In the following we give a more general definition of μ' that holds in uncertain soft CSPs. In order to show that it is reasonable, we show that the characterization property for μ' continues to hold also in this general framework. Let us consider an uncertain soft CSP $SQ = \langle S, V_c, V_u, C\rangle$, where S is a generic c-semiring $\langle A, +, \times, 0, 1\rangle$ and $\leq_S$ is the c-semiring ordering on A induced by the additive operator of S. We generalize the preference function of every constraint $\langle \mu', con'\rangle$ described above, obtained by removing uncontrollable variables, as follows. Notice that $+$ refers to the additive operator of the c-semiring.

$$\mu'(d) = inf_{a \in A_Z}(\mu(d,a) + c_S(\pi_Z(a))) \tag{2.2}$$

where

- inf returns one of the bottom elements of A_Z with respect to the c-semiring ordering (i.e., $a \in A_Z$ such that $\forall a' \in A_Z$ with $a' \neq a$ then $a' >_S a$ or $a' \bowtie_S a$, where $\bowtie_S$ means incomparable with respect to c-semiring ordering);

- $[0,1] \subseteq A$. This allows us to apply the operator $+$ between $\mu(d,a)$ and $c_S(\pi_Z(a))$;

- c_S is a order-reversing map with respect to c-semiring S, that is bijection from [0,1] to [0,1] such that, $\forall a_1, a_2 \in [0,1]$, $a_1 \leq a_2$ if and only if $c_S(a_1) \geq_S c_S(a_2)$ and $c_S(c_S(a)) = a$, $\forall a$.

In what follows we show that the new preference function μ' defined for uncertain soft CSPs satisfies the characterization property given for uncertain FCSPs. More precisely, we show that, if the set of preferences, A, of the c-semiring S is totally ordered, the same characterization of μ' holds, while, if A is partially ordered, a slightly weaker characterization of μ', that depends on the fact that preferences of A can be also incomparable, holds.

Proposition 11 *Consider an uncertain soft CSP $SQ = \langle S, V_c, V_u, C\rangle$, where S is a generic c-semiring $\langle A, +, \times, 0, 1\rangle$. Every constraint, $\langle \mu, con\rangle \in C$, such that $con \cap V_c = X$ and $con \cap V_u = Z$, with possibility distribution π_Z, can be translated in a new constraint, $\langle \mu', con'\rangle$, where $con' = X$ and μ' is such that, if d is an assignment to X, and a an assignment to Z,*

- *if* A is totally ordered,
  $\mu'(d) \geq_S \alpha$ *if and only if,* $\forall a$ *such that* $\pi_Z(a) > c_S(\alpha)$*,* $\mu(d, a) \geq_S \alpha$*.*
- *if* A is partially ordered,
  $\mu'(d) \not\leq_S \alpha$ *if and only if,* $\forall a$ *such that* $\pi_Z(a) \geq c_S(\alpha)$*,* $\mu(d, a) \not\leq_S \alpha$
  ($\not\leq_S$ *means* $>_S$ *or* $\bowtie_S$*).*

where c_S is an order reversing map with respect to c-semiring S.

Proof: Let us recall that $\mu'(d) = inf_{a\in A_Z}(\mu(d, a) + c_S(\pi_Z(a)))$.

A totally ordered. (⇒) We assume $inf_{a\in A_Z}(\mu(d, a) + c_S(\pi_Z(a))) \geq_S \alpha$. Since $inf_{a\in A_Z}(\mu(d, a)+c_S(\pi_Z(a))) \leq_S (\mu(d, a)+c_S(\pi_Z(a)))$, $\forall a \in A_Z$, then $(\mu(d, a)+c_S(\pi_Z(a))) \geq_S \alpha$, $\forall a \in A_Z$. For every a such that $\pi_Z(a) > c_S(\alpha)$, then, since c_S is an order-reversing map with respect to c-semiring S, such that $c_S(c_S(p)) = p$, we have $c_S(\pi_Z(a)) <_S c_S(c_S(\alpha)) = \alpha$. Since A is totally ordered, for any two elements of the c-semiring we have $a + b = a$ or b, then, for every a such that $\pi_Z(a) > c_S(\alpha)$, we have $\mu(d, a) = (\mu(d, a) + c_S(\pi_Z(a))) \geq_S \alpha$.

(⇐) We assume that, for every a such that $\pi_Z(a) > c_S(\alpha)$, we have $\mu(d, a) \geq_S \alpha$. Then, for such a, $(\mu(d, a) + c_S(\pi_Z(a))) \geq_S \mu(d, a) \geq_S \alpha$. On the other hand, for every a such that $\pi_Z(a) \leq c_S(\alpha)$, we have $c_S(\pi_Z(a)) \geq_S \alpha$ and so $(\mu(d, a) + (\pi_Z(a))) \geq_S \alpha$. Thus for every $a \in A_Z$, $(\mu(d, a) + c_S(\pi_Z(a))) \geq_S \alpha$. Therefore, since the inf operator applied to elements of the c-semiring returns one of these elements, we have that $inf_{a\in A_Z}(\mu(d, a) + c_S(\pi_Z(a))) \geq_S \alpha$, i.e., $\mu'(d) \geq_S \alpha$.

A partially ordered. (⇒) We assume that $inf_{a\in A_Z}(\mu(d, a) + c_S(\pi_Z(a))) \not\leq_S \alpha$, then $(\mu(d, a) + c_S(\pi_Z(a))) \not\leq_S \alpha$, $\forall a \in A_Z$. For every a such that $\pi_Z(a) \geq c_S(\alpha)$ we have, as above, $c_S(\pi_Z(a)) \leq_S \alpha$ and so, since $(\mu(d, a) + c(\pi_Z(a))) \not\leq_S \alpha$, we have $\mu(d, a) \not\leq_S \alpha$. In fact, if we assume $\mu(d, a) \leq_S \alpha$, then we obtain $(\mu(d, a)+c(\pi_Z(a))) \leq_S \alpha+\alpha$, for monotonicity of $+$, and, for idempotency of $+$, $(\mu(d, a) + c(\pi_Z(a))) \leq_S \alpha$, that is a contradiction.

(⇐) We assume that, for every a such that $\pi_Z(a) \geq c(\alpha)$, $\mu(d, a) \not\leq_S \alpha$. Then, for such a, we have $(\mu(d, a)+c(\pi_Z(a))) \not\leq_S \alpha$. In fact, if $(\mu(d, a)+c(\pi_Z(a))) \leq_S \alpha$, then, for monotonicity and idempotency of $+$, we have $\mu(d, a) \leq_S (\mu(d, a) + c(\pi_Z(a))) \leq_S \alpha + \alpha = \alpha$,

that is a contradiction. On the other hand, for every a such that $\pi_Z(a) < c(\alpha)$, we have $c(\pi_Z(a)) >_S \alpha$ and so $(\mu(d,a) + c(\pi_Z(a))) \geq_S c(\pi_Z(a)) >_S \alpha$. Thus for every $a \in A_Z$, $(\mu(d,a) + c(\pi_Z(a))) \not\leq_S \alpha$. Therefore, since the inf operator applied to elements of the c-semiring returns one of these elements, we have that $inf_{a \in A_Z}(\mu(d,a) + c(\pi_Z(a))) \not\leq_S \alpha$.
□

Now we are ready to show how algorithm generalizing SP, that we call *G-SP*, works. It starts from an uncertain soft CSP $SQ = \langle S, V_c, V_u, C = C_f \cup C_{fu} \rangle$, where S is a generic c-semiring $\langle A, +, \times, 0, 1 \rangle$, $\leq_S$ is the c-semiring ordering on A induced by the additive operator of S, C_f is the set of constraints of SQ defined only on controllable variables, C_{fu} is the set of constraints of SQ defined on both controllable and uncontrollable variables. Then, it obtains a soft CSP $SQ' = \langle S, V_c, C' = C_{control} \cup C_u \rangle$, where $C_{control} = C_f \cup C_p$, C_p is the set of constraints obtained by projecting (as described in Definition 5 of Section 2.2) the constraints C_{fu} on the controllable variables and C_u is the set of constraints, defined only on controllable variables, obtained from the constraints C_{fu} applying the method described at the beginning of this section.

2.13.2 Desired properties on robustness

We will now show that the new value U, which is computed by exploiting Formula 2.2, satisfies the properties on the robustness presented in Section 2.3. As done in Section 2.13.1, we will distinguish the cases where the set of preferences of the c-semiring is totally ordered and where it is partially ordered, and we will give slightly modified properties in this second case, in order to take incomparability into account.

Proposition 12 *Consider two uncertain soft CSPs:* $SQ_1 = \langle S, V_c, V_u, C_1 = C_{f_1} \cup C_{fu_1} \rangle$ *and* $SQ_2 = \langle S, V_c, V_u, C_2 = C_{f_2} \cup C_{fu_2} \rangle$, *where* S *is a generic c-semiring* $\langle A, +, \times, 0, 1 \rangle$, C_1 *and* C_2 *differ only by the preference functions of constraints involving variables in* V_u, *i.e.,* $C_{f_1} = C_{f_2}$, $C_{fu_1} = \bigcup_i \langle \mu_1^i, con^i \rangle$ *and* $C_{fu_2} = \bigcup_i \langle \mu_2^i, con^i \rangle$. *In particular, for every such constraint,* $c^i = \langle \mu^i, con^i \rangle$, *such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$, *with possibility distribution* π_{Z^i}, *let* $\mu_1^i(d,a) \leq_S \mu_2^i(d,a)$, *for all* a *assignments to* Z^i *and for all* d *assignments to* X^i. *Then, given solution* s *of* SQ_1 *and* SQ_2, *such that* $s \downarrow_{X^i} = d$,

- *if* A is totally ordered, $U_1(s) \leq_S U_2(s)$;
- *if* A is partially ordered, $U_1(s) \not>_S U_2(s)$.

Proof: We recall that, for every constraint $c^i = \langle \mu^i, con^i \rangle$ in the statement of this proposition, $\mu_1'^i(d) = inf_{a \in A_{z^i}} (\mu_1^i(d,a) + c_S(\pi_{Z^i}(a)))$ and $\mu_2'^i(d) = inf_{a \in A_{z^i}} (\mu_2^i(d,a) + c_S(\pi_{Z^i}(a)))$, where A_{z^i} is the Cartesian product of the domains of the variables in Z^i. Since $\mu_1^i(d,a) \leq_S \mu_2^i(d,a)$, $\forall a, d$, then, by monotonicity of $+$, $(\mu_1^i(d,a) + c_S(\pi_{Z^i}(a))) \leq_S \mu_2^i(d,a) + c_S(\pi_{Z^i}(a)))$, $\forall a, d$.

If A *totally ordered,* then we have $inf_{a \in A_{z^i}} (\mu_1^i(d,a) + c_S(\pi_{Z^i}(a))) \leq_S (\mu_1^i(d,a) + c_S(\pi_{Z^i}(a)))$ $\leq_S (\mu_2^i(d,\ a) + \ c_S(\pi_{Z^i}(a)))$, $\forall a$, $\forall d$. Therefore, since $s \downarrow_{X^i} = d$, $\mu_1'^i(s \downarrow_{X^i}) = inf_{a \in A_{z^i}}$ $(\mu_1(s \downarrow_{X^i}, a) + c_S(\pi_{Z^i}(a))) \leq_S inf_{a \in A_{z^i}} (\mu_2(s \downarrow_{X^i}, a) + c_S(\pi_{Z^i}(a))) = \mu_2'^i(s \downarrow_{X^i})$. The fact that $U_1(s) = \prod_i \mu_1'^i(s \downarrow_{X_i})$ and $U_2(s) = \prod_i \mu_2'(s \downarrow_{X_i})$, and the monotonicity of $\times$ allow to conclude.

If A *is partially ordered,* then $\mu_1'^i(s \downarrow_{X^i}) = inf_{a \in A_{z^i}} (\mu_1(s \downarrow_{X^i},\ a) + \ c_S(\pi_{Z^i}(a))) \not>_S$ $inf_{a \in A_{z^i}} (\mu_2(s \downarrow_{X^i},\ a) + \ c_S(\pi_{Z^i}(a))) = \mu_2'^i(s \downarrow_{X^i})$. In fact, if we assume that $inf_{a \in A_{z^i}}$ $(\mu_1(s \downarrow_{X^i},\ a) + \ c_S(\pi_{Z^i}(a))) >_S inf_{a \in A_{z^i}} (\mu_2(s \downarrow_{X^i},\ a) + \ c_S(\pi_{Z^i}(a))) = \mu_2'^i(s \downarrow_{X^i})$, then $\exists \bar{a} \in A_Z$, $inf_{a \in A_{z^i}} (\mu_1(s \downarrow_{X^i}, a) + \ c_S(\pi_{Z^i}(a))) >_S (\mu_2(s \downarrow_{X^i},\ \bar{a}) + \ c_S(\pi_{Z^i}(\bar{a})))$, since the operator inf returns one of the elements on which it is applied. But we know that $(\mu_1(s \downarrow_{X^i}, a) + c_S(\pi_{Z^i}(a))) \leq_S (\mu_2(s \downarrow_{X^i}, a) + c_S(\pi_{Z^i}(a)))$, $\forall a$, and so, $inf_{a \in A_{z^i}} (\mu_1(s \downarrow_{X^i},$ $a) + \ c_S(\pi_{Z^i}(a))) >_S (\mu_2(s \downarrow_{X^i},\ \bar{a}) + \ c_S(\pi_{Z^i}(\bar{a}))) \geq_S (\mu_1(s \downarrow_{X^i},\ \bar{a}) + \ c_S(\pi_{Z^i}(\bar{a})))$, that is a contradiction, since we find that $inf_{a \in A_{z^i}} (\mu_1(s \downarrow_{X^i}, a) + c_S(\pi_{Z^i}(a))) >_S (\mu_1(s \downarrow_{X^i}, \bar{a}) +$ $c_S(\pi_{Z^i}(\bar{a})))$. The fact that $U_1(s) = \prod_i \mu_1'^i(s \downarrow_{X_i})$ and $U_2(s) = \prod_i \mu_2'(s \downarrow_{X_i})$, and the monotonicity of $\times$ allow to conclude. $\square$

Proposition 13 *Consider two uncertain soft CSPs:* $SQ_1 = \langle S, V_c, V_u, C \rangle$ *and* $SQ_2 = \langle S, V_c, V_u', C \rangle$*, where* S *is a generic c-semiring* $\langle A, +, \times, 0, 1 \rangle$*,* V_u *and* V_u' *are the same set of uncontrollable variables described, however, by different possibility distributions. In particular, for every constraint,* $c^i = \langle \mu^i, con^i \rangle$*, such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$*, let* $\pi^1_{Z^i}(a) \geq \pi^2_{Z^i}(a)$*, for all* a *assignments to* Z^i*. Then, given solution* s *of* SQ_1 *and* SQ_2*, such that* $s \downarrow_{X^i} = d$*,*

- *if* A is totally ordered, $U_1(s) \leq_S U_2(s)$*;*
- *if* A is partially ordered, $U_1(s) \not>_S U_2(s)$.

Proof: For every constraint $c^i = \langle \mu^i, con^i \rangle$ in the statement of this proposition we have $\mu_1'^i(d) = inf_{a \in A_{z^i}} (\mu^i(d,a) + \ c_S(\pi^1_{Z^i}(a)))$ and $\mu_2'^i(d) = inf_{a \in A_{z^i}} (\mu^i(d,a) + \ c_S(\pi^2_{Z^i}(a)))$.

Since c_S is an order-reversing map with respect to c-semiring ordering, if $\pi^1_{Z^i}(a) \geq_S \pi^2_{Z^i}(a)$, $\forall a$ then $c_S(\pi^1_{Z^i}(a)) \leq_S c_S(\pi^2_{Z^i}(a))$, $\forall a$. Thus, by monotonicity of $+$, $(\mu^i(d,a)+ c_S(\pi^1_{Z^i}(a)))$ $\leq_S (\mu^i(d,a)+ c_S(\pi^2_{Z^i}(a)))$, $\forall a, d$. From here we can conclude as in the proof of Proposition 12. $\square$

2.13.3 Semantics

In this section we show how to compute the preference of a solution of an uncertain soft CSP and how to adapt the semantics defined in Section 2.7 to this more general context.

We recall that, given an uncertain SCSP $SQ = \langle S, V_c, V_u, C = C_f \cup C_{fu}\rangle$, algorithm G-SP obtains a SCSP $SQ' = \langle S, V_c, C = C_f \cup C_p \cup C_u\rangle$. Given a solution s of SQ, i.e., a complete assignment to V_c, we can compute its preference. This preference, which is obtained by computing the values $F(s)$, $P(s)$ and $U(s)$, that are respectively the result of the combination (via the operator $\times$ of the c-semiring S) of the preferences over the constraints in C_f, C_p and C_u, is given by the two values $F_P(s) = (F(s) \times P(s))$ and $U(s)$.

Once a solution of an uncertain soft CSP based on a c-semiring $S = \langle A, +, \times, 0, 1\rangle$ is associated to a pair, we can generalize the semantics introduced in Section 2.7 for fuzzy preferences. We perform this generalization by taking into account the fact that preferences could also be incomparable.

Definition 19 (P-semantics) Given an uncertain soft SCSP $SQ = \langle S, V_c, V_u, C\rangle$, where S is a semiring $\langle A, +, \times, 0, 1\rangle$ such that A is partially ordered, consider a solution s with corresponding satisfaction degree $F_P(s)$ and robustness $U(s)$. Each semantics associates to s the ordered pair $\langle a_s, b_s\rangle$ as follows:

- **P-Risky** (pR), **P-Diplomatic** (pD): $\langle a_s, b_s\rangle$=$\langle F_P(s), U(s)\rangle$;
- **P-Safe** (pS): $\langle a_s, b_s\rangle$=$\langle U(s), F_P(s)\rangle$;
- **P-Risky1** (pR1): $\langle a_s, b_s\rangle$= $\langle F_P(s) \times U(s), F_P(s)\rangle$;
- **P-Safe1** (pS1): $\langle a_s, b_s\rangle$= $\langle F_P(s) \times U(s), U(s)\rangle$.

Given two solutions s and s', let $\langle a_s, b_s\rangle$ and $\langle a_{s'}, b_{s'}\rangle$ represent the pairs associated to the solutions by each semantics in turn. The **P-Risky, P-Safe, P-Risky1, P-Safe1** semantics work as follows:

- if $a_1 >_S a_2$ then $\langle a_1, b_1\rangle >_J \langle a_2, b_2\rangle$ (and the opposite for $a_2 >_S a_1$)

- if $a_1 = a_2$ then
  - if $b_1 >_S b_2$ then $\langle a_1, b_1\rangle >_J \langle a_2, b_2\rangle$ (and the opposite for $b_2 > b_1$)
  - if $b_1 = b_2$ then $\langle a_1, b_1\rangle = \langle a_2, b_2\rangle$;
  - if $b_1 \bowtie_S b_2$ then $\langle a_1, b_1\rangle \bowtie_J \langle a_2, b_2\rangle$,
- if $a_1 \bowtie_S a_2$ then
  - if $b_1 >_S b_2$ then $\langle a_1, b_1\rangle >_J \langle a_2, b_2\rangle$ (and the opposite for $b_2 > b_1$)
  - else $\langle a_1, b_1\rangle \bowtie_J \langle a_2, b_2\rangle$;

where $J = pR, pS, pR1, pS1$.

The **P-Diplomatic** (pD) semantics works as follows:

- if $a_1 \leq a_2$ and $b_1 \leq b_2$ then $\langle a_1, b_1\rangle \leq_{pD} \langle a_2, b_2\rangle$ (and the opposite for $a_2 \leq a_1$ and $b_2 \leq b_1$);
- if $a_1 = a_2$ and $b_1 = b_2$ then $\langle a_1, b_1\rangle = \langle a_2, b_2\rangle$;
- else $\langle a_1, b_1\rangle \bowtie_{pD} \langle a_2, b_2\rangle$.

Notice that all the semantics defined above in the case of partially ordered preferences can produce a partial order over solutions.

If the preferences in A are totally ordered, according to the ordering induced by the additive operator of S, then in the definition of the semantics above we must not consider cases $a_1 \bowtie_S a_2$ and $b_1 \bowtie_S b_2$, since they can't happen. In this case all the semantics except P-Diplomatic semantics produce a total order over the solutions.

2.13.4 Two instances of G-SP

In this section we add uncertainty to two very common classes of soft CSPs, which are Probabilistic CSPs [FLS96] and Weighted CSPs [BMR97], and we instantiate G-SP in these two classes in a way that all the assumptions required by G-SP are satisfied.

We recall that, given a USCSP SQ based on a generic c-semiring $S = \langle A, +, \times, 0, 1\rangle$, G-SP translates every constraint of sQ involving controllable and uncontrollable variables in a constraint only among their controllables, which is characterized by a preference function μ such that, if d is an assignment to the controllables and a is an assignment to the uncontrollables, $\mu'(d) = inf_{a \in A_Z}(\mu(d, a) + c_S(\pi_Z(a)))$, where the assumptions are: (i) inf

returns one of the bottom elements of A_Z with respect to the c-semiring ordering, (ii) $[0,1]$ is subset of the set of preferences A of the c-semiring and (iii) c_S is a bijection from $[0,1]$ to $[0,1]$ which reverses the ordering with respect to c-semiring and such that $c(c(p)) = p$, $\forall p \in [0,1]$.

Probabilistic CSPs with uncertainty

In several real-life scenarios, fuzzy constraints are not the ideal setting. In fact, they suffer for the well-known drowning effect which makes solutions with the same minimum preference but very different higher preferences not distinguished.

Probabilistic CSPs (PCSPs) [FLS96] model those situations where to each constraint is assigned the probability to be in the real problem. Here preferences are interpreted as probabilities ranging from 0 to 1 and, as expected, they are combined using the product and compared using the maximum operator. The goal is to maximize the joint probability. Therefore, the semiring to be used is $S_{PCSP} = \langle [0,1], max, \times, 0, 1\}$.

PCSPs with uncertainty are PCSPs defined by a set of controllable and uncontrollable variables and by a set of constraints relating these variables. Moreover, every value in the domain of a uncontrollable variable is associated with a possibility degree, stating how much is possible for that variable assuming that value.

In the following we show how to instantiate the Formula 2.2 in the case of PCSPs with uncertainty and we show that this instantiation satisfies the required assumptions.

Proposition 14 *Consider a Probabilistic CSP with uncertainty* $SQ = \langle S_{PCSP}, V_c, V_u, C\rangle$, *where* $S_{PCSP} = \langle [0,1], max, \times, 0, 1\rangle$. *Algorithm G-SP can be instantiated by translating every constraint,* $c = \langle \mu, con\rangle \in C$, *such that* $con \cap V_c = X$ *and* $con \cap V_u = Z$ *with possibility distribution* π_Z, *in a new constraint,* $\langle \mu', con' = X\rangle$, *where* μ' *is such that, if* d *is an assignment to* X, *and* a *an assignment to* Z,

$$\mu'(d) = min_{a\in A_Z} max(\mu(d,a), 1 - \pi_Z(a)).$$

Proof: (i) The assumption stating that inf returns one of the bottom elements of A_Z is trivially true, since the inf operator used here is min. (ii) $[0,1] \subseteq A$ is trivially true, since $A = [0,1]$. (iii) The mapping c_S used here is a bijection from $[0,1]$ to $[0,1]$, which associates to every element $a \in [0,1]$ an element $c_S(a) = (1-a) \in [0,1]$. This mapping reverses the ordering with respect to the c-semiring S_{PCSP}. In fact, given $a_1, a_2 \in [0,1]$, with $a_1 \leq a_2$, then $1 - a_1 \geq_S 1 - a_2$, since $max(1-a_1, 1-a_2) = 1 - a_1$. Moreover, $\forall a \in [0,1]$, we have that $1 - (1-a) = a$. $\square$

Example 12 Figure 2.9 shows the result of applying *G-SP* to the Probabilistic CSP with uncertainty $SQ = \langle S_{PCSP}, \{x, y, w\}, \{z\}, C = C_{xyz} \cup C_{xw} \rangle$ in Figure 2.8, where C_{xyz} is the constraint $\langle \{x, y, z\}, \mu \rangle$, C_{xw} is the constraint $\langle \{x, w\}, \mu_1 \rangle$ and the values of uncontrollable variables are described by the possibility distribution π_Z. In particular, Figure 2.9 (a) shows the resulting probabilistic CSP SQ' obtained by instantiating algorithm *G-SP* as described above in this section. Figure 2.9 (b) shows all the solutions, together with their associated preferences, that have been computed following the procedure described in Section 2.13.1 with the new instantiated formula presented in Proposition 14. Notice that in this example the optimal solution is the same for all the general semantics defined in Section 2.13.3. □

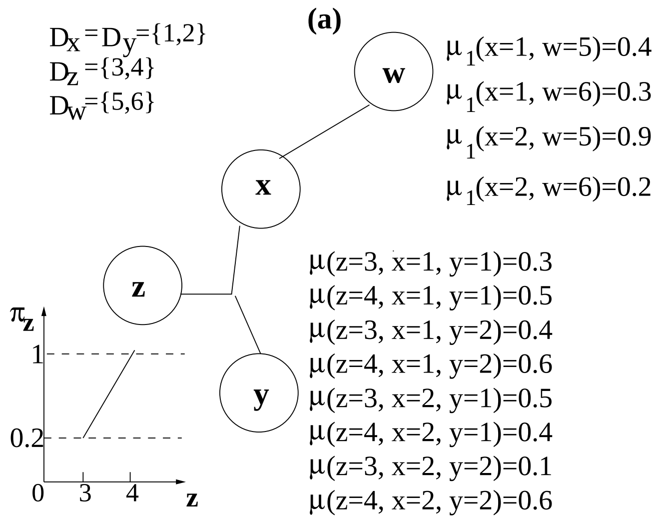

Figure 2.8: An uncertain soft CSP.

Weighted CSPs with uncertainty

In several situations where neither fuzzy nor probabilistic constraints are ideal, weighted constraints can be useful to model preferences. For example, when dealing with costs which are naturally combined by a sum. In this setting, preferences are penalties (or costs) to be added, and the best solutions are those with the smallest preference. Therefore, Weighted CSPs (WCSP) are characterized by the c-semiring $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0 \rangle$.

Weighted CSPs with uncertainty are WCSPs defined by a set of controllable and uncontrollable variables and by a set of constraints relating these variables. Moreover, every value in the domain of a uncontrollable variable is associated with a possibility degree, stating how much is possible for that variable assuming that value.

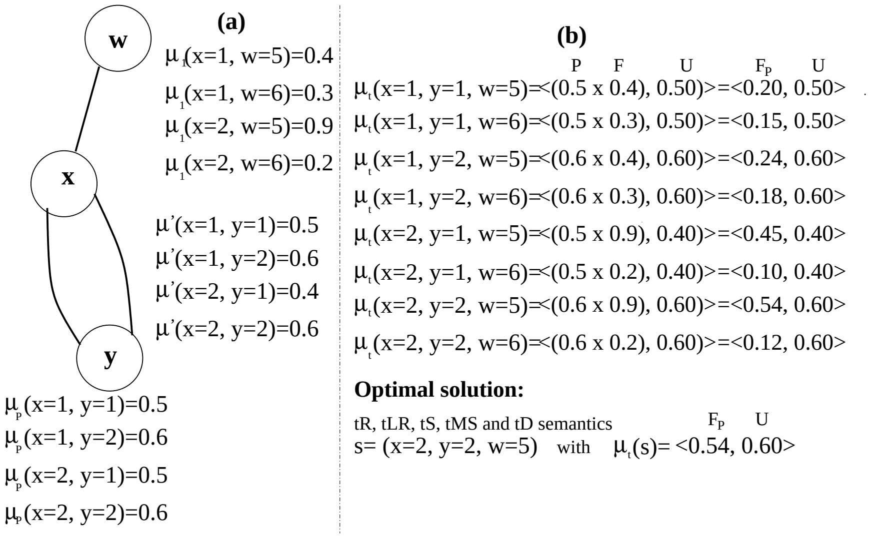

Figure 2.9: Result of algorithm *G-SP* on the USCSP in Figure 2.8, seen as UPCSP.

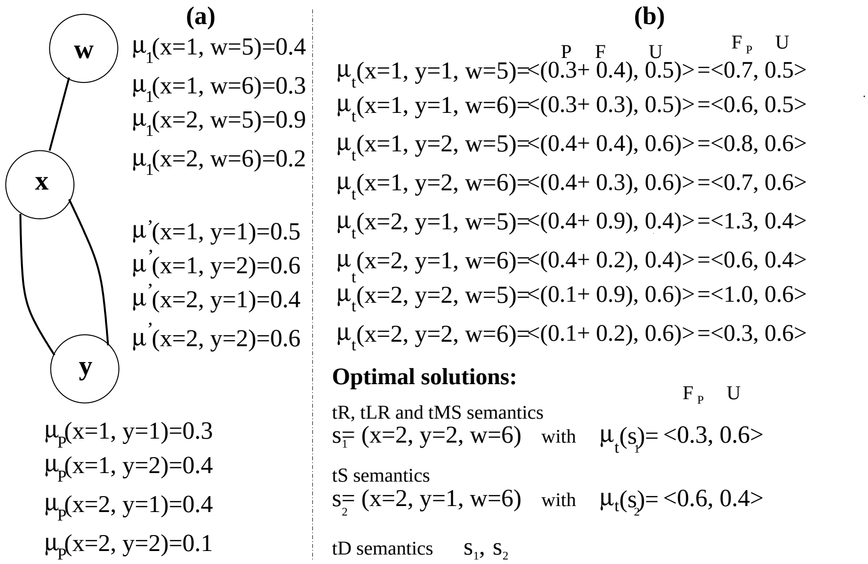

Figure 2.10: Result of algorithm *G-SP* on USCSP in Figure 2.8, seen as UWCSP.

As done above for Probabilistic CSPs with uncertainty, we show how to instantiate the Formula 2.2 in the case of WCSPs with uncertainty and we prove that this instantiation satisfies the required assumptions.

Proposition 15 *Consider a Weighted CSP with uncertainty* $SQ = \langle S_{WCSP}, V_c, V_u, C\rangle$, *where* $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0\rangle$. *Algorithm G-SP can be instantiated by translating every constraint,* $c = \langle \mu, con\rangle \in C$, *such that* $con \cap V_c = X$ *and* $con \cap V_u = Z$ *with possibility distribution* π_Z, *in a new constraint,* $\langle \mu', con' = X\rangle$, *where* μ' *is such that, if* d *is an assignment to* X, *and* a *an assignment to* Z,

$$\mu'(d) = max_{a \in A_z} min(\mu(d, a), \pi_Z(a))).$$

Proof: (i) The assumption stating that inf returns one of the bottom elements of A_Z is true, since the inf operator used here is max, that returns the highest and so the worst cost. (ii) $[0, 1] \subseteq \mathcal{R}^+$. (iii) The function c_S used here is the identity map in $[0, 1]$. Hence $c(c(a)) = c(a) = a$, $\forall a \in [0, 1]$. Moreover it reverses the order with respect to c-semiring S_{WCSP}. In fact, given $a_1, a_2 \in [0, 1]$ such that $a_1 \leq a_2$, then $c(a_1) = a_1$ and $c(a_2) = a_2$, then since $min(a_1, a_2) = a_1$, $a_1 \geq_S a_2$. $\square$

Example 13 Figure 2.10 shows how the instantiation of G-SP to uncertain WCSPs works on the Weighted CSP with uncertainty $SQ = \langle S_{WCSP}, \{x, y, w\}, \{z\}, C = C_{xyz} \cup C_{xw}\rangle$ shown in Figure 2.8, where preferences are interpreted as costs and where the values of uncontrollable variables are described by the possibility distribution π_Z. Figure 2.10 (a) shows the resulting Weighted CSP SQ' returned by the instantiated G-SP, while Figure 2.10 (b) shows all the solutions, together with the associated pair. In this problem the optimal solution obtained using P-Risky, P-Risky1 and P-Safe1 semantics is different from the one obtained using P-Safe. For P-Diplomatic these two solutions are both optimal. $\square$

2.13.5 Desired properties on the solution ordering

In this section we check if new general semantics, that we have defined in Section 2.13.3, satisfy the desired properties on the solution ordering presented in Section 2.3. Notice that if a semantics in fuzzy case doesn't satisfy a property, than surely the corresponding general semantics doesn't satisfy that property in general, since every new semantics generalizes the corresponding one defined for UFCSPs. However, since the non-satisfaction of the property could depend on the idempotency of the combination operator of the fuzzy c-semiring, then that property could be satisfied by the general semantics in UPCSPs or in UWCSPs, since in their c-semirings the combination operator is not idempotent. Hence, when we will show that a semantics doesn't satisfy a property in general, since this happens for UFCSPs, we will also check if the same happens also for UPCSPs and UWCSPs.

In Section 2.8 we have proved that Property 3 is satisfied only by Risky and Risky1 for fuzzy preferences. Now we show that in general and also for UPCSPs and UWCSPs this holds only for P-Risky semantics.

Proposition 16 *Consider an uncertain SCSP $SQ = \langle S, V_c, V_u, C = C_f \cup C_u \rangle$, where S is a c-semiring $\langle A, +, \times, 0, 1 \rangle$. Given two solutions s and s' of SQ, i.e., assignments to V_c, if $\forall a$ assignments to V_u in SQ, $pref(s,a) >_S pref(s',a)$ (where $>_S$ refers to the ordering induced by operator $+$ of S), then $s >_{pR} s'$, whereas it could happen that $s \not>_J s'$ for $J = pR1, pS, pS1, pD$. The same result holds also if $S = S_{PCSP}$ or if $S = S_{WCSP}$.*

Proof:

- *P-Risky.* Similarly to the proof of Proposition 3, from USCSP SQ we can obtain an equivalent USCSP $SQP = \langle S, \{V^c\}, \{V^u\}, C_1 \cup C_2 \cup C_3 \rangle$ where: V^c is a controllable variable and V^u is an uncontrollable variable, representing respectively all the variables in V_c and V_u, having as domains the corresponding Cartesian products. The uncontrollable variable V^u is described by a possibility distribution, π, which is the joint possibility (see Section 2.2.4) of all the possibility distributions of the uncontrollable variables in V_u. Constraints $C_1 = \langle \mu_1, V^c \rangle$ and $C_2 = \langle \mu_2, \{V^c, V^u\} \rangle$ are, respectively, defined as the combination of all the constraints in C connecting variables in V_c and as the combination of all the constraints in C connecting variables in V_c to variables in V_u. Constraint $C_3 = \langle \mu_3, V^c \rangle$ is defined as the combination of all the constraints obtained from constraints in C_2 by projecting them (by using operator $+$ of the c-semiring S) over the controllable variables in V_c (i.e., $C_3 = C_2 \Downarrow_{V_c}$). Notice that all these combinations are obtained using operator $\times$ of the c-semiring S. Thus, given assignment $V^c = s$ in SQP, which corresponds to an assignment to all the variables in V_c, its preference on constraint C_1 is $\mu_1(s) = F(s)$, on C_3 is $\mu_3(s) = P(s)$ and on $C_1 \bigotimes C_3$ is $\mu_1(s) \times \mu_3(s) = (F(s) \times P(s)) = F_P(s)$. Given assignment $(V^c = s, V^u = a_i)$, instead, which corresponds to a complete assignment to variables in V_c and V_u, its preference, $\mu_2(s, a_i)$, is obtained performing the combination of the preferences associated to all the subtuples of (s, a_i) by the constraints in C involving at least one variable in V_u. Using this new notation we have that $\forall s, a_i$ assignments to V^c and V^u, $pref(s, a_i) = (\mu_1(s) \times \mu_2(s, a_i)) = (F(s) \times \mu_2(s, a_i))$.

If $pref(s, a_i) >_S pref(s', a_i)$, $\forall a_i$ assignment to V^u, then $F_P(s) >_S F_P(s')$ and so, by the definition of P-Risky semantics, we can conclude that $s >_{pR} s'$. In fact, if $pref(s, a_i) > pref(s', a_i)$, $\forall a_i$, then this holds also for a_{i*} such that $P(s') = \mu_2(s, a_{i*})$. Then we have $(F(s)\times \mu_2(s, a_{i*})) > (F(s')\times \mu_2(s', a_{i*})) = F(s')\times P(s') = F_P(s')$. Since $P(s) \geq \mu_2(s, a_{i*}))$, then, by monotonicity of $\times$, $F_P(s) = (F(s) \times P(s)) \geq (F(s) \times \mu_2(s, a_{i*})) > F_P(s')$, and so $F_P(s) > F_P(s')$.

- *P-Risky1, P-Safe, P-Diplomatic and P-Safe1.* For these semantics it can happen that $s \not> s'$. This holds also if $S = S_{PCSP}$ or if $S = S_{WCSP}$ as shown in the following.

  $S = S_{PCSP}$. Let us consider an UPCSP $SQ = \langle S_{PCSP}, V_c, V_u, C\rangle$ where $S_{PCSP} = \langle [0, 1], max, \times, 0, 1\rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.5$, $\mu_2(s, a_2) = 0.35$, $\mu_2(s', a_1) = 0.6$, $\mu_2(s', a_2) = 0.4$, $\mu_1(s) = 0.5$ and $\mu_1(s') = 0.4$. Then the overall preferences are: $pref(s, a_1) = 0.5 \times 0.5 = 0.25$, $pref(s, a_2) = 0.5\times 0.35 = 0.175$, $pref(s', a_1) = 0.4\times 0.6 = 0.24$, $pref(s', a_2) = 0.4\times 0.4 = 0.16$, i.e., $pref(s, a_i) >_S pref(s', a_i), \forall i$, $i = 1, 2$, where $\leq_S$ is the ordering induced by maximum, hence s and s' satisfy the hypothesis. The robustness values for s and s' are $U(s) = 0.35$, $U(s') = 0.4$ and the satisfaction degrees are $F_P(s) = 0.5 \times 0.5 = 0.25$ and $F_P(s') = 0.6\times 0.4 = 0.24$. Since $U(s) < U(s')$, then $s <_{pS} s'$. Since we have also that $F_P(s) > F_P(s')$, then $s \bowtie_{pD} s'$. Moreover, since $F_P(s) \times U(s) = 0.25 \times 0.35 = 0.0875 < F_P(s') \times U(s') = 0.24 \times 0.4 = 0.096$, then $s <_{pR1,pS1} s'$.

  $S = S_{WCSP}$. Let us consider an UPCSP $SQ = \langle S_{WCSP}, V_c, V_u, C\rangle$ where $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0\rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.64$, $\mu_2(s, a_2) = 0.4$, $\mu_2(s', a_1) = 0.55$, $\mu_2(s', a_2) = 0.35$, $\mu_1(s) = 0.4$ and $\mu_1(s') = 0.5$. Then the overall preferences are: $pref(s, a_1) = 0.4 + 0.64 = 1.04$, $pref(s, a_2) = 0.4+0.4 = 0.8$, $pref(s', a_1) = 0.5+0.55 = 1.05$, $pref(s', a_2) = 0.5+0.35 = 0.85$, i.e., $pref(s, a_i) >_S pref(s', a_i), \forall i$, $i = 1, 2$, since we consider the ordering $\leq_S$ induced by the minimum operator, hence s and s' satisfy the hypothesis. The robustness values for s and s' are $U(s) = 0.64$, $U(s') = 0.55$ and the satisfaction degrees are $F_P(s) = 0.4 + 0.4 = 0.8$ and $F_P(s') = 0.5 + 0.35 = 0.85$. Since $U(s) <_S U(s')$, then $s <_{pS} s'$.

Since $F_P(s) > F_P(s')$, then $s \bowtie_{pD} s'$. Since $F_P(s) + U(s) = 0.8 + 0.64 = 1.44 <_S F_P(s') \times U(s') = 0.85 + 0.55 = 1.4$, then $s <_{pR1,pS1} s'$.

□

In Section 2.8 we have proved that Property 4 and Property 5 are satisfied by all the new semantics except respectively Safe1 and Risky1 semantics. We show that the same results hold also for the new general semantics. Moreover we give a new relevant result stating that Property 4 and Property 5 are satisfied by all the general semantics if we assume that the combination operator of the c-semiring on which USCSP is based is strictly monotone.

Proposition 17 *Consider an uncertain SCSP* $SQ = \langle S, V_c, V_u, C\rangle$, *where* S *is a c-semiring* $\langle A, +, \times, 0, 1\rangle$. *Given two solutions* s *and* s' *of* SQ, *if* $U(s) = U(s')$ *and* $F_P(s) >_S F_P(s')$, *then* $s >_J s'$, *where* $J = pR, pS, pD, pR1$. *If* $\times$ *is strictly monotone, then* $s >_{pS1} s'$, *otherwise it could happen that* $s \not>_{pS1} s'$. *Hence if* $S = S_{PCSP}$ *or* $S = S_{WCSP}$, *then* $s >_{pS1} s'$ *when* $U(s) = U(s') \neq 0$.

Proof:

- *P-Risky*, *P-Safe*, *P-Diplomatic* satisfy this property by definition.

  Also *P-Risky1* satisfies this property. Since $F_P(s) >_S F_P(s')$ then, by monotonicity of $\times$, $(F_P(s) \times U(s)) \geq_S (F_P(s') \times U(s))$, and, since $U(s) = U(s')$, $(F_P(s) \times U(s)) \geq_S (F_P(s') \times U(s'))$. If $(F_P(s) \times U(s)) >_S (F_P(s') \times U(s'))$, then we conclude immediately. If $(F_P(s) \times U(s)) = (F_P(s') \times U(s'))$ we conclude by observing that $F_P(s) >_S F_P(s')$.

- *P-Safe1*. If $\times$ is strictly monotone, then $(F_P(s) \times U(s)) > (F_P(s') \times U(s'))$ and so $s >_{pS1} s'$. If $S = S_{PCSP}$ and $U(s) = U(s') \neq 0$, then $\times$, which is the product that is by definition monotone, is also strictly monotone. If $S = S_{WCSP}$ and $U(s) = U(s') \neq \{0, +\infty\}$, then the combination operator of S, i.e. the sum, is strictly monotone. Since, by construction $U(s)$ is always different from $+\infty$, then it is sufficient that $U(s) = U(s') \neq 0$ in order to have the sum strictly monotone.

  Otherwise, it could happen $s \not>_{pS1} s'$ as shown in the proof of Proposition 4 for Safe1 semantics. For showing that this holds also in UPCSPs and in UWCSPs we can consider and example of UPCSP where $U(s) = U(s') = 0$.

□

We recall that constraints in C_{fu} are constraints between a set X of controllable variables and a set Z of uncontrollable variables. If $S = S_{PCSP}$, stating $U(s) = 0$ means that we have associated in a constraint of C_{fu} the value 0 to an assignment $(X = d, Z = a)$ where the possibility of a is equal to 1. If $S = S_{WCSP}$ stating $U(s) = 0$, means that in every constraint of C_{fu} for every assignment $(X = d, Z = a)$ or its preference is equal to 0 or the possibility of a is zero. Hence, since this last condition is very specific to obtain, we can say that in general all the semantics satisfy Property 4 in UWCSPs.

Proposition 18 *Consider a USCSP* $SQ = \langle S, V_c, V_u, C\rangle$*, where* S *is a c-semiring* $\langle A, +, \times, 0, 1\rangle$*. Given two solutions* s *and* s' *of* SQ*, if* $F_P(s) = F_P(s')$ *and* $U(s) > U(s')$*, then* $s >_J s'$*, where* $J = pR, pS, pD, pS1$*. If* $\times$ *is strictly monotone, then* $s >_{pR1} s'$*, otherwise it could happen that* $s \not>_{pR1} s'$*. Hence if* $S = S_{PCSP}$*, then* $s >_{pS1} s'$ *when* $F_P(s) = F_P(s') \neq 0$ *and if* $S = S_{WCSP}$ *then* $s >_{pS1} s'$ *when* $F_P(s) = F_P(s') \notin \{0, +\infty\}$*.*

Proof:

- *P-Risky*, *P-Safe* and *P-Diplomatic* satisfy this property by definition.

  Also *P-Safe1* satisfies this property. If $F_P(s) = F_P(s)$ and $U(s) > U(s')$, then by monotonicity of $\times$, $(F_P(s) \times U(s)) \geq (F_P(s') \times U(s'))$. If $(F_P(s) \times U(s)) > (F_P(s') \times U(s'))$, then we conclude immediately. If $(F_P(s) \times U(s)) = (F_P(s') \times U(s'))$, we conclude by observing that $U(s) > U(s')$.

- *P-Risky1*. If $\times$ is strictly monotone, then $(F_P(s) \times U(s)) > (F_P(s') \times U(s'))$ and so $s >_{pR1} s'$. If $S = S_{PCSP}$ and $F_P(s) = F_P(s') \neq 0$, then $\times$, which is the product, that is by definition monotone, is also strictly monotone. If $S = S_{WCSP}$ and $F_P(s) = F_P(s') \notin \{0, +\infty\}$, then the combination operator of S, i.e. the sum, is strictly monotone.

  Otherwise, it could happen $s \not>_{pR1} s'$. For showing this we can use the same counterexample used in the proof of Proposition 5 for Risky1 semantics. For showing that this holds also in UPCSPs and in UWCSPs we can consider an example where $F_P(s) = F_P(s') = 0$.

□

In $UPSCPs$ it could be $s \not> s'$ when $F_P(s) = 0$, i.e., when there is at least a constraint where a sub-tuple of s appears such that its associated preference is zero. In $UWSCPs$ it could be $s \not>_{pR1} s'$ when $F_P(s) = 0$, i.e., when in every constraint where a sub-tuple of s

appears its associated preference is zero, and when $F_P(s) = +\infty$, i.e., when in at least a constraint where a subtuple of s appears its associated preference is $+\infty$.

Summarizing, P-Risky satisfies all the desired properties on the solution ordering (i.e., Properties 3, 4 and 5); P-Safe and P-Diplomatic satisfy Properties 4 and 5; P-Safe1 satisfies Property 5 and, if the combination operator of the c-semiring is strictly monotone, also Property 4; P-Risky1 satisfies Property 4 and, if the combination operator of the c-semiring is strictly monotone, also Property 5.

2.13.6 Desired properties in general DFP

Let us now briefly reconsider the solution ordering produced by DFP algorithm [DFP96a] (see Section 2.2.4). We recall that in fuzzy case, according to DFP, the preference of a solution s is a single value equal to $min(F(s), U(s))$. Thus, given two solutions s and s', $s >_{DFP} s'$ if and only if $min(F(s), U(s)) > min(F(s), U(s))$. We can generalize this approach to USCSPs stating that, given two solutions s and s', $s >_{pDFP} s'$ if and only if $(F(s) \times U(s)) > (F(s) \times U(s))$, where $\times$ is the combination operator of the c-semiring on which the USCSP is based.

We will show that the solution ordering produced by the above semantics, that we call P-DFP, doesn't satisfy any desired properties regarding the solution ordering also in UPCSPs and in UWCSPs. This is due to the fact that if forgets information derived by the projections constraints, which is instead useful to recall.

Proposition 19 *Consider an uncertain SCSP $SQ = \langle S, V_c, V_u, C\rangle$. Given two solutions s and s' of SQ, i.e., assignments to V_c, if $\forall a$ assignments to V_u in SQ, $pref(s, a) >_S pref(s', a)$, then it could happen that $s \not>_{gDFP} s'$. The same result holds also if $S = S_{PCSP}$ or if $S = S_{WCSP}$.*

Proof: For showing that $s <_{pDFP} s'$ in some case, we can use the same example considered in the proof of Proposition 6 in Section 2.9 for UFCSPs. Now we show that $s <_{pDFP} s'$ can happen also in UPCSPs and in UWCSP.

- $S = S_{PCSP}$. Let us consider an UPCSP $SQ = \langle S_{PCSP}, V_c, V_u, C\rangle$ where $S_{PCSP} = \langle [0, 1], max, \times, 0, 1\rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.8$, $\mu_2(s, a_2) = 0.3$, $\mu_2(s', a_1) = 0.5$, $\mu_2(s', a_2) = 0.2$, $\mu_1(s) = 0.5$ and $\mu_1(s') = 0.7$. Then the overall preferences are: $pref(s, a_1) = 0.5 \times 0.8 = 0.4$, $pref(s, a_2) = 0.5 \times 0.3 = 0.15$, $pref(s', a_1) = 0.7 \times 0.5 = 0.35$, $pref(s', a_2) = 0.7 \times 0.2 = 0.14$, i.e., $pref(s, a_i) >_S pref(s', a_i), \forall i, i = 1, 2$, hence s and s' satisfy the hypothesis. The robustness values for s and s' are $U(s) = 0.3$, $U(s') = 0.3$. Since $F(s) \times U(s) = 0.5 \times 0.3 = 0.15 <_S F(s') \times U(s') = 0.7 \times 0.3 = 0.21$, then $s <_{pDFP} s'$.

- $S = S_{WCSP}$. Let us consider an UWCSP $SQ = \langle S_{WCSP}, V_c, V_u, C\rangle$ where $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0\rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.05$, $\mu_2(s, a_2) = 0.6$, $\mu_2(s', a_1) = 0.4$, $\mu_2(s', a_2) = 1$, $\mu_1(s) = 0.8$ and $\mu_1(s') = 0.5$. Then the overall preferences are: $pref(s, a_1) = 0.8 + 0.05 = 0.85$, $pref(s, a_2) = 0.8 + 0.6 = 1.4$, $pref(s', a_1) = 0.5 + 0.4 = 0.9$, $pref(s', a_2) = 0.5 + 1 = 1.5$, i.e., $pref(s, a_i) >_S pref(s', a_i), \forall i, i = 1, 2$, where $>_S$ is the ordering induced by the minimum operator, hence s and s' satisfy the hypothesis. The robustness values for s and s' are $U(s) = 0.6$, $U(s') = 0.7$. Since $F(s) + U(s) = 0.8 + 0.6 = 1.4 <_S F(s') + U(s') = 0.5 + 0.7 = 1.2$, then $s <_{pDFP} s'$.

□

Proposition 20 *Consider a USCSP $SQ = \langle S, V_c, V_u, C\rangle$. Given two solutions s and s' of SQ, if $U(s) = U(s')$ and $F_P(s) >_S F_P(s')$, then it could happen that $s \not>_{gDFP} s'$. The same result holds also if $S = S_{PCSP}$ or if $S = S_{WCSP}$.*

Proof: For showing that $s <_{pDFP} s'$ in some case, we can use the same example considered in the proof of Proposition 7 in Section 2.9 for UFCSPs. Now we show that $s <_{pDFP} s'$ can happen also when $S = S_{PCSP}$ or if $S = S_{WCSP}$.

- $S = S_{PCSP}$. Let us consider an UPCSP $SQ = \langle S_{PCSP}, V_c, V_u, C\rangle$ where $S_{PCSP} = \langle [0, 1], max, \times, 0, 1\rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.8$, $\mu_2(s, a_2) = 0.1$, $\mu_2(s', a_1) = 0.5$, $\mu_2(s', a_2) = 0.2$, $\mu_1(s) = 0.5$ and $\mu_1(s') = 0.7$.

$F_P(s) = 0.5 \times 0.8 = 0.4 >_S F_P(s') = 0.7 \times 0,5 = 0.35$ and and $U(s) = 0.3 = U(s')$. Hence s and s' satisfy the hypothesis. Since $F(s) \times U(s) = 0.5 \times 0.3 = 0.15 <_S F(s') \times U(s') = 0.7 \times 0.3 = 0.21$, then $s <_{pDFP} s'$.

- $S = S_{WCSP}$. Let us consider an UWCSP $SQ = \langle S_{WCSP}, V_c, V_u, C \rangle$ where $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0 \rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.6$. Let us assume moreover that $\mu_2(s, a_1) = 0.05$, $\mu_2(s, a_2) = 0.6$, $\mu_2(s', a_1) = 0.4$, $\mu_2(s', a_2) = 1$, $\mu_1(s) = 0.8$ and $\mu_1(s') = 0.5$. Then $F_P(s) = 0.8 + 0.05 = 0.85 >_S F_P(s') = 0.5 + 0.4 = 0.9$ and $U(s) = U(s') = 0.6$ hence s and s' satisfy the hypothesis. Since $F(s) + U(s) = 0.8 + 0.6 = 1.4 <_S F(s') + U(s') = 0.5 + 0.6 = 1.1$, then $s <_{pDFP} s'$.

□

Proposition 21 *Consider a USCSP $SQ = \langle S, V_c, V_u, C \rangle$. Given two solutions s and s' of SQ, if $F_P(s) = F_P(s')$ and $U(s) >_S U(s')$, then it could happen that $s \not>_{pDFP} s'$. The same result holds also if $S = S_{PCSP}$ or if $S = S_{WCSP}$.*

Proof: For showing that $s <_{pDFP} s'$ in some case, we can use the same example considered in the proof of Proposition 8 in Section 2.9 for UFCSPs. Now we show that $s <_{pDFP} s'$ can happen also when $S = S_{PCSP}$ or if $S = S_{WCSP}$.

- $S = S_{PCSP}$. Let us consider an UPCSP $SQ = \langle S_{PCSP}, V_c, V_u, C \rangle$ where $S_{PCSP} = \langle [0, 1], max, \times, 0, 1 \rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.4$, $\mu_2(s, a_2) = 0.8$, $\mu_2(s', a_1) = 0.5$, $\mu_2(s', a_2) = 0.3$, $\mu_1(s) = 0.5$ and $\mu_1(s') = 0.8$. $F_P(s) = 0.5 \times 0.8 = 0.4 = F_P(s')$ and and $U(s) = 0.4 >_S U(s') = 0.3$. Hence s and s' satisfy the hypothesis. Since $F(s) \times U(s) = 0.5 \times 0.4 = 0.20 <_S F(s') \times U(s') = 0.8 \times 0.3 = 0.24$, then $s <_{pDFP} s'$.

- $S = S_{WCSP}$. Let us consider an UWCSP $SQ = \langle S_{WCSP}, V_c, V_u, C \rangle$ where $S_{WCSP} = \langle \mathcal{R}^+, min, +, +\infty, 0 \rangle$, $V_c = \{x\}$, $V_u = \{z\}$, C is composed by two constraints: $c_1 =< \mu_1, \{x\} >$ and $c_2 =< \mu_2, \{x, z\} >$ and where $D_z = \{a_1, a_2\}$ and $D_x = \{s, s'\}$ are respectively the domain of z and x. Let us assume that the possibility distribution on z is such that $\pi(a_1) = 1$ and $\pi(a_2) = 0.7$. Let us assume moreover that $\mu_2(s, a_1) = 0.1$, $\mu_2(s, a_2) = 0.6$, $\mu_2(s', a_1) = 0.4$, $\mu_2(s', a_2) = 1$, $\mu_1(s) = 0.8$ and $\mu_1(s') = 0.5$. Then $F_P(s) = 0.8 + 0.1 = 0.9 = F_P(s')$ and $U(s) = 0.7 >_S U(s') = 0.6$ hence s and s' satisfy the hypothesis. Since $F(s) + U(s) = 0.8 + 0.6 = 1.4 <_S F(s') + U(s') = 0.5 + 0.7 = 1.2$, then $s <_{pDFP} s'$.

□

	P-DFP	P-Risky	P-Safe	P-Dipl.	P-Safe1	P-Risky1
P1	X	X	X	X	X	X
P2	X	X	X	X	X	X
P3	-	X	-	-	-	-
P4	-	X	X	X	Xif	X
P5	-	X	X	X	X	Xif

Table 2.3: Properties satisfied in the various general semantics. 'X' means satisfied, 'Xif' means satisfied if $\times$ of the c-semiring is strictly monotone and '-' means satisfied neither in UPCSPs nor in UWCSPs.

Table 2.3 summarizes which properties hold in the new general semantics. In particular, all the properties, which hold for the various semantics in the fuzzy case, except Property 3 for Risky1, continue to hold also for the corresponding general semantics. Moreover, if we consider USCSPs defined by c-semirings with the combination operator which is strictly monotonic, then P-Risky1 satisfies also Property 5 and P-Safe1 satisfies also Property 4. Hence, under that assumption, Properties 4 and 5 are satisfied by all the new general semantics. Instead, the general semantics induced by DFP satisfies the desired properties on the solution ordering neither in UPCSPs nor in UWCSPs.

2.14 Related work

We have defined a new way for integrating preferences and possibilistic uncertainty, that assumes commensurability between preferences and possibilities scales. This new method allows us to discriminate the satisfaction level and the robustness value of a solution and so to obtain a solution ordering which better reflects the desirability and the robustness of a solution.

Many other approaches have studied procedures for reasoning with preferences and uncertainty, such as the ones in [DFP02, FS03, BT96, BT97]. However, they don't mix preferences and uncertainty since they do not assume commensurability. Moreover, their approaches are based on the qualitative decision theory, while our procedure is based on the quantitative decision theory.

Some effort has been done for unifying preferences and possibilistic uncertainty [AP04] assuming commensurability. In [AP04] two different approaches for integrating fuzzy preferences and possibilistic uncertainty are proposed: a pessimistic and an optimistic one, however both methods propose to mix robustness and the satisfaction level, and so it isn't possible to make a tight comparison between our semantics that assume every solution is associated with two parameters and those of [AP04] where every solution is associated with only one value, which is the combination of the satisfaction level and the robustness one. However, we can show some differences concerning the degree of satisfaction. In our framework we calculate the degree of satisfaction combining in pessimistic way all preferences, but some of them, i.e., the preferences obtained projecting on the controllable variables the constraints involving both controllable and uncontrollable variables are computed in optimistic way. In fact, performing projection of a constraint C linking a set of variables $V = V_1 \cup V_2$ on the subset V_1 means choosing the assignment $v1$ of variables in V_1 such that the assignment (v_1, v_2) in V has the best preference in C. Hence in the computation of our satisfaction degree we are *less pessimistic* than pessimistic approach in [AP04], where all preferences of the solutions are calculated and combined in pessimistic way; and *less optimistic* than optimistic approach in [AP04], where all the preferences of the solutions are calculated and combined in optimistic way. Therefore the degree of satisfaction in our framework is in the middle between optimistic and pessimistic ones proposed in [AP04].

Many approaches have been proposed for dealing with probabilistic uncertainty in decision problems (some of them are [Wal02, FLS96]). Notice that approaches considering probability distributions instead of possibility ones, make the assumption of the independence of the uncontrollable variables, which isn't required in our possibilistic framework. In the following we will present some of them. In [Wal02] it is proposed a procedure for dealing with probabilistic uncertainty in decision problems. It considers *stochastic constraint programming*, where a stochastic constraint program contains both stochastic variables, which follows a probability distribution, and controllable variables, which can be set dynamically. In this case a complete assignment of controllable variables satisfies the stochastic constraint program if the product of its associated probabilities is greater than or equal to a fixed threshold. A possible future direction of search could be the application of the stochastic constraint programming to the our possibilistic framework. The idea is assigning values to decision variables in a dynamic way, considering in every stage all the possible situations that can happen. The uncertainty is given by a probability distribution also in [FLS96]. This paper extends constraint satisfaction problem framework to deal with some decision problems under uncertainty, that they call mixed CSPs. The basis of this extension consists in a distinction between controllable and uncontrollable variables, like in our work, but uncontrollable variables are characterized by a probabilistic distribution and not by a possibilistic one. Moreover, differently from our approach, a solution gives a conditional decision. In particular, a solution depends on the assumptions concerning the agent's awareness of the uncontrollable variables at the time the decision must be made. It would be interesting to modify our framework in order to take into account these assumptions.

2.15 Future work

In this chapter we have presented a method for handling Soft CSPs with uncertainty that allows us to discriminate the satisfaction level and the robustness value of every solution and that satisfies some desired properties. We have first considered the method on Fuzzy CSPs with uncertainty, i.e., soft CSPs defined by a set of controllable and uncontrollable variables and by a set of fuzzy constraints involving these variables, where we have assumed that the values in the domain of the uncontrollable variables are characterized by possibility distributions. Then we have generalized the approach for dealing with other classes of soft constraints, not necessarily fuzzy. We plan to implement the solver which we have defined for such problems, and to perform experiments on benchmarks.

We plan to generalize the framework for what concerns uncertainty. In particular, we want to study USCSPs where the values in domain of uncontrollable variables are defined by probability distributions and not by possibility distributions. Then, starting from this, we would like also to study USCSPs where the values in the domain of some uncontrollable variables are specified by a possibility distribution and the values in the domain of other uncontrollables are characterized by a probability distribution. In this case we could replace possibilities with probabilities, or vice versa. In [DP98] it is presented a way to do this. Thus we could use such a method to obtain only one kind of distribution and then, if we have only possibility distributions, we can use the same general approach described in this chapter, while if we have only probabilities we can apply the procedure that we want to define for USCSPs with uncontrollable variables defined only by probability distributions. However, if we transform a probability into a possibility distribution we loose information, and solutions have a lower robustness. In fact, by Property 2, if we use possibilities, which are higher than probabilities, we get a smaller robustness value. Thus we can say that the robustness value obtained in this way is a *lower bound* to the robustness. On the other hand, if we transform possibilities into probabilities, we get smaller values, and thus, by Property 2, a higher robustness value, which can be seen as an *upper bound* to the robustness degree of a solution. Thus, for avoiding loss of information, we can still use a similar approach to the one used in this chapter, except that we have to define the robustness by an interval, whose bouderies are the lower and upper bound defined above. Since now the robustness is defined by an interval, in order to extend to this case the general semantics presented in this chapter, we must define an ordering over intervals. A possible ordering defines two intervals incomparable if one is strictly contained in the other one, and both lower and upper bounds are different and an interval better than another one when it is different from the other and its lower bound is greater or equal than the other lower bound and its upper bound is greater or equal than the other upper bound. Notice that this ordering is partial over the robustness values, while before we had a total order, hence this yields more incomparability in the ordering over solutions induced by the various semantics.

Another line of research, that we plan to investigate, regards the comparison between our notion of robustness with the different notions of controllability which have been introduced in the literature [VF99, YSVR03]. Controllability is, in general, defined as the ability of the agent to assign values to the controllable variables guaranteeing different levels of consistency or preference with the possible assignments to uncontrollable variables. In [VF99] three levels of controllability have been introduced in the context of hard quantitative temporal constraints. In [YSVR03] such definitions have been extended to deal with preferences. An assumption which is fixed is that of complete ignorance on the values taken by the uncertain variables. We are interested in the strongest notions of controllability which can be defined as follows. Given an uncertain SCSP $\langle S, V_c, V_u, C\rangle$, it is said to be *Optimally Strongly Controllable* if there is a fixed assignment to the variables in V_c, d, such that, given any assignment a to the variables in V_u, $pref(d, a)$ is optimal, that is, there is no other assignment to V_c, d', such that $pref(d', a) > pref(d, a)$. An uncertain SCSP is said to be *α-Strongly Controllable* (α-SC), where α is a preference level, if there is a fixed assignment to the variables in V_c, d, such that, given an assignment to the variables in V_u, a, $pref(d, a)$ is optimal if the optimal level which can be achieved given a is smaller or equal to α, and $pref(d, a) \geq \alpha$ otherwise. We think that it is possible to show that, if we make the assumption of complete ignorance, then an optimal solution according to the DFP semantics is a witness of α-SC. Making the assumption of complete ignorance in our context means to assume that all the possibilities degrees associated to values of uncontrollable variables are equal to one and this means, for example in fuzzy case, to reduce robustness with respect to uncertainty to the more intuitive notion of robustness, that considers the minimum preference which can be obtained in any possible case. In fact, given an UFCSP Q, the robustness of a solution s of Q is obtained by performing the minimum of all the preferences associated to subtuples of s in the new constraints computed by SP. We recall that the preference in these new constraints is $\mu'(d) = min_{\{a \in A_Z\}} max(\mu(d,a), 1 - \pi_Z(a)))$, where d is an assignment to controllable variables, μ is the preference function of the corresponding constraint involving also uncontrollables and π_Z is the possibility distribution associated with uncontrollables. If we assume complete ignorance, then $\pi_Z(a) = 1$ for any assignment a to uncontrollables, hence $\mu'(d) = min_{\{a \in A_Z\}} max(\mu(d,a), 1 - 1)) = min_{\{a \in A_Z\}} \mu(d,a)$. We plan to compare our notion of robustness with the various definitions of controllability mentioned before.

Chapter 3

Bipolar preferences

Real-life problems present several kinds of preferences. We focus on problems with both positive and negative preferences, that we call *bipolar problems*. Although seemingly specular notions, these two kinds of preferences should be dealt with differently to obtain the desired natural behaviour. We technically address this by generalizing the soft constraint formalism, which is able to model problems with one kind of preferences. We show that soft constraints model only negative preferences, and we define a new mathematical structure which allows to handle positive preferences as well. We also address the issue of the compensation between positive and negative preferences, studying the properties of this operation. Then, we extend the notion of arc consistency to bipolar problems, and we show how branch and bound (with or without constraint propagation) can be easily adapted to solve such problems. Finally, we define bipolar problems with uncertainty, where some variables are uncontrollable. We call such problems *uncertain bipolar problems* (UBPs) and we propose to handle them by extending existing techniques to handle bipolar problems (BPs) and problems with uncertainty. In particular, we first eliminate the uncertainty of the problem, transforming a UBP into a BP. Then we associate to each solution of BP both a degree of preference and a degree of robustness. Suitable semantics are then defined to order the solutions according to different attitudes with respect to these two notions.

3.1 Motivations and chapter structure

Many real-life problems contain statements which can be expressed as preferences. Moreover, preferences can be of many kinds: qualitative, (as in ”I like A more than B”), quantitative, (as in ”I like A at level 10 and B at level 11”), conditional, (as in ”If A happens, then I prefer B to C”), positive, (as in ”I like A, and I like B even more than A”), or negative (as in ”I don’t like A, and I really don’t like B”). Our long-term goal is to define a framework where many all such kinds of preferences can be naturally modelled and dealt with. In this chapter, we focus on problems which present positive and negative (quantitative and non-conditional) preferences, that we call *bipolar problems*. For example, when buying a house, we may like very much to live in the country, but we may also don’t like to have to take a bus to go to work, and be indifferent to the color of the house. Thus we will give a preference level (either positive, or negative, or indifference) to each feature of the house, and then we will look for a house which overall has the best combined preference.

Positive and negative preferences could be thought as two symmetric concepts, and thus one could think that they can be dealt with via the same operators. However, it is easy to see that this would not model what one usually expects in real scenarios. For example, when we have a scenario with two objects A and B, if we like both A and B, then the overall scenario should be more preferred than having just A or B alone. On the other hand, if we don’t like A nor B, then the preference of the overall scenario should be smaller than the preferences of A or B alone. Thus, usually combination of positive preferences should produce a higher (positive) preference, while combination of negative preferences should give a lower (negative) preference.

When dealing with both kinds of preferences, it is natural to express also indifference, which means that we express neither a positive nor a negative preference over an object. A desired behaviour of indifference is that, when combined with any preference (either positive or negative), it should not influence the overall preference.

Besides combining preferences of the same type, we also want to be able to combine positive with negative preferences. We strongly believe that the most natural and intuitive way to do so is to allow for compensation. Confronting positive against negative aspects and compensating them with respect to their strength is one of the core features of decision-making processes, and is, undoubtedly, a tactic universally applied to solve many real-life problems. For example, if we have a meal with meat (which we like very much) and wine (which we don’t like), then what should be the preference of the meal? To know that, we should be able to compensate the positive preference given to meat with the negative preference given to wine. The expected result is a preference which is between the two, and which should be positive if the positive preference is ”stronger” than the negative one.

Positive and negative preferences might seem as just two different criteria to reason with, and thus techniques such as those usually adopted by multi-criteria optimization [EG02] could appear suitable for dealing with them. However, this interpretation would hide the fundamental nature of bipolar preferences, that is, positive preferences are naturally the opposite of negative preferences. Moreover, in multi-criteria optimization it is often reasonable to use a Pareto-like approach, thus associating tuples of values to each solution, and comparing solutions according to tuple dominance. Instead, in bipolar problems, it would be very unnatural to force such an approach in all contexts, or to associate to a solution a preference which is neither a positive nor a negative one.

Soft constraints [BMR97] are a useful formalism to model problems with quantitative preferences. However, they can only model just one kind of preferences. Technically, they can model just negative preferences, since in this framework preference combination returns lower preferences, which, as mentioned above, is natural when using negative preferences. In this chapter we adopt the soft constraint formalism based on semirings, to model negative preferences. We then define a new algebraic structure to model positive preferences. To model bipolar problems, we link these two structures and we set the highest negative preference to coincide with the lowest positive preference to model indifference. We then define a combination operator between positive and negative preferences to model preference compensation, and we study its properties.

Non-associativity of preference compensation occurs often in many contexts, thus we think it is too restrictive to focus just on associative environments. Our framework allows for non-associativity, since we want to give complete freedom to choose the positive and negative algebraic structures. However, we describe a technique that, given a negative preference structure, builds a corresponding positive preference structure and so a bipolar preference structure where the compensation operator is associative.

Next, we consider the problem of finding optimal solutions of bipolar problems, by suggesting a possible adaptation of constraint propagation and branch and bound to the generalized scenario.

Finally, since many real-life situations contain some form of uncertainty, we focus on bipolar problems with uncertainty. We call such problems *uncertain bipolar problems* (UBPs). We model uncertainty by the presence of so-called *uncontrollable* variables. This means that the value of such variables will not be decided by us, but by Nature. A typical example, in the context of satellite scheduling or weather prediction, is a variable representing the time when clouds will disappear. Although we cannot choose the value for such variables, usually we have some information on the plausibility of the different values. This is modelled in this chapter by a possibility distribution over the domains of such variables. We tackle such problems by adapting and extending the techniques we will propose to handle bipolar problems and the techniques described Chapter 2 for solving problems with preferences and uncertainty.

When we have only negative preferences, uncertainty can be eliminated by transforming constraints among controllable and uncontrollable variables into suitable constraints on controllable variables only. When we consider also positive preferences, a similar technique can be used, while maintaining similar properties, despite the fact that positive and negative preferences are combined by different operators.

The resulting problem is then a bipolar problem (BP) where however each partial instantiation can have both a positive and a negative preference. Such a pair of elements is then used to associate to each solution an overall preference level and an overall robustness level.

Compensation of positive with negative preferences can be done via an operator which is not associative. This does not allow for preference compensation within the constraints. However, preference compensation can be performed at the level of complete solutions, thus allowing us to associate two elements to each solution: a preference degree and a robustness degree. Depending on the attitude we have towards risk, we can then order solutions by using a Pareto or a lexicographic approach over such two degrees.

The work presented in this chapter has appeared in the proceedings of the following conferences and workshops.

- S. Bistarelli, M. S. Pini, F. Rossi and K. B. Venable. Positive and negative preferences, *In Proceedings of the 7th International Workshop on Preferences and Soft Constraints*, held in conjunction with the 11th International Conference on Principles and Practice of Constraint Programming (CP 2005), Sitges, Spain, October 2005.

- S. Bistarelli, M. S. Pini, F. Rossi and K. B. Venable. Modelling and solving bipolar preference problems. *In Proceedings of 11th Annual ERCIM Workshop on Constraint Solving and Constraint Logic Programming (CSCLP 2006)*, Lisbon, Portugal, June 2006.

- M. S. Pini, F. Rossi and K. B. Venable. Bipolar preference problems. *In Proceedings of the 17th European Conference on Artificial Intelligence (ECAI 2006)*, IOS Press, vol.141, pp. 705-706, Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi and K. B. Venable. Uncertainty in bipolar preference problems. *In Proceedings of the 8th International Workshop on Preferences and Soft Constraints*, held in conjunction with the 12th International Conference on Principles and Practice of Constraint Programming (CP 2006), Nantes, France, September 2006.

- M. S. Pini and F. Rossi. Reasoning on bipolar preference problems. *In Proceedings of the CP 2006 Doctoral Programme*, Nantes, France, September 2006.

The chapter is organized as follows.

- In Section 3.2 we describe how to model negative preferences using classical soft constraint formalism based on c-semirings.
- In Section 3.3 we introduce a new algebraic structure for modelling positive preferences, which has similar properties to a c-semiring, except that the combination of positive preferences gives a higher positive preference as desired and not a lower one.
- In Section 3.4 we give a formal definition of an algebraic structure to model bipolar preferences.
- In Section 3.5 we study the notion of compensation and of its properties (such as associativity). Moreover, we present a technique to build a bipolar preference structure with an associative compensation operator.
- In Section 3.6 starting from bipolar preference structures we define bipolar preference problems and we give a semantics for ordering solutions in a bipolar problem.
- In Section 3.7 we present a real-life problem and we show how to model it as a bipolar problem.
- In Section 3.8 we propose how to adapt branch and bound to solve bipolar problems. We also give the definition of of bipolar propagation and we show its use within a branch and bound solver.
- In Section 3.9 we define bipolar problems with uncertainty and we propose to handle them by extending the procedure illustrated in Chapter 2 for removing uncontrollable variables in preference problems with uncertainty.
- In Sections 3.10 and 3.11 we describe respectively related and future work.

3.2 Negative preferences

The structure we use to model negative preferences is exactly a c-semiring, as defined in Section 2.2. In fact, in a c-semiring the element which acts as indifference is $\mathbf{1}$, since $\forall a \in A$, $a \times \mathbf{1} = a$. Notice that such element, denoted as $\mathbf{1}$ is not necessarily number 1 and in general it can be any element or number (0,1,100,X). This element is the best in the ordering, which is consistent with the fact that indifference is the best preference we can express when using only negative preferences.

Moreover, in a c-semiring, combination goes down in the ordering, since $a \times b \leq a, b$. This can be naturally interpreted as the fact that combining negative preferences worsens the overall preference.

Example 14 This interpretation is very natural when considering, for example, the weighted c-semiring $\langle R^+, min, +, +\infty, 0\rangle$. In fact, in this case the real numbers are costs and thus negative preferences. The sum of different costs is worse in general with respect to the ordering induced by the additive operator (that is, min) of the c-semiring. □

Example 15 Let us now consider the fuzzy c-semiring $\langle [0, 1], max, min, 0, 1\rangle$. According to this interpretation, giving a preference equal to 1 to a tuple means that there is nothing negative about such a tuple. Instead, giving a preference strictly less than 1 (e.g., 0.6) means that there is at least a constraint which such tuple doesn't satisfy at the best. Moreover, combining two fuzzy preferences means taking the minimum and thus the worst among them. □

From now on, we will use a standard c-semiring to model negative preferences, denoted as: $\langle N, +_n, \times_n, \perp_n, \top_n\rangle$.

3.3 Positive preferences

When dealing with positive preferences, we want two main properties to hold: combination should bring to better preferences, and indifference should be lower than all the other positive preferences. These properties can be found in the following structure.

Definition 20 (positive preference structure) A positive preference structure is a tuple $\langle P$, $+_p$, $\times_p$, $\perp_p$, $\top_p\rangle$ such that

- P is a set and $\top_p, \perp_p \in P$;
- $+_p$, the additive operator, is commutative, associative, idempotent, with $\perp_p$ as its unit element ($\forall a \in P, a +_p \perp_p = a$) and $\top_p$ as its absorbing element ($\forall a \in P, a +_p \top_p = \top_p$);
- $\times_p$, the multiplicative operator, is associative, commutative and distributes over $+_p$ ($a \times_p (b +_p c) = (a \times_p b) +_p (a \times_p c)$), with $\perp_p$ as its unit element and $\top_p$ as its

absorbing element[1].

Notice that the additive operator of this structure has the same properties as the corresponding one in c-semirings, and thus it induces a partial order over P in the usual way: $a \leq_p b$ if and only if $a +_p b = b$. This allows to prove that $+_p$ is monotone over $\leq_p$ and that it is the least upper bound in the lattice $(P, \leq_p)$.

On the other hand, the multiplicative operator has different properties. More precisely, the best element in the ordering $(\top_p)$ is now its absorbing element, while the worst element $(\bot_p)$ is its unit element. This reflects the desired behavior of the combination of positive preferences.

Theorem 2 *Given the positive preference structure $\langle P, +_p, \times_p, \bot_p, \top_p \rangle$, consider the relation $\leq_p$ over P. Then:*

- *$\times_p$ is monotone over $\leq_p$. That is, for any $a, b \in P$ such that $a \leq_p b$, then $a \times_p d \leq_p b \times_p d$, $\forall d \in P$.*

- *For any pair $a, b \in P$, $a \times_p b \geq_p a +_p b \geq_p a, b$.*

Proof: Since $a \leq_p b$ if and only if $a +_p b = b$, then $b \times_p d = (a +_p b) \times_p d = (a \times_p d) +_p (b \times_p d)$. Thus $a \times_p d \leq_p b \times_p d$. Also, $a \times_p b$ = $a \times_p (b + \bot_p)$ = $(a \times_p b) + (a \times_p \bot_p)$ = $(a \times_p b) + a$. Thus $a \times_p b \geq_p a$ (the same for b). Finally, $a \times_p b \geq a, b$. Thus $a \times_p b \geq lub(a, b) = a +_p b$. □

In a positive preference structure, $\bot_p$ is the element modelling indifference. In fact, it is the worst one in the ordering and it is the unit element for the combination operator $\times_p$. These are exactly the desired properties for indifference with respect to positive preferences.

The role of $\top_p$ is to model a very high preference, much higher than all the others. It is analogous to the absorbing bottom element in the set of negative preferences: when complete inconsistency is present in one constraint, many preference aggregation frameworks declare the whole problem inconsistent (for example the conjunctive fuzzy framework). Dually, an absorbing top element is a preference so high that the presence of any other positive preference cannot improve the situation. For example, when buying a house, we could say that having at least two bedrooms is so important that any other positive feature does not result in a higher preference. Of course negative features, combined with the top, could return in a lower preference than the top.

[1]The absorbing nature of $\top_p$ can be derived from the other properties.

Example 16 As a first example of a positive preference structure, consider $P_1 = \langle R^+, max, +, 0, +\infty\rangle$, where preferences are positive reals. The smallest preference that can be assigned is 0. It represents the lack of any positive aspect and can thus be regarded as indifference. Preferences are aggregated taking the sum and are compared taking the max. □

Example 17 Another example is $P_2 = \langle [0, 1], max, max, 0, 1\rangle$. In this case preferences are reals between 0 and 1, as in the fuzzy semiring for negative preferences. However, the combination operator is max, which gives, as a resulting preference, the highest one among all those combined. □

Example 18 As an example of a partially ordered positive preference structure consider the Cartesian product of the two described above: $\langle R^+ \times [0, 1], \langle max, max\rangle, \langle +, max\rangle, \langle 0, 0\rangle, \langle +\infty, 1\rangle\rangle$. Positive preferences, here, are ordered pairs where the first element is a positive preference of type P_1 and the second one is a positive preference of type P_2. Consider for example the (incomparable) pairs $(8, 0.1)$ and $(3, 0.8)$. Applying the multiplicative operator will give pair $(11, 0.8)$ which, as expected, is better than both pairs since both $max(8, 3, 11) = 11$ and $max(0.1, 0.8, 0.8) = 0.8$. □

3.4 Bipolar preference structures

Once we are given a positive and a negative preference structure, a first, naive, way to combine them is by performing the Cartesian product of the two structures. For example, if we have positive structure $\langle P, +_p, \times_p, \bot_p, \top_p\rangle$ and negative structure $\langle N, +_n, \times_n, \bot_n, \top_n\rangle$ the Cartesian product would be $\langle P \times N, \langle +_p, +_n\rangle, \langle \times_p, \times_n\rangle, \langle \bot_p, \bot_n\rangle, \langle \top_p, \top_n\rangle\rangle$. In this setting, given a solution, it will be associated with a pair $\langle p, n\rangle$, where p is the overall positive preference and n is the overall negative preference. Such a pair is an element of the carrier of the new structure. Clearly, the new structure is not a positive nor a negative preference structure, and, in fact, some pairs will be neither clearly positive nor negative. The ordering induced over the pairs is the well known Pareto ordering, which declares as incomparable any two solutions defeating each other on one component. Although simple, this criterion is not satisfactory in practice since we cannot compensate positive and negative preferences, which are two symmetric concepts.

This ability is, instead, one of the key features of another, more sophisticated, bipolar structure which we will now describe.

Definition 21 (bipolar preference structure) A bipolar preference structure is a tuple $\langle N, P, +, \times, \bot, \Box, \top\rangle$ where

- $\langle P, +_{|P}, \times_{|P}, \Box, \top\rangle$ is a positive preference structure;
- $\langle N, +_{|N}, \times_{|N}, \bot, \Box\rangle$ is a c-semiring;
- $+ : (N \cup P)^2 \longrightarrow (N \cup P)$ is such that $a_n + a_p = a_p$ for any $a_n \in N$ and $a_p \in P$; this operator induces as partial ordering on $N \cup P$: $\forall a, b \in P \cup N$, $a \leq b$ if and only if $a + b = b$;
- $\times : (N \cup P)^2 \longrightarrow (N \cup P)$ is an operator (called the *compensation operator*) that, for all $a, b, c \in N \cup P$, satisfies the following properties:
  - commutativity: $a \times b = b \times a$;
  - monotonicity: if $a \leq b$, then $a \times c \leq b \times c$.

In the following, we will write $+_n$ instead of $+_{|N}$ and $+_p$ instead of $+_{|P}$. Similarly for $\times_n$ and $\times_p$. Moreover, we will sometimes write $\times_{np}$ when operator $\times$ will be applied to a pair in $(N \times P)$.

Bipolar preference structures generalize c-semirings. In fact, a c-semiring is just a bipolar preference structure with a single positive preference: the indifference element, which, in such a case, is also the top element of the structure.

Similarly, bipolar preference structures generalize positive structures. In fact, the latter are just bipolar preference structures with a single negative preference: the indifference element. By symmetry, in such cases the indifference element coincides with the bottom element of the structure.

Given the way the ordering is induced by $+$ on $N \cup P$, easily, we have $\bot \leq \Box \leq \top$. Thus, there is a unique maximum element (that is, $\top$), a unique minimum element (that is, $\bot$); the element $\Box$ is smaller than any positive preference and greater than any negative preference, and it is used to model indifference. The shape of a bipolar preference structure is shown in Figure 3.1.

Despite the ordering suggested by Figure 3.1, which places all positive preferences strictly above negative preferences, our framework does not prevent from using the same scale to represent both positive and negative preferences. Such a case can be easily handled by using two isomorphisms: one between an instance of the scale and the positive preference structure, and another one between another instance of the same scale and the negative preference structure. The same holds also when one wishes to use partially overlapping scales.

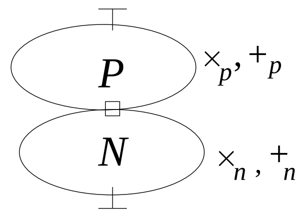

Figure 3.1: The shape of a bipolar preference structure.

A bipolar preference structure allows us to have different ways to model and reason about positive and negative preferences. In fact, we can have different lattices $(P, \leq_p)$ and $(N, \leq_n)$. For example, we can have a richer structure for one kind of preference. This is common in real-life problems, where negative and positive statements are not necessarily expressed using the same granularity. For example, we could be satisfied with just two levels of negative preferences, while requiring ten levels of positive preferences. Nevertheless, our framework allows to model cases in which the two structures are isomorphic, as well.

It is easy to show that the combination of a positive and a negative preference is a preference which is higher than, or equal to, the negative one and lower than, or equal to, the positive one. The following theorems hold when a bipolar preference structure $\langle N, P, +, \times, \bot, \Box, \top \rangle$ is given.

Theorem 3 *For all* $p \in P$ *and* $n \in N$, $n \leq p \times n \leq p$.

Proof: For any $n \in N$ and $p \in P$, $\Box \leq p$ and $n \leq \Box$. By monotonicity of $\times$, we have: $n \times \Box \leq n \times p$ and $n \times p \leq \Box \times p$. Hence: $n = n \times \Box \leq n \times p \leq \Box \times p = p$. $\Box$

This means that the compensation of positive and negative preferences must lie in one of the chains between the two combined preferences. Notice that all such chains pass through the indifference element $\Box$. Possible choices for combining strictly positive with strictly negative preferences are thus the average or the median operator.

Moreover, by monotonicity, we can show that if $\top \times \bot = \bot$, then the result of the compensation between any positive preference and the bottom element is the bottom element, i.e., if there is an event which is so negative that any other positive event doesn't matter, for example the work loss, then every scenario including this event will be in set of the most refused ones. Similarly, if $\top \times \bot = \top$, then the compensation between any negative preference and the top element is the top element, i.e., if there is an event which is so positive that any other negative event doesn't matter, for example winning the lottery, then every scenario including winning the lottery will be in the set of the most preferred ones.

N,P	$+_p$, $\times_p$	$+_n$, $\times_n$	$\times_{np}$	$\bot$, $\Box$, $\top$
R^-, R^+	max, sum	max, sum	sum	$-\infty$, 0, $+\infty$
$[-1,0]$, $[0,1]$	max, max	max, min	sum	-1, 0, 1
$[0,1]$, $[1,+\infty]$	max, prod	max, prod	prod	0, 1, +∞

Table 3.1: Examples of bipolar preference structures.

Theorem 4 *Given bipolar preference structure* $\langle N, P, +, \times, \bot, \Box, \top\rangle$*:*

- *if* $\top \times \bot = \bot$*, then* $\forall p \in P$*,* $p \times \bot = \bot$*;*
- *if* $\top \times \bot = \top$*, then* $\forall n \in N$*,* $n \times \top = \top$.

Proof: Assume $\top \times \bot = \bot$. Since for all $p \in P$, $p \leq \top$, then, by monotonicity of $\times$, $p \times \bot \leq \top \times \bot = \bot$, hence $p \times \bot = \bot$.

Assume $\top \times \bot = \top$. Since for all $n \in N$, $\bot \leq n$, then, by monotonicity of $\times$, $\top = \top \times \bot \leq \top \times n$, hence $\top \times n = \top$. $\Box$

Example 19 In Table 3.1 each row corresponds to a bipolar preference structure. The structure described in the first row uses positive real numbers as positive preferences and negative reals as negative preferences. Compensation is obtained by summing the preferences, while the ordering is given by the max operator, i.e., the most preferred preferences are the highest ones. In the second structure we have positive preferences between 0 and 1 and negative preferences between -1 and 0. The compensation operator between positive preferences is max, between negative preferences is min, between positive and negative preferences is sum and the order is given by max. In the third structure we use positive preferences between 1 and +∞ and negative preferences between 0 and 1. Compensation is obtained by multiplying the preferences and ordering is obtained again via max. The compensation in the first and in the third structure is associative. $\Box$

3.5 Associativity of preference compensation

In general, the compensation operator $\times$ may be not associative. First, we list some sufficient conditions for the non-associativity of the $\times$ operator, then we show how to build a bipolar preference structure with an associative combination operator.

Theorem 5 *Given a bipolar preference structure $\langle P, N, +, \times, \bot, \Box, \top \rangle$, $\times$ is not associative if at least one of the following two conditions is satisfied:*

- $\top \times \bot = c \in (N \cup P) - \{\top, \bot\}$*;*
- $\exists p \in P - \{\top, \Box\}$ *and* $n \in N - \{\bot, \Box\}$ *s.t.* $p \times n = \Box$ *and at least one of the following conditions holds:*
  - $\times_p$ *or* $\times_n$ *is idempotent;*
  - $\exists p' \in P - \{p, \top\}$ *s.t.* $p' \times n = \Box$ *or* $\exists n' \in N - \{n, \bot\}$ *s.t.* $p \times n' = \Box$*;*
  - $\top \times \bot = \bot$ *and* $\exists n' \in N - \{\bot\}$ *s.t.* $n \times n' = \bot$*;*
  - $\top \times \bot = \top$ *and* $\exists p' \in P - \{\top\}$ *s.t.* $p \times p' = \top$*;*
  - $\exists a, c \in N \cup P$ *s.t.* $a \times p = c$ *if and only if* $c \times n \neq a$ *(or* $\exists a, c \in N \cup P$ *s.t.* $a \times n = c$ *if and only if* $c \times p \neq a$*),*

Proof:

- If $c \in P - \{\top\}$, then $\top \times (\top \times \bot) = \top \times c = \top$, while $(\top \times \top) \times \bot = \top \times \bot = c$. If $c \in N - \{\bot\}$, then $\bot \times (\bot \times \top) = \bot \times c = \bot$, while $(\bot \times \bot) \times \top = \bot \times \top = c$.
- Assume that $\exists p \in P - \{\top, \Box\}$ and $n \in N - \{\bot, \Box\}$ such that $p \times n = \Box$.
  - If $\times_p$ is idempotent, then $p \times (p \times n) = p \times \Box = p$, while $(p \times p) \times n = p \times n = \Box$. Similarly if $\times_n$ is idempotent.
  - If $\exists p' \in P - \{p, \top\}$ such that $p' \times n = \Box$, then $(p \times n) \times p' = p'$, while $p \times (n \times p') = p$. Analogously, if $\exists n' \in N - \{n, \bot\}$ such that $p \times n' = \Box$.
  - If $\top \times \bot = \bot$, then, by Theorem 4, $p \times \bot = \bot$. If $\exists n' \in N - \{\bot\}$ such that $n \times n' = \bot$, then $(p \times n) \times n' = \Box \times n' = n'$, while $p \times (n \times n') = p \times \bot = \bot \neq n'$.
  - If $\top \times \bot = \top$, then, by Theorem 4, $n \times \top = \top$. If $\exists p' \in P - \{\top\}$ such that $p \times p' = \top$, then $(n \times p) \times p' = \Box \times p' = p'$, while $n \times (p \times p') = n \times \top = \top \neq p'$.
  - If $c \times n \neq a$, then $(a \times p) \times n = c \times n \neq a$, but $a \times (p \times n) = a \times \Box = a$. Analogously if $c \times p \neq a$.

$\Box$

Notice that sufficient conditions for the non-associativity of the compensation operator presented in Theorem 5 refer to various aspects of a bipolar preference structure such as the properties of operators and the relation between $\times$ and the other operators. Since some of these conditions often occur in practice, it is not reasonable to require associativity of $\times$, we prefer to let the users the freedom of choosing operators as they want on condition that they satisfy the properties required in Definition 21.

For example, $\times$ is not associative when the combination between $\top$ and $\bot$ is different from $\top$ or $\bot$, or when there are two preferences, a positive and a negative one, whose compensation produces the indifference element and the combination operator of either the positive or the negative preferences is idempotent. This result depends on the fact that the proposed framework allows one to choose freely the result of the compensation between $\top$ and $\bot$, and the operators $\times_n$ and $\times_p$, as long as the monotonicity of $\times$ is respected. However, there are also cases in which both $\times_p$ and $\times_n$ are not idempotent, and still $\times$ is not associative. For example, this happens when there are two different positive (respectively negative) preferences that combined with the same negative (respectively positive) preference give the indifference element. Another sufficient condition for the non-associativity of the compensation operator concerns the presence of at least two negative (respectively positive) preferences different from $\bot$ (respectively $\top$), such that their combination is $\bot$ (respectively $\top$). Consider, for example, a bipolar preference structure where N=[-50,0], P=[0,100], +=max, $\times$ = bounded-sum, $\bot = -50, \Box = 0$, and $\top = 100$. In this case, there are preferences such as 50 and 60 which are not equal to the top (100) but such that their bounded sum obtains 100. As expected, $-10 + (50 + 60) = -10 + 100 = 90$, while $(-10+50)+60 = 40+60 = 100$. Another case that leads to non-associativity of $\times$ is when there are two preference values that don't behave like inverse elements in ordinary algebra.

It is however useful to be able to build bipolar preference structures where compensation is associative. It is obvious that, if we are free to choose any positive and any negative preference structure when building the bipolar framework, we will never be able to assure associativity of the compensation operator. Thus, to assure this, we must pose some restrictions on the way a bipolar preference structure is built.

We describe now how to build a positive preference structure from a given negative one where $\times_n$ is not idempotent, such that the resulting bipolar preference structure has an associative compensation operator. The methodology is called *localization* and represents a standard systematic technique for adding multiplicative inverses to a (semi)ring [BH98].

Given a (semi)ring with carrier set N (representing, in our context, a negative preference structure), and a subset $S \subseteq N$, we can construct another structure with carrier set P (representing, for us, a positive preference structure), and a mapping from N to P which makes all elements in the image of S invertible in P. The localization of N by S is also denoted by $S^{-1}N$.

We can select any subset S of N. However, it is usual to select a subset S of N which is closed under $\times_n$, such that $\mathbf{1} \in S$ ($\mathbf{1}$ is the unit for $\times_n$, which represents indifference), and $\mathbf{0} \notin S$.

Given N and S, let us consider the quotient field of N with respect to S. This is denoted by $Quot(N, S)$, and will represent the carrier set of our bipolar structure. One can construct $Quot(N, S)$ by just taking the set of equivalence classes of pairs (n, d), where n and d are elements of N and S respectively, and the equivalence relation is: $(n, d) \equiv (m, b) \iff n \times_n b = m \times_n d$. We can think of the class of (n, d) as the fraction $\frac{n}{d}$.

The embedding of N in $Quot(N, S)$ is given by the mapping $f(n) = (n, \mathbf{1})$, thus the (semi)ring N is a subring of $S^{-1}N$ via the identification $f(a) = \frac{a}{\mathbf{1}}$.

The next step is to define the $+$ and $\times$ operator in $Quot(N, S)$, as function of the operators $+_n$ and $\times_n$ of N. We define $(n, d) + (m, b) = ((n \times_n b) +_n (m \times_n d), d \times_n b)$ and $(n, d) \times (m, b) = (m \times_n n, d \times_n b)$. By using the fraction representation we obtain the usual form where the addition and the multiplication of the formal fractions are defined according to the natural rules: $\frac{a}{s} + \frac{b}{t} = \frac{(a \times_n t) +_n (b \times_n s)}{s \times_n t}$ and $\frac{a}{s} \times \frac{b}{t} = \frac{a \times_n b}{s \times_n t}$.

It can be shown that the structure $\langle P, +_p, \times_p, \frac{\mathbf{1}}{\mathbf{1}}, \frac{\mathbf{1}}{0} \rangle$, where $P = \{\frac{\mathbf{1}}{a} \text{ s.t. } a \in (S \cup \{\mathbf{0}\})\}$, $+_p$ and $\times_p$ are the operators $+$ and $\times$ restricted over $\frac{\mathbf{1}}{S} \times \frac{\mathbf{1}}{S}$, $\frac{\mathbf{1}}{\mathbf{1}}$ is the bottom element in the induced order (notice that the element coincide with $\mathbf{1}$), and $\frac{\mathbf{1}}{\mathbf{0}}$ is the top element of the structure[2], is a positive preference structure. Moreover, $Quot(N, S) = P \cup N$, and it is the carrier of a bipolar preference structure $\langle P, N, +, \times, \mathbf{0}, \frac{\mathbf{1}}{\mathbf{1}}, \frac{\mathbf{1}}{\mathbf{0}} \rangle$ where $\times$ is an associative compensation operator by construction.

Notice that the first example of the table in Section 3.4, as well as the third example restricted to rational numbers, can be obtained via the localization procedure.

3.6 Bipolar preference problems

Once we have defined bipolar preference structures, we can define a notion of bipolar constraint, which is just a constraint where each assignment of values to its variables is associated to one of the elements in a bipolar preference structure.

Definition 22 (bipolar constraints) Given a bipolar preference structure $\langle N, P, +, \times, \bot, \Box, \top \rangle$, a finite set D (the domain of the variables), and an ordered set of variables V, a

[2]This element is introduced ad hoc because $\mathbf{0}$ is not an unit and cannot be used to build its inverse.

constraint is a pair $\langle def, con \rangle$ where $con \subseteq V$ and $def : D^{|con|} \rightarrow (N \cup P)$.

Given a set of bipolar constraints and a set of variables we can define a bipolar constraint satisfaction problem.

Definition 23 (bipolar CSP) A bipolar CSP (V, C) is a set of variables V and a set of bipolar constraints C over V.

There could be many ways of defining the optimal solutions of a bipolar CSP. Here we propose a simple one which compensates only preferences of complete instantiations. This avoids problems due to the possible non-associativity of the compensation operator, since compensation never involves more than two preference values. Thus the preference of a solution does not depend on the order in which the preferences of its constraints are aggregated.

Definition 24 (optimal solutions) A solution of a bipolar CSP (V, C) is a complete assignment to all variables in V, say s, and an associated preference which is computed as follows: $pref(s) = (p_1 \times_p \ldots \times_p p_k) \times (n_1 \times_n \ldots \times_n n_l)$, where $p_i \in P$ and $n_j \in N$ and where $\exists \ \langle def_i, con_i \rangle \in C$ for $i := 1, \ldots, k$ s.t. $p_i = def_i(s \downarrow_{con})$, and $\exists \langle def_j, con_j \rangle \in C$ for $j := 1, \ldots, l$ s.t. $n_j = def_j(s \downarrow_{con})$. A solution s is an optimal solution if there is no other solution s' with $pref(s') > pref(s)$.

In this definition, the preference of a solution s is obtained by combining all the positive preferences associated to its projections over the constraints, by using $\times_p$, combining all the negative preferences associated to its projections over the constraints, by using $\times_n$, and then, combining the two preferences obtained so far (one positive and one negative) by using the operator $\times_{np}$.

If $\times$ is associative, then other definitions of solution preference could be used while giving the same result. In fact, any combination of aggregation and compensation, applied to the preferences of the constraints of the problem, would lead to the same overall preference, and thus to the same solution ordering.

3.7 An example of bipolar CSP

We now show how a real-life problem can be modelled as a bipolar CSP. Consider the scenario in which we want to buy a car and we have preferences over some features. In terms of color, we like red, we are indifferent to white, and we hate black. Also, we like convertible cars a lot and we don't care much for big cars (e.g., SUVs). In terms of engines, we like diesel. However, we don't want a diesel convertible.

We represent positive preferences via positive integers, negative preferences via negative integers and we maximize the sum of all kinds of preferences. This can be modelled by a bipolar preference structure where $N = [-\infty, 0]$, $P = [0, +\infty]$, $+$ =max, $\times$=sum, $\bot = -\infty$, $\Box = 0, \top = +\infty$.

We have three variables: variable T (type) with domain {convertible, big}, variable E (engine) with domain {diesel, gasoline}, and variable C (color) with domain {red, white, black}. For the preferences over the colors, we define a constraint $c_1 = \langle def_1, \{C\}\rangle$ where, for example, we set def_1(red) $= +10$, def_1(black) $= -10$, and def_1(white) $= 0$. We also have a constraint over car types, say $c_2 = \langle def_2, \{T\}\rangle$, where we set def_2(convertible) $= +20$ and def_2(big) $= -3$. The constraint over engines can then be $c_3 = \langle def_3, \{E\}\rangle$, where we can set def_3(diesel) $= +10$ and def_3(gasoline) $= 0$. Finally, the last preference can be modelled by a constraint $c_4 = \langle def_4, \{T, E\}\rangle$, where we can set def_4(convertible, diesel) $= -20$ and $def_4(a, b) = 0$ for $(a, b) \neq$(convertible, diesel). Figure 3.2 shows the structure (variables, domains, constraints, and preferences) of such a bipolar CSP, where preferences have been chosen to fit the informal specification above, and 0 is used to model indifference (also when tuples are not shown).

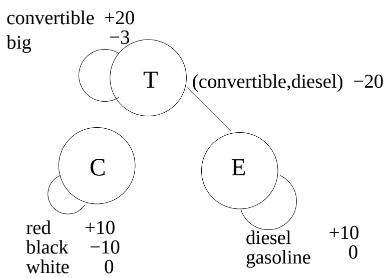

Figure 3.2: A bipolar CSP modelling car's preferences.

Notice that we have set the preference values in a way that models the intuitive strength of the preferences described informally in the example.

Consider solution s_1=(red,convertible,diesel). $pref(s_1) = (def_1$(red)$\times$ def_2(convertible) $\times def_3$(diesel)$)$ $\times$ def_4(convertible, diesel) $= (10 + 20 + 10) + (-20) = 20$. Analogously, we can compute the preference of all other solutions and see that the optimal solution is (red, convertible, gasoline) with global preference of 30.

Consider now a different bipolar preference structure, which differs from the previous one only for $\times_p$, which is now max. Now solution s_1 has preference $pref(s_1)$ =(def_1(red)

$\times def_2$(convertible)$\times def_3$(diesel))$\times def_4$(convertible, diesel) $= max(10, 20, 10)+(-20) = 0$. It is easy to see that now an optimal solution has preference 20. There are two of such solutions: one is the same as the optimal solution above, and the other one is (white,convertible, gasoline). The two cars have the same features except for the color. A white convertible is just as good as a red convertible because we decided to aggregate positive preference by taking the maximum elements rather than by summing them.

3.8 Solving bipolar CSPs

Bipolar problems are NP-complete, since they generalize both classical and soft constraints, which are already known to be difficult problems [BMR97]. In this section we will consider how to adapt some usual techniques for soft constraints to bipolar problems.

3.8.1 A branch and bound solver

Preference problems based on c-semirings can be solved via a branch and bound technique, possibly augmented via soft constraint propagation, which may lower the preferences and thus allow for the computation of better bounds [BMR97].

In bipolar CSPs, we have both positive and negative preferences. We propose to use an algorithm similar to Branch and Bound algorithm (BB) [Dec03] used for unipolar preferences. Being able to do so is a good point since it allows to handle bipolar preferences without much additional effort.

Following BB, whenever a solution is found, its preference, if higher that those found before, is kept as a lower bound, L, for the optimal preference in the maximization task. Moreover, for each partial solution t an upper bound, $ub(t)$, is computed by overestimating the best preference of a solution extending t. If $ub(t) \leq L$, i.e. the preference of the best solution in the subtree below t is worse than the preference of the best solution found so far, then the subtree below t is pruned.

Our algorithm is different from standard BB since it allows the compensation operator to be non-associative. This may require to consider some total completions of t in order to compute $ub(t)$.

More precisely, we adapt BB to compute, at each search node k corresponding to a partial assignment t, an upper bound to the preferences of all the solutions in the k-rooted subtree as follows.

- If $\times$ is not associative, then each node is associated to a positive and a negative preference, say p and n, which are obtained by aggregating all preferences of the same type obtained in the instantiated part of the problem. Next all the best preferences (which may be positive or negative) in the uninstantiated part of the problem are considered. By aggregating those of the same type, we get a positive and a negative preference, say p' and n', which can be combined with the ones associated to the current node. This produces the following upper bound $ub = (p \times_p p') \times (n \times_n n')$, where $p' = p_1 \times_p \ldots \times_p p_w$, $n' = n_1 \times_n \ldots \times_n n_s$, with $w + s = r$, where r is the number of uninstantiated variables/constraints. Hence ub can be computed via $r - 1$ aggregation steps and one compensation step.

- If $\times$ is associative, then we don't need to postpone compensation until all constraints have been considered. This means that we can keep just one preference value for each search node, $v = p \times n$, that can be positive or negative, which is obtained by aggregating all preferences (both positive and negative) obtained in the instantiated part of the problem. The same can be done considering the best preferences in the uninstantiated part of the problem, obtaining a value v'. Thus, ub can now be written as $ub = v \times v'$, where $v' = a_1 \times \ldots \times a_r$, where $a_i \in N \cup P$ is the best preference found in a constraint of the uninstantiated part of the problem. Thus now ub can be computed via at most $r - 1$ steps among which there can be many compensation steps. A compensation step can generate the indifference element $\square$, which is the unit element for the compensation operator. Thus, when $\square$ is generated, the successive computation step can be avoided.

Algorithm 5 shows the pseudocode of the procedure we propose to compute the upper bound within the BB algorithm. The input is a partial assignment t to a subset $X = \{x_1 \ldots, x_k\}$ of the set of variables $V = \{x_1, \ldots x_n\}$ and the bipolar CSP, P', obtained from the initial bipolar CSP by reducing the domains of the variables in X to the singleton corresponding to their assignment in t.

For every constraint $c =< def, con >\in C$, we compute the constraint $c \Downarrow_{X,t}$, which is obtained by projecting c on X and considering only the subtuple $t \downarrow_{X \cap con}$. We will denote $c \Downarrow_{X,t}$ with c' and we will denote with C' the union set of all such constraints. Note that, by the definition of projection constraint (Section 2.2), $c \Downarrow_{X,t}$ associates to subtuple $t \downarrow_{X \cap con}$ the best preference associated by def to any of its completions to variables in con.

If $\times$ is not associative, then the algorithm computes the aggregation $p(t)$ of all the best preferences that are positive, i.e., the preferences obtained on each constraint $c^+ =<$

$def_{c^+}, con_{c^+} >\in C'$ such that $def_{c^+}(t \downarrow_{con_{c^+}}) \in P$ and the aggregation $n(t)$ of all the best preferences that are negative, i.e. the preferences obtained on each constraint $c^- =< def_{c^-}, con_{c^-} >\in C'$ s. t. $def_{c^-}(t \downarrow_{con_{c^-}}) \in N$. The final step compensates between $p(t)$ and $n(t)$ and returns the result, $ub(t)$, of this compensation.

If $\times$ is associative then the algorithm aggregates directly the best preferences that can be positive or negative and it returns the result of this aggregation, i.e. $ub(t)$.

Algorithm 5: Upper Bound computation

Input: t: assignment to variables in $X = \{x_1, \ldots, x_k\}$

P': bipolar CSP;

Output: ub(t): preference;

foreach $c \in C$ **do**

  compute $c' = c \Downarrow_{X,t}$

$C' \leftarrow \cup_{c \in C} c'$;

if $\times$ *is not associative* **then**

  $p(t) \leftarrow \prod_{p\ \{c^+ \in C'\}} def_{c^+}(t \downarrow_{con_{c^+}})$;

  $n(t) \leftarrow \prod_{n\ \{c^- \in C'\}} def_{c^-}(t \downarrow_{con_{c^-}})$;

  $ub(t) \leftarrow p(t) \times n(t)$

else

  $ub(t) \leftarrow \prod_{\{c' \in C'\}} def_{c'}(t \downarrow_{con_{c'}})$;

return $ub(t)$;

3.8.2 Bipolar propagation

When looking for an optimal solution, BB can be helped by some form of partial or full constraint propagation. To see whether this can be done when solving bipolar problems as well, we must first understand what constraint propagation means in such problems. For sake of simplicity, we will focus here on arc-consistency.

Given any bipolar constraint, let us first define its negative version $neg(c)$, which is obtained by just replacing the positive preferences via indifference. Similarly, the positive version $pos(c)$ is obtained by replacing negative preferences via indifference. We now define when a constraint is negative arc-consistent and when it is positive arc-consistent. In the following definitions we consider a bipolar preference structure $\langle N, P, +, \times, \bot, \Box, \top \rangle$, we denote with c_{XY} a binary bipolar constraint connecting two variables X and Y, with c_X the soft domain of X and with c_Y the soft domain of Y.

Definition 25 (NAC) A binary bipolar constraint c_{XY} is Negatively Arc-Consistent (NAC) if and only if $neg(c_{XY})$ is soft arc-consistent, i.e., if and only if $neg(c_X) = (neg(c_X) \times_n neg(c_Y) \times_n neg(c_{XY})) \Downarrow_X$ and $neg(c_Y) = (neg(c_X) \times_n neg(c_Y) \times_n neg(c_{XY})) \Downarrow_Y$.

If a binary bipolar constraint is not soft arc-consistent, we can make it NAC by modifying the soft domains of its two variables such that the two equations above hold. The modifications required can only decrease some preference values. Thus some negative preferences can become more negative than before. If operator $\times_n$ is idempotent, then such modifications generate a new constraint which is equivalent to the given one [BMR97].

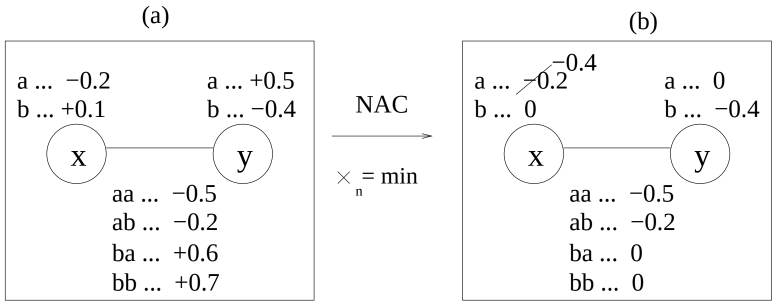

Figure 3.3: How to make a bipolar constraint NAC.

Example 20 In Figure 3.3 it is shown how to make a bipolar constraint Negatively Arc-Consistent. Part (a) shows a bipolar constraint, named c_{XY}, linking two variables, named X and Y, where positive preferences are defined in interval $[0, 1]$ and negative preferences in interval $[-1, 0]$. Part (b) presents the negative version of c_{XY}, that becomes NAC, if we assume to combine negative preferences via minimum operator, by decreasing the negative preference associated to $X = a$ from -0.2 to -0.4. □

Let us now consider the positive version of a bipolar constraint.

Definition 26 (PAC) A binary bipolar constraint c_{XY} is Positively Arc-Consistent (PAC) if and only if $c_X = glb_X(pos(c_X) \times_p pos(c_Y) \times_p pos(c_{XY}))$ and $c_Y = glb_Y(pos(c_X) \times_p pos(c_Y) \times_p pos(c_{XY}))$, where glb_X is an operator which, taken any constraint c_S over variables S such that $X \in S$, computes a new constraint over X as follows: for every value a in the domain of X, its preference is computed by taking the greatest lower bound of all preferences given by c_S to tuples containing $X = a$.

If a binary bipolar constraint is not positive arc-consistent, we can make it PAC by modifying the soft domains of its two variables such that the two equations above hold. The modifications required can only involve the increase of some preference values. Thus some positive preferences can become more positive than before. If operator $\times_p$ is idempotent, such modifications generate a new constraint which is equivalent to the given one.

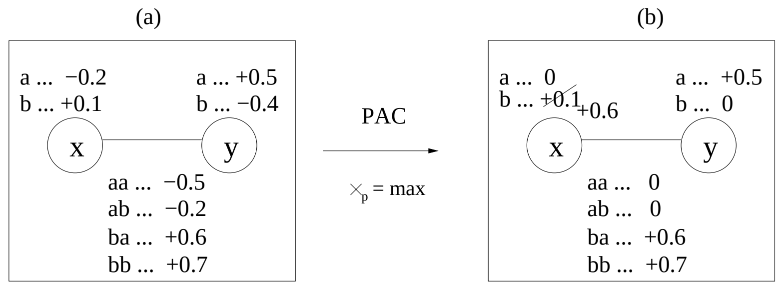

Figure 3.4: How to make a bipolar constraint PAC.

Example 21 In Figure 3.4 it is shown how to make a bipolar constraint Positively Arc-Consistent. In Part (a) we show the same bipolar constraint, c_{XY}, presented in Figure 3.3 (a). In Part (b) we present the positive version of c_{XY}, that becomes PAC, if we assume to combine positive preferences via maximum operator, by increasing the positive preference associated to $X = b$ from $+0.1$ to $+0.6$. □

We now explain when a binary bipolar constraint is Bipolar Arc-Consistent and when a bipolar problem is Bipolar Arc-Consistent.

Definition 27 (BAC) A binary bipolar constraint is Bipolar Arc-Consistent (BAC) if and only if it is both NAC and PAC. A bipolar constraint problem is BAC if and only if all its constraints are BAC.

If a bipolar constraint problem is not BAC, we can consider its negative and positive versions and achieve PAC and NAC on them. If both $\times_n$ and $\times_p$ are idempotent, this can be seen as the application of functions which are monotone, inflationary, and idempotent on a suitable partial order. Thus usual algorithms based on chaotic iterations [Apt03] can be used, with the assurance of terminating and having a unique equivalent result which is independent of the order in which constraints are considered. However, this can generate two versions of the problem (of which one is NAC and the other one is PAC) which could be impossible to reconcile. The problem can be solved by achieving only partial forms of PAC and NAC in a bipolar problem. The basic idea is to consider the given bipolar problem, apply the NAC and PAC algorithms to its negative and positive versions, and then modify the preferences of the original problem only when the two new versions can be reconciled, that is, when at least one of the two new preferences is the indifference element. In fact, this means that, in one of the two consistency algorithms, no change has been made. If this holds, the other preference is used to modify the original one. This algorithm achieves a partial form of BAC, that we call p-BAC, and assures equivalence.

Notice that this algorithm will possibly decrease some negative preferences and increase some positive preferences. Therefore, if we use constraint propagation to improve the bounds in a branch and bound algorithm, it will actually sometimes produce worse bounds, due to the increase of the positive preferences. We will thus use only the propagation of negative preferences (that is, NAC) within a BB algorithm. Since the upper bound is just a combination of several preferences, and since preference combination is monotonic, lower preferences give a lower, and thus better, upper bound.

Example 22 In Figure 3.5 it is shown how to make a bipolar constraint partially Bipolar Arc-Consistent. We consider in Part (a) the same bipolar constraint, c_{XY}, illustrated in Part (a) of Figures 3.3 and 3.4. In Part (b) and Part (c) we recall how to make positive version of c_{XY} PAC and how to make negative version of c_{XY} NAC, assuming to combine negative preferences via minimum operator and positive preferences via maximum operator. In Part (d) we show how to achieve p-BAC of c_{XY}. For obtaining p-BAC we must reconcile the modified preferences obtained in Part (b) and in Part (c) when it is possible. Since in this example it is always possible to reconcile such preferences, we obtain a bipolar constraint which is not only p-BAC, but also BAC. □

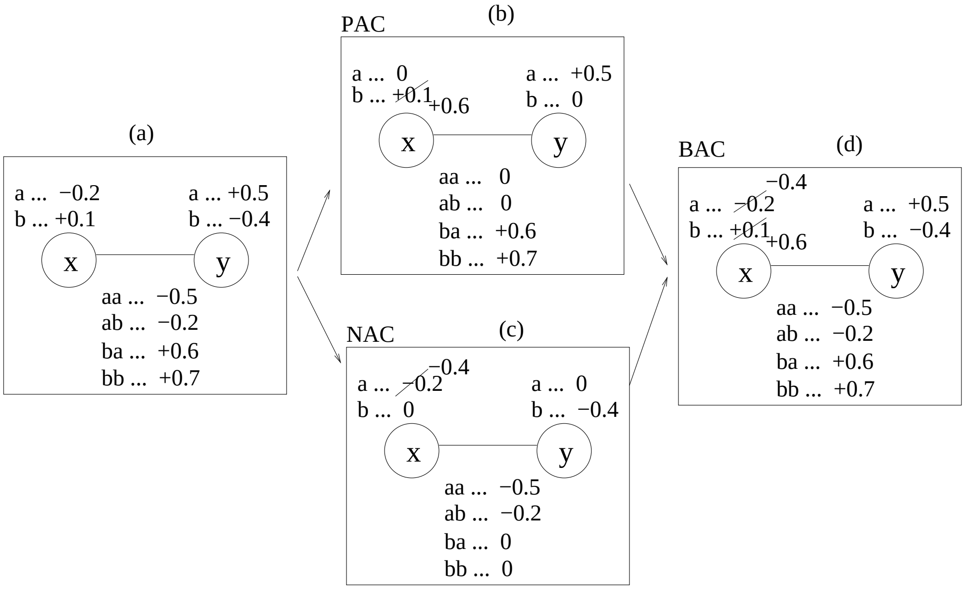

Figure 3.5: How to make a bipolar constraint p-BAC.

3.9 Bipolar preferences and uncertainty

In this section we define bipolar preference problems with uncertainty and we give an algorithm for translating them in a new kind of bipolar preference problems without uncertainty. Then we give a way for computing the preference of a solution of these new problems and we show that the same semantics mentioned in Section 2.13.3 can be used here for ordering the solutions.

Uncertain bipolar preference problems are problems that are characterized by a set of variables, which can be controllable or uncontrollable, and by a set of bipolar constraints. We assume that the domain of every uncontrollable variable is equipped with a possibility distribution, that specifies, for every value in the domain, the degree of plausibility that the variable takes that value.

Definition 28 (UBCSP) An uncertain bipolar CSP (UBCSP) is a tuple $\langle bS,\, V_c,\, V_u,\, bC = bC_f \cup bC_{fu}\rangle$, where

- $bS = \langle N,\, P,\, +,\, \times,\, \bot,\, \Box,\, \top\rangle$ is a bipolar preference structure;
- $V_c = \{x_1, \ldots x_n\}$ is the set of controllable variables,
- $V_u = \{z_1, \ldots z_k\}$ is the set of uncontrollable variables with possibility distributions $\{\pi_1, \ldots \pi_k\}$,
- $bC = bC_f \cup bC_{fu}$ is the set of bipolar constraints, that may involve any subset of variables of $V_c \cup V_u$. More precisely, constraints in bC_f involve only on a subset of controllable variables of V_c, while constraints in bC_{fu} involve both a subset of variables of V_c and a subset of variables in V_u.

3.9.1 Removing uncertainty from UBCSPs

We now describe an algorithm, that we call *B-SP*, for handling UBCSPs, that generalizes algorithm *SP*, described in Section 2.4 for fuzzy preferences, to the case of positive and negative totally ordered preferences. This algorithm takes in input an uncertain bipolar preference problem $BQ = \langle bS, V_c, V_u, BC = BC_f \cup BC_{fu}\rangle$, where $bS = \langle N,\, P,\, +,\, \times,\, \bot,\, \Box,\, \top\rangle$, N and P are totally ordered sets with respect to the ordering induced by $+$ and it returns a new kind of bipolar preference problem without uncertainty. The algorithm is mainly characterized by two steps: in the first one it transforms the given UBCSP in a new kind of bipolar problem with uncertainty, in order to be able to handle separately the positive and the negative preferences, and in the second one it removes uncertainty from this problem.

1st step: translation into a new kind of UBCSP.

Since we are not assuming that the compensation operator $\times$ of bS is associative, then, for avoiding problems due to non-associativity, we translate the given UBCSP BQ into a new kind of bipolar preference problem, that allows to handle separately its positive and the negative preferences. In order to get this, we introduce 2-bipolar constraints, that are similar to bipolar constraints, except that they associate to each assignment not a unique (positive or negative) value, but a pair of values, that is, a positive and a negative one. We consider also 2-bipolar CSPs, that are just a set of variables and a set of 2-bipolar constraints over these variables.

The first step of *B-SP* regards the translation of every constraint $Bc = \langle \mu, con \rangle$ in BC into a corresponding 2-bipolar constraint $bc = \langle b\mu, con \rangle$ as follows. For every assignment d to variables in con, if $\mu(d) \in P$, then $b\mu(d) = (\mu(d), \Box)$, whereas if $\mu(d) \in N$, then $b\mu(d) = (\Box, \mu(d))$, i.e., if the starting preference of d is positive, then we put that preference in the first component of the pair, and indifference in the other component, otherwise, we put starting negative preference in the second component of the pair and indifference in the other one.

Doing so for every constraint of bC, we obtain an uncertain 2-bipolar CSP $bQ = \langle bS, V_c, V_u, bC = bC_f \cup bC_{fu} \rangle$, which is like the uncertain bipolar preference problem BQ except that every constraint respectively in BC_f, and BC_{fu} is translated in the corresponding 2-bipolar constraint respectively in bC_f and bC_{fu}. Since now bQ is a problem with uncertainty that keep separate positive and negative preferences, then we can reason separately with these two kinds of preferences.

2nd step: elimination of uncertainty.

The next step is characterized by the translation of the 2-bipolar CSP bQ with uncertainty in a 2-bipolar CSP without uncertainty $bQ' = \langle bS, V_c, bC' = bC_f \cup bC_p \cup bC_u \rangle$. This is obtained by eliminating the uncontrollable variables and the 2-bipolar constraints in bC_{fu} relating controllable and uncontrollable variables and by adding new 2-bipolar constraints only among these controllable variables. These new constraints, that can be classified in two sets of constraints, that we call bC_u and bC_p, generalize the constraints in C_u and C_p computed by *SP*. We recall that in *SP* constraints in C_u are obtained by applying a specific procedure for removing uncontrollability and constraints in C_p are computed for recalling the best preference that can be obtained in the removed constraints.

Constraints in bC_u. Every 2-bipolar constraint $bc = \langle b\mu, con \rangle$ in bC_{fu}, i.e. such that $con \cap V_c = X$ and $con \cap V_u = Z$ is translated into a 2-bipolar constraint $bc' = \langle b\mu', con' \rangle$ in bC_u, where $con' = X$, such that for every assignment (d, a) to $X \times Z$, with $b\mu(d, a) = (b\mu_{pos}(d, a), b\mu_{neg}(d, a))$, $b\mu'(d) = (b\mu'_{pos}(d), b\mu'_{neg}(d))$, where $b\mu'_{pos}(d)$ and $b\mu'_{neg}(d)$ are obtained by applying a formula similar to the one presented in Section 2.4 considering respectively $b\mu_{pos}(d, a)$ and $b\mu_{neg}(d, a)$ instead of $\mu(d, a)$.

Recall that in *SP* every constraint $\langle \mu, con \rangle$ in C_{fu}, i.e. such that $con \cap V_c = X$ and $con \cap V_u = Z$, is translated in a constraint $\langle \mu', con' \rangle$ in C_u, where $con' = X$ and for every assignment d to X, μ' is defined as follows [DFP96a]: $\mu'(d) = inf_{a \in A_Z} max(\mu(d, a), c(\pi_Z(a)))$, where c is the order reversing map in $[0, 1]$ such that $c(p) = 1 - p$ and where π_Z is the possibility distribution of Z, which has domain A_Z. This definition depends on the assumption of commensurability between preferences and possibilities, that can be done since fuzzy preferences and possibilities are defined in the same scale (i.e., in $[0, 1]$). It depends also on the fact that the maximum operator is the additive operator of the fuzzy c-semiring and on the fact that c is an order reversing map in $[0, 1]$ with respect to the ordering induced by the maximum operator such that $c(c(p)) = p, \forall p \in [0, 1]$.

Since we want to use a similar formula for both positive and negative preferences, but the set of positive and negative preferences, i.e., P and N, are not necessarily the interval $[0, 1]$, we propose to map in $[0, 1]$ the positive and the negative preferences of every assignment $(d, a) \in X \times Z$ in every constraint $bc \in bC_{fu}$. We perform this mapping via functions, that we call respectively f_p and f_n, that are strictly monotone functions with respect to $\leq_S$. More precisely, if $P = [a, b]$ (respectively $N = [a, b]$) with $a < b$, then f_p (respectively f_n): $[a, b] \to [0, 1]$ associates to every $x \in [a, b]$ the value $\frac{x+|a|}{b+|a|} \in [0, 1]$. Then we can apply the formula recalled above, by replacing the maximum operator with operator $+$ of bS, the map c with a map c_S which reverses the ordering in $[0, 1]$ with respect to the ordering $\leq_S$ induced by $+$ of bS and by assuming that operator inf applied to a set A returns the worst element of A with respect to the ordering $\leq_S$. Since all the other preferences in the problems are in P and N, then we map again in P and N the values returned by the formula, by using respectively the inverse functions f_p^{-1} and f_n^{-1}. f_p^{-1} (respectively f_n^{-1}): $[0, 1] \to [a, b]$ associates to every $y \in [0, 1]$ the value $[y(b + |a|) - |a|] \in [a, b]$. Notice that the fact that f_p and f_n are strictly monotone functions with respect to the ordering $\leq_S$ induced by the operator $+$ of bS, implies that they are invertible and their inverse functions are monotone with respect to the same ordering [Mar95].

More formally, we build the set bC_u from bC_{fu} as follows.

1. Every 2-bipolar constraint $bc = \langle b\mu, con \rangle$ in bC_{fu} such that $con \cap V_c = X$ and

$con \cap V_u = Z$, is translated in a 2-bipolar constraint with preferences in $[0,1]$, $bc^* = \langle b\mu^*, con\rangle$, where, for every assignment (d,a) to $X \times Z$, $b\mu^*(d,a) = (b\mu^*_{pos}(d,a),$ $b\mu^*_{neg}(d,a))$ and $b\mu^*_{pos}(d,a) = f_p(b\mu_{pos}(d,a)) \in [0,1]$ and $b\mu^*_{neg}(d,a) = f_n(b\mu_{neg}(d,a))$ $\in [0,1]$.

2. Then bc^* is translated into the 2-bipolar constraint $bc^{*'} = \langle b\mu^{*'}, con^{*'} = X\rangle$, only on controllable variables, where for every assignment d to X, $b\mu^{*'}(d) = (b\mu^{*'}_{pos}(d), b\mu^{*'}_{neg}(d))$ and $b\mu^{*'}_{pos}(d)$ and $b\mu^{*'}_{neg}(d)$ are computed by following a procedure similar to the one described above, that is, $b\mu^{*'}_{pos}(d) = inf_{a\in A_Z}(b\mu^*_{pos}(d,a) + c_S(\pi_Z(a)))$ and $b\mu^{*'}_{neg}(d) = inf_{a\in A_Z}(b\mu^*_{neg}(d,a) + c_S(\pi_Z(a)))$, where c_S is an order reversing map with respect to $\leq_S$ in $[0,1]$, such that $c_S(c_S(p)) = p$.

3. Finally, $bc^{*'}$ is translated in a new 2-bipolar constraint $bc' = \langle b\mu', con' = X\rangle$ in bC_u where for every assignment d to X, $b\mu'(d) = (b\mu'_{pos}(d), b\mu'_{neg}(d)) \in P \times N$, where $b\mu'_{pos}(d) = f_p^{-1}(b\mu^{*'}_{pos}(d))$ and $\mu'_{neg}(d) = f_n^{-1}(b\mu^{*'}_{neg}(d))$.

In the following we show formally that the considered functions f_p and f_n are strictly monotone with respect to the ordering induced by $+$ of bS. Then we show that, in the general framework that we have defined above, properties which are similar to the ones presented for fuzzy preferences continue to hold.

Lemma 1 *Given $a, b \in \mathcal{R}$, with $a < b$, the function f_p (respectively f_n): $[a,b] \to [0,1]$ associating to every $x \in [a,b]$ the value $\frac{x+|a|}{b+|a|} \in [0,1]$ is strictly monotone. Moreover, if we consider a bipolar preference structure $bS = (N, P, +, \times, \bot, \square, \top)$ where P and N are totally ordered sets, then f_p and f_n are strictly monotone with respect to the ordering induced by the additive operator of bS.*

Proof: Given $x \in [a,b]$, $f_p(x) = \frac{x}{b+|a|} + \frac{|a|}{b+|a|}$. Let us denote with K the value $\frac{1}{b+|a|}$ and with G the value $\frac{|a|}{b+|a|}$. Then $f_p(x) = Kx + G$. We have $K = \frac{1}{b+|a|} > 0$, since $1 > 0$ and $b + |a| > 0$. In fact, if $b > 0$ then $b + |a| \geq b + 0 > 0 + 0 = 0$ and if $b \leq 0$, then $a < b \leq 0$, hence $-a > -b \geq 0$ and so $b + |a| = b + (-a) > b + (-b) = 0$. We have also $G \geq 0$, since $G = \frac{|a|}{b+|a|} = |a|K$ where $K > 0$ and $|a| \geq 0$. Hence we have $f_p(x) = Kx + G$, where $K > 0$ and $G \geq 0$. Let us consider x_1 and x_2 in $[a,b]$. If $x_1 < x_2$, then $Kx_1 < Kx_2$, by the strict monotonicity of the product over real numbers when we multiply for a real number $K > 0$, and so $Kx_1 + G < Kx_2 + G$, where $G \geq 0$, for strict monotonicity of the sum over real numbers.

Given the bipolar preference structure bS, then f_p is strictly monotone with respect to the ordering $\leq_S$ induced by its additive operator $+$. In fact, let us consider x_1 and x_2

in $P = [a, b]$. If $x_1 <_S x_2$, then since P is totally ordered, $x_1 < x_2$ or $x_1 > x_2$. If $x_1 > x_2$ then, for the first part of the proof, $f_p(x_1) > f_p(x_2)$. Analogously, if $x_1 < x_2$, then $f_p(x_1) < f_p(x_2)$. Hence, if $x_1 <_S x_2$, then $f_p(x_1) <_S f_p(x_2)$. Similarly we can show that f_n is strictly monotone with respect to $\leq_S$. $\square$

Notice that the property described in Section 2.4 characterizing the preference function μ' of every constraint in C_u (i.e., $\mu'(d) \geq \alpha$ if and only if, when $\pi_Z(a) > c(\alpha)$, then $\mu(d, a) \geq \alpha$, where a is the actual value of z and c is the order reversing map in $[0, 1]$ s.t. $c(p) = 1 - p$) holds also in our framework for both $b\mu'_{pos}(d)$ and $b\mu'_{neg}(d)$.

Proposition 22 *Consider an uncertain 2-bipolar CSP $\langle bS, V_c, V_u, bC\rangle$, where $bS = \langle N, P, +, \times, \bot, \square, \top\rangle$ is a bipolar preference structure where P and N are totally ordered sets. Every 2-bipolar constraint, $\langle b\mu, con\rangle \in bC$, such that $con \cap V_c = X$ and $con \cap V_u = Z$, with possibility distribution π_Z, such that if d is an assignment to X, and a an assignment to Z, its preference is $b\mu(d, a) = (b\mu_{pos}(d, a), b\mu_{neg}(d, a))$, can be translated in a new constraint, $\langle b\mu', con'\rangle$, where $con' = X$ and $b\mu'$ is such that,*

- *$b\mu'_{pos}(d) \geq_S \beta \in P$ if and only if, when $\pi_Z(a) > c_S(f_p(\beta))$, then $b\mu_{pos}(d, a) \geq_S \beta$;*
- *$b\mu'_{neg}(d) \geq_S \alpha \in N$ if and only if, when $\pi_Z(a) > c_S(f_n(\alpha))$, then $b\mu_{neg}(d, a) \geq_S \alpha$,*

where c_S is an order reversing map with respect to ordering $\leq_S$ in $[0, 1]$ such that $c_S(c_S(p)) = p, \forall p \in [0, 1]$

Proof: We show the first statement concerning $b\mu'_{pos}(d)$. The second one, concerning $b\mu'_{neg}(d)$, can be proved analogously, since by construction f_n and f_n^{-1} have the same properties respectively of f_p and f_p^{-1}.

We recall that $b\mu'_{pos}(d) = f_p^{-1}(inf_{a\in A_Z}\ (f_p(b\mu_{pos}(d, a)) + c_S(\pi_Z(a))))$.

($\Rightarrow$) We assume that $b\mu'_{pos}(d) \geq_S \beta$. If $b\mu'_{pos}(d) \geq_S \beta$, since f_p is monotone with respect to the ordering $\leq_S$, then $f_p(b\mu'_{pos}(d)) \geq_S f_p(\beta)$, i.e., $f_p(f_p^{-1}(inf_{a\in A_Z}\ (f_p(b\mu_{pos}(d, a)) + c_S(\pi_Z(a))))) \geq_S f_p(\beta)$, that is, since f_p is the inverse function of f_p^{-1}, $inf_{a\in A_Z}\ (f_p(b\mu_{pos}(d, a) + c_S(\pi_Z(a))) \geq_S f_p(\beta)$. Since we are considering totally ordered preferences, this implies that $(f_p(b\mu_{pos}(d, a)) + c_S(\pi_Z(a))) \geq f_p(\beta), \forall a \in A_Z$. For a with $\pi_Z(a) > c_S(f_p(\beta))$, since c_S is an order reversing map with respect to $\leq_S$ such that $c_S(c_S(p)) = p$, we have $c_S(\pi_Z(a)) <_S c_S(c_S(f_p(\beta)) = f_p(\beta)$. Therefore for such a value a, $f_p(b\mu_{pos}(d, a)) = (f_p(b\mu_{pos}(d, a)) + c_S(\pi_Z(a))) \geq_S f_p(\beta)$ and, since f_p^{-1} is monotone, we have $f_p^{-1}(f_p(b\mu_{pos}(d, a))) \geq_S f_p^{-1}(f_p(\beta))$, i.e., $b\mu_{pos}(d, a) \geq_S \beta$.

($\Leftarrow$) We assume that $\forall a$ with $\pi_Z(a) > c_S(f_p(\beta))$, $b\mu_{pos}(d, a) \geq_S \beta$. Then, for such a, since f_p is monotone with respect to $\leq_S$, $f_p(b\mu_{pos}(d,a)) \geq_S f_p(\beta)$ and so, $(f_p(b\mu_{pos}(d,a))+ c_S(\pi_Z(a))) \geq_S f_p(\beta)$. On the other hand, for every a such that $\pi_Z(a) < c(f_p(\beta))$, we have $c(\pi_Z(a)) >_S f_p(\beta)$ and so $(f_p(b\mu_{pos}(d,a))+ c_S(\pi_Z(a))) >_S f_p(\beta)$. Thus for every $a \in A_Z$, $(f_p(b\mu_{pos}(d,a))+ c_S(\pi_Z(a))) \geq_S f_p(\beta)$ and so $inf_{a\in A_Z}(f_p(b\mu_{pos}(d,a)) + c_S(\pi_Z(a))) \geq_S f_p(\beta)$. Hence, since f_p^{-1} is monotone, $f_p^{-1}(inf_{a\in A_Z}\ (f_p(b\mu_{pos}(d,\ a)) + c_S(\pi_Z(a)))) \geq_S f_p^{-1}(f_p(\beta))$, i.e., $b\mu'_{pos}(d) \geq_S \beta$. $\square$

Notice also that the procedure above for removing uncontrollability holds both for positive and negative preferences, since it is not based on the combination operators ($\times_p$ and $\times_n$) of positive and negative preferences, which have different behaviours, but only on the positive and negative operators (i.e., $+_p$ and $+_n$) inducing the ordering which satisfy similar properties.

Constraints in bC_p. Constraints in bC_p generalize constraints in C_p of *SP*. Recall that constraints in C_p are added to the resulting problem without uncertainty, in order to avoid having solutions with satisfaction degree F strictly better than the best one in the original problem. In the case of fuzzy preferences, adding these constraints is useful, since the aggregation of fuzzy preferences goes down in the ordering. This is also reasonable for the negative preferences whose combination follows the same behaviour. For the positive preferences, instead, where the combination goes up in the ordering, is reasonable to save the worst positive preference obtained in the original problem, in order to avoid to give a solution with positive degree of satisfaction that is strictly lower than the ones that can be effectively obtained.

Hence, we define the set of constraints bC_p as follows. Given a 2-bipolar constraint $bc =< b\mu, con >$ in bC_{fu}, such that $con \cap V_c = X$ and $con \cap V_u = Z$, then the corresponding 2-bipolar constraint in bC_p is $bc_p =< b\mu_p, con_p = X >$, and μ_p is such that for every assignment d to X, $b\mu_p(d) = (b\mu_{p_{pos}}(d), b\mu_{p_{neg}}(d)) \in P \times N$, where $b\mu_{p_{neg}}(d)$ (respectively $b\mu_{p_{pos}}(d)$) is the best negative (respectively the worst positive) preference that can be reached for d in bc when we consider the various values a in the domain of the uncontrollable variables in con, i.e., $b\mu_{p_{neg}}(d) = \sum_{n_{\{a\in A_Z\}}} b\mu_{neg}(d,a)$ and $b\mu_{p_{pos}}(d) = \inf_{P_{\{a\in A_Z\}}} b\mu_{pos}(d,a)$, where A_Z is the domain of Z, $\sum_n$ is the operator $+_n$ of the negative preferences applied to more than two negative preferences that returns the best negative preference and inf_p is the operator that, applied to a set of positive preferences, returns its worst positive preference with respect to ordering induced by $+_p$.

3.9.2 Solution ordering

Once we have the problem without uncertainty bQ' returned by *B-SP*, we can associate to each solution of bQ' both a degree of satisfaction and a degree of robustness. More precisely, for every solution s of bQ', i.e. for every complete assignment to V_c, we compute $F_{pos}(s)$, $P_{pos}(s)$, $U_{pos}(s)$, that are respectively obtained by combining, via operator $\times_p$, all the positive preferences of the projections of s over the constraints in bC_f, bC_p and bC_u, and $F_{neg}(s)$, $P_{neg}(s)$, $U_{neg}(s)$, that are respectively obtained by combining, via operator $\times_n$, all the negative preferences of the projections of s over the constraints in bC_f, bC_p and bC_u. Hence, we compute two satisfaction levels, a positive one, i.e., $F_{P_{pos}}(s) = F_{pos}(s) \times_p P_{pos}(s)$ and a negative one, i.e., $F_{P_{neg}}(s) = F_{neg}(s) \times_n P_{neg}(s)$ and two degrees of robustness, i.e., $U_{pos}(s)$ and $U_{neg}(s)$, that characterize respectively the positive and the negative robustness degree. Then we can compensate the two degrees of satisfactions and the two degrees of robustness. Hence, we can associate to every solution a degree of satisfaction $F_P(s) = F_{P_{pos}}(s) \times F_{P_{neg}}(s)$ and a robustness degree $U(s) = U_{pos}(s) \times U_{neg}(s)$. Since every solution is associated to a pair composed by a satisfaction degree and a robustness degree, in order to compare solutions, we can use the same semantics (i.e., P-Risky, P-Risky1, P-Safe, P-Safe1 and P-Diplomatic) described in Section 2.13.3.

Notice that the derived properties presented in Section 2.3 continue to hold. The first one states that if we fix, for every constraint in C_{fu} linking controllable and uncontrollable variables, the possibilities of its uncontrollable variables, and if we increase preferences of a given assignment to its controllable and uncontrollable variables for every value in the domain of the uncontrollable variables, then we obtain a higher value of robustness.

Proposition 23 *Consider two uncertain 2-bipolar CSPs* $BQ_1 = \langle bS, V_c, V_u, bC_1 = bC_{f_1} \cup bC_{fu_1}\rangle$, *and* $BQ_2 = \langle bS, V_c, V_u, bC_2 = bC_{f_2} \cup bC_{fu_2}\rangle$, *where* $bS = \langle N, P, +, \times, \bot, \Box, \top\rangle$ *is a bipolar preference structure,* P *and* N *are totally ordered sets and* bC_1 *and* bC_2 *differ only by the preference functions of constraints involving variables in* V_u, *i.e.,* $bC_{f_1} = bC_{f_2}$, $bC_{fu_1} = \bigcup_i \langle b\mu_1^i, con^i\rangle$ *and* $bC_{fu_2} = \bigcup_i \langle b\mu_2^i, con^i\rangle$. *In particular, for every such constraint,* $bc^i = \langle b\mu^i, con^i\rangle$, *such that* $con^i \cap V_c = X^i$ *and* $con^i \cap V_u = Z^i$, *with possibility distribution* π_{Z^i}, *let* $b\mu^i_{1_{pos}}(d,a) \leq_S b\mu^i_{2_{pos}}(d,a)$, *and* $b\mu^i_{1_{neg}}(d,a) \leq_S b\mu^i_{2_{neg}}(d,a)$, *for all* a *assignments to* Z^i *and for all* d *assignments to* X^i, *where* $\leq_S$ *is the order induced by the operator* $+$ *of* bS. *Then, given solution* s *of* BQ_1 *and* BQ_2, *such that* $s \downarrow_{X^i}= d$, $U_1(s) \leq_S U_2(s)$.

Proof: We recall that, for every constraint $bc^i = \langle b\mu^i, con^i\rangle \in C_{fu_1}$, $b\mu'^i_{1_{pos}}(d) = f_p^{-1}(inf_{a\in A_{z^i}}$ $(f_p(b\mu^i_{1_{pos}}(d,a)) + c_S(\pi_{Z^i}(a))))$ and $b\mu'^i_{2_{pos}}(d) = f_p^{-1}(inf_{a\in A_{z^i}}$ $(f_p(b\mu^i_{2_{pos}}(d,a)) + c_S(\pi_{Z^i}(a))))$ where A_{z^i} is the Cartesian product of the domains of the variables in Z^i. By hypothesis,

$b\mu^i_{1_{pos}}(d,a) \leq_S b\mu^i_{2_{pos}}(d,a)$, $\forall a, d$, then, since f is monotone with respect to ordering $\leq_S$, $f_p(b\mu^i_{1_{pos}}(d,a)) \leq_S f_p(b\mu^i_{2_{pos}}(d,a))$, $\forall a, d$. Hence, by monotonicity of $+$, $(f_p(b\mu^i_{1_{pos}}(d,a))+ c_S(\pi_{Z^i}(a)))) \leq_S (f_p(b\mu^i_{2_{pos}}(d,a))+ c_S(\pi_{Z^i}(a)))$, $\forall a, d$, then $inf_{a\in A_{z^i}}$ $(f_p(b\mu^i_{1pos}(d,a))+ c_S(\pi_{Z^i}(a))) \leq_S (f_p(b\mu^i_{1_{pos}}(d,a))+ c(\pi_{Z^i}(a))) \leq_S (f_p(b\mu^i_{2_{pos}}(d,a))+ c_S(\pi_{Z^i}(a)))$, $\forall a$, $\forall d$. Therefore we have that $b\mu'^i_{1_{pos}}(s\downarrow_{X^i}) = inf_{a\in A_{z^i}}$ $(f_p(b\mu_{1_{pos}}(s\downarrow_{X^i},a))+ c_S(\pi_{Z^i}(a))) \leq_S inf_{a\in A_{z^i}}$ $(f_p(b\mu_{2_{pos}}(s\downarrow_{X^i},a)+c_S(\pi_{Z^i}(a)))) = b\mu'^i_{2_{pos}}(s\downarrow_{X^i})$. Hence we have that $U_{1_{pos}}(s) = \prod_{p\ i} b\mu'^i_{1_{pos}}(s\downarrow_{X_i}) \leq_S U_{2_{pos}}(s) = \prod_{p\ i} b\mu'^i_{2_{pos}}(s\downarrow_{X_i})$, by monotonicity of $\times_p$.

Analogously we can prove that $U_{1_{neg}}(s) = \prod_{p\ i} b\mu'^i_{1_{neg}}(s\downarrow_{X_i}) \leq_S U_{2_{pos}}(s) = \prod_{p\ i} b\mu'^2_{2_{neg}}$ $(s\downarrow_{X_i})$, by monotonicity of $\times_n$.

The monotonicity of the compensation operator $\times$ of bS allows us to conclude. In fact, $U_1(s) = U_{1_{pos}}(s) \times U_{1_{neg}}(s) \leq_S U_{2_{pos}}(s) \times U_{2_{neg}}(s) = U_2(s)$. □

The other property presented in Section 2.3 states that if we fix preferences in every constraint in C_{fu} and if we decreases possibilities of the uncontrollable variables, then we obtain an higher value of robustness. This continues to hold also in our scenario for both U_{pos} and U_{neg}.

Proposition 24 *Consider two uncertain 2-bipolar CSPs $bQ_1 = \langle bS, V_c, V_u, bC_1 = bC_f \cup bC_{fu}\rangle$, and $BQ_2 = \langle bS, V_c, V'_u, bC_2 = bC_f \cup bC_{fu}\rangle$, where $bS = \langle N, P, +, \times, \bot, \Box, \top\rangle$ is a bipolar preference structure, P and N are totally ordered sets and V_u and V'_u are the same set of uncontrollable variables described, however, by different possibility distributions. In particular, for every constraint, $bc^i = \langle b\mu^i, con^i\rangle$, such that $con^i \cap V_c = X^i$ and $con^i \cap V_u = Z^i$, let $\pi^1_{Z^i}(a) \geq \pi^2_{Z^i}(a)$, for all a assignments to Z^i. Then, given solution s of bQ_1 and bQ_2, such that $s\downarrow_{X^i}= d$, $U_1(s) \leq_S U_2(s)$, where $\leq_S$ is the order induced by the operator $+$ of bS.*

Proof: For every constraint $bc^i = \langle b\mu^i, con^i\rangle \in bC_{fu}$, $b\mu'^i_{1_{pos}}(d) = inf_{a\in A_{z^i}}$ $(b\mu^i_{pos}(d,a)+ c_S(\pi^1_{Z^i}(a)))$ and $\mu'^i_{2_{pos}}(d) = inf_{a\in A_{z^i}}$ $(b\mu^i_{pos}(d,a)+ c_S(\pi^2_{Z^i}(a)))$. Moreover, $b\mu'^i_{1_{neg}}(d) = inf_{a\in A_{z^i}}$ $(b\mu^i_{neg}(d,a)+ c_S(\pi^1_{Z^i}(a)))$ and $b\mu'^i_{2_{neg}}(d) = inf_{a\in A_{z^i}}$ $(b\mu^i_{neg}(d,a)+ c_S(\pi^2_{Z^i}(a)))$. Since c_S is an order-reversing map with respect to $\leq_S$, if $\pi^1_{Z^i}(a) \geq \pi^2_{Z^i}(a)$, $\forall a$ then $c_S(\pi^1_{Z^i}(a)) \leq_S c_S(\pi^2_{Z^i}(a))$, $\forall a$. Thus, by monotonicity of $+$, $(b\mu^i_{pos}(d,a)+ c_S(\pi^1_{Z^i}(a))) \leq_S (b\mu^i_{pos}(d,a)+ c_S(\pi^2_{Z^i}(a)))$ and $(b\mu^i_{neg}(d,a)+ c_S(\pi^1_{Z^i}(a))) \leq_S (b\mu^i_{neg}(d,a)+ c_S(\pi^2_{Z^i}(a)))$, $\forall a, d$. From here we can conclude as in the proof of Proposition 23. □

3.9.3 An example

In this section we show via an example how to remove uncertainty from an uncertain bipolar CSP, how to compute the preference of a solution and how to order the solutions according to the semantics described in Section 2.13.3.

Let us consider the uncertain bipolar CSP in Figure 3.6, that we call BQ, defined as follows: $\langle bS, V_c = \{x, y\}, V_u = \{z_1, z_2\}, BC = BC_f \cup BC_{fu}\rangle$. The bipolar structure bS is $\langle N = [-1, 0]$, $P = [0, 1]$, $+ = max$, $\times$, $\perp = -1$, $\Box = 0$, $\top = 1\rangle$, where $\times$ is s. t. $\times_p = max$, $\times_n = min$ and $\times_{np} = sum$. The set of constraints bC_{fu} contains $c1 = \langle\mu_1, \{x, z_1\}\rangle$ and $c2 = \langle\mu_2, \{x, z_2\}\rangle$, while bC_f contains $c3 = \langle\mu_3, \{x, y\}\rangle$. Figure 3.6 shows the positive and the negative preferences within such constraints, as well as the possibility distributions π_1 and π_2 over domains of z_1 and z_2.

Figure 3.7 (a) shows the uncertain 2-bipolar CSP $bQ = \langle bS, V_c = \{x, y\}, V_u = \{z_1, z_2\}$, $bC = bC_f \cup bC_{fu}\rangle$ built in the 1st step of *B-SP*. Figure 3.7 (b) shows the 2-bipolar CSP without uncertainty $bQ' = \langle bS, V_c = \{x, y\}, bC' = bC_f \cup bC_p \cup bC_u\rangle$, built in the 2nd step of *B-SP*. bC_f is composed by $c3 = \langle\mu_3, \{x, y\}\rangle$, bC_p by $cp1 = \langle\mu_{p1}, \{x\}\rangle$ and $cp2 = \langle\mu_{p2}, \{x\}\rangle$ and bC_u by $c1' = \langle\mu'_1, \{x\}\rangle$ and $c2' = \langle\mu'_2, \{x\}\rangle$. $c1'$ and $c2'$ are obtained by using functions $f_n : N = [-1, 0] \rightarrow [0, 1]$ mapping every value $n \in [-1, 0]$ into the value $(n + 1) \in [0, 1]$, $f_n^{-1} : [0, 1] \rightarrow [-1, 0]$ mapping every value $t \in [0, 1]$ into the value $(t - 1) \in [-1, 0]$ and c_S mapping every $p \in [0, 1]$ in $1 - p$.

Figure 3.7 (c) shows all the solutions of the UBCSP BQ, i.e., all the complete assignments to the controllable variables (thus x and y). To compute the preference of a solution s, we need the positive satisfaction degree $F_{P_{pos}}(s)$ (respectively, the negative satisfaction degree $F_{P_{neg}}(s)$) obtained by combining via $\times_p = max$ (respectively, $\times_n = min$) all the positive preferences associated to the projections of s in constraints on $bC_f \cup bC_p$, i.e., $c3$, $cp1$ and $cp2$. We need also to compute the positive robustness $U_{pos}(s)$ (respectively, the negative robustness $U_{neg}(s)$) obtained by combining via $\times_p = max$ (respectively, $\times_n = min$) all the positive preferences associated to the projections of s in constraints in bC_u, i.e., in this case in $c1'$ and $c2'$. Then we obtain a unique satisfaction degree $F_P(s)$ for s by compensating (via $\times_{np} = sum$) $F_{P_{pos}}(s)$ and $F_{Pneg}(s)$ and a unique robustness value by compensating $U_{pos}(s)$ and $U_{neg}(s)$.

The optimal solution for P-Risky semantics is $s_2 = (y = b, x = a)$, which has preference $(F_P = 0.8, U = -0, 2)$ and for P-Safe, P-Risky1 and P-Safe1 semantics is $s_4 = (y = b, x = b)$, which has preference $(F_P = 0.7, U = 0.1)$ For the P-Diplomatic semantics s_2 and s_4 are equally optimal. Notice that the solutions chosen by the various semantics differ on the attitude toward risk they implement. In particular, the P-Risky semantics is risky, since it disregards almost completely the uncertain part of the problem. In fact, in this example it chooses the solution that gives an high positive preference in the controllable part, even if the uncontrollable part, which must be decided by Nature, will give with high possibility a negative preference. On the other hand, for the P-Safe semantics is better to select the solution with a higher robustness, i.e., that guarantees a higher number of scenarios with a higher preference. The P-Risky1 and P-Safe1 semantics try combine the preference given in the controllable part with that one given in the uncontrollable part. In this example, P-Safe, P-Risky1 and P-Safe1 choose a solution with a lower preference with respect to P-Risky, but that will have with high possibility a positive preference in the part involving uncontrollable variables.

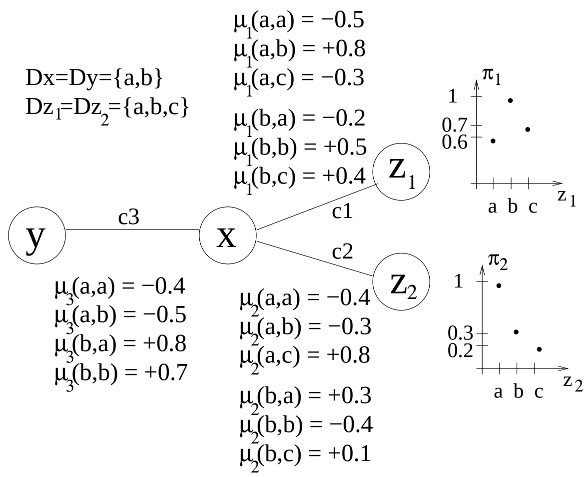

Figure 3.6: An uncertain bipolar CSP.

3.10 Related work

Bipolar reasoning has already considered in the AI community, and it has been handled in many different ways.

In [CS04], fair preference structures are introduced. In such a structure, which is an ordered set with an operation, $\oplus$, the key concept is that of difference of two elements. In particular, a structure is said to be fair if for each pair of ordered elements, $\alpha \leq \beta$, there exists a maximal element, γ, such that $\alpha \oplus \gamma = \beta$ called the difference of β and α. Although there is some similarity with the behaviour of our compensation operator, in [CS04], the setting is unipolar and the goal is mainly algorithmic (extension of arc consistency to Valued CSPs), rather than concerned with modelling new types of preferences.

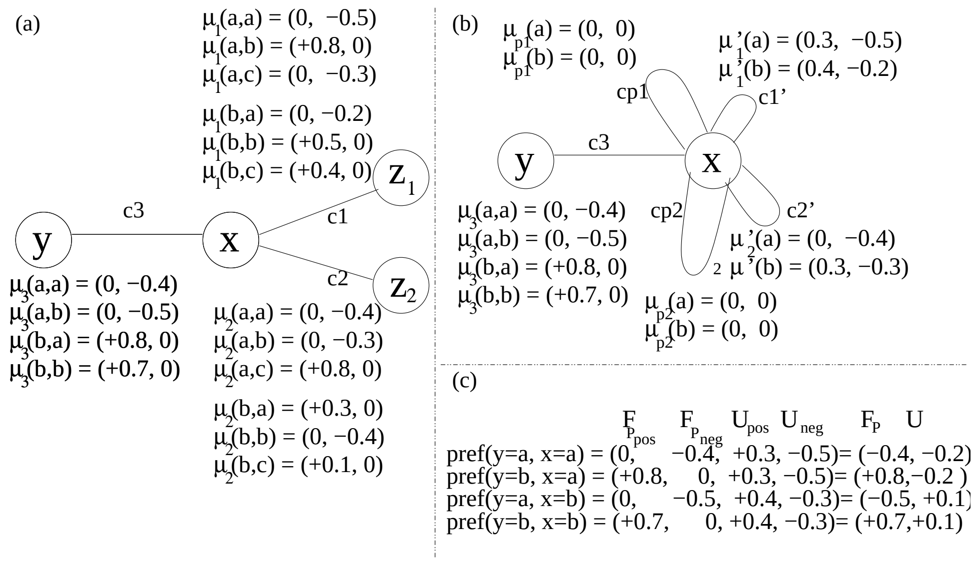

Figure 3.7: How algorithm *B-SP* works on the UBSCP of Figure 3.6.

In [GdBF03] the authors consider totally ordered unipolar and bipolar preference scales. In this chapter we present a method to deal with partially ordered bipolar scales. When the preference set is totally ordered, operators $\times_n$ and $\times_p$ described here correspond respectively to the $t - norm$ and $t - conorm$ used in [GdBF03]. Moreover, in [GdBF03] an operator, the $uninorm$, similar to the compensation operator but with the restriction of always being associative is considered. Due to the associativity requirement, our compensation operator is more general and may not be a uninorm when restricted to totally ordered scales.

In [BDKP02, BDKP06] a bipolar preference model based on a fuzzy possibilistic approach is described. The main differences with the framework presented in this chapter are the fact that only fuzzy preferences are considered and that negative preferences are interpreted as violations of constraints. In particular, the approach followed to combine negative and positive preferences in [BDKP02, BDKP06] is that of giving precedence to the negative preference optimization and resorting to positive preferences only to distinguish among the optimals found in the first step. Positive and negative preferences are, thus, kept separate and no compensation is allowed. In [BDKP06] the negative preferences are interpreted as strong constraints, i.e., as ordinary constraints which cannot be violated, whereas the positive preferences as criteria which can take any rate without leading to the rejection of potential solutions. Hence their feasible solutions are the complete assignments satisfying all the negative constraints and the optimal solutions are the feasible solutions satisfying the most of the positive criteria. Notice that a feasible solution can be optimal even if it satisfies only one of the positive criteria. Hence in [BDKP06] stating that an object is preferred or stating that its opposite is refused is very different. In fact the first statement is not important for finding feasible solutions, because it is considered as a criteria that can or not be satisfied, whereas the second one is very relevant, since a solution for being feasible must satisfy it. In our framework both the negative preferences and the positive preferences are considered as criteria, i.e., we assume that they can be compensated.

Another difference between our work and the one in [BDKP06] is that in our structure we assume that, given a certain situation, each agent gives or a positive or a negative preference over it, whereas in [BDKP06] they assume that an agent gives for every situation both a positive preference, expressing how much he would like that this situation happens, and a negative preference, defining how much we refuse that this situation does not happen. Hence for comparing directly our approach with the one in [BDKP06], we have to introduce in our framework something new. More precisely we set for every situation the preference that is lacking to the indifference level, assuming that if the agent doesn't give an explicit preference over something it means that he is indifferent with it. For example if an agent says that he would like at level 100 a red car, we translate this preference into a positive preference equal to 100 expressing he likes having a red car at level 100 and in a negative preference equal to the indifference element $\square$, showing that he is indifferent on the refusing not to have a red car. The agents have two ways for saying the same thing: a positive way and a negative way. In fact the agent can say that he prefers to have a red car at level 100 and that he refuses to not have a red car at level $\square$ (positive way) or he can say that he prefers to have a red car at level $\square$ and that he refuses not to have a red car at level -100 (negative way). Translating a situation defined by a positive preference in its opposite in negative side, both in our structure and in [BDKP06] produces a lowering of the values of the solutions, but as said before, in [BDKP06] this translation gives much more relevance to the considered situation, since it translate a desire into a strong constraint, and so it changes feasible solutions, whereas in our structure this doesn't increase the importance of this situation in the problem, even if it can change the ranking of the feasible solutions and the set of the optimal solutions.

3.11 Future work

We plan to develop a solver for bipolar CSPs which should be flexible enough to accommodate for both associative and non-associative compensation operators, by following the algorithm in Section 3.8. We also intend to implement the outlined algorithms for BB, NAC, PAC, and p-BAC and to test them over classes of bipolar problems.

We plan to consider new semantics for computing the preference of a solution of a bipolar problem and to order the various solutions according to several design principles. When we compute the preference of a solution, we aggregate positive and negative preferences via the two operators $\times_p$ and $\times_n$. However, this may lead to poor discrimination among solutions. In fact, if we have a finite preference scale with few elements, then aggregating means obtaining one of the preferences in the scale to associate to a solution. Thus, if the number of solutions is much higher than the number of the element of the scale, many solutions will end up in the same evaluation and will thus result indistinguishable. To solve this problem, we plan to adapt the formalism for bipolar problems to allow for *no aggregation,* and to maintain, for each solution, the tuple of all preferences given by the single constraints to the solution. Thus a greater discriminating power is achieved. Assuming no aggregation, a solution is associated to a tuple of positive preferences and a tuple of negative preferences. Different solutions can then be compared by ordering the elements of their tuples (according to $+_p$ and $+_n$) for each solution, and then by comparing the ordered tuples by a lexicographic order.

Moreover, we want to generalize our bipolar structure in order to deal with problems, where there are some negative statements, which are so negative that we would not like them to be compensated even by the best positive statements. For example, if we are allergic to the ingredients of a medicine, then, even if the medicine would solve our health problem, we don't want to use it. Moreover, there are also statements that need to be expressed as hard constraints, which have to be satisfied for a scenario to be feasible. For example, if a classroom cannot fit more than 100 students, then, no matter the other features of the room, we cannot choose it for a class of 150 students. It is important to provide a framework where such situations can be expressed. To do that, we plan to consider an extension of the bipolar preference structure defined in this chapter, where it is present an additional structure, that represents negative statements that cannot be compensated by any positive preference.

We intend to reason with bipolar preferences in terms of multi-criteria methods. In particular, we plan to define a unique more general bipolar preference structure where it is possible to choose whether to perform compensation of positive and negative preferences, and to use classical multi-criteria methods if we don't allow for compensation.

We plan to generalize the BB algorithm for bipolar problems to bipolar problems with uncertainty, by adapting to bipolar preferences the BB algorithm described in Section 2.12 for solving fuzzy CSPs with uncertainty. We also plan to extend the procedure for removing uncontrollability to bipolar preference problems with uncertainty where the set of positive and negative preferences are partially ordered. To do so, we intend to use a procedure similar to the one described in Section 2.13 for removing uncertainty from soft CSPs with uncertainty.

Another topic that we want to investigate regards the relationship between bipolar preferences and importance between pairs of variables [BD02]. In particular, we want to study if we can assimilate importance with preferences or if it is better to keep it separate from preferences like done with CP-nets [BD02]. We want also to analyse the relations with trade-off methods [BO], which are methods proposed in literature for solving over-constrainedness in interactive constraint-based tools that reason about user preferences.

Moreover, we plan to consider the possible connections between our work and non-monotonic concurrent constraints [BdBC97], where removing a constraint is related to adding a positive preference and where constraints must be considered in a fixed ordered and so combined with a non-associative operator.

Finally, we intend to study the concept of bipolarity in the area of voting theory as done in [BS06], which presents a voting method according to which each voter submits a set of candidates he approves and a set of candidates he disapproves.

Chapter 4

Preference aggregation: fairness and strategy proofness

In this chapter we want to consider even more general scenarios than those we have tackled up to here. In fact, here we are going to study preferences expressed by multiple agents. This means that we must consider ways of reasoning and aggregating preferences in order to choose outcomes that satisfy all the agents.

In this chapter, we consider a multi-agent framework preference reasoning where each agent expresses his own preferences using a partial order and we study how to aggregate the agent's preferences once they have been collected. In particular, we adapt the most popular aggregating criteria of social choice theory [Kel87] in our context and we push the bridge between social choice theory and aggregation of preferences obtained using AI representations, by considering the fairness [MS77, Arr51] and non-manipulability [Gib73, Sat75] of the aggregating criteria we propose. The main difference from the context in which the famous Arrow's theorem [Arr51], Muller-Satterthwaite's theorem [MS77] and Gibbard-Satterthwaite's theorem [Gib73, Sat75] were originally written and our scenario is that we don't have total orders, that is, we allow incomparability. We thus extend Muller-Satterthwaite's impossibility theorem [MS77] and Gibbard-Satterthwaite's theorem [Gib73, Sat75] to the situation in which the ordering given by each agent is a partial order.

4.1 Motivations and chapter structure

Many problems require us to combine the preferences of different agents. For example, when planning a wedding, we must combine the preferences of the bride, the groom and possibly some or all of the in-laws. Incomparability is an useful mechanism to resolve conflict when aggregating such preferences. If half of the agents prefers a to b and the other half prefers b to a, then it may be best to say that a and b are incomparable. In addition, an agent's preferences are not necessarily total. For example, while it is easy and reasonable to compare two apartments, it may be difficult to compare an apartment and a house. We may wish simply to declare them incomparable. Moreover, an agent may have several possibly conflicting preference criteria he wants to follow, and their combination can naturally lead to a partial order. For example, one may want a cheap but large apartment, so an 80 square metre apartment which costs 100.000 euros is incomparable to a 50 square metre apartment which costs 60.000 euros.

In [Ven05] they assume that both the preferences of an agent and the result of preference aggregation can be a partial order. In this context, it is natural to ask if we can combine partially ordered preferences *fairly*. For total orders, Arrow's theorem shows this is impossible [Arr51]. In [Ven05] they show that this result can be generalized to partial orders under certain conditions. Moreover, they identify two cases where fairness of social welfare functions over partial orders is possible, one of which is a generalization of Sen's theorem [Sen70]. The two cases correspond to two extremes of the amount of partiality of the partial orders. One result considers partially ordered profiles which are very ordered (that is, very close to be total orders), while the other concerns profiles with no chain of ordered pairs, and the ordering relation contains a very small number of pairs.

These results assume that one is interested in obtaining a partial order over the different scenarios as the outcome of preference aggregation. One may wonder if the situation is easier when we are only interested in the most preferred outcomes in the aggregated preferences. In this chapter we show that even in this case (that is, when considering social choice functions over partial orders) it is impossible to be fair, i.e. Arrow's impossibility theorem holds also in this case. This is a generalization of Muller-Satterthwaite's theorem [MS77] to partial orders.

We then consider the notion of strategy proofness, which denotes the non-manipulability of a social choice function. For totally ordered preferences, the Gibbard-Satterthwaite's result [Gib73] tells us that it is not possible for a social choice function to be at same time non-manipulable and have no dictators. We prove that this result holds also in the partially ordered scenario.

The work described in this chapter has appeared in the proceedings of the following conferences:

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Aggregating partially ordered preferences: possibility and impossibility results. *In Proceedings of 10th Conference on*

Theoretical Aspects of Rationality and Knowledge (TARK X), ACM Digital Library, National University of Singapore, pp. 193-206, Singapore, June 2005.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Strategic voting when aggregating partially ordered preferences. *In Proceedings of the 5th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2006)*, ACM Press, pp. 685-687, Hakodate, Japan, May 2006.

The chapter is organized as follows.

- In Section 4.2 we give some background notions on orderings (Section 4.2.1) and on the formalisms for representing agents' preferences compactly, which typically induce a partial order (Section 4.2.2). Next we present some mechanisms for aggregating such preferences which have been proposed in the literature, i.e., social welfare functions and social choice functions (Section 4.2.3 and Section 4.2.4). Then, we present impossibility results concerning fairness in social welfare functions over partially ordered preferences (Section 4.2.5), and possibility results in which the majority rule can be fair (Section 4.2.5).

- In Section 4.3 we present an impossibility result concerning fairness in social choice functions over partially ordered preferences. In particular, we define social choice functions over partial orders and their properties, and we prove that they cannot be fair.

- In Section 4.4 we introduce the notion of strategy proofness for partially ordered social choice functions (Section 4.4.1) and we prove the generalization of the Gibbard-Satterthwaite's result (Section 4.4.2).

- In Section 4.5 we present the work related to what is presented in this chapter. In Section 4.6 we summarize the main results described in this chapter and in Section 4.7 we present the future directions of research.

4.2 Background

We start reviewing some basic notions on orders and preferences, which will be useful in what follows. In Section 4.2.3 we describe the main notions of voting theory, those which we will consider in this chapter, and we describe briefly the fundamental result contained in Arrow's impossibility theorem, in Sen's possibility theorem, in Muller-Satterthwaite's impossibility theorem and in Gibbard-Satterthwaite's impossibility theorem. Moreover, we present a generalization of Arrow's impossibility theorem and a generalization of Sen's possibility theorem to partially ordered preferences.

4.2.1 Orderings

A preference ordering can be described by a binary relation on outcomes where x is preferred to y if and only if (x, y) is in the relation. Such relations may satisfy a number of properties. A binary relation R on a set S (that is, $R \subseteq S \times S$) is:

- *reflexive* if and only if $\forall x \in S$, $(x, x) \in R$;
- *transitive* if and only if $\forall x, y, z \in S$, $(x, y) \in R$ and $(y, z) \in R$ implies $(x, z) \in R$;
- *antisymmetric* if and only if $\forall x, y \in S$, $(x, y) \in R$ and $(y, x) \in R$ implies $x = y$;
- *complete* if and only if $\forall x, y \in S$, either $(x, y) \in R$ or $(y, x) \in R$.

Definition 29 (total order) A *total order* (TO) is a binary relation which is reflexive, transitive, antisymmetric, and complete.

A total order has an unique *optimal* element, that is an element $o \in S$ such that $\forall x \in S, (o, x) \notin S$. We say that this element is *undominated*.

Definition 30 (partial order) A *partial order* (PO) is a binary relation which is reflexive, transitive and antisymmetric but may be not complete.

There may be pairs of elements (x, y) of S which are not in the partial order relation, that is, such that neither $(x, y) \in R$ nor $(y, x) \in R$. Such elements are *incomparable* (written $x \bowtie y$). A partial order can have several optimal and mutually incomparable elements. Again, we say that these elements are *undominated*. Undominated elements will also be called *top* elements. The set of all top elements of a partial order o will be called $top(o)$. Elements which are below or incomparable to every other element will be called *bottom* elements.

Definition 31 (strict order) Given any relation R which is either a total or a partial order, if $(x, y) \in R$, it can be that $x = y$ or that $x \neq y$. If R is such that $(x, y) \in R$ implies $x \neq y$, then R is said to be *strict*. This means that reflexivity does not hold.

Both total and partial orders can be extended to deal with ties, that is, sets of elements which are equally positioned in the ordering. Two elements which belong to a tie will be said to be *indifferent*. To summarize, in a total order with ties, two elements can be either ordered or indifferent. On the other hand, in a partial order with ties, two elements can be either ordered, indifferent, or incomparable. Notice that, while incomparability is not transitive in general, indifference is transitive, reflexive, and symmetric.

In the following we will sometimes need to consider partial orders with some restrictions. In particular, we will call a **rPO** a partial order where the top elements are all indifferent, or the bottom elements are all indifferent. In both POs and rPOs, ties are allowed everywhere, except where explicitly stated otherwise.

4.2.2 Preferences

An agent's preferences are not necessarily total. For example, while it is easy and reasonable to compare two apartments, it may be difficult to compare an apartment and a house. We may wish simply to declare them incomparable. Moreover, an agent may have several possibly conflicting preference criteria he wants to follow, and their combination can naturally lead to a partial order. For example, one may want a cheap but big apartment, so an 80 square meters apartment which costs 100.000 euros is incomparable to a 50 square meters apartment which costs 60.000 euros. We assume therefore that the preferences of an agent can be a partial order.

A number of formalisms have been proposed for compactly representing and efficiently reasoning about preferences of a single agent. Common to all is that they induce some sort of partial or total ordering, possibly with ties, on the outcomes. For example, soft constraints, described in Section 2.2.1, can model quantitative preferences [BMR97, Sch92]. We recall that each constraint associates a preference value to each assignment of its variables. To model preference ordering and aggregation, the set of possible preference values is the carrier of a semiring, whose two operations state how to order values in the set and how to combine values to obtain new preferences. A complete assignment of values to variables is associated to a preference value by combining the preferences of each partial assignment in each constraint via the combination operation of the semiring. In general, the order induced on the preferences via this approach is a partial order with ties. Assignments with the same preference are naturally interpreted as ties.

Soft constraints can also represent hard statements, as in “I need to be back before 8pm”: it is enough to take a set with just two preferences values (that can be interpreted as true and false), order them via logical or (thus true is better than false and we have a total order), and combine them via logical and (so an assignment has preference true if all constraints have preference true, and it is said to be consistent; an assignment has preference false, and it is said to be inconsistent, if some of the constraints have preference false). In this case, the ordering induced over the complete assignments is a total order with ties: all consistent assignments have preference true (thus they are all indifferent) and are better than all inconsistent assignments (which again are indifferent among them).

Another formalism for representing preferences is CP nets [BBHP99, DB02]. They are a compact mechanism to model conditional qualitative preferences (as in "If I take the fish course, I prefer white wine over red") which satisfy the *ceteris paribus* or "all other things being equal" property. A dependency graph in a CP net states the relation among the features of the problem. Each feature X has a domain of possible values and some parent features $Pa(X)$ on which it depends on: given any complete assignment to $Pa(X)$, CP nets state a total order for the values in the domain of X (in a structure called a CP conditional preference table). Such a total order represents the preference order on the values of X given the values of its parents, all else being equal. A CP net induces an ordering over the complete assignments of all its features: an assignment O is better than another one O' if there is a chain of improving flips from O to O', where an improving flip is a change of the value of one feature that improves the preference according to some preference table in the CP net. Such an ordering is in general partial and does not have ties.

Partial CP nets [RVW04] do not require that all features are ranked. This allows one to represent situations as in "I am indifferent to the color of the car". This means that the ordering induced by a partial CP net over its outcomes is in general a partial ordering with ties. In fact, there could be flips which are neither improving nor worsening, since they change the value of a non-ranked feature.

A number of mechanisms have been proposed for aggregating such preferences [Doy91, Sen70]. One possibility is to run an election in which each agent votes on how they rank every pair of outcomes. In [RVW04], each agent represents their preferences with a partial CP net and then votes on how outcomes should be ordered. However, the agents can represent their individual preferences with soft constraints or any other formalism for representing preferences. We need, however, to specify how their votes are collected together into a result.

As in voting theory, the orderings of the agents is called a profile. A social welfare function is then a function mapping profiles onto a result (a partial ordering). In [RVW04], a number of different social welfare functions for when agents vote with partial orders are described.

Pareto: One outcome α is better than another β (written $\alpha \succ_p \beta$) if and only if every agent says α is better than or equal to β (written $\alpha \succ \beta$ or $\alpha \approx \beta$) and at least one of them says α is better than β. Two outcomes are incomparable if and only if they are not ordered either way. An outcome is Pareto optimal if and only if no other outcome is better.

Majority: One outcome α is majority better than another β (written $\alpha \succ_{maj} \beta$) if and only if the number of agents which say that α is better than β is greater than the number of agents which say the opposite plus the number of those that say that α and β are incomparable. Two outcomes are majority incomparable if and only if they are not ordered either way. An outcome is majority optimal if and only if no other outcome is majority better.

Max: One outcome α is max better than another β (written $\alpha \succ_{max} \beta$) if and only if more agents vote in favor than against or for incomparability. Two outcomes are max incomparable if and only if they are not ordered either way. An outcome is max optimal if and only if no other outcome is max better.

Lex: This rule assumes the agents are ordered in importance. One outcome α is lexicographically better than another β (written $\alpha \succ_{lex} \beta$) if and only if there exists some agent A such that all agents higher in the order say $\alpha \approx \beta$ and A says $\alpha \succ \beta$. Two outcomes are lexicographically incomparable iff there exists some distinguished agent such that all agents higher in the ordering are indifferent between the two outcomes and the outcomes are incomparable to the distinguished agent. Finally, an outcome is lexicographically optimal if and only if no other outcome is lexicographically better.

Rank: Each agent gives a numerical rank to each outcome. For example, in a partial CP net, the rank of an outcome is zero if the outcome is optimal, otherwise it is the length of the shortest chain of worsening flips between one of the optimal outcomes and it. We say that one outcome α is rank better than another β (written $\alpha \succ_r \beta$) if and only if the sum of the ranks assigned to α is smaller than that assigned to β. Two outcomes are rank indifferent iff the sum of the ranks assigned to them are equal. Either two outcomes are rank indifferent or one must be rank better than the other. Finally, an outcome is rank optimal if and only if no other outcome is rank better.

The Pareto and Lex rules define strict partial orderings if the agents have a strict partial order, while if the agents have a partial order with ties then these rules give a partial order without ties. The Rank rule, instead, gives a total order with ties. Maj and Max are irreflexive and antisymmetric but may be not transitive. However, they all have at least one optimal element. Notice that in all the five rules, except Rank, it is not possible for two outcomes to be indifferent, since we assume that each feature is ranked by at least one of the partial CP nets, while indifference in the qualitative relations (Pareto, Max, Majority, and Lex) means indifference for everybody. In all these voting rules, except Rank, the result of aggregating preferences is itself a partial order. For each of these social welfare function, we can define a corresponding social choice function by taking just the top elements in their result.

4.2.3 Social welfare theory

In classical social welfare theory [Kel87, Arr51, Str80], individuals state their preferences in terms of total orders. In this section we will define the main concepts and properties in this context.

Definition 32 (profile) Given a set of n individuals and a set of outcomes O, a *profile* is a sequence of n orderings over O, one for each individual.

Definition 33 (social welfare function) A *social welfare function* is a function from profiles to orderings over O.

Thus social welfare functions provide a way to aggregate the preferences of the n individuals into an ordering of the outcomes.

Several properties of social welfare functions can be considered:

- *Freeness*: if the social welfare function can produce any ordering;
- *Unanimity*: if all agents agree that an outcome a is preferable to another outcome b, then the resulting order must agree as well;
- *Independence to irrelevant alternatives*: if the ordering between a and b in the result depends only on the relation between a and b given by the agents;
- *Monotonicity*: if, whenever an agent moves up the position of one outcome in his ordering, then (all else being equal) such an outcome cannot move down in the result;
- *Dictatoriality*: if there is at least an agent such that, no matter what the others vote, if he says a is better than b then the resulting ordering says the same. Such an agent is then called a *dictator*.

- *Fairness*: if the social welfare function is free, unanimous, independent to irrelevant alternatives, and non-dictatorial.

These properties are all very reasonable and desirable also for preference aggregation. Unfortunately, a fundamental result in voting theory is Arrow's impossibility theorem [Arr51, Kel78] which shows that no social welfare function on total orders with ties can be fair. In particular, the usual proofs of this result show that, given at least two voters and three outcomes, and a social welfare function which is free, monotone, and independent of irrelevant alternatives, then there must be at least one dictator.

It is possible to prove that monotonicity and independence to irrelevant alternatives imply unanimity. Social welfare functions can be free, unanimous and independent to irrelevant alternatives but not monotonic [Sen70]. Therefore a stronger version of Arrow's result can be obtained by proving that freeness, unanimity and independence of irrelevant assumptions implies that there must be at least one dictator [Gea01].

A very reasonable social welfare function which is often used in elections is pairwise majority voting. In this voting function, for each pair of outcomes, the order between them in the result is what the majority says on this pair. This function, however, can produce an ordering which is cyclic. A sufficient condition to avoid generating cycles via a pairwise majority voting function is *triplewise value-restriction* [Sen70], which means that, for every triple of outcomes x_1, x_2, x_3, there exists $x_i \in \{x_1, x_2, x_3\}$ and $r \in \{1, 2, 3\}$ such that no agent ranks x_i as his r-th preference among x_1, x_2, x_3. A typical example where this condition is not satisfied and there are cycles in the result is one where agent 1 ranks $x_1 > x_2 > x_3$, agent 2 ranks $x_2 > x_3 > x_1$, and agent 3 ranks $x_3 > x_1 > x_2$. In this example, x_1 is ranked 1st by agent 1, 2nd by agent 3, and 3rd by agent 2, x_2 is ranked 1st by agent 2, 2nd by agent 1, and 3rd by agent 3, and x_3 is ranked 1st by agent 3, 2nd by agent 2, and 3rd by agent 1. The resulting order obtained by majority voting has the cycle $x_1 > x_2 > x_3 > x_1$.

4.2.4 Social choice theory

In many situations we may only be interested in the best outcome for all the agents. Such a situation can be described by means of a social choice function.

Definition 34 (social choice function) A *social choice* function is a mapping from a profile to one outcome, the optimal outcome.

A social choice function f is

- *unanimous* if and only if for any profile p with $a = top(p_i)$ for every agent i, (i.e., a is at top of every individual i's ranking) then $f(p) = a$;

- *monotonic* iff, given two profiles p and p', if $f(p) = a$ and, for any other alternative b, $a >_{p_i} b$ implies $a >_{p'_i} b$, for all agents i, then $f(p') = a$;

- a *dictatorship* if for some agent i, $f(p) = a$ if and only if $a = top(p_i)$. Agent i is then called a *dictator*.

The Muller-Satterthwaite's theorem is a generalization of Arrow's theorem on total orders which shows that a dictator is inevitable if we have two agents, three or more outcomes and the social choice function collecting votes is unanimous and monotonic [MS77].

An interesting result in social choice theory is Gibbard-Statterthwaite's theorem [Gib73]. That is, there are inevitable dictators, if we have at least two agents and three outcomes, and the social choices function is strategy proof and onto.

A social choice function f is

- *onto* if and only if it is surjective, i.e., for every profile p, $f(p)$ can be any outcome.

- *strategy proof* if and only if for every profile $p = (p_1, \ldots, p_i, \ldots, p_n)$ and for every ranking p_i, $f(p_1, \ldots, p'_i, \ldots, p_n) \geq_i f(p_1, \ldots, p_i, \ldots, p_n)$, where $a \geq_i b$ if and only if $a = b$ or $a >_i b$. Intuitively, a social choice function is strategy proof if it is best for each agent to order outcomes as they prefer and not to try to vote tactically.

4.2.5 Fairness for social welfare functions over partial orders

Preferences typically define a partial ordering over outcomes. For situations involving multiple agents, it is necessary to combine the preferences of several individuals. In this section, we consider each agent as voting on whether they prefer one outcome to another. In [Ven05] they prove that, under certain conditions on the kind of partial orders that are allowed to express the preferences of the agents and of the result, if there are at least two agents and three outcomes to order, no preference aggregation system can be fair. That is, no preference aggregation system can be free (give any possible partial order in the result), monotonic (improving a vote for an outcome only ever helps), independent to irrelevant assumptions (the result between two outcomes only depends on how the agents vote on these two outcomes), and non-dictatorial (there is not one agent who is never contradicted). This result generalizes Arrow's impossibility theorem for combining total orders [Arr51].

The formalisms for representing preferences [BBHP99, DB02] provide an ordering on outcomes. In general, this ordering is partial as outcomes may be incomparable. For example, when comparing wines, we might prefer a white wine grape like chardonnay to the sauvignon blanc grape, but we might not want to order chardonnay compared to a red wine grape like merlot.

The result of aggregating the preferences of multiple agents is itself naturally a partial order. If two outcomes are incomparable for each agent, it is reasonable for them to remain incomparable in the final order. Incomparability can also help us deal with disagreement between the agents. If some agents prefer A to B and others prefer B to A, then it may be best to say that A and B are incomparable (as in the Pareto semantics). In [Ven05] they consider this kind of scenario. They assume each agent has a preference ordering on outcomes represented via soft constraints, CP-nets or any other mechanism. A preference aggregation procedure then combines these partial orders to produce an overall preference ordering, and this again can be a partial order. The question they address here is: can we combine such preferences fairly? They show that, if each agent can order the outcomes via a partial order with unique top and bottom, and if the result is a partial order with a unique top or a unique bottom, then any preference aggregation procedure is ultimately unfair. They assume that each agent's preference specify a partial order over the possible outcomes.

Definition 35 (profile) Given a set of n individuals and a set of outcomes O, a *profile* is a sequence of n partial orderings over O, one for each individual.

They aggregate the preferences of a number of agents using a social welfare function.

Definition 36 (social welfare function over POs) A *social welfare function over partial orders* is a function from profiles to partial orderings over O.

As we said in Section 4.2.3, one property of preference aggregation which is highly desirable is fairness. In [Ven05] they consider fairness also in the context of partially ordered preferences. Arrow's theorem does not directly apply to aggregating preferences in such a scenario, since agents are assumed to express preferences in terms of total orders with ties. They show that all the properties defined in Section 4.2.3 for social welfare functions over total orders can be used also for partially ordered ones. Except in the case of dictator, they are straightforward generalizations of the corresponding properties for social welfare functions for total orders [ASS02].

A social welfare function over partial orders is free if it can produce any partial ordering over outcomes.

Definition 37 (freeness) A social welfare function f over partial orders satisfies *freeness* if and only if f is surjective, that is, for every profile p, $f(p)$ can be any partial ordering.

A social welfare function over partial orders is unanimous if, when all agents agree that an outcome a is preferable to an outcome b, then the resulting order agrees as well.

Definition 38 (unanimity) A social welfare function f over partial orders satisfies *unanimity* iff, for every profile p, if $a >_{p_i} b$ for all agents i, then $a >_{f(p)} b$.

A social welfare function over partial orders is independent to irrelevant alternatives if the ordering between two outcomes a and b in the result depends only on the relation between a and b given by the agents.

Definition 39 (independence to irrelevant alternatives) A social welfare function f over partial orders satisfies *independence to irrelevant alternatives* iff, for all profiles p and p', if $p_i(a, b) = p'_i(a, b)$ for every pair of outcomes a and b, for all agents i, then $f(p)(a, b) = f(p')(a, b)$, where, given an ordering o, $o(a, b)$ is the restriction of o on a and b.

A social welfare function over partial orders is monotonic if, whenever an agent moves up the position of one outcome in his ordering, then (all else being equal) such an outcome cannot move down in the result.

Definition 40 (monotonicity) A social welfare function f over partial orders satisfies *monotonicity* if and only if for any two profiles p and p', if b improves with respect to a in passing from p to p' in one agent i and $p_j = p'_j$ for all $j \neq i$, then, in passing from $f(p)$ to $f(p')$, b improves with respect to a, where b improves with respect to a if and only if the relationship between a and b does not move to the left along the following sequence: $>, \geq, (\bowtie \text{ or } =), \leq, <$.

Another desirable property of social welfare functions is the absence of a dictator. With partial orders, there are several possible notions of dictator [Ven05].

Definition 41 (dictatorship) Given a social welfare function f over partial orders,

- a *strong dictator* is an agent i such that, in every profile p, $f(p) = p_i$, that is, his ordering is the result.
- a *dictator* is an agent i such that, in every profile p, if $a \geq_{p_i} b$ then $a \geq_{f(p)} b$.
- a *weak dictator* is an agent i such that, in every profile p, if $a \geq_{p_i} b$, then $a \not< _{f(p)} b$.

Nothing is said about the result if a is incomparable or indifferent to b for the dictator or weak dictator. A strong dictator is a dictator, and a dictator is a weak dictator. Moreover, whilst there can only be one strong dictator or dictator, there can be any number of weak dictators.

With partial orders, there are several possible notions of fairness.

Definition 42 (fairness) A social welfare function over partial orders is

- *strongly fair* if it is unanimous, independent to irrelevant alternatives, and does not have a strong dictator;
- *fair* if it is unanimous, independent to irrelevant alternatives, and does not have a dictator;
- *weakly fair* if it is unanimous, independent to irrelevant alternatives, and does not have a weak dictator

Arrow's impossibility theorem [Arr51, Kel78] shows that, if a social welfare function on total orders with ties is unanimous and independent to irrelevant alternatives and there are at least two voters and three outcomes, then there must be at least one dictator. In [Ven05] they prove that freeness, monotonicity and independence to irrelevant alternatives imply unanimity. On the other hand, there are social welfare functions which are free, unanimous and independent to irrelevant alternatives but not monotonic [Sen70]. Therefore a weaker version of Arrow's result on total orders with ties states that freeness, monotonicity and independence of irrelevant assumptions imply that there must be at least one dictator [Gea01].

Proposition 25 [Ven05] *A social welfare function over partial orders can be fair.*

For example, the Pareto rule in which the outcome is ordered if every agent agrees, but is incomparable otherwise, is fair. A social welfare function that is fair is also strongly fair. Hence a social welfare function on partial orders can be strongly fair. Therefore, strong fairness is a very weak property to demand. Even voting rules which appears very "unfair" may not have a strong dictator. For example, Lex rule, in which the agents are ordered, and two outcomes are ordered according to the first agent who is not indifferent, is not fair, but it is strongly fair.

Fairness: impossibility results

In [Ven05] they show that, under certain conditions, it is impossible for a social welfare function over partial orders to be weakly fair. The conditions involve the shape of the partial orders. In fact, they assume the partial orders of the agents to be general (PO), but the resulting partial order must have all top or all bottom elements indifferent (rPO).

Theorem 6 [Ven05] *Given a social welfare function f over partial orders, assume the result is a rPO, there are at least 2 agents and 3 outcomes, and f is unanimous and independent to irrelevant alternatives. Then there is at least one weak dictator.*

As with total orders, they also prove a weaker result in which they replace unanimity by monotonicity and freeness.

Corollary 1 [Ven05] *Given a social welfare function f over partial orders, assume the result is a rPO, there are at least 2 agents and 3 outcomes, and f is free, monotonic, and independent to irrelevant alternatives. Then there is at least one weak dictator.*

They consider, for example, the Pareto rule. With this rule, every agent is a weak dictator since no agent can be contradicted. They note that we could consider a social welfare function which modifies Pareto by applying the rule only to a strict subset of the agents, and ignores the rest. The agents in the subset will then all be weak dictators. A number of results follow from these theorems. They denote the class of all social welfare functions from profiles made with orders of type A to orders of type B by $A^n \mapsto B$, and they prove the impossibility of being weakly fair for functions in $PO^n \mapsto rPO$. The first result concerns the restriction of the codomain of the social welfare functions.

Theorem 7 [Ven05] *If all functions in $A^n \mapsto B$ are not weakly fair, then also functions in $A^n \mapsto B'$, where B' is a subset of B, are not weakly fair.*

This theorem implies, for example, that the functions in $PO^n \mapsto O$, where O is anything more ordered than a rPO, cannot be weakly fair. For example, it can be deduced that functions in $PO^n \mapsto TO$ cannot be weakly fair.

In [Ven05] they consider the restriction of the domain of the functions, that is, let us pass from $A^n \mapsto B$ to $A'^n \mapsto B$ where A' is a subtype of A. They are interested in understanding whether the impossibility result holds also when performing such a restriction. In general, this is not true. However, passing from $PO^n \mapsto rPO$ to $TO^n \mapsto rPO$, they note that the proof of Theorem 6 still works, since it does not assumes incomparability in the preferences of the agents.

Summarizing, they prove the same impossibility results for all functions with the following types:

- $PO^n \mapsto rPO$ (by Theorem 6);
- $PO^n \mapsto TO$ (by Theorems 6 and 7);
- $TO^n \mapsto rPO$ (by the same proof as Theorem 6);
- $TO^n \mapsto TO$, that is, Arrows' theorem (by the result for $TO^n \mapsto rPO$ and Theorem 7);

They arrange these four impossibility results in a lattice where the ordering is given by either domain or codomain subset, as it can be seen in Figure 4.1.

$$
\begin{array}{ccccc}
 & & PO^n \mapsto rPO & & \\
 & \swarrow & & \searrow & \\
PO^n \mapsto TO & & & & TO^n \mapsto rPO \\
 & \searrow & & \swarrow & \\
 & & TO^n \mapsto TO & &
\end{array}
$$

Figure 4.1: **Lattice of impossibility results.** rPO stands for partial order where top elements or bottom elements are all indifferent, PO stands for partial order, TO stands for total order. Arrow's theorem applies to $TO^n \mapsto TO$. $\swarrow$ and $\searrow$ stand for the lattice ordering, which is either domain or codomain subset.

Fairness: possibility results

In [Ven05] they consider ways of assuring that a social welfare function is weakly fair. In fact, they identify situations where the well known *majority rule* is transitive, which makes it weakly fair since it has all the other properties. The majority rule they consider says a is better than b if and only if the number of agents which say that $a > b$ is greater than the number of agents which say that $b > a$ plus the number of those that say that a and b are incomparable. Notice that ties are ignored by this rule. They focus on the condition that Sen has proved sufficient for fairness in the case of total orders, namely triplewise value-restriction [Sen70].

Definition 43 (triplewise value-restriction) A total order profile satisfies triplewise value-restriction if and only if for every triple of outcomes x_1, x_2, x_3, there exists $x_i \in \{x_1, x_2, x_3\}$ and $r \in \{1, 2, 3\}$ such that no agent ranks x_i as his r-th preference among x_1, x_2, x_3.

To apply Sen's theorem to this context, they consider linearizations of partially ordered profiles. They note that, as they have partial orders, to assure transitivity in the resulting order, they must avoid both cycles (as in the total order case) and incomparability in the wrong places. More precisely, if the result has $a > b > c$, it is not possible to have $c > a$, which would create a cycle, and not even $a \bowtie c$, since in both cases transitivity would not hold. In this case they define a generalized triplewise value-restriction and they generalize Sen's theorem to partial orders without ties.

Definition 44 (generalized triplewise value-restriction) A partial order profile p satisfies the *generalized triplewise value-restriction* if all the profiles obtained from p by linearizing any PO to a TO have the triplewise value-restriction property.

Theorem 8 [Ven05] *If all profiles satisfy the generalized triplewise value-restriction and are without ties, then the majority rule is weakly fair.*

This result is useful when the profiles are highly ordered, and, within each profile, the agents have similar orders.

On the other extreme, in [Ven05] they give another possibility result which can be applied to profiles which order few outcomes. This result assures transitivity of the resulting ordering by a more rough approach: it just avoids the presence of chains in the result. That is, for any triple x_1, x_2, x_3 of outcomes, it makes sure that the result cannot contain $x_i > x_j > x_k$ where i, j, k is any permutation of $\{1, 2, 3\}$. This is done by restricting the classes of orderings allowed for the agents.

Definition 45 (non-chaining) A profile is *non-chaining* iff, for any triple of outcomes, only one of the following situations, that we call respectively situations α and β, can happen:

- the outcomes are all incomparable,
- only two of them are ordered, or
- there is one of them, which is more preferred than the other two.

Or:

- the outcomes are all incomparable;
- only two of them are ordered, or
- there is one of them, which is less preferred than the other two.

Theorem 9 [Ven05] *If all profiles are non-chaining and without ties, then the majority rule is weakly fair.*

4.3 Fairness for social choice functions over partial orders

In some situations, the result of aggregating the preferences of a number of agents might not need to be an order over outcomes. It might be enough to know the "most preferred" outcomes. For example, when aggregating the preferences of two people who want to buy an apartment, we don't need to know whether they prefer an 80 square metre apartment at the ground floor or a 50 square metre apartment at the 2nd floor, if they both prefer a 100 square metre apartment at the 3rd floor. They would just buy the 3rd floor apartment without trying to order the other two apartments. Social choice functions identify such most preferred outcomes, and do not care about the ordering on the other outcomes.

A social choice function on total orders is a mapping from a profile to the optimal outcome, or winner. With partial orders, there can be several outcomes which are incomparable and optimal. We have therefore defined the following generalization.

Definition 46 (social choice function over POs) A *social choice function over partial orders* is a mapping from a profile to a non-empty set of outcomes, called the optimal outcomes, or the winners.

We need to modify slightly the usual notions to deal with this generalization.

Definition 47 (unanimity) A social choice function f over partial orders is *unanimous* if and only if

- given any profile p where outcome $a \in top(p_i)$ for every agent i, then $a \in f(p)$;
- given any profile p where $\{a\} = top(p_i)$ for every agent i, then $f(p) = \{a\}$;

Definition 48 (monotonicity) A social choice function f over partial orders is *monotonic* iff, given two profiles p and p',

- if $a \in f(p)$ and for any other alternative b, $a >_{p_i} b$ implies $a >_{p'_i} b$ and $a \bowtie_{p_i} b$ implies $a \bowtie_{p_i} b$ or $a >_{p'_i} b$, for all agents i, then we have $a \in f(p')$;
- if $f(p) = A$ and for all $a \in A$, for all b, $a >_{p_i} b$ implies $a >_{p'_i} b$ and $a \bowtie_{p_i} b$ implies $a \bowtie_{p'_i} b$ or $a >_{p'_i} b$, for all agents i, then $f(p') = A$.

Definition 49 (ontoness) A social choice function f over partial orders is *onto* iff, for every subset of alternatives S, there is a profile p such that $f(p) = S$.

As for social welfare functions over partial orders, we will define three notions of dictators.

Definition 50 (dictatorship) Given a social choice function f over partial orders,

- a *strong dictator* is an agent i such that, for all profiles p, $f(p) = top(p_i)$;
- a *dictator* is an agent i such that, for all profiles p, $f(p) \subseteq top(p_i)$;
- a *weak dictator* is an agent i such that, for all profiles p, $f(p) \cap top(p_i) \neq \emptyset$.

Notice that, in any profile p, if a is the unique top of a weak dictator i, then $a \in f(p)$. However, this is not true if a is not the unique top of i.

Notice also that these three notions are consistent with the corresponding ones for social welfare functions. More precisely, a dictator (respectively, weak, strong) for a social welfare function f is also a dictator (respectively weak, strong) for the social choice function f' obtained by f by $f'(p) = top(f(p))$ for every profile p.

Proposition 26 *A social choice function over partial orders can be at the same time unanimous, monotonic, and have no dictators.*

For example, the social choice function corresponding to the Pareto rule is unanimous, monotonic, and has no dictators. However, all the agents are weak dictators. Another example is the social choice function which returns $\bigcup_i top(p_i)$, which again is unanimous, monotonic, and has no dictators (but all agents are weak dictators). On the other hand, the Lex rule has a strong dictator (which is the first agent).

The Muller-Satterthwaite's theorem [MS77], which is the Arrow's impossibility theorem in the case of social choice functions, can be generalized to social choice functions over partial orders without ties, for weak dictators. This means that, even if we are only interested in obtaining a set of winners, rather than a whole preference ordering over all the outcomes, it is impossible to be weakly fair.

Theorem 10 *If we have at least two agents and at least three outcomes, and the social choice function on partial order without ties is unanimous and monotonic, then there is at least one weak dictator.*

b	...	b	$a \bowtie h$	$a \bowtie h$	...	$a \bowtie h$	b	...	b	b	$a \bowtie h$	...	$a \bowtie h$
$a \bowtie h$	...	$a \bowtie h$	b	.	...	.	$a \bowtie h$	...	$a \bowtie h$	$a \bowtie h$	.	...	.
.		.	.	.		.			.	.	.		.
.		.	.	.		.			.	.	.		.
.		.	.	.		.			.	.	.		.
.	...	.	.	b	...	b	.	...	.	.	b	...	b
1	...	$i-1$	i	$i+1$	...	n	1	...	$i-1$	i	$i+1$	...	n

Figure 4.2: Profiles p_1 and p_2.

Proof: The proof follows the scheme of the proof of the Muller-Satterthwaite's theorem that can be found in [Ren01].

Consider three alternatives a, b, and h, and a profile p where $a \bowtie h$ (where $\bowtie$ means incomparability) is at the top, above every other element, and b is the unique bottom, for all agents. By unanimity, $f(p)$ contains both a and h, and so $f(p)$ can be $\{a, h\}$ or $\{a, h, d\}$ where d is an alternative different from a, b, h, if any.

Let us now rise b one position at time in agent 1's ranking. By monotonicity, the set of winners still contains both a and h, as long as $b < a$ and $b < h$. When b is risen above a and h, by monotonicity the set of winners may contain b. If we continue this with the other agents in the order, at the end we must have b as the only winner by unanimity. Thus at some point b must appear in the set of winners.

Step 1. Consider profiles p_1 and p_2. p_1 is the last profile where the set of winners is still $\{a, h\}$ or $\{a, h, d\}$, where d is one or more other elements, whereas p_2 is the first profile such that the set of winners contain b.

If $d \in f(p_1)$, then by monotonicity on profiles p_1 and p_2, we have $d \in f(p_2)$. If instead $d \notin f(p_1)$, then $d \notin f(p_2)$. In fact, assume $d \in f(p_2)$; then, by monotonicity on p_2 and p_1, $d \in f(p_1)$ as well, which is a contradiction. Therefore,

- if $f(p_1) = \{a, h\}$ then $f(p_2)$ can be $\{b\}$, $\{a, b\}$, $\{h, b\}$ or $\{a, h, b\}$;
- if $f(p_1) = \{a, h, d\}$ then $f(p_2)$ can be $\{b, d\}$, $\{a, b, d\}$, $\{h, b, d\}$ or $\{a, h, b, d\}$.

Step 2. Consider the new profiles p'_1 and p'_2 in Figure 4.3.

Notice that $f(p'_2)$ must contain b, by monotonicity on p_2 and p'_2.

If $d \in f(p_2)$, then $d \in f(p'_2)$ by monotonicity on p_2 and p'_2. If $h \notin f(p_2)$, then $h \notin f(p'_2)$. In fact, assume $h \in f(p'_2)$; then monotonicity on p'_2 and p_2 implies $h \in f(p_2)$, that is a contradiction. Analogously, if $a \notin f(p_2)$, then $a \notin f(p'_2)$.

If $f(p_1) = \{a, h, d\}$, we know from Step 1 that $f(p_2)$ contains d. Then, for the reasoning above, $d \in f(p'_2)$. Hence $f(p'_2)$ can be $\{b, d\}$, $\{a, b, d\}$, $\{h, b, d\}$ or $\{a, h, b, d\}$. Whereas, if

b	$\dots$	b	$a \bowtie h$	.	$\dots$	.	b	$\dots$	b	b	.	$\dots$	.
.	$\dots$	.	b	.	$\dots$	.	.	$\dots$	.	$a \bowtie h$	.	$\dots$	.
.		.	.	.		.	.		.	.	.		.
.		.	.	.		.	.		.	.	.		.
.		.	.	$a \bowtie h$	$\dots$	$a \bowtie h$	.		.	.	$a \bowtie h$		$a \bowtie h$
$a \bowtie h$	$\dots$	$a \bowtie h$	.	b	$\dots$	b	$a \bowtie h$	$\dots$	$a \bowtie h$	.	b	$\dots$	b
1	$\dots$	$i-1$	i	$i+1$	$\dots$	n	1	$\dots$	$i-1$	i	$i+1$	$\dots$	n

Figure 4.3: Profiles p'_1 and p'_2.

$f(p_1) = \{a, h\}$, then we know only that $f(p'_2)$ must contain b, therefore $f(p'_2)$ can be $\{b\}$, $\{b, d\}$, $\{a, b\}$, $\{a, b, d\}$, $\{h, b\}$, $\{h, b, d\}$, $\{a, h, b\}$ or $\{a, h, b, d\}$.

In particular, if $f(p_2)$ is $\{b, d\}$ or $\{b\}$, then by monotonicity on p_2 and p'_2, $f(p'_2)$ is respectively $\{b, d\}$ or $\{b\}$, and if $f(p_2) \neq \{b\}$, then $f(p'_2) \neq \{b\}$. In fact, suppose $f(p'_2) = \{b\}$. Then by monotonicity on p'_2 and p_2, $f(p_2) = \{b\}$, that is a contradiction. Moreover, for the reasoning above, if $f(p_2)$ is $\{a, b, d\}$ or $\{a, b\}$, i.e., $h \notin f(p_2)$, then $h \notin f(p'_2)$, and analogously, if $f(p_2)$ is $\{h, b, d\}$ or $\{h, b\}$, i.e., $a \notin f(p_2)$, then $a \notin f(p'_2)$.

Summarizing,

- if $f(p_1) = \{a, h, d\}$,
  - if $f(p_2) = \{b, d\}$, then $f(p'_2) = \{b, d\}$;
  - if $f(p_2) = \{a, b, d\}$, then $f(p'_2) = \{b, d\}$ or $\{a, b, d\}$;
  - if $f(p_2) = \{h, b, d\}$, then $f(p'_2) = \{b, d\}$ or $\{h, b, d\}$;
  - if $f(p_2) = \{a, h, b, d\}$, then $f(p'_2)$ can be $\{b, d\}$, $\{a, b, d\}$, $\{h, b, d\}$ or $\{a, h, b, d\}$;
- if $f(p_1) = \{a, h\}$,
  - if $f(p_2) = \{b\}$, then $f(p'_2) = \{b\}$;
  - if $f(p_2) = \{a, b\}$, then $f(p'_2) = \{b, d\}$, $\{a, b\}$, $\{a, b, d\}$;
  - if $f(p_2) = \{h, b\}$, then $f(p'_2) = \{b, d\}$, $\{h, b\}$, $\{h, b, d\}$;
  - if $f(p_2) = \{a, b, h\}$, then $f(p'_2) = \{b, d\}$, $\{a, b\}$, $\{a, b, d\}$, $\{h, b\}$, $\{h, b, d\}$, $\{a, h, b\}$ or $\{a, h, b, d\}$.

Hence, $f(p'_2)$ can be $\{b\}$, $\{b, d\}$, $\{a, b\}$, $\{a, b, d\}$, $\{h, b\}$, $\{h, b, d\}$, $\{a, h, b\}$ or $\{a, h, b, d\}$.

Notice that $f(p'_1)$ doesn't contain b. In fact, if we suppose $b \in f(p'_1)$, then by monotonicity on p'_1 and p_1, also $f(p_1)$ should contain b. But this is a contradiction, since $f(p_1)$ doesn't contain b.

Moreover, $f(p'_1) \neq \{d\}$. In fact, if $f(p'_1) = \{d\}$, then by strict monotonicity on p'_1 and p'_2, $f(p'_2) = \{d\}$, that is not one of the possible cases for $f(p'_2)$. Hence, $f(p'_1)$ can be $\{a\}$, $\{h\}$, $\{a, h\}$, $\{a, d\}$, $\{h, d\}$, $\{a, h, d\}$.

If $d \in f(p'_2)$, then $d \in f(p'_1)$ for monotonicity on profiles p'_2 and p'_1. If $d \notin f(p'_2)$, then $d \notin f(p'_1)$. In fact, if we suppose $d \in f(p'_1)$, then monotonicity on p'_1 and p'_2, implies $d \in f(p'_2)$, that is a contradiction.

Moreover, if $a \in f(p'_2)$, then, for monotonicity on profile p'_2 and p'_1, $a \in f(p'_1)$ and, for the same reason, if $h \in f(p'_2)$, then for monotonicity on profile p'_2 and p'_1, $h \in f(p'_1)$.

If $f(p'_2) = \{b, d\}$, $\{a, b, d\}$, $\{h, b, d\}$ or $\{a, h, b, d\}$, then for the reasoning above, $f(p'_1)$ must contain d and so it can be $\{a, d\}$, $\{h, d\}$, $\{a, h, d\}$, whereas if $f(p'_2) = \{b\}$, $\{a, b\}$, $\{h, b\}$, or $\{a, h, b\}$, then $f(p'_1)$ cannot contain d and so it can be $\{a\}$, $\{h\}$ or $\{a, h\}$.

More precisely,

- if $f(p'_2) = \{b\}$, then $f(p'_1)$ can be $\{a\}$, $\{h\}$ or $\{a, h\}$;
- if $f(p'_2) = \{a, b\}$, then $f(p'_1)$ can be $\{a\}$ or $\{a, h\}$;
- if $f(p'_2) = \{h, b\}$, then $f(p'_1)$ can be $\{h\}$ or $\{a, h\}$;
- if $f(p'_2) = \{a, h, b\}$, then $f(p'_1) = \{a, h\}$;
- if $f(p'_2) = \{b, d\}$, then $f(p'_1)$ can be $\{a, d\}$, $\{h, d\}$ or $\{a, h, d\}$;
- if $f(p'_2) = \{a, b, d\}$, then $f(p'_1)$ can be $\{a, d\}$ or $\{a, h, d\}$;
- if $f(p'_2) = \{h, b, d\}$, then $f(p'_1)$ can be $\{h, d\}$ or $\{a, h, d\}$;
- if $f(p'_2) = \{a, h, b, d\}$, then $f(p'_1) = \{a, h, d\}$;

Step 3. Consider an alternative e, distinct from a, h, and b, and the arbitrary profile p_3 in Figure 4.4, obtained from the profile p'_1 without changing the ranking of a and h versus any other alternative in all agents' rankings, bringing b just above a and h (which are at the bottom) for agents $j < i$, and inserting the alternative e just above b for $j \leq i$ and just above a and h for $j > i$.

Notice that $f(p_3)$ must not contain b, in fact if $b \in f(p_3)$ then, by monotonicity on p_3 and p_1, $b \in f(p_1)$, that is a contradiction. Hence $f(p_3)$ can be $\{a\}$, $\{h\}$, $\{d\}$, $\{a, h\}$, $\{a, d\}$, $\{h, d\}$, $\{a, h, d\}$.

By monotonicity on profiles p'_1 and p_3, if $a \in f(p'_1)$ then $a \in f(p_3)$ and if $h \in f(p'_1)$ then $h \in f(p_3)$. Moreover if $a \notin f(p'_1)$ then $a \notin f(p_3)$, in fact if we assume $a \in f(p_3)$, then

.	...	.	$a \bowtie h$	.	...	.
.	...	.	e	.	...	.
.		.	b	.		.
e		e	.	e		e
b		b	.	$a \bowtie h$		$a \bowtie h$
$a \bowtie h$	...	$a \bowtie h$	.	b	...	b
1	...	$i-1$	i	$i+1$	...	n

.	...	.	$a \bowtie h$	.	...	.
.	...	.	e	.	...	.
.		.	b	.		.
e		e	.	e		e
b		b	.	b		b
$a \bowtie h$	...	$a \bowtie h$	.	$a \bowtie h$	...	$a \bowtie h$
1	...	$i-1$	i	$i+1$	...	n

Figure 4.4: Profiles p_3 and p_4.

monotonicity on profiles p_3 and p'_1 produces $a \in f(p'_1)$, that is a contradiction. Analogously, if $h \notin f(p'_1)$ then $h \notin f(p_3)$.

By monotonicity on profiles p'_1 and p_3, if $f(p'_1) = \{a\}$ then $f(p_3) = \{a\}$, if $f(p'_1) = \{h\}$ then $f(p_3) = \{h\}$ and if $f(p'_1) = \{a, h\}$ then $f(p_3) = \{a, h\}$. In particular, if $f(p'_1) \neq \{a\}$ then $f(p_3) \neq \{a\}$. In fact if $f(p_3) = \{a\}$, then by monotonicity on p_3 and p'_1, $f(p'_1)$ must be $\{a\}$, that is a contradiction. Analogously, if $f(p'_1) \neq \{h\}$ then $f(p_3) \neq \{h\}$ and if $f(p'_1) \neq \{a, h\}$ then $f(p_3) \neq \{a, h\}$.

By Step 2 we know that $f(p'_1)$ can be $\{a\}$, $\{h\}$, $\{a, h\}$, $\{a, d\}$, $\{h, d\}$ or $\{a, h, d\}$, therefore, applying the reasoning above, we have that $f(p'_1) = f(p_3)$.

Step 4. Consider profile p_4 derived from profile p_3 by swapping the ranking of alternatives a and b for agents $j > i$, and profile p'_4 obtained from p_4 by bringing alternative e at the unique top of every agent's ranking. By unanimity, $f(p'_4) = \{e\}$. Note that $f(p_4)$ does not contain b. In fact, if $b \in f(p_4)$, by monotonicity on profiles p_4 and p'_4, $b \in f(p'_4)$, that is a contradiction since $f(p'_4) = \{e\}$. If $d \notin f(p_3)$, then $d \notin f(p_4)$ and if $d \in f(p_3)$, then $d \in f(p_4)$. Moreover, if $h \notin f(p_3)$, then $h \notin f(p_4)$ and analogously if $a \notin f(p_3)$, then $a \notin f(p_4)$. In fact, if $a \in f(p_4)$, by monotonicity on p_4 and p_3, then $a \in f(p_3)$, that is a contradiction. Notice that if $f(p_3) \neq \{a\}$, then $f(p_4) \neq \{a\}$. In fact, if we assume $f(p_4) = \{a\}$, then by monotonicity on profiles p_4 and p_3, $f(p_3)$ must be $\{a\}$, that is a contradiction. Analogously, if $f(p_3) \neq \{h\}$ then $f(p_4) \neq \{h\}$, if $f(p_3) \neq \{a, h\}$ then $f(p_4) \neq \{a, h\}$, if $f(p_3) \neq \{a, d\}$ then $f(p_4) \neq \{a, d\}$ and if $f(p_3) \neq \{a, h, d\}$ then $f(p_4) \neq \{a, h, d\}$.

By Step 3, we know that $f(p_3)$ can be $\{a\}$, $\{h\}$, $\{a, h\}$, $\{a, d\}$, $\{h, d\}$ or $\{a, h, d\}$. Therefore, by reasoning above, $f(p_3) = f(p_4)$.

Step 5. Consider an arbitrary profile p_5, with a and h the only top elements of agent i's ranking. It can be obtained from profile p_4 without reducing the ranking of a and h versus any other alternative in any agent's ranking. Remember that, by step 4, $f(p_4)$ can be $\{a\}$, $\{h\}$, $\{a, h\}$, $\{a, d\}$, $\{h, d\}$ or $\{a, h, d\}$. By monotonicity on profiles p_4 and p_5, if $a \in f(p_4)$, then $a \in f(p_5)$, and if $h \in f(p_4)$, then $h \in f(p_5)$. Therefore, since in all possible cases

$f(p_4)$ contains a or h (where or is not exclusive), the set of winners of an arbitrary profile, i.e. $f(p_5)$, must contain at least one (a or h) of the tops of the agent i. Thus agent i is a weak dictator.

It is easy to see that this proof can be easily generalized to the case of *more than two tops* for agent i. Moreover, the case of just one top for agent i can be proved via a simpler version of this proof. □

4.4 Strategy proofness in preference aggregation

In the previous part of this chapter we have shown that many problems require us to combine the preferences of different agents. Moreover, we have explained that, when aggregating preference orders, one may be interested in obtaining a combined ordering among the outcomes, or just the set of most preferred outcomes. In the first choice the aggregation function, which is a social welfare function, provides more information about the combined preference ordering since it also tells us the ordering of two outcomes which are not among the best ones. In the second scenario the aggregation function, which is a social choice function, is less informative but often enough when we are just interested in choosing one of the most preferred outcomes. In Section 4.2.5 we have also considered a property that has been studied by many people, both for social choice and for social welfare functions, the fairness property.

In this section we focus on another very desirable property of preference aggregation, which is non-manipulability (also called strategy proofness). It should not be possible for agents to manipulate the election by voting strategically. Strategic voting is when agents express preferences which are different from their real ones, to get the result they want. If this is possible, then the preference aggregation rule is said to be manipulable. In this section we focus on social choice functions. Thus the result of aggregating preferences will be just a set of outcomes. For social choice rules on *totally* ordered preferences, the Gibbard-Satterthwaite's result [Gib73] tells us that it is not possible to be at same time non-manipulable and have no dictators. Either there is a dictator (that is, an agent who gets what he wants by voting sincerely) or a manipulator (that is, an agent who gets what he wants by lying). In either case, there is an agent who gets what he wants no matter what the other agents say.

We extend this result to *partially* ordered preferences. Even in this more general case, we prove that it is impossible for a social choice function to have no dictator and be non-manipulable at the same time. As with total orders, we conjecture that there will be ways around this negative result. For example, it may be that certain social choice functions on partial orders are computationally hard to manipulate. As another example, it may be that certain restrictions on the way agents vote (like single-peaked preferences for total orders) guarantee strategy-proofness.

4.4.1 Strategy proofness for partial orders

The Gibbard-Satterthwaite's theorem on totally ordered preferences [Gib73, Sat75] proves that either we have a dictator or the social choice function can be manipulated. That is, agents can manipulate the result using tactical voting. Once we know that fairness is impossible for partial orders, we may now wonder if a similar relationship holds between weak dictators and non-manipulability. To answer this question, we generalize the notion of non-manipulability (also called strategy proofness) to social choice functions on partially ordered preferences, and we show that the Gibbard-Satterthwaite's theorem [Gib73, Sat75] holds in this more general context.

In particular, we show that weak dictators are inevitable if we have at least two agents and three outcomes, and the social choice function is strategy proof and onto. These conditions are identical to those in the Gibbard-Satterthwaite's theorem for total orders. A social choice function is strategy-proof if it is best for each agent to order outcomes as he really prefers and not to try to order them tactically, with the hope of getting a better result. More precisely, the social choice function must never allow an agent to get a preferred outcome among the winners by ordering outcomes in a way that contradicts his true preferences [ASS02].

If we assume that each agent's preference specifies a partial order over the possible outcomes then we can generalize the notion of strategy proofness as follows.

Definition 51 (strategy proofness) A social choice function f over partial orders is *strategy proof* if, for every agent i, for every pair of profiles p and p', which differ only for the agent i ranking, that is, p_i, $\forall a \in f(p) - f(p')$ and $\forall b \in f(p')$,

- if $a \bowtie_{p_i} b$, then $a \bowtie_{p'_i} b$ or $a <_{p'_i} b$,
- if $a <_{p_i} b$, then $a <_{p'_i} b$,

and $\exists b \in f(p')$ such that one of the following holds:

- $(a >_{p_i} b)$ and $(a \bowtie_{p'_i} b$ or $a <_{p'_i} b)$,
- $(a \bowtie_{p_i} b)$ and $(a <_{p'_i} b)$.

In other words, a social choice function is strategy proof if an agent can remove an element a from the set of winners only by worsening its rank with respect to at least one the new winners b. This means that it is not possible for an agent to make a disappear from the set of winners by improving its ranking in his preference ordering. In fact, this would be tactical voting: lowering the rank of an outcome to make it a winner.

In general, even in the totally ordered case, most voting procedures involving three or more alternatives are not strategy proof [ASS02]. This is true also in the partially ordered case.

Example 23 Consider the following two social choice functions:

- f_1 is such that $f_2(p) = \bigcup_i top(p_i)$, that is, this function returns the union of the sets of optimal elements of each agent;
- f_2 is the Pareto function, which returns the optimal elements of the ordering returned by the Pareto social welfare function (where $a > b$ is all agents say $a > b$, otherwise $a \bowtie b$);

Let us now consider two profiles p and p' on three alternatives a, b and c such that $p = (p_1 = (c > a \wedge c \bowtie b \wedge a \bowtie b)$, $p_2 = (c \bowtie a \wedge c \bowtie b \wedge a \bowtie b)$, $p_3 = p_1)$ and $p' = (p'_1 = p_1$, $p'_2 =(c > a > b)$, $p'_3 = p_3)$. Then, for both such social choice functions, the set of winners in profile p is $\{a, b, c\}$, while in profile p' is $\{b, c\}$. Thus a has disappeared by passing from p to p' but its ranking has improved with respect to b in agent i. Thus both social choice functions are not strategy-proof. □

4.4.2 Strategy-proofness: an impossibility result

We now generalize the Gibbard-Satterthwaite's theorem to the partially ordered case. In particular, we show that if a social choice function is strategy proof and onto, then there is at least a weak dictator. For showing this result, first we prove that if a social choice function is strategy proof and onto then it is unanimous and monotonic, then we conclude by using Theorem 10 which states that if a social choice function is unanimous and monotonic then there is at least a weak dictator.

Theorem 11 *If a social choice function f over partial orders is strategy proof and onto, then it is unanimous and monotonic.*

Proof: The proof is composed by two parts. Part 1 shows that if f is strategy proof then it is monotonic, while Part 2 shows that if f is onto and monotonic then it is unanimous.

Part 1. Consider profiles p and p', which differ only for the ranking of *agent* i.

Assume that $a \in f(p)$ and that for any other alternative b, $a >_{p_i} b$ implies $a >_{p'_i} b$ and $a \bowtie_{p_i} b$ implies $a \bowtie_{p'_i} b$ or $a >_{p'_i} b$. We want to show that $a \in f(p')$. For the sake of contradiction, assume that $a \notin f(p')$. Since f is strategy proof, then $\exists c \in f(p')$ such that one of the following holds:

- ($a >_{p_i} c$) and ($a \bowtie_{p'_i} c$ or $a <_{p'_i} c$),
- ($a \bowtie_{p_i} c$) and ($a <_{p'_i} c$).

If the first holds then there is an element c which is worse than a in p_i and that becomes strictly better or incomparable than a in p'_i. This contradicts the fact that for any other alternative b, $a >_{p_i} b$ implies $a >_{p'_i} b$. If the second holds then there is an element c which is incomparable with a in p_i and becomes strictly better then a in p'_i. This contradicts the fact that for any other alternative b, $a \bowtie_{p_i} b$ implies $a \bowtie_{p'_i} b$ or $a >_{p'_i} b$.

Assume now that $\forall a \in f(p)$, for any other alternative b, $a >_{p_i} b$ implies $a >_{p'_i} b$ and $a \bowtie_{p_i} b$ implies $a \bowtie_{p_i} b$ or $a >_{p'_i} b$. We want to show that $f(p) = f(p')$. For the sake of contradiction, we can assume that $\exists a$ such that $a \in f(p)$ and $a \notin f(p')$ or that $\exists a$ such that $a \in f(p')$ and $a \notin f(p)$. If $\exists a$ such that $a \in f(p)$ and $a \notin f(p')$ then, since f is strategy proof, the same reasoning above leads to the same contradictions. If instead $\exists a$ such that $a \in f(p')$ and $a \notin f(p)$, then since f is strategy proof, then $\exists c \in f(p)$ such that one of the following holds:

- ($a >_{p'_i} c$) and ($a \bowtie_{p_i} c$ or $a <_{p_i} c$),
- ($a \bowtie_{p'_i} c$) and ($a <_{p_i} c$).

In the first case there is an element a which is strictly smaller than or incomparable to c in p_i that becomes strictly greater than c in p'_i. This is in contradiction either with the fact that for any other alternative b, $c \bowtie_{p_i} b$ implies $c \bowtie_{p'_i} b$ or $c >_{p'_i} b$ or with the fact that $c >_{p_i} b$ implies $c >_{p'_i} b$. If the second case holds then there is an element a that is smaller than c in p_i and that becomes incomparable with c in p'_i. This contradicts the assumption that for any other alternative b, $c >_{p_i} b$ implies $c >_{p'_i} b$.

Consider a profile q such that $f(q) \supseteq \{a\}$ and a profile q' such that *for every agent* i and for every alternative b, $a >_{q_i} b$ implies $a >_{q'_i} b$. We want prove that $f(q') \supseteq \{a\}$, that is the first part of the definition of the monotonicity for social choice functions. Since we can move from $q = (q_1, ...q_n)$ to $q' = (q'_1, ...q'_n)$, passing from $q = (q_1, q_2..., q_n)$ to $(q'_1, q_2..., q_n)$, and $(q'_1 q'_2, ..., q_n)$ to $(q'_1, q'_2, ..., q_n)$ and so on, and we have shown above that at each step a remains in the set of winners, $a \in f(q')$. The same reasoning holds for profiles q such that $f(q) = A$ and q' such that for every agent i, $\forall a \in A$, for every other alternative b, such that $a >_{q_i} b$ implies $a >_{q'_i} b$. In this case we conclude that $f(q') = A$. This is the second part of the definition of monotonicity for social choice function. We have thus shown that f is monotonic.

Part 2. Since f is onto, then for every subset S of alternatives there is a profile p such that $f(p) = S$.

If $S = \{a\}$, where is a is an alternative, since f is onto, there is a profile p such that $f(p) = \{a\}$. If we consider the profile p', obtained from profile p bringing a to the very top of every agent, then for strict monotonicity on profiles p and p', that we have just proved, $f(p') = a$. Therefore, whenever a is the unique top of every agent's ranking in a profile $\bar{p}$, then $f(\bar{p}) = \{a\}$. Because a is arbitrary then f satisfies pareto efficiency in the case of unique top for every agent.

If $S \supset \{a\}$, since f is onto, then there is a profile p_1 such that $f(p_1) = S \supset \{a\}$. If we consider a profile p'_1 where a is at the top (not unique) for every agent, then, for monotonicity on profiles p_1 and p'_1, $f(p'_1)$ must contain a. Therefore, whenever a is one of the tops of every agent's ranking in a profile $\bar{p}$, then $f(\bar{p}) \supseteq \{a\}$. Because a is arbitrary, then f satisfies pareto efficiency in the case of not unique top for every agent. □

We now use this result, together with the extension of the Muller-Satterthwaite's result to easily prove the main theorem.

Theorem 12 *If there are at least two agents and at least three outcomes and the social choice function over partial orders is strategy-proof and onto, then there is at least one weak dictator.*

Proof: By Theorem 11, if a social choice is onto and strategy proof then it is monotonic and unanimous. We can conclude via Theorem 10, which states that if a social choice is monotonic and unanimous then there is at least one weak dictator. □

This means that weak dictators are present not only when the function is unanimous and monotonic, but also when it is strategy-proof and onto. In other words, it is not possible for a social choice function to be at the same time strategy proof and onto, and have no weak dictators.

4.5 Related work

Preference aggregation has attracted interest from the AI community fairly recently. This has motivated work on both preference representation and aggregation. In particular, voting theory has been considered in light of AI specifications and in terms of combinatorial sets of candidates. The work by Jerome Lang in [Lan02, Lan04] is related to ours since it considers a scenario where there are multiple agents expressing preferences which must be combined in a unifying result. This work is a very interesting survey in the field since it presents the various logical preference representation approaches and considers them in terms of complexity for answering the usual queries of interest.

Since the original theorem by Arrow on combining total orders, some effort has been made to weaken its conditions. Both the domain and the codomain of a social welfare function have been the subject of more relaxed assumptions in several Arrow-like impossibility theorems.

- In [Fis74], the codomain is a partial order, and profiles are allowed to be strict weak orders, which are negatively transitive and asymmetric. This structure is more general than total orders but less general than partial orders, since, for example, it does not allow situations where $A > B$ and C is incomparable to both A and B.

- In [Bar82], social orders can be partial, and agents are allowed to vote using a partial order. However, the set of profiles must be regular, meaning that for any three alternatives, every configuration of their orders must be present in a profile.

- In [Wey84] agents must vote using total orders. However, the social order can be a quasi-ordering, which is reflexive and transitive. A similar setting is considered in [DFP02], where agents use total orders with some additional requirements (such as the discrimination axiom which requires that each agent orders strictly at least one triple of candidates).

- In [DW91] each agent models his preferences using a non monotonic logic. Thus, it gives a preorder on the outcomes. The concept of *aggregation policy* is defined as a function that specifies the global preorder corresponding to any given set of individual preorders. A result similar to the Arrow's impossibility theorem [Arr51] is given for this scenario where the agents use preorders and also the result is a preorder. However an additional hypothesis is added, namely *conflict resolution*. This property requires that if a pair is ordered in some way by at least an agent, then it must be ordered also in the global preorder. In other words, all pairs that are comparable for some agent cannot be incomparable in the resulting preorder. In [DW91] it is shown that no aggregation policy can be free, unanimous, independent of irrelevant alternatives, non dictatorial and respect also conflict resolution. It should be noticed that the definition of non dictatorship used in [DW91] corresponds to that of dictator which we have given in Section 4.2.5. This result is thus different from the one we propose here. On one side, they are more general, since they allow preorders while we allow only partial orders. However conflict resolution is a very strong property to require. For example the Pareto semantics defined in 4.2.2 does not respect it. Moreover we use a weaker version of dictatorship while they use a stronger version not adapted to deal with incomparability.

With respect to all these approaches, our profiles are more general, since in our results a profile can be any set of partial orders. However, the resulting order of a social welfare function is required to be a restricted partial order, that is, a partial order with a unique top or a unique bottom. Thus our result is incomparable to these previous results. In addition, our possibility theorem for the majority rule is, to our knowledge, the first result of this kind for partially ordered profiles in social welfare functions. The same holds also for our impossibility result about fairness for social choice functions.

Similarly, efforts have been made to weaken the conditions of the Gibbard-Satterthwaite's theorem.

- In [Bar83], the domain of the social choice function has been generalized to preferences over sets of outcomes and an impossibility result proved. In addition, efforts have been made to identify specific situations where social choice rules are strategy proof.

- In [NP05], the domain is restricted to single-peaked preferences and it is shown that social choice can be non-dictatorial and strategy-proof in this situation. However, the Gibbard-Satterthwaite's theorem has been shown to be robust to several other restrictions of the domain of the social choice function.

4.6 Conclusions

We considered how the preferences of multiple agents can be aggregated together to give the set of most preferred outcomes. We viewed the agents as voting for their preferred outcomes, and preference aggregation in terms therefore of a social choice voting rule. We proved that social choice functions over partial orders cannot be weakly fair (that is, they cannot be at the same time unanimous, independent to irrelevant alternatives, and with no weak dictator). This result generalizes Arrow's impossibility theorem for combining total orders [Arr51] in the case of social choice functions [MS77].

We introduced the notion of strategy proofness (or non-manipulability) for the aggregation of partially ordered preferences. We proved that when the social choice function is strategy proof and onto, weak dictators are also present. This result shows that the fact that agents can use partial orders to express their preferences does not change one of the fundamental properties of a social choice function: the absence of dictators implies the social choice function can be manipulated. In other words, when aggregating preferences, whether they are totally or partially ordered, no matter which aggregation function we use, there will always be one agent who can get what he wants, either by voting sincerely or by lying about his preferences.

4.7 Future work

Partiality in the preference ordering is interpreted here as incomparability. However, it is sometimes useful to interpret it as lack of knowledge, for example in contexts where the agents don't want to reveal all their preferences, or when we are eliciting preferences and agents do not reveal their preferences all at once. Moreover, the two interpretations of partiality in the preference ordering can be combined, since agents may want to express that some objects are really incomparable, while they may want to not say the actual relationship among other objects. In this more general context, the notion of winners can be generalized to the notions of *possible* and *necessary* winners [KL05]. We plan to study in this scenario the fairness and the non-manipulability properties presented in this chapter.

We proved an impossibility result for aggregating partially ordered preferences. It is likely that there are ways around this negative result. We plan for example to investigate certain social choice functions on partial orders which may be computationally hard to manipulate. As another example, we intend to find certain restrictions on the way agents vote (like single-peaked preferences for total orders) which may guarantee strategy-proofness. We are also interested in studying other computational aspects of preference aggregation. For example, we want to investigate how difficult it is to manipulate a preference aggregation function.

Chapter 5

Preference aggregation with uncertainty: complexity of winner determination

In this chapter we consider how to combine the preferences of multiple agents despite the presence of incompleteness and incomparability in their preference orderings. An agent's preference ordering may be incomplete because, for example, there is an ongoing preference elicitation process. It may also contain incomparability, which can be useful, for example, in multi-criteria scenarios. We focus on the problem of computing the *possible* and *necessary* winners, that is, those outcomes which can be or always are the most preferred for the agents. Possible and necessary winners are useful in many scenarios, including preference elicitation. We show that computing the sets of possible and necessary winners is in general a difficult problem as it is providing a good approximation of such sets. We identify sufficient conditions, related to general properties of the preference aggregation function, where such sets can be computed in polynomial time. We show how possible and necessary winners can be used to focus preference elicitation.

Then, we focus on specific voting rules which perform a sequence of pairwise comparisons between two candidates, where the result of each is computed by a majority vote. The winner thus depends on the chosen sequence of comparisons, which can be represented by a binary tree. In this case there are candidates that will win in some trees (called *possible winners*) or in all trees (called *Condorcet winners*). While it is easy to find the possible and Condorcet winners, we prove that it is difficult if we insist that the tree is balanced. This restriction is therefore enough to make voting difficult for the chair to manipulate. We also consider the situation where we lack complete information about preferences and we determine the computational complexity of computing possible and Condorcet winners in this extended case considering balanced and unbalanced trees.

5.1 Motivations and chapter structure

We consider a multi-agent setting where each agent specifies his preferences by means of an ordering over the possible outcomes. A pair of outcomes can be ordered, incomparable, in a tie, or the relationship between them may not yet be specified. Incomparability and incompleteness represent very different concepts. Outcomes may be incomparable because the agent does not wish very dissimilar outcomes to be compared. For example, we might not want to compare a biography with a novel as the criteria along which we judge them are just too different. Outcomes can also be incomparable because the agent has multiple criteria to optimize. For example, we might not wish to compare a faster but more expensive laptop with a slower and cheaper one. Incompleteness, on the other hand, represents simply an absence of knowledge about the relationship between certain pairs of outcomes. Incompleteness arises naturally when we have not fully elicited an agent's preferences or when agents have privacy concerns which prevent them from revealing their complete preference ordering.

As we wish to aggregate together the agents' preferences into a single preference ordering, we must modify preference aggregation functions to deal with incompleteness. One possibility is to consider all possible ways in which the incomplete preference orders can be consistently completed. In each possible completion, preference aggregation may give different optimal elements (or *winners*). This leads to the idea of the *possible winners* (those outcomes which are winners in at least one possible completion) and the *necessary winners* (those outcomes which are winners in all possible completions) [KL05].

While voting theory has been mainly interested in possibility or impossibility results about social choice or social welfare functions, recently there has been some interest also in computational properties of preference aggregation [RVW04, Lan04, KL05, CS02b]. It has also been noted that the complexity of deciding whether there is a manipulation in an election is closely related to the complexity of computing possible winners [KL05, CS02a].

In this chapter we start by considering the complexity of computing the necessary and the possible winners. We show that both tasks are hard in general, even to approximate. Then we identify sufficient conditions that assure tractability. Such conditions concern properties of the preference aggregation function, such as monotonicity and independence to irrelevant alternatives (IIA) [ASS02], which are natural properties to require.

We show how possible and necessary winners are useful in many scenarios including preference elicitation [CP04]. For example, elicitation is over when the set of possible winners coincides with that of the necessary winners [CS02b]. However, recognizing when such a condition is satisfied is hard in general. We show that, if the preference aggregation function is IIA, preference elicitation can focus just on the incompleteness concerning those outcomes which are possible and necessary winners, allowing us to ignore all other outcomes and to complete preference elicitation in polynomial time.

Finally, we focus on specific voting rules which are incompletely specified. In particular, we consider a well-known family of voting rules based on *sequential majority comparisons*, where the winner is computed from a series of majority comparisons along a binary voting tree. The winner thus depends on the chosen binary voting tree. There are candidates that will win in some trees (called *possible winners*) or in all trees (called *Condorcet winners*). We study the impact, on these voting rules, of this new kind of incompleteness, which derives from the voting rule itself, and of the previous form of incompleteness, which derives from voters' preferences.

The work described in this chapter has appeared in the proceedings of the following conferences and international workshops:

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Computing possible and necessary winners from incomplete partially-ordered preferences. *In Proceedings of the 17th European Conference on Artificial Intelligence (ECAI 2006), Best poster Award*, IOS Press, vol.141, pp. 767-768, Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Incompleteness and incomparability in preference aggregation. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, held in conjunction of the 17th European Conference on Artificial Intelligence (ECAI 2006), Riva del Garda, Italy, August 2006.

- J. Lang, M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Winner determination in sequential majority voting with incomplete preferences. *In Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, held in conjunction of the 17th European Conference on Artificial Intelligence (ECAI 2006), Riva del Garda, Italy, August 2006.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Preference aggregation and elicitation: tractability in the presence of incompleteness and incomparability. *In Proceedings of DIMACS/LAMSADE Workshop on Computer Science and Decision Theory II*, Paris, France, October 2006.

- J. Lang, M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Winner determination in sequential majority voting. *In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007),* to appear, Hyderabad, India, January 2007.

- M. S. Pini, F. Rossi, K. B. Venable and T. Walsh. Incompleteness and incomparability in preference aggregation. *In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007),* to appear, Hyderabad, India, January 2007.

The chapter is organized as follows.

- In Section 5.2 we give some basic definitions on which this work is based. In particular, we define incomplete preferences and incomplete profiles, incomplete preference aggregation functions, possible and necessary winners, and we present the combined result, which is a way for compactly representing the set of the results of a preference aggregation function.
- In Section 5.3 we show that computing the sets of possible and necessary winners and good approximations of such sets is a difficult problem.
- In Section 5.4 we show that computing the combined result is a difficult problem and we identify properties on preference aggregation functions, which allows us to compute a good approximation of the combined result in polynomial time.
- In Section 5.5 we show that, starting from this good approximation of the combined result, it is polynomial to compute the sets of possible and necessary winners.
- In Section 5.6 we show how possible and necessary winners can be used in preference elicitation.
- In Section 5.7 we focus on a specific voting rule which is itself incompletely specified, named sequential majority voting rule, which consists on sequential majority comparisons, which is represented by a binary tree. We recall some basics on such a rule (Section 5.7.1). We deal with incompleteness in this voting rule (Section 5.7.2), and we study the computational difficulty of computing candidates that win in some or all possible binary trees. Then, we focus on binary trees where the number of competitions for each candidate is as balanced as possible, and we show that winner determination in this context is hard. It is however possible to build in polynomial time a tree featuring a bounded level of imbalance where a particular candidate A wins, if such a tree exists (Section 5.7.3). Finally, we consider the other kind of uncertainty where the agents have only partially revealed their preferences (Section 5.7.4).

- In Sections 5.8 and 5.9 we describe respectively related and future work.

5.2 Basic notions

In this section we give some basic notions that we will use in this chapter. We give the definitions of preferences and preference aggregation functions in presence of incompleteness. We define the necessary and the possible winners in the case of partially ordered preferences. Finally, we show a compact way for representing the set of results given by the preference aggregation function.

5.2.1 Incomplete preferences and profiles

Agent's preferences can be specified via a (possibly incomplete) partial order with ties over the set of possible outcomes, that we will denote by Ω. An incomplete partial order is a partial order where some relation between pairs of outcomes is unknown.

Definition 52 (IPO) An *incomplete partial order with ties* is a partial order with ties such that the relation between any pair of outcomes A and $B \in \Omega$ can be $A < B$, $A > B$, $A = B$, $A \bowtie B$, or $A?B$, where $A \bowtie B$ means that A and B are incomparable, and $A?B$ that the relation between A and B can be any element of $\{=, >, <, \bowtie\}$.

Example 24 Given outcomes A, B, and C, an agent may state preferences such as $A > B$, $B \bowtie C$, and $A > C$, or also $A > B$, $B \bowtie C$ and $A?C$. However, an agent cannot state preferences such as $A > B$, $B > C$, $C > A$, or also $A > B$, $B > C$, $A \bowtie C$ since neither are POs. □

If we consider a sequence of partial orders, one for every agent, where at least one of the partial orders is incomplete, then we obtain an incomplete PO profile.

Definition 53 (IPO profile) An *incomplete PO profile* ip is a sequence of partial orders $ip_1, \ldots, ip_n$ over a set of outcomes, such that $\exists i \in \{1, \ldots, n\}$, ip_i is incomplete.

5.2.2 Aggregation functions of incomplete preferences

We consider how to combine the preferences of multiple agents despite the presence of incompleteness and incomparability in their preference orderings. In particular, we define a preference aggregation function in presence of incompleteness starting from a social welfare function [ASS02]. We recall that, a social welfare function, as said in Section 4.2.3, maps profiles into partial orders with ties.

Definition 54 (preference aggregation function with incompleteness) A *preference aggregation function with incompleteness* is a function mapping IPO profiles into a sets of partial orders with ties (POs).

Given a social welfare function f, the corresponding preference aggregation function, written pa_f, works as follows. Given an IPO profile $ip = (ip_1, \ldots, ip_n)$, where the ip_i's are IPOs, we consider all the profiles, say $p_1, \ldots, p_k$, obtained from ip by replacing any occurrence of $?$ in the ip_i's with either $<$, $>$, $=$, or $\bowtie$ which is consistent with a partial order. Then we set $pa_f(ip) = \{f(p_1), \ldots, f(p_k)\}$. This set will be called the *set of results* of f on profile ip.

Example 25 In Figure 5.1 we present an IPO profile $ip = (ip_1, \ldots, ip_3)$, which expresses the preferences of three agents over three outcomes A, B, and C, and we show how the preference aggregation function, built starting from the Pareto social welfare function, works. We recall that the Pareto social welfare function, that we have described in Section 4.2.2, is such that, given a profile p, for any pair of outcomes A and B, if all agents say $A > B$ or $A = B$ and at least one says $A > B$ in p, then $A > B \in f(p)$; otherwise, $A \bowtie B \in f(p)$. □

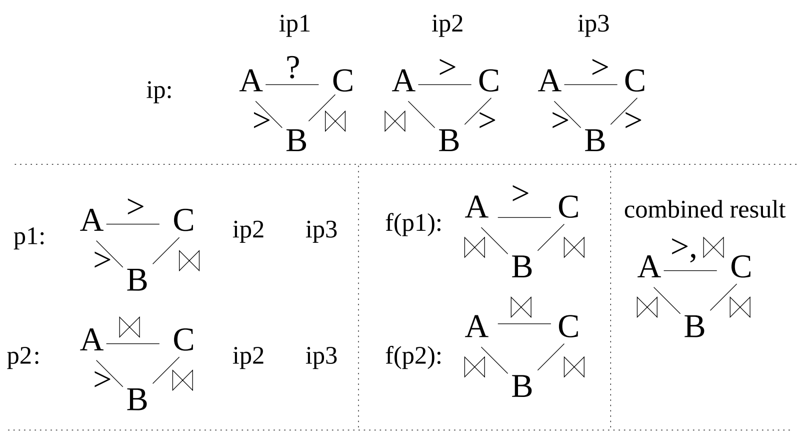

Figure 5.1: An IPO profile ip, its completions p_1 and p_2, the results $f(p_1)$ and $f(p_2)$, and the combined result $cr(f, ip)$.

5.2.3 Combined result

Unfortunately, the set of results can be exponentially large. We will therefore consider a compact representation that is polynomial in size. This may throw away information by compacting together results into a single combined result.

Definition 55 (combined result) Given a social welfare function f and an IPO profile ip, consider a graph, whose nodes are the outcomes, and whose arcs are labeled by non-empty subsets of $\{<, >, =, \bowtie\}$. Label l is on the arc between outcomes A and B if and only if there exists a PO in $pa_f(ip)$ where A and B are related by l. This graph is the *combined result* of f on ip, and it is denoted by $cr(f, ip)$. If an arc is labeled by set $\{<, >, =, \bowtie\}$, the combined result is *fully incomplete*, otherwise it is *partially incomplete*.

We denote the set of labels on the arc between the outcomes A and B with $rel(A, B)$.

Example 26 Consider the preference aggregation function shown in Example 25. Its combined result is shown in Figure 5.1. □

5.2.4 Possible and necessary winners for partial orders

We extend to the case of partial orders the notions of possible and necessary winners defined in [KL05] for total orders.

Definition 56 (necessary winners) Given a social welfare function f and an IPO profile ip, the *necessary winners* of f given ip are all those outcomes which are maximal elements in all POs in $pa_f(ip)$.

A necessary winner must be a winner, no matter how incompleteness is resolved in the IPO profile.

Definition 57 (possible winners) Given a social welfare function f and an IPO profile ip, the *possible winners* are all those outcomes which are maximal elements in at least one of the POs in $pa_f(ip)$.

A possible winner is a winner in at least one possible completion of the IPO profile.

We will write $NW(f, ip)$ and $PW(f, ip)$ for the set of necessary and possible winners of f on IPO profile ip. We will sometimes omit f and/or ip, and just write NW and PW when they will be obvious or irrelevant.

Example 27 In Example 25, A and B are necessary winners, since they are top elements in all the completions of the IPO profile ip, i.e., both in $f(p_1)$ and in $f(p_2)$. The outcome C is a possible winner, since it wins in $f(p_2)$, but it is not a necessary winner, since it doesn't win in $f(p_1)$. □

5.3 Computing possible and necessary winners: complexity results

We focus on the problem of computing the sets of possible and necessary winners in presence of incompleteness and incomparability. We prove that this computation is in general a difficult problem, as it is providing a good approximation of such sets. In particular, we show that computing the set of necessary and possible winners of a social welfare function is, in general, $\mathcal{NP}$-hard even if we restrict ourselves to incomplete but *total* orders. To do this, we will consider the following, well known, voting rule.

Single Transferable Vote

In the STV rule each voter provides a total order on candidates and, initially, an individual's vote is allocated to their most preferred candidate. The *quota* of the election is the minimum number of votes necessary to get elected. If only one candidate is to be elected then the quota is $\lfloor n/2 \rfloor + 1$, where n is the number of voters. If no candidate exceeds the quota, then, the candidate with the fewest votes is eliminated, and his votes are equally distributed among the second choices of the voters who had selected him as first choice. In what follows we consider STV elections in which some total orders, provided by the voters, are incomplete.

In general, given an IPO profile and a candidate a, we say PossibleWinner holds if and only if a is a possible winner of the election.

Theorem 13 `PossibleWinner` *is $\mathcal{NP}$-complete.*

Proof: In fact, membership in $\mathcal{NP}$ follows by giving a completion of the profile in which a wins. Completeness follows from the result that Effective Preference (determining if a particular candidate can win an election with one vote unknown) for STV is $\mathcal{NP}$-complete [BO91]. □

This result allows us to conclude that, in general, finding possible winners of an election is difficult. However, it should be noticed that for many rules used in practice including some positional scoring rules [KL05], answering `Possible Winner` is polynomial. The complexity of computing possible winners is related to the complexity of deciding whether there is a manipulation [KL05]. For instance, it is $\mathcal{NP}$-complete to determine for the Borda, Copeland, Maximin and STV rules if a coalition can cast weighted votes to ensure a given winner [CS02a]. It follows therefore that with weighted votes, `PossibleWinner` is $\mathcal{NP}$-hard for these rules.

Given an IPO profile and a candidate a, we say NECESSARYWINNER holds if and only if a is a necessary winner of the election.

Theorem 14 `NecessaryWinner` *is co$\mathcal{NP}$-complete.*

Proof: The complement problem is in $\mathcal{NP}$ since we can show membership by giving a completion of the profile in which some b different to a wins. To show completeness, we give a reduction from EFFECTIVE PREFERENCE with STV in which a appears at least once in first place in one vote. This restricted form of EFFECTIVE PREFERENCE is $\mathcal{NP}$-complete [BO91]. Consider an incomplete profile Π in which $n + 1$ votes have been cast, a has at least one first place vote, one vote remains unknown, and we wish to decide if a can win. We construct a new election from Π with n new additional votes, and one new candidate b. We put b at the top of each of these new votes, and rank the other candidates in any order within these n votes. We place b in last place in the original $n + 1$ votes, except for one vote where a is in first place (by assumption, one such vote must exist) where we place b in second place and shift all other candidates down. We observe that b will survive till the last round as b has at least n votes and no other candidate can have as many till the last round. We also observe that if a remains in the election, then the score given to each candidate by STV remains the same as in the original election so the candidates are eliminated in the same order up till the point a is eliminated. If a is eliminated before the last round, the second choice vote for b is transferred. Since b now has $n + 1$ votes, b is unbeatable and must win the election. If a survives, on the other hand, to the last round, we can assume b is ranked at the bottom of the unknown vote. All the other candidates but a and b have been eliminated so a has $n + 1$ votes and is unbeatable. Hence, if a is not a possible winner in the original election, b is the necessary winner of this new election. Thus determining the necessary winner of this new election decides if a is a possible winner of the original election. □

Given these results, we might wonder if it is easy to compute a reasonable approximation of the sets of possible and necessary winners. Unfortunately this is not the case. The reduction described in the proof of previous theorem shows that we cannot approximate the set of possible winners within a factor of two. In fact, we can show that we cannot approximate efficiently the set of possible winners within any constant factor.

Theorem 15 *It is $\mathcal{NP}$-hard to return a superset of the possible winners, PW^* in which we guarantee $|PW^*| < k|PW|$ for some given positive integer k, with $k > 1$.*

Proof: We again give a reduction from EFFECTIVE PREFERENCE for STV in which a appears at least once in first place in one vote. Consider an incomplete profile Π in which $n+1$ votes have been cast, a has at least one first place vote, one vote remains unknown, and we wish to decide if a can win. We construct a new election from Π. We make k copies of Π. In the ith copy Π_i, we subscript each candidate with the integer i. We add n new additional votes, and one new candidate b. We put b at the top of each of these new votes, and rank all the other candidates except a_i in any order within these n votes. The ranking of the candidates a_i is left unknown but beneath b. In each Π_i, we place b in last place except for one vote where a_i is in first place (by assumption, one such vote must exist) where we place b in second place and shift all other candidates down. Finally, for each candidate in Π_j not in Π_i except for a_j, we rank them in any order at the bottom of the votes in Π_i. The ranking of the candidates a_i is again left unknown but beneath b. We observe that b will survive till all but one candidate has been eliminated from one of the Π_i. We also observe that if a_i remains in the election, then the score given to each candidate by STV remains the same as in the original election so the candidates in Π_i are eliminated in the same order up till the point a_i is eliminated. Suppose a cannot win the original election. Then a_i will always be eliminated before the final round. The second choice vote for b is transferred. Since b now has at least $n+1$ votes, b is unbeatable and must win the election. Suppose, on the other hand, that a can win the original election. Then a_i can survive to be the last remaining candidate in Π_i. We can assume b is ranked at the bottom of the unknown votes of all the candidates with an index i and above all the candidates with an index j different to i. Thus a_i has $n+1$ votes. If we have the corresponding ranking in the other unknown votes, a_j for $j \neq i$ will also survive. Since b has only n votes, b will be eliminated. It is now possible for any of the candidates, a_i where $1 \leq i \leq k$ to win depending on how exactly the a_i are ranked in the different votes. Thus the set of possible winners is $\{a_i \mid 1 \leq i \leq k\}$ plus b if a is not a necessary winner in the original election. Hence, if a is a possible winner in the original election, the size of the set of possible winners is greater than or equal to k, whilst if it is not, the set is of size 1. If we know that $|PW^*| < k|PW|$, then $|PW^*| < k$ guarantees that $|PW| = 1$, b is the necessary winner and hence that a is not a possible winner in the original election. □

Similarly, we cannot approximate efficiently the set of necessary winners within some fixed ratio.

Theorem 16 *It is $\mathcal{NP}$-hard to return a subset of the necessary winners, NW^* in which we guarantee $|NW^*| > \frac{1}{k}|NW|$ whenever $|NW| > 0$ for some given positive integer k.*

Proof: In the reduction used in the last proof, $|NW| = 1$ if a is a possible winner in the original election and 0 otherwise. Suppose a is a possible winner. Then in the new election, $|NW| = 1$. As $|NW^*| > \frac{1}{k}|NW|$, it follows that $|NW^*| = 1$. Thus, the size of NW^* will determine if a is possible winner. □

5.4 Complexity of computing the combined result

We now consider the problem of computing the combined result. We show that, while in general it is difficult, there are some restrictions which allow us to compute an approximation of the combined result in polynomial time. In the next section, we will show how it is possible to compute in polynomial time the set of possible and necessary winners starting from this approximation of the combined result.

Theorem 17 *Given an IPO profile, determining if a label is in the combined result for STV is $\mathcal{NP}$-complete.*

Proof: In fact, a polynomial witness is a completion of the incomplete profile. To show completeness, we use a polynomial number of calls to this problem to determine if a given candidate is a possible winner. □

From this result we immediately get the following corollary.

Corollary 2 *Given an IPO profile and a social welfare function, computing the combined result is $\mathcal{NP}$-hard.*

We now give some properties of preference aggregation functions which allow us to compute an upper approximation to the combined result in polynomial time. We recall that the set of labels of an arc between A and B in the combined result is called $rel(A, B)$.

The first property we consider is *independence to irrelevant alternatives* (IIA). We recall that a social welfare function is said to be IIA when, for any pair of outcomes A and B, the ordering between A and B in the result depends only on the relation between A and B

given by the agents [ASS02]. Many preference aggregation functions are IIA, and this is a desirable property which is related to the notion of fairness in voting theory [ASS02]. Given a function which is IIA, to compute the set $rel(A, B)$, we just need to ask each agent their preference over the pair A and B, and then use f to compute all possible results between A and B. However, if agents have incompleteness between A and B, f has to consider all the possible completions, which is exponential in the number of such agents.

Assume now that f is also *monotonic*. We recall that a social welfare function f is monotonic when, for any two profiles p and p' and any two outcomes A and B if passing from p to p' B improves with respect to A in one agent i and $p_j = p'_j$ for all $j \neq i$, then, passing from $f(p)$ to $f(p')$, B improves with respect to A. Saying that B improves with respect to A means that the relationship between A and B does not move left along the following sequence: $>, \geq$, ($\bowtie$ or $=$), $\leq, <$.

Consider now any two outcomes A and B. To compute $rel(A, B)$ under IIA and monotonicity, again, since f is IIA, we just need to consider the agents' preferences over the pair A and B. However, now we don't need to consider all possible completions for all agents with incompleteness between A and B, but just two completions: $A < B$ and $A > B$. Function f will return a result for each of these two completions, say AxB and AyB, where $x, y \in \{<, >, =, \bowtie\}$. Since f is monotonic, the results of all the other completions will necessarily be between x and y in the ordering $>$, $\geq$, ($\bowtie$ or $=$), $\leq$, $<$. By taking all such relations, we obtain a superset of $rel(A, B)$, that we call $rel^*(A, B)$. In fact, monotonicity of f assures that, if we consider profile $A < B$ and we get a certain result, then considering profiles where A is in a better position with respect to B (that is, $A > B$, $A = B$, or $A \bowtie B$), will give an equal or better situation for A in the result. Thus we have obtained an approximation of the combined result, that we call $cr^*(f, ip)$. We will now give a characterization of this approximation.

Theorem 18 *Given two outcomes A and B, $rel^*(A, B) \supseteq rel(A, B)$. Moreover, if $rel^*(A, B) = \{<, >, \bowtie, =\}$, then either $rel^*(A, B) = rel(A, B)$ or $rel^*(A, B) - rel(A, B) = \{\bowtie, =\}$.*

Example 28 Consider the Lex rule [ASS02], in which agents are ordered and, given any two outcomes A and B, the relation between A and B in the result is the relation given by the first agent in the order that does not declare a tie between A and B. Consider the incomplete profile $ip = (ip_1 = (A > C, B > C, A?B), ip_2 = (A > B > C))$. Then $rel^*(A, B) = \{<, =, \sim, >\}$, whereas $rel(A, B) = \{<, \sim, >\}$.

By following the procedure informally described above, this approximation can be computed polynomially, since we only need to consider two completions.

Theorem 19 *Given a preference aggregation function f which is IIA and monotonic, and an IPO profile ip, computing $cr^*(f, ip)$ is polynomial in the number of agents.*

5.5 Computing possible and necessary winners: tractable cases

We will now show how to determine the possible and necessary winners, given $cr^*(f, ip)$. Consider the arc between an outcome A and an outcome C in $cr^*(f, ip)$. Then, if this arc has the label $A < C$, then A is not a necessary winner, since there is an outcome C which is better than A in some result. If this arc *only* has the label $A < C$, then A is not a possible winner since we must have $A < C$ in all results. Moreover, consider all the arcs between A and every other outcome C. Then, if no such arc has label $A < C$, then A is a necessary winner. Notice, however, that in general, even if none of the arcs connecting A have just a single label $A < C$, A could not be possible winner. A could be better than some outcomes in every completion, but there might be no completion where it is better than all of them. We will show that this is not the case if f is IIA and monotonic.

We now define Algorithm 6, which, given $cr^*(f, ip)$, computes NW and PW, in polynomial time.

Algorithm 6: Computing NW and PW

Input: $cr^*(f, ip)$, where f: IIA and monotonic preference aggregation function, ip: IPO profile;
Output: P, N: sets of outcomes;
$P \leftarrow \Omega$;
$N \leftarrow \Omega$;
foreach $A \in \Omega$ **do**
  if $\exists\, C \in \Omega$ *such that* $\{<\} \subseteq rel^*(A, C)$ **then**
    $N \leftarrow N - A$;
  if $\exists\, C \in \Omega$ *such that* $\{<\} = rel^*(A, C)$ **then**
    $P \leftarrow P - A$;
return P, N;

Theorem 20 *Given $cr^*(f, ip)$, Algorithm 6 terminates in $O(m^2)$ time, where $m = |\Omega|$, returning $N = NW$ and $P = PW$.*

Proof: Algorithm 6 considers, in the worst case, each arc exactly once, thus we have $O(m^2)$.

N=NW. By construction of $cr^*(f, ip)$, $<\notin rel^*(A, C)$ if and only if $<\notin rel(A, C)$. By Algorithm 6, $A \in N$ if and only if $\forall C, <\notin rel\{A, C\}$, and this implies that there is no result in which there exists an outcome C that beats A. Thus, $A \in NW$. On the contrary, $A \in NW$ if and only if $A \not< C, \forall C \in \Omega$, for all results, from which, $A \in N$.

$P = PW$. An outcome A is in PW if there is no other outcome which beats it in all results. Thus, there cannot exist any other outcome C such that $<$ is the only label in $rel\{A, C\}$ and, thus by construction, also in $rel^*\{A, C\}$. Thus, $PW \subseteq P$. To show the other inclusion we consider $A \in P$ and we construct a completion of ip such that A wins in its result. First, let us point out that for any outcome A, $A \in P$ if and only if $\nexists C \in \Omega$, $rel^*(A, C) = \{<\}$. If $\forall C \in \Omega$, $<\notin rel^*(A, C)$, then A is never beaten by any other outcome C and A is NW and, thus, a PW. Secondly, let us consider the case in which A is such that whenever $<\in rel^*(A, C)$, either $\bowtie$ or $>$ (or both) are also in $rel^*(A, C)$ and let us denote with X such set of outcomes. Then for every outcome in $C \in X$ we choose $>$ whenever available and $\bowtie$ otherwise. This corresponds to replacing $A?C$ with $A > C$ in the incomplete profile. Such a choice on AC arcs cannot cause a transitivity inconsistency and thus can be completed to a result in which A is a winner. Finally, let us consider the case in which there is at least a C such that $rel^*(A, C) = \{<, =\}$. If for every other outcome C', $rel^*(A, C')$ contains exactly one label from the set: $\{>, \bowtie, =\}$ then we can safely set AC to $=$ since there is, for sure, a result with that labeling. Moreover, in such a result A is a winner. Assume, instead, that there is at least an outcome C' such that $|rel^*(A, C') > 1|$. This means that there is at least an agent which has not declared his preference on AC' and that such preference cannot be induced by transitivity closure. We replace $A?C'$ with $A > C'$ everywhere in the profile, we perform the transitive closure of all the modified $IPOs$, and we apply f. We will prove that such transitive closure does not force label $<$ on AC. After the procedure, due to monotonicity, $rel(A, C')$ will contain exactly one label from the set: $\{>, \bowtie, =\}$. Let us assume that, after the procedure, $A = C'$ and let us now consider $rel(C', C)$. Had it been $rel^*(C', C) = \{<\}$ from the start, this would have forced $rel(A, C) = \{<\}$. However, this is not possible since $A \in P$. This allows us to conclude that $(rel^*(C', C) \cap \{>, \bowtie, =\}) \neq \emptyset$ and any of such additional labels together with $A = C'$ can never force $A < C$. Clearly, if $A > C'$ or $A \bowtie C'$, there is no labeling of $C'C$ which can force $A < C$. It should be noticed that any available choice on $C'C$ can always be made safely due to the fact that the function is IIA and that the transitive closure of the profiles has already ruled out inconsistent choices. By iterating the procedure until every ? in the incomplete profile is replaced, we can construct a result of the function in which A is a winner. $\square$

Pareto rule and Lex rule, described in Section 4.2.2, are examples of preference aggregation functions which are both IIA and monotonic. Another example is the approval voting rule, in which each voter can vote for as many or as few candidates as the voter chooses and the winner is the candidate with the highest number of votes. For this last rule a tractability result for computing NW and PW is given in [KL05] since it is a positional scoring rule.

5.6 Preference elicitation

One use of necessary and possible winners is in eliciting preferences [CP04]. Preference elicitation is the process of asking queries to agents in order to determine their preferences over outcomes. At each stage in eliciting agents' preferences, there is a set of possible and necessary winners. When $NW = PW$, preference elicitation can be stopped since we have enough information to declare the winners, no matter how the remaining incompleteness is resolved [CS02b]. At the beginning, NW is empty and PW contains all outcomes. As preferences are declared, NW grows and PW shrinks. At each step, an outcome in PW can either pass to NW or become a loser. When PW is larger than NW, we can use these two sets to guide preference elicitation and avoid useless work.

If the preference aggregation function is IIA, then to determine if an outcome $A \in PW - NW$ is a loser or a necessary winner, it is enough to ask agents to declare their preferences over all pairs involving A and another outcome, say B, in PW. Moreover, IIA allows us to consider just one profile when computing the relations between A and B in the result, and guarantees that the result is a precise relation, that is, either $<$, or $>$, or $=$, or $\bowtie$. Thus we need to know all possible relations $A?B$ for $A \in PW - NW$ and $B \in PW - \{A\}$. In the worst case, we need to consider all such pairs. This bound is tight as there are examples where we may not be sure till all the relations are given. Algorithm 7, in $O(|PW|^2)$ steps eliminates enough incompleteness to determine the winners. At each step, the algorithm asks each agent to express their preferences on a pair of outcomes (via procedure $ask(A, B)$) and aggregates such preferences via function f. If function f is polynomially computable, the whole computation is polynomial in the number of agents and outcomes.

Theorem 21 *If f is IIA and polynomially computable, then determining the set of winners via preference elicitation is polynomial in the number of agents and outcomes.*

Using the results of the previous sections, under certain conditions we know how to compute efficiently the necessary winners and the possible winners. Thus Algorithm 7 can be given as input the outputs of Algorithm 6.

Algorithm 7: Winner determination

```
Input: PW, NW: sets of outcomes; f: preference aggregation function;
Output: W: set of outcomes;
wins: bool;
P ← PW; N ← NW;
while P ≠ N do
    choose A ∈ P − N;
    wins ← true; P_A ← P − {A};
    repeat
        choose B ∈ P_A;
        if ∃ an agent such that A?B then
            ask(A,B);
            compute f(A,B);
        if f(A, B) = (A > B) then
            P ← P − {B};
        if f(A, B) = (A < B) then
            P ← P − {A}; wins ← false;
        P_A ← P_A − {B};
    until f(A, B) = (A < B) or P_A = ∅ ;
    if wins = true then
        N ← N ∪ {A};
W ← N;
return W;
```

It should be noticed that deciding when elicitation is over, that is checking if $P = N$, is hard in general since, in [CS02b] such a result has been proved for STV.

5.7 Winner determination in sequential majority voting

In this section we consider a specific preference aggregation function, the sequential majority voting rule, which performs a sequence of pairwise comparisons between two candidates, where the result of each is computed by a majority vote. The winner thus depends on the chosen sequence of comparisons, which can be represented by a binary tree. Hence, there are candidates that will win in some trees (called possible winners) or in all trees (called Condorcet winners). We study the computational complexity of determining such winners in the situations where we have complete and incomplete information about voters' preferences.

While it is easy to find the possible and Condorcet winners for such a rule, we prove that it is difficult if we insist that the tree is balanced. This restriction is therefore enough to make voting difficult for the chair to manipulate. We then consider the situation where we lack complete information about preferences, and we determine the computational complexity of computing possible and Condorcet winners in this extended case.

5.7.1 Basic notions

We now give some basic notions on sequential majority voting rules. In particular, we start by defining preferences in such a scenario and the compact way for representing the result of all the majority comparisons between pairs of outcomes. We define the binary voting trees that we can use for inducing sequential majority voting rules and we define candidates that win in some trees (possible winners) or in all trees (Condorcet winners). In this scenario we will use sometimes the terms voters and candidates instead of agents and outcomes.

Incomplete preferences and profiles

We assume that each agent's preferences are specified by a (possibly incomplete) strict total order (that is, by an asymmetric, irreflexive and transitive order) over a set of candidates Ω, then given two candidates, $A, B \in \Omega$, an agent will specify exactly one of the following: $A < B$, $A > B$, or $A?B$, where $A?B$ means that the relation between A and B has not yet been revealed.

Definition 58 (ITO) An *incomplete total order* without ties is a total order without ties such that the relation between any pair of outcomes A and B can be $A < B$, $A > B$ or $A?B$, where $A?B$ means that the relation between A and B is unknown and it can be $>$ or $<$.

If we consider a sequence of total orders without ties, one for every agent, where at least one of total order is incomplete, then we obtain an incomplete TO profile.

Definition 59 (ITO profile) An *incomplete TO profile* ip is a sequence of total orders $ip_1, \ldots,$ ip_n over a set of outcomes, such that $\exists i \in \{1, \ldots, n\}$, ip_i is incomplete.

Majority graph

We start by defining the majority graph, which is a compact representation of voters's preferences for several voting rules, such as sequential majority voting rule.

Definition 60 (majority graph) Given a TO profile P, *the majority graph* $M(P)$ induced by P is the graph whose set of vertices is the set of the candidates Ω and in which for all $A, B \in \Omega$, there is a directed edge from A to B in $M(P)$ (denoted by $A >_m B$) if and only if a strict majority of voters prefers A to B.

The majority graph is asymmetric and irreflexive, but it is not necessarily transitive. *For the sake of simplicity we assume that the number n of voters is odd.* Then, $M(P)$ is complete: for each A, $B \neq A$, either $A >_m B$ or $B >_m A$ holds. Therefore, $M(P)$ is a complete, irreflexive and asymmetric graph, also called a *tournament* on Ω [Las97].

If we associate weights to the edges of the majority graph, we obtain weighted majority graphs, which are widely used in social choice theory. Weights measure the amount of disagreement (e.g. the number of voters preferring A to B). When we want to use standard majority graphs, we just consider weights to be identical, and we call them just majority graphs.

Definition 61 (weighted majority graph) The *weighted majority graph* associated with a TO profile P is the graph $M_W(P)$ whose set of vertices is Ω and in which for all $A, B \in \Omega$, there is a directed edge from A to B weighted by the number of voters who prefer A to B in P.

Majority graphs could also be incomplete. This happens when some arc between two candidates is missing.

Definition 62 (incomplete majority graph) Given an ITO profile P, the *incomplete majority graph* $M(P)$ induced by P is the graph whose set of vertices is Ω and containing an edge from A to B if and only if the number of voters who prefer A to B is greater than $n/2$. $M(P)$ is called a *partial tournament* over Ω.

The set of all (complete) tournaments extending $M(P)$ corresponds to a superset of the set of majority graphs induced by all possible completions of P.

Example 29 The majority graph induced by the 3-voter profile $((A > B > C), (B > C > A), (B > A > C))$ has the three edges $B >_m A$, $B >_m C$, and $A >_m C$, and so it is complete, while the graph induced by the 3-voter profile $((A > B > C), (A > C), (A > B, C))$ is incomplete because it has only the two edges $A >_m B$ and $A >_m C$.

Binary voting trees

Suitable structures for representing sequential majority voting rules are binary voting trees, that can be balanced or unbalanced.

Definition 63 (binary voting tree) [Mou88] Given a set of candidates Ω, a *binary voting tree* T is a binary tree where each internal node (including the root) has two children, each node is labelled by a candidate (element of Ω), and the leaves contain all candidates in Ω (one in each leaf). Given an internal node x and its two children x_1 and x_2, the candidate associated to x is the winner of the competition between the candidates associated to x_1 and x_2.

Definition 64 (balanced voting tree) A binary voting tree T is *balanced* iff the difference between the maximum and the minimum depth among the leaves is less than or equal to 1. In general, such a difference denotes the level of imbalance of the tree.

Example 30 Figure 5.2 gives two examples of balanced voting trees. In this figure, W returns the winner between two candidates. Figure 5.2 (a) shows a binary tree with $n = 2^2 = 4$ leaves where every leave is at depth $log_2(4) = 2$. In the part (b) there is a tree with $n = 5$ leaves which is balanced since the distance between the maximum and the minimum depth of the leaves is $3 - 2 = 1$. □

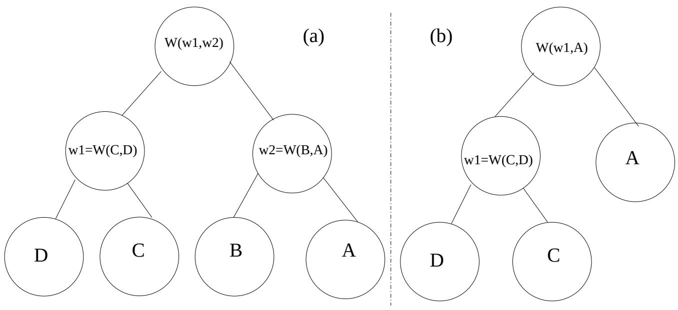

Figure 5.2: Balanced voting trees.

Sequential majority voting rules induced by voting trees

We now show how to induce sequential majority voting rules from voting trees.

Given a binary voting tree T, the voting rule r_T induced by T maps each tournament G to the candidate returned from the following procedure (called a *knock-out competition*):

1. Pick a nonterminal node x in T whose successors p, q are terminal nodes; let P and Q be the candidates associated to p and q respectively.

2. Delete the branches $x \to p$ and $x \to q$ from T, thus making x a terminal node with candidate P (respectively Q) if $P >_m Q$ (respectively $Q >_m P$) is in G.

3. Repeat this operation until T consists of a single node. Then the candidate associated to this node is returned.

Example 31 Given the majority graph in Figure 5.3 (a), Figure 5.3 (b) shows the voting tree corresponding to the sequence of competitions $((A, C), (B, A))$ where the winner is the outcome B. □

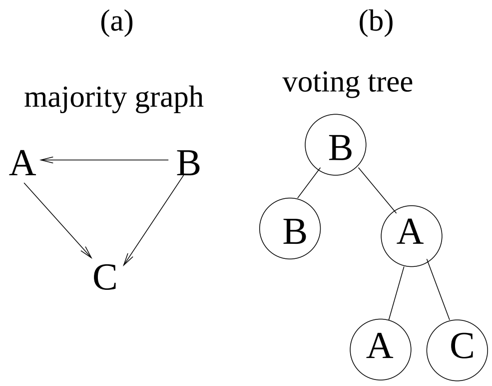

Figure 5.3: Majority graph and a resulting voting tree.

Possible and Condorcet winners for sequential majority voting rules

Sequential majority voting rules are incompletely specified, since the winner depends on the chosen binary voting tree. We can define Condorcet winners, which are those candidates that win in every binary voting tree and possible winners, which are those candidates that win in at least one of them.

Definition 65 (Condorcet winner) A candidate A is a *Condorcet winner* if and only if, for any other candidate B, we have $A >_m B$.

Thus, a Condorcet winner corresponds to a vertex of the majority graph with outgoing edges only. There are profiles for which no Condorcet winner exists; however, when a Condorcet winner exists, it is unique. If there is a Condorcet winner, then it is the sequential majority winner for each binary voting tree.

Definition 66 (possible winner) A candidate A is a *possible winner* if and only if there exists at least one binary voting tree for which A is the winner.

5.7.2 Computing possible winners

The set of possible winners coincides with the *top cycle* of the majority graph [Mou88]. The top cycle of a majority graph G is the set of maximal elements of the reflexive and transitive closure G^* of G. An equivalent characterization of possible winners is in terms of paths in the majority graph.

Theorem 22 (see e.g. [Mou88, Las97]) *Given a complete majority graph G, a candidate A is a possible winner iff for every other candidate C, there exists a path from A to C in G.*

This gives us a polynomial method to compute possible winners.

Corollary 3 *Given a complete majority graph and a candidate A, checking whether A is a possible winner and, if so, finding a tree where A wins, is polynomial .*

Proof: Since path finding is polynomial [Bel58], we can check in polynomial time whether A is a possible winner. For these paths, we can construct in polynomial time a tree in which A wins. □

Example 32 Assume that, given a majority graph G over candidates A, B_2, B_3, and C, candidate A is a possible winner and that only C beats A. Then, for Theorem 22, there must be a path in G from A to every other candidate. Assume $A \rightarrow B_2 \rightarrow B_3 \rightarrow C$ is a path from A to C in G. Figure 5.4 shows the voting tree corresponding to such a path. □

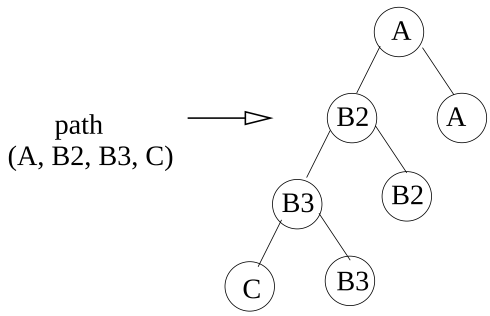

Figure 5.4: A path and the corresponding voting tree.

5.7.3 Fair possible winners

Possible winners are candidates who win in at least one voting tree. However, such a tree may be very unbalanced, thus representing a sequence of competitions where the winner may compete with few other candidates. This may be considered unfair. In the following, we will consider a competition *fair* if it has a balanced voting tree, and we will call such winners *fair possible winners*.

Definition 67 (fair possible winner) A candidate A is a *fair possible winner* if and only if there is a balanced voting tree where A wins.

For simplicity, we will assume that are 2^k candidates but results can easily be lifted to situations where the number of candidates is not a power of 2. Notice that a Condorcet winner is a fair possible winner, since it wins in all trees, thus also in balanced ones.

We will show that a candidate is a fair possible winner when the nodes of the majority graph can be covered by a binomial tree [CLR90], i.e., the nodes of the majority graph are the terminal nodes of a balanced voting tree.

Definition 68 (binomial tree) A binomial tree is defined inductively as follows.

- A binomial tree of order 0, written T_0, is a tree with only one node.
- A binomial tree of order k, with $k > 0$, written T_k, is the tree where the root has k children, and for $i = 1, \dots, k$ the i-th child is a binomial tree of order $k - i$.

It is easy to see that, in a binomial tree of order k, there are 2^k nodes and the tree has height k.

Given a majority graph with 2^k nodes, and given a candidate A, it is possible to find a covering of the nodes which is a binomial tree of order k with root A. In this situation, we have a balanced voting tree where A wins. Thus A is a fair possible winner.

Example 33 Consider the majority graph G over candidates A, B, C and D depicted in Figure 5.5. Since such a majority graph is covered by the binomial tree T_2 with root A, we can conclude that A is a fair possible winner. □

Theorem 23 *Given a complete majority graph G with 2^k nodes, and a candidate A, A is a fair possible winner if and only if there is a binomial tree T_k covering all nodes of G with root A.*

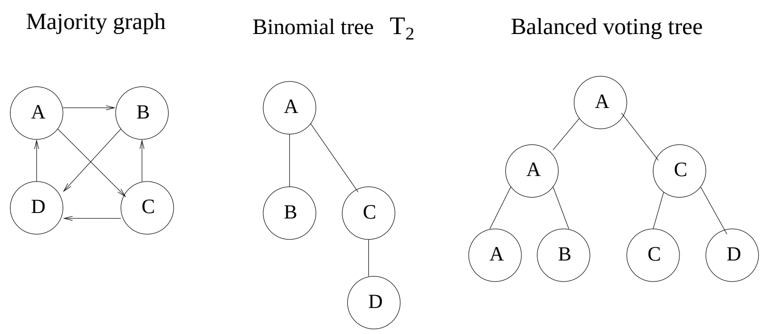

Figure 5.5: From a majority graph to a balanced voting tree via a binomial tree.

Proof: Assume there is a binomial tree T_k satisfying the statement of the theorem. Then each node of T_k is associated to a candidate. For node n, we will write $cand(n)$ to denote such a candidate. We will show that it is possible to define, starting from T_k, a balanced voting tree $b(T_k)$ where A wins. Each node of $b(T_k)$ is associated to a candidate as well, with the same notation as above. The definition of $b(T_k)$ is given by induction:

- If $k = 0$, $b(T_0) = T_0$;
- If $k > 0$, $b(T_k)$ is the balanced tree built from two instances of $b(T_{k-1})$, corresponding to the two instances of T_{k-1} which are part of T_k by definition; the roots of such trees are the children of the root of $b(T_k)$; the candidate of the root of $b(T_k)$ is the candidate of the root of T_k.

It is easy to see that $b(T_k)$ is a balanced binary voting tree, where the winner is A if A is the candidate associated to the root of the binomial tree.

Proof of the opposite direction is similar. □

Notice that the set of possible winners contains the set of fair possible winners, which in turn contains the set of Condorcet winners. However, while there could be no Condorcet winner, there is always at least one fair possible winner, since we can always build a balanced tree where an outcome wins. Thus, a voting rule accepting only fair possible winners is well-defined.

Unfortunately, the complexity of deciding if A is a fair possible winner is an open problem. However, for *weighted* majority graphs, it is difficult to check whether a candidate A is a fair possible winner. This means that, if we restrict the voting trees only to balanced ones, it is difficult for the chair (if they can choose the voting tree) to manipulate the election. The problem of chair manipulation (also called “control”) has been considered first in [BTT95].

Theorem 24 *Given a complete oriented and weighted majority graph G and a candidate A, it is $\mathcal{NP}$-complete to check whether A is a fair possible winner for G.*

Proof: (Sketch) To prove the statement of the theorem, we prove that the Exact Cover problem reduces polynomially to the problem of finding a minimum binomial tree with root A covering G. The proof is similar to that used in [PY82] to show that Exact Cover reduces polynomially to the problem of finding a minimum spanning tree for a graph from a class of trees which are sets of disjoint flowers of type 2 and where, for each tree t in the class, the number of flowers of type 2 is $d \geq c|t|^{\epsilon}$.

A flower is a tree where all nodes but the root have degree at most 2 and are at distance 1 or 2 from the root. A flower is of type 2 if at least a node has distance 2 from the root.

Binomial trees are trees which consist of disjoint flowers of type 2. In fact, the binomial tree T_2 is a flower of type 2, and all bigger binomial trees T_k, with 2^k nodes, consist of 2^{k-2} instances of such flowers of type 2, all disjoint[1]. Thus we have $d = 2^{k-2} = \frac{|T_k|}{2} \geq c|T_k|^{\epsilon}$.

Given an instance of the Exact Cover Problem, consisting of $3k$ sets of size 3 and $3k$ elements, each of which appears three times in the sets, the proof proceeds by constructing a graph such that the instance has an exact cover if and only if the graph has a minimum covering binomial tree. $\square$

On the other hand, given a possible winner A, it is easy to find a tree, with a *bounded* level of imbalance in which A wins. We define $D(T, A)$ as the length of the path in the tree T from the root labelled A to the only leaf labelled A. We define $\Delta(A)$ as the maximum $D(T, A)$ over all voting trees T where A wins. If $\Delta(A) = m - 1$, where $m = |\Omega|$, then A is a Condorcet winner, and vice versa. In fact, this means that there is a tree where A competes against everybody else, and wins. This can be seen as an alternative characterization of Condorcet winners. Moreover, we will now show that, if A is a possible winner then there exists a voting tree where A wins with level of imbalance at most $m - \Delta(A) - 1$. Notice that there could also be more balanced trees in which A wins.

Theorem 25 *Given a complete majority graph G and a possible winner A, there is a voting tree with level of imbalance smaller than or equal to $m - \Delta(A) - 1$, where A wins. This tree can be built in polynomial time.*

Proof: Since A beats $\Delta(A)$ candidates, we can easily build a balanced voting tree BT involving only A and the candidates beaten by A and this tree will have level of imbalance

[1] We thank Claude Guy-Quimper for this observation and for suggesting we look at [PY82].

equal to 0. Then we have to add the remaining $m - \Delta(A) - 1$ candidates to BT. Since A is a possible winner, there must be a path from one of the candidates beaten by A to each of the remaining candidates. In the worst case, this adds a subtree of depth $m - \Delta(A) - 1$ rooted at one of the nodes beaten by A in BT.

The procedure illustrated above is clearly polynomial. In fact BT can be built in linear time in $\Delta(A)$ since A beats every candidate of BT and thus the order of the competitions among the other candidates in BT can be set in any way respecting the balance constraint. □

If unfair tournaments are undesirable, we can consider those possible winners for which there are voting trees which are as balanced as possible. Theorem 25 helps us in this respect: if A is a possible winner, knowing $\Delta(A)$, we can compute an upper bound to the minimum imbalance of a tree where A wins. In general, if $\Delta(A) \geq k$, then there is a tree with imbalance level smaller or equal than $m - k - 1$, hence the higher $\Delta(A)$ is, the lower is the upper bound to the level of imbalance of a tree where A wins.

It is thus important to be able to compute $\Delta(A)$. This is an easy task. In fact, once we know that a candidate A is a possible winner, $\Delta(A)$ coincides with the number of outgoing edges from A in the majority graph.

Theorem 26 *Given a majority graph G and a possible winner A, $\Delta(A)$ is the number of outgoing edges from A in G.*

Proof: If A has k outgoing edges, no voting tree where A wins can have A appearing at depth larger than k. In fact, to win, A must win in all competitions scheduled by the tree, so such competitions must be at most k. Thus $\Delta(A) \leq k$.

Moreover, it is possible to build a voting tree where A wins and appears at depth k. Let us first consider the linear tree, T_1, in which A competes against all and only the $D_1, \ldots, D_k$ candidates which it defeats directly in G. Clearly in such tree, T_1, $D(T_1, A) = k$. However T_1 may not contain all candidates. In particular it will not contain candidates defeating A in G. We will now consider, one after the other, each candidate C such that $C \rightarrow A$ in G. For each such candidate we will add a subtree to the current tree. The current tree at the beginning is T_1. Let us consider the first C and the path, which we know exists, which connects A to C, say $A \rightarrow B_1 \rightarrow B_2 \ldots \rightarrow B_h \rightarrow C$. Let $j \in \{1, \ldots, h\}$ be such that B_j belongs to the current tree and $\forall i > j$, B_i does not belong to the current tree. Notice that such candidate B_j always exists, since any path from A must start with an edge to one of the $D_1, \ldots, D_k$ candidates. We then attach to the current tree the subtree corresponding to the path $B_j \rightarrow \ldots \rightarrow C$ at node B_j obtaining a new tree in which only new candidates have been added. After having considered all candidates defeating C in A, the tree obtained is a voting tree, in which A wins and has depth exactly k. Thus $\Delta(A) = k$. $\square$

If a candidate is a possible winner with the maximum number of outgoing edges in the majority graph (i.e., it is a *Copeland winner*), we can give a smaller upper bound on the amount of imbalance in the fairest/most balanced tree in which it wins.

Theorem 27 *If a candidate A is a Copeland winner, then the imbalance of a fairest/most balanced tree in which A wins is smaller or equal than $log(m - \Delta(A) - 1)$.*

Proof: If A is Copeland winner then every candidate beating A must be beaten by at least one candidate beaten by A. Consider the balanced tree BT defined in the proof of Theorem 25 rooted at A and involving all and only the candidates beaten by A. In the worst case all the remaining $m - \Delta(A) - 1$ candidates are beaten only by the same candidate in BT, say B'. In such a case, however, we can add to BT a balanced subtree with depth $log(m - \Delta(A) - 1)$ rooted at B', involving all the remaining candidates. $\square$

5.7.4 Incomplete preferences

Up till now, voters have defined all their preferences over candidates, thus the majority graph is complete; uncertainty comes only from the tree, i.e., from the voting rule itself. Another source of uncertainty is that the voters' preferences may only be partially known. In this case, the profiles and so the majority graph can be incomplete. We would like to reason about the winners in such two scenarios. In this section we show how to extend the notions of possible and Condorcet winners for incomplete majority graphs and for incomplete profiles.

Incomplete majority graphs

Assume the incomplete information about preferences is given by an incomplete majority graph (IMG). We can extend the notions of possible and Condorcet winners in such a scenario as follows.

Definition 69 (weak/strong possible/Condorcet winners for IMGs) Let G be an incomplete majority graph and A a candidate.

- A is a *weak possible winner* for G if and only if there exists a completion of G and a tree for which A wins.

- A is a *strong possible winner* for G if and only if for every completion of G there is a tree for which A wins.

- A is a *weak Condorcet winner* for G if and only if there is a completion of G for which A is a Condorcet winner.

- A is a *strong Condorcet winner* for G if and only if for every completion of G, A is a Condorcet winner.

When the majority graph is complete, strong and weak Condorcet winners coincide, and coincide also with the notion of Condorcet winners given before.

We denote by $WP(G)$, $SP(G)$, $WC(G)$ and $SC(G)$ the sets of, respectively, weak possible winners, strong possible winners, weak Condorcet winners and strong Condorcet winners for G. We have the following inclusions:

$$SC(G) \subseteq WC(G) \cap SP(G)$$
$$WC(G) \cup SP(G) \subseteq WP(G)$$

We now give a characterization for each of the four notions above.

Theorem 28 *Given an incomplete majority graph G and a candidate A, A is a strong possible winner if and only if for every other candidate B, there is a path from A to B in G.*

Proof: ($\Leftarrow$) Suppose that for each $B \neq A$ there is a path from A to B in G. Then these paths remain in every completion of G. Therefore, using Theorem 22, A is a possible winner in every completion of G, i.e., it is a strong possible winner.
($\Rightarrow$) Suppose there is no path from A to B in G, $\exists B$. Let us define the following three subsets of the set of candidates Ω: $R(A)$ is the set of candidates reachable from A in G (including A); $R^{-1}(B)$ is the the set of candidates from which B is reachable in G (including B); and $Others = \Omega \setminus (R(A) \cup R^{-1}(B))$. Because there is no path from A to B in G, we have that $R(A) \cap R^{-1}(B) = \emptyset$ and therefore $\{R(A), R^{-1}(B), Others\}$ is a partition of Ω. Now, let us build the complete tournament $\hat{G}$ as follows:

1. $\hat{G} := G$;

2. $\forall\, x \in R(A), \forall\, y \in R^{-1}(B)$, add (y, x) to $\hat{G}$;

3. $\forall\, x \in R(A), \forall\, y \in Others$, add (y, x) to $\hat{G}$;

4. $\forall\, x \in Others, \forall\, y \in R^{-1}(B)$, add (y, x) to $\hat{G}$;

5. $\forall\, x, y$ belonging to the same element of the partition: if neither (x, y) nor (y, x) in G then add one of them (arbitrarily) in $\hat{G}$.

We will show that G is a complete tournament and that there is no path from A to B in $\hat{G}$. Let us first show that $\hat{G}$ is a complete tournament. If $x \in R(A)$ and $y \in R^{-1}(B)$, then $(x, y) \notin G$ (otherwise there would be a path from A to B in G). If $x \in R(A)$ and $y \in Others$, then $(x, y) \notin G$, otherwise y would be in $R(A)$. If $x \in Others$ and $y \in R^{-1}(B)$, then $(x, y) \notin G$, otherwise x would be in $R^{-1}(B)$. Therefore, whenever x and y belong to two distinct elements of the partition, $\hat{G}$ contains (y, x) and not (x, y). Now, if x and y belong to the same element of the partition, by Step 5, $\hat{G}$ contains exactly one edge among $\{(x, y), (y, x)\}$. Therefore, $\hat{G}$ is a complete tournament. Let us show now that there is no path from A to B in $\hat{G}$. Suppose there is one, that is, there exist $z_0 = A, z_1, \ldots, z_{m-1}, z_m = B$ such that $\{(z_0, z_1), (z_1, z_2), \ldots, (z_{m-1}, z_m)\} \subseteq \hat{G}$. Now, for all $x \in R(A)$ and all y such that $(x, y) \in \hat{G}$, by construction of $\hat{G}$ we necessarily have $y \in R(A)$. Therefore, for all $i < m$, if $z_i \in R(A)$ then $z_{i+1} \in R(A)$. Now, since $z_0 = A \in R(A)$, by induction we have $z_i \in R(A)$ for all i, thus $B \in R(A)$, which is impossible. Therefore, there is no path from A to B in $\hat{G}$. Thus, $\hat{G}$ is a complete tournament with no path from A to B, which implies that A is not a possible winner with respect to $\hat{G}$. Lastly, by construction, $\hat{G}$ contains G. So $\hat{G}$ is a complete extension of G for which A is not a possible winner. This shows that A is not a strong possible winner for G. $\square$

Clearly, a procedure based on the previous theorem gives us a polynomial algorithm to find strong possible winners.

Let G be an asymmetric incomplete graph, Ω the set of candidates, and $A \in \Omega$. Let us call $f(G, A)$ the set Σ returned by Algorithm 8 on G and A. Then we have the following result:

Theorem 29 $f(G, A) = \Omega$ *if and only if* A *is a weak possible winner for* G.

Proof: We first make the following observation: the graph G' obtained at the end of the algorithm is asymmetric and extends G. It is asymmetric because it is asymmetric at the start of the algorithm (since G is) and then, when an edge $Y \to Z$ is added to G' when $Z \to Y$ is not already in G'.

($\Rightarrow$) Now, assume $f(G, A) = \Omega$. Let G'' be a tournament extending G' (and, a fortiori, G). Such a G'' exists (because G' is asymmetric). By construction of G', there is a path in G' from A to every node of $f(G, A) \setminus \{A\}$, hence to every node of $\Omega \setminus \{A\}$; since G'' extends

Algorithm 8: Weak possible winners determination

Input: G: asymmetric incomplete graph; Ω: set of candidates; $A \in \Omega$;
Output: Σ: set of outcomes;
$\Sigma \leftarrow \{A\} \cup \{X|$ there is a path from A to X in $G\}$;
$G' \leftarrow G$;
repeat
  foreach $(Y, Z) \in \Sigma \times (\Omega \setminus \Sigma)$ **do**
    if $(Z \rightarrow Y) \notin G'$ **then**
      add $(Y \rightarrow Z)$ to G'
  foreach $Z \in \Omega \setminus \Sigma$ **do**
    if *there is a path from A to Z in G'* **then**
      add Z to Σ
until $\Sigma = \Omega$ *or there is no* $(Y, Z) \in \Sigma \times (\Omega \setminus \Sigma)$ *s. t.* $(Z \rightarrow Y) \in G'$;
return Σ.

G', this holds a fortiori for G'', hence A is a possible winner in G'' and therefore a weak possible winner for G.
($\Leftarrow$) Conversely, assume $f(G, A) = \Sigma \neq \Omega$. Denote $\Theta = \Omega \backslash \Sigma$. Then, for all $(Y, Z) \in \Sigma \times \Theta$ we have $Z \rightarrow Y \in G'$. Now, $Z \in \Theta$ means that no edge $Z \rightarrow Y$ (for $Z \in \Theta$ and $Y \in \Sigma$) was added to G'; hence, for every $Y \in \Sigma$ and $Z \in \Theta$, we have that $Z \rightarrow Y \in G'$ if and only if $Z \rightarrow Y \in G$. This implies that for all $(Y, Z) \in \Sigma \times \Theta$ we have $Z \rightarrow Y \in G$, therefore, in every tournament G'' extending G, every candidate of Θ beats every candidate of Σ, and in particular A. Therefore, there cannot be a path in G'' from A to a candidate in Z, which implies that A is not a possible winner in G''. Since the latter holds for every tournament G'' extending G, A is not a weak possible winner for G. □

Since the algorithm computing $f(G, A)$ runs in time $O(|\Omega|^2)$, we get, as a corollary, that weak possible winners can be computed in polynomial time.

Given an asymmetric graph G, Θ is said to be a *dominant subset* of G if and only if for every $Z \in \Theta$ and every $X \in \Omega \setminus \Theta$ we have $(Z, X) \in G$.

Then we have an alternative characterization of weak possible winners:

Theorem 30 *A is a weak possible winner with respect to G if and only if A belongs to all* *dominant subsets of G.*

Proof: Suppose there exists a dominant subset Θ of G such that $A \notin \Theta$. Then there can be no extension of G in which there is a path from A to a candidate $Z \in \Theta$. Hence A is not a weak possible winner for G. Conversely, suppose that A is not a weak possible winner for G. Then the algorithm for computing $f(G, A)$ stops with $f(G, A) \neq \Omega$ and $\Omega \setminus f(G, A)$ being a dominant subset of G. Since $A \in f(G, A)$, $\Omega \setminus f(G, A)$ is a dominant subset of G to which A does not belong. □

In the following we will characterize the weak/strong Condorcet winners and then we will use this characterization for showing that it is linear to compute the exact set of weak/strong Condorcet winners.

Theorem 31 *Given an incomplete majority graph G and a candidate A, A is the strong Condorcet winner if and only if A has $m - 1$ outgoing edges in G.*

Proof: Follows directly from the fact that, given a majority graph, A is a Condorcet winner if and only if A has only outgoing edges in G. □

Theorem 32 *Given an incomplete majority graph G and a candidate A, A is a weak Condorcet winner if and only if A has no ingoing edges in G.*

Given an incomplete majority graph, the set of weak/strong Condorcet winners can therefore be computed in polynomial time from the majority graph.

We end up this section by giving the bounds on the number of weak/strong possible/ Condorcet winners.

Theorem 33 *Let $|\Omega| = m$. The following inequalities hold, and for each of them the bounds are reached:* $0 \leq |SC(G)| \leq 1$; $0 \leq |WC(G)| \leq m$; $0 \leq |SP(G)| \leq m$; $1 \leq |WP(G)| \leq m$.

Incomplete profiles

Reasoning with incomplete majority graphs is useful for computing weak/strong possible/ Condorcet winners in polynomial time, however, it can lead to a loss of information. In fact, we can define as winners some outcomes that don't win in any completion of the incomplete profiles inducing such majority graphs, as shown in the following example.

Example 34 Assume to have a unique voter and three candidates A, B and C and that the voter orders only A and B saying $A > B$. Then the incomplete majority graph has only one arc between A and B. In this case B is a weak possible winner since there is a completion of the majority graph and a voting tree where B wins. In fact, if we consider the completion of the majority graph where B beats C and C beats A, then we can build a voting tree where A wins, by performing first the competition between A and C, where the winner is C, and then the competition between C and B and so the winner is B. But if we start from the incomplete profile, then there are no completions of this profile where B wins. Notice that the completion of the majority graph considered before cannot be a completion of the voter's preferences since it is not a partial order. □

Thus, in this section we assume that incompleteness is given by an incomplete profile (IP). In such a scenario we can extend notions of possible and Condorcet winners as follows.

Definition 70 (weak/strong possible/Condorcet winners for IPs) Let P be an incomplete profile and A a candidate.

- A is a *weak possible winner* for P if and only if there exists a completion of P and a tree for which A wins.
- A is a *strong possible winner* for P if and only if for every completion of P there is a tree for which A wins.
- A is a *weak Condorcet winner* for P if and only if there is a completion of P for which A is a Condorcet winner.
- A is a *strong Condorcet winner* for P if and only if for every completion of P A is a Condorcet winner.

We denote by $WP(P)$, $SP(P)$, $WC(P)$ and $SC(P)$ the sets of, respectively, weak possible winners, strong possible winners, weak Condorcet winners and strong Condorcet winners for the incomplete profile P.

When the profile is complete, strong and weak Condorcet winners coincide, and coincide also with the notion of Condorcet winners given before.

When, like in [KL05], the voting rule is fixed, i.e. when the voting tree is fixed, then weak possible winners coincide with weak Condorcet winners and strong possible winners coincide with strong Condorcet winners. Hence we have only two kinds of winners, that we call weak winners and strong winners.

Definition 71 (weak/strong winners for IPs) Let P be an incomplete profile, A a candidate and the voting rule fixed (i.e., the tree fixed).

- A is a *weak winner* for P if and only if there exists a completion of P for which A wins in the fixed tree.
- A is a *strong winner* for P if and only if for every completion of P A wins in the fixed tree.

We denote by $WW(P)$ and $SW(P)$ the sets of weak and strong winners for the incomplete profile P.

Incomplete majority graph with respect to incomplete profiles

We now compare the winners obtained considering an incomplete profile with those ones defined considering its induced incomplete majority graph.

Given an incomplete profile P and the incomplete majority graph G induced by P, then the set of the completions of G is a superset of the set of the majority graphs induced by all possible completions of P. The following inclusions hold.

Theorem 34 *Let P be an incomplete profile and let G be the incomplete majority graph induced by P. Then,*

- $WP(G) \supseteq WP(P)$*;*
- $SP(G) \subseteq SP(P)$*;*
- $WC(G) = WC(P)$*;*
- $SC(G) = SC(P)$.

Proof:

- $WP(G) \supseteq WP(P)$. In fact, if an outcome A belongs to $WP(P)$, then there is a completion P' of P such that A is a possible winner and so A will be a possible winner for the complete majority graph G' induced by P'. Since G' is one of all the possible completions of G, then $A \in WP(G)$. Notice that $WP(G) \not\subseteq WP(P)$, since an outcome could be a possible winner for a completion of G, which is not induced by any completion of P.

- $SP(G) \subseteq SP(P)$. In fact, if an outcome is a possible winner for every completion of G, then it will be a possible winner also for the majority graphs induced by the completion of P, since the set of all the majority graphs induced by completions of P are a subset of all the completions of G. Notice that $SP(G) \not\supseteq SP(P)$, since there could be an outcome that is a possible winner for every completion of P and so for every majority graph induced by completions of P, but not for the completions of G which are not induced by completions of P.

- $WC(G) = WC(P)$. In fact, as above, $WC(G) \supseteq WC(P)$. Moreover, $WC(G) \subseteq WC(P)$. In fact, if an outcome A belongs to $WC(G)$, there is a completion of G where A is a Condorcet winner. This complete majority graph is the majority graph induced by the completion of P, where, we replace every $A?C$ (where C is an outcome different from A) with $A > C$. Then $A \in WC(P)$.

- $SC(G) = SC(P)$. In fact, as above, $SC(G) \subseteq SC(P)$. Moreover, $SC(G) \supseteq SC(P)$. In fact, if an outcome belongs to $SC(P)$ then it is a Condorcet winner, i.e. it beats every other outcome, for every completion of P. Hence he must beat every other outcome in the certain part, hence in the incomplete majority graph G induced by P there are only outgoing edges from this outcome and so this outcome must belong to $SC(G)$.

□

Theorem 35 *Given a profile P, the sets $WC(P)$, $SC(P)$, $WW(P)$ and $SW(P)$ are easy to compute.*

Proof: Since $WC(G) = WC(P)$ and $SC(G) = SC(P)$ and since in Section 5.7.4 we have shown that, given an incomplete majority graph, it is easy to compute $WC(G)$ and $SC(G)$, then it is also easy to compute $WC(P)$ and $SC(P)$ for every profile P.

If the voting rule is fixed, then computing the weak winners and strong winners from an incomplete profile is easy. In fact, given an incomplete profile P, since $WW(P) = WP(P) = WC(P)$, $SW(P) = SP(P) = SC(P)$ and since we have shown that $WC(P)$ and $SC(P)$ are easy to compute, then also $WW(P)$ and $SW(P)$ are easy to compute. □

Given an incomplete profile P, we think that is difficult to find $WP(P)$ and $SP(P)$, but we don't have a proof yet. If this conjecture is confirmed, then in order to have approximations of these sets, it is useful to compute weak/strong possible winners for incomplete majority graphs.

5.7.5 Fair possible winners with incomplete preferences

In this section we extend the notion of fair possible winner to the case of incomplete preferences. We consider incomplete profiles and not incomplete majority graphs, since we have shown previously in this section that is more reasonable, because considering incomplete majority graphs can lead to a loss of information.

Definition 72 (fair weak/strong possible winners) Let P be an incomplete profile,

- A is a *fair weak possible winner* for P if and only if there exists a completion of the profile and a balanced tree for which A wins;
- A is a *fair strong possible winner* for P if and only if for every completion of the profile there is a balanced tree for which A wins.

In the case of complete preferences, we have shown that for *weighted* majority graphs it is difficult to check whether a candidate A is a fair possible winner. We will now show that, if we assume that $\mathcal{P} \neq \mathcal{NP}$, the same result holds also if we consider incompleteness. Hence if we restrict the voting trees only to balanced ones, it is difficult for the chair to manipulate the election.

Theorem 36 *Given an incomplete profile P and a candidate A, if $\mathcal{P} \neq \mathcal{NP}$, it is $\mathcal{NP}$-complete to check whether A is a fair weak possible winner for P and also to check whether A is a fair strong possible winner for P.*

Proof: The proof is given by contradiction and considers the complexity of the problem of checking if an outcome is a fair weak possible winner. A similar proof can be used for fair strong possible winners.

Assume that the problem of checking whether an outcome is a fair weak possible winner for P is in $\mathcal{P}$. Then there must be a polynomial algorithm that takes in input an outcome and an incomplete profile, and that says if this outcome is a fair weak possible winner. If we give in input to this algorithm a complete profile and an outcome, then in polynomial time the algorithm says if this outcome is a fair possible winner. Thus determining fair possible winners is in $\mathcal{P}$. However, Theorem 24 shows that this problem is in $\mathcal{NP}$. $\square$

5.8 Related work

In [KL05] preference aggregation functions for combining incomplete total orders are considered. In this setting, it is shown that determining the possible winners is in $\mathcal{NP}$ and the necessary winners is in $co\mathcal{NP}$. However, it is tractable to compute possible and necessary winners for positional scoring voting procedures like the Borda and plurality procedures, as well as for a non-positional procedure like Condorcet. Compared to our work, we permit both incompleteness and incomparability, while [KL05] allows only for incompleteness. We recall that incomparability is an important aspect of preferences, especially when agents have multiple criteria to optimize. Second, [KL05] considers social choice functions which return the (non-empty) set of winners. Instead, we consider social welfare functions which return a partial order. Social welfare functions give a finer grained view of the result. Third, [KL05] considers specific voting rules like the Borda procedure whilst we give, in the first part of this chapter, general properties that ensure tractability.

The general properties found in this chapter could be useful, not just for combining preferences from multiple agents, but also for combining multiple conflicting preferences from a single agent. Recent work addressing the combination of multiple complex preferences is presented in [Cho04] and [Kie05].

In the second part of this chapter we focused on a specific preference aggregation function, the sequential majority voting. Sequential majority voting rule can be represented by a voting tree and the winner depends on the chosen tree. We dealt with uncertainty about the choice of the binary tree. Because the choice of the tree is under the control of the chairman, our results can be interpreted in terms of difficulty of manipulation by the chairman. This issue has been considered first in [BTT95] that has pointed out that, even doing the non realistic assumption that the chairman knows the preferences of every voter and knows that they will vote sincerely, some voting schemes, which are in principle susceptible to control (i.e., chair manipulation), are resistant in practice due to excessive computational costs, whereas other voting schemes are vulnerable to control. For example, it may be possible to influence the result of an election by specifying the sequence in which alternatives will be considered (which is our case), or by specifying the composition of subcommittees that nominate candidates. With regard to chair manipulation, we proved that sequential majority voting is easy to manipulate, except if we require that the binary tree used for defining the winner is balanced. Having found a case where sequential majority voting is difficult to manipulate by the chairman is a relevant result since this voting scheme had always been considered computationally vulnerable to control (see, for example [Ban95, HM66, Mil80]). In fact, the chairman can determine a sequence resulting in victory for his preferred candidate, or conclude that none exists, within a polynomial number of computational steps.

In [CS02a] there is another result regarding chair manipulation for a voting protocol, called "Cup rule", which is similar to our sequential voting where the tree must be balanced. In the Cup protocol there is a balanced binary tree with one leaf per candidate, each non-leaf node is assigned the candidate that is winner of the pairwise election of the node's children and the candidate assigned to the root wins. [CS02a] shows that this protocol, that requires a schedule to be instantiated, is easy to manipulate if the schedule is known in advance and for making manipulation difficult it is sufficient, even if there are few candidates, to randomize over these schedules.

We also dealt with uncertainty about voters' preferences. We showed that in this case, it is easy to compute a lower and an upper bound of the set of candidates winning for some binary tree. These results apply to manipulation by coalitions of voters [CS02a] and elicitation by the chair [CS02b] as these are two situations where we have to reason with incomplete preferences.

5.9 Future work

Possible and necessary winners are only the first step. In our future work, we will consider the probability that a candidate is a winner assuming some probability distribution over the possible completions. In addition, we plan to consider notions other than the winner (e.g. possible rankings, possible dominances, ...).

We intend also to consider the addition of constraints to agents' preferences. This means that preference aggregation must take into account the feasibility of the outcomes. Thus possible and necessary winners must now be feasible. Consider for example the problem of configuring a family car. We have various product constraints. We also have preferences of the multiple agents who will use the car ("I prefer four doors to two doors", "You prefer soft top to hard top", . . .). We may therefore want to reason about a constrained preferentially optimization problem in which we have incompleteness and incomparability in the preferences. We can define the *constrained necessary winners* as those feasible outcomes which are not dominated by any other feasible outcome in all possible completions, and the *constrained possible winners* as those feasible outcomes which are not dominated by any other feasible outcome in at least one possible completion. Note that the necessary and possible winners may not themselves be feasible. We may therefore need to look lower in the result than just the winners.

It is also important to consider compact knowledge representation formalisms to express agents' preferences, such as CP-nets and soft constraints. Possible and necessary winners should then be defined directly from such compact representations, and preference elicitation should concern statements allowed in the representation language.

Some of our results rely on IIA assumption, which is a strong assumption. However, we use it just to show intractability is not inevitable on incomplete partial orders. Nevertheless, it is important to show that these intractability results do not always hold but that there are cases (e.g. monotonic and IIA rules on partial orders [as shown here], and positional scoring rules on total orders [as shown in [KL05]]) where computing possible and necessary winners is tractable. In the cases where computing possible and necessary winners is tractable, our novel preference elicitation algorithm (which focuses the questions on just PW-NW) is useful. We plan however to relax the IIA assumption in the future tractability results.

For the scenario where agents express their preferences over a set of alternatives and their preferences are aggregated by a sequential majority rule, we characterized the class of fair possible winners; that is, possible winners that win in a balanced voting tree and proved that, for weighted majority graphs, it is difficult to compute fair possible winners. Balance is therefore enough to make voting difficult for the chair to manipulate. We intend to investigate if the complexity of computing fair possible winners with unweighted majority graphs has the same complexity. We intend also to discover the complexity of computing weak and strong fair possible winners in presence of incompleteness in voters' preferences.

Finally, we plan to study incomplete preferences in the context of strategic games. In fact, the notion of incomplete information is a well-understood topic in game theory and dealt with by means of Bayesian games, but games with incomplete preferences have not been analyzed.

Chapter 6

Conclusions

In this chapter we discuss what has been achieved in this thesis and we describe a number of possible directions we would like to pursue in the future.

6.1 Summary

Preferences and uncertainty occur in many real-life problems. Thus, it is important to model faithfully these two aspects both for problems with a single agent and for problems with several agents. A long-term goal is to define a framework where many kinds of preferences and many kinds of uncertainty can be naturally modelled and dealt with. In this thesis we have given some contribution in this direction, both for single-agent and for multi-agent settings.

We have started considering problems with fuzzy preferences, expressed by a single agent, and we have considered also the presence of uncertainty in these problems. We have then defined a formalism for handling fuzzy preferences and uncertainty, by integrating these two aspects via possibility theory, so that some desirable properties are satisfied. Moreover, we have defined suitable semantics for ordering the solutions according to different attitudes to the risk and, by following these semantics, we have defined a solver for handling this kind of problems. Next, we have generalized the approach to general soft preferences, thus obtaining a more powerful engine for reasoning with preferences and uncertainty, which can handle problems with uncertainty and with general preferences.

We have then considered scenarios where a single agent can express both positive and negative preferences, defining a formalism that produces the desired natural behaviour for what regarding the combination of these two kinds of preferences. Starting from the semiring-based soft constraint formalism, we have defined a structure for handling problems with bipolar preferences, which allows to compensate positive and negative preferences, and we have presented a solver for handling such problems. Moreover, in order to obtain a more powerful formalism, we have generalized such an approach to the case of problems with bipolar preferences and uncertainty.

We have then enlarged our target even more, encompassing also multi-agent scenarios. We have considered preference aggregation when the users reason about their own preferences, which can be partially ordered. In particular, the issues of fairness and non-manipulability of the aggregation schemes have risen. We have reconsidered the main results on these topics from social choice theory, namely Muller-Satterthwaite's theorem, which is Arrow's impossibility theorem for social choice functions, and Gibbard-Satterthwaite's theorem, with respect to partial orders. We have shown that the results continue to hold if the properties required are suitably adapted to partial orders.

Finally, we have considered a more general scenario, in which agents can partially express their preferences, analyzing the computational complexity of computing possible and necessary winners. We have shown that it is difficult both computing them exactly and approximating them. However, we have identified sufficient conditions on the preference aggregation function that allow us to compute the sets of possible and necessary winners in polynomial time. Moreover, we have shown the usefulness of possible and necessary winners in the preference elicitation process. We have then analyzed a specific class of preference aggregation functions, i.e., sequential majority voting, which performs a sequence of pairwise comparisons between two candidates along a binary tree. For such a voting rule, we have dealt with uncertainty which derives from the choice of the tree. We have characterized possible and Condorcet winners and we have shown that it is difficult to find them if we require that the tree must be balanced. In this case, it is thus difficult for the chair to manipulate the voting system. Finally, we have characterized winners of sequential majority voting in a more general scenario, where agents can hide some of their preferences, and we have shown that, if we don't require that the tree must be balanced, it is easy to find the winners in this scenario.

All the results presented in this thesis contribute to a general framework where users can model their problems in a natural and flexible way. For example, problems may have both hard and soft requirements. Thus it is reasonable to allow for both constraints and preferences in the modelling framework. Also, preferences can mean desires or rejection levels, so it is important to allow for both such notions when modelling a real-life problem. Finally, often problems contain preferences coming from several sources, so it is crucial to study how such preferences are combined to satisfy all sources of information.

Moreover, very often the users have to consider problems which are affected by uncertainty. However, uncertainty can be characterized in several ways, which depend on the information that the user has on the uncertain events. For example, the user can be completely ignorant about the occurrence of an event, he can know how probable is that the event will happen, or, in absence of a probability information, he can have a vague information over the occurrence of event and know only how much it is possible for the event to happen. It is useful to give users the freedom of representing uncertainty as they prefer. With this thesis, we allow users to model uncertainty coming from lack of probabilistic information on uncontrollable events. Thus users can either model a complete ignorance situation, or a situation where only possibilistic information is provided about the uncertain part of the problem.

Much work is still needed to achieve the long-term goal of a single framework where many kinds of preferences and uncertainty can be modelled by one or several agents, and where the underlying machinery is able to find efficiently the best solutions in all the available scenarios. This thesis tries to set the bases to move towards this goal.

6.2 Future directions

There are many future directions which will be interesting to pursue. The results of this thesis are a strong motivation to continue exploring fields of knowledge representation, searching for formalisms that can handle preferences and uncertainty both in the case of a single agent and in the case of several agents.

In the context of preferences expressed by a single agent, we plan to implement the solvers that we have presented for bipolar and not bipolar preferences with and without uncertainty, and we intend to perform experiments on benchmarks. We plan also to generalize the framework for what concerns uncertainty. In particular, we want to study problems with preferences and uncertainty, where uncertainty is not only expressed via possibility theory, but also via probability theory. Moreover, we plan to generalize the formalism of bipolar preferences for allowing no compensation of positive and negative preferences, in order to model classical multi-criteria approaches. We want also to strengthen the formalism for handling bipolar preferences introducing the notion of importance between pairs of variables.

Moreover, we plan to investigate further preference aggregation in multi-agent scenarios. The study of societies of artificial agents is a topic which is attracting an increasing amount of attention. One of the main goals of AI is to build tools that allow agents to reason in increasingly sophisticated ways. Moreover, when embedded into a distributed system, such agents must interact with others and negotiate common decisions while pursuing their personal goals. We believe that a powerful preference reasoning engine can make a difference, since it allows the representation of the agent's goal in a way that is amenable to negotiation and coordination with other agents, avoiding deadlocks. A considerable gain can be obtained by reconsidering many important results of social welfare theory and social choice theory in light of this new perspective. In fact, many properties which are desirable in human societies, as unanimity, strategy-proofness, monotonicity and many others are desirable also in all automated agents scenarios.

We also intend to investigate formalisms handling uncertainty in multi-agent preference aggregation systems, by defining new more sophisticated notions of possible and necessary winners. In particular, we plan to add constraints to agents' preferences, and so to consider possible and necessary winners which must be also feasible. Moreover, we plan to express preferences via compact knowledge representation formalisms, such as CP-nets and soft constraints, and to define possible and necessary winners directly from these compact formalisms. We also intend to add possibility distributions over the completions of an incomplete preference relation between outcomes, and to define winners in such scenarios. Finally, we plan to find new tractability results to compute possible and necessary winners both in general and for particular preference aggregation systems, possibly relaxing the IIA assumption.

Bibliography

[ABP05] L. Amgoud, J. F. Bonnenfon, and H. Prade. An argumentation-based approach to multiple criteria decision. In *ECSQARU'05*, pages 10–17, 2005.

[AP04] L. Amgoud and H. Prade. Using arguments for making decisions: a possibilistic logic approach. In *UAI 2004*, pages 10–17, 2004.

[Apt03] K. R. Apt. *Principles of Constraint Programming*. Cambridge University Press, 2003.

[Arr51] K. J. Arrow. *Social Choice and Individual Values.* John Wiley and Sons, 1951.

[ASS02] K. J. Arrow, A. K. Sen, and K. Suzumara. *Handbook of Social Choice and Welfare.* North-Holland, Elsevier, 2002.

[Ban95] J. S. Banks. Sophisticated voting outcomes and agenda control. *Social Choice and Welfare*, 1(4):295–306, 1995.

[Bar82] J. P. Barthelemy. Arrow's theorem: unusual domains and extended codomains. *Matematical Social Sciences*, 3:79–89, 1982.

[Bar83] S. Barbera. Strategy-proofness and pivotal voters: a direct proof of the Gibbard-Satterthwaite theorem. *International Economic Review*, 24(2):413–17, June 1983. Available at http://ideas.repec.org/a/ier/iecrev/v24y1983i2p413-17.html.

[BBHP99] C. Boutilier, R. Brafman, H. Hoos, and D. Poole. Reasoning with conditional ceteris paribus preference statements. In *UAI-99*, pages 71–80, 1999.

[BD02] R. Brafman and C. Domshlak. Introducing variable importance tradeoffs into cp-nets. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI-02)*, pages 69–76, San Francisco, CA, 2002. Morgan Kaufmann Publishers.

[BdBC97] E. Best, F.S. de Boer, and C.Palamidessi. Partial order and sos semantics for linear constraint programs. In *Coordination 97*, volume 1282 of *LNCS*, pages 256–273. Springer, 1997.

[BDKP02] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Bipolar representation and fusion of preferences in the possibilistic logic framework. In *KR 2002*. Morgan Kaufmann, 2002.

[BDKP06] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Bipolar possibility theory in preference modeling: representation, fusion and optimal solutions. *Information Fusion*, 7(1):135–50, 2006.

[Bel58] R. Bellman. On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90, 1958.

[BFM$^+$96] S. Bistarelli, H. Fargier, U. Montanari, F. Rossi, T. Schiex, and G. Verfaillie. Semiring-based CSPs and Valued CSPs: Basic Properties and Comparison. In M. Jampel, E. Freuder, and M. Maher, editors, *Over-Constrained Systems (Selected papers from the Workshop on Over-Constrained Systems at CP'95, reprints and background papers)*, volume 1106, pages 111–150. Springer, 1996.

[BH98] W. Bruns and J. Herzog. *Cohen-Macaulay Rings*. Cambridge University Press, 1998.

[BMR95] S. Bistarelli, U. Montanari, and F. Rossi. Constraint solving over semirings. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95*, pages 624–630, Montreal, Quebec, Canada, August 20-25, 1995. Morgan Kaufmann.

[BMR97] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint solving and optimization. *Journal of the ACM*, 44(2):201–236, mar 1997.

[BO] S. Bistarelli and B. O'Sullivan. A theoretical framework for tradeoff generation using soft constraints.

[BO91] J. J. Bartholdi and J. B. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8(4):341–354, 1991.

[Bou92a] D. Bouyssou. Democracy and efficiency – a note on Arrow's theorem is not a surprising result'. *European Journal of Operational Research*, 58:427–430, 1992.

[Bou92b] D. Bouyssou. Democracy and efficiency - a note on 'arrow's theorem is not a surprising result'. *European Journal of Operational Research*, 58:427–430, 1992.

[BS06] S. J. Brams and M. R. Sanver. Voting systems that combine approval and preference. In *The Eighth International Meeting of the Society for Social Choice and Welfare*, 2006.

[BT96] R. I. Brafman and M. Tennenholtz. On the foundations of qualitative decision theory. In *AAAI'96*, pages 1291–1296, 1996.

[BT97] R. I. Brafman and M. Tennenholtz. On the axiomatization of qualitative decision criteria. In *AAAI'97*, 1997.

[BTT95] J. J. Bartholdi, C.A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(3):157–165, 1995.

[CGB97] J. T. Cacioppo, W. L. Gardner, and G. G. Berntson. Beyound bipolar conceptualizations and measures: the case of attitudes and evaluative space. *Personality and Social Psychology Review*, 1:3–25, 1997.

[Cho04] J. Chomicki. Semantic optimization of preference queries. In *CDB*, pages 133–148, 2004.

[CLR90] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[CP04] L. Chen and P. Pu. Survey of preference elicitation methods. Technical Report IC/200467, Swiss Federal Institute of Technology in Lausanne (EPFL), 2004.

[CS02a] V. Conitzer and T. Sandholm. Complexity of manipulating an election with few candidates. In *Proceedings of AAAI-02*, pages 314–319, 2002.

[CS02b] V. Conitzer and T. Sandholm. Vote elicitation: complexity and strategy-proofness. In *Proceedings of AAAI-02*, pages 392–397, 2002.

[CS04] M. Cooper and T. Schiex. Arc consistency for soft constraints. *AI Journal*, 154(1-2):199–227, 2004.

[DB02] C. Domshlak and R. Brafman. Cp-nets - reasoning and consistency testing. In *KR-02*, pages 121–132, 2002.

[Dec03] R. Dechter. *Constraint processing*. Morgan Kaufmann, 2003.

[DF05] D. Dubois and H. Fargier. On the qualitative comparison of sets of positive and negative affects. In *ECSQARU'05*, pages 305–316, 2005.

[DF06] D. Dubois and H. Fargier. Qualitative decision making with bipolar information. In *KR'06*, pages 175–186, 2006.

[DFP95] D. Dubois, H. Fargier, and H. Prade. Fuzzy constraints in job shop scheduling. *Intelligent Manufacturing.*, 6(4):215–34, 1995.

[DFP96a] D. Dubois, H. Fargier, and H. Prade. Possibility theory in constraint satisfaction problems: Handling priority, preference and uncertainty. *Appl. Intell.*, 6(4):287–309, 1996.

[DFP96b] D. Dubois, H. Fargier, and H. Prade. Possibility theory in constraint satisfaction problems: handling priority, preference and uncertainty. *Applied Intelligence*, 6:287–309, 1996.

[DFP02] D. Dubois, H. Fargier, and P. Perny. On the limitations of ordinal approaches to decision making. In *KR 2002*, pages 133–144, 2002.

[Doy91] J. Doyle. Rational belief revision. In James F. Allen, Richard Fikes, and Erik Sandewall, editors, *KR'91: Principles of Knowledge Representation and Reasoning*, pages 163–174. Morgan Kaufmann, San Mateo, California, 1991.

[DP80] D. Dubois and H. Prade. *Fuzzy sets and Systems - Theory and Applications*. Academic Press, 1980.

[DP88] D. Dubois and H. Prade. *Possibility theory*. Plenum Press, 1988.

[DP93] D. Dubois and H. Prade. Belief revision and updates in numerical formalisms: An overview, with new results for the possibilistic framework. In *IJCAI 1993*, pages 620–625, 1993.

[DP98] D. Dubois and H. Prade. Possibility theory: quantitative and qualitative aspects. In *Handbook of defeasible reasoning and uncertainty management systems*, volume 1, pages 169–226. Kluwer, 1998.

[DP00] D. Dubois and H. Prade. *Fundamentals of Fuzzy Sets - Handbooks of Fuzzy Sets Series*. Kluwer Press, 2000.

[DT99] J. Doyle and R. H. Thomason. Background to qualitative decision theory. *AI Magazine*, 20(2):55–68, 1999.

[DW91] J. Doyle and M. P. Wellman. Impediments to universal preference-based default theories. *Artif. Intell.*, 49(1-3):97–128, 1991.

[EG02] M. Ehrgott and X. Gandibleux. *Multiple Criteria Optimization. State of the art annotated bibliographic surveys*. Kluwer Academic, Dordrecht, 2002.

[Fis74] P. C. Fishburn. Impossibility theorems without the social completeness axiom. *Econometrica*, 42:695–704, 1974.

[FL93] H. Fargier and J. Lang. Uncertainty in constraint satisfaction problems: a probalistic approach. In M. Clarke, R. Kruse, and S. Moral, editors, *Symbolic and Quantitative Approaches to Reasoning and Uncertainty, European Conference, ECSQARU'93, Granada, Spain, November 8-10, Proceedings*, volume 747 of *Lecture Notes in Computer Science*, pages 97–104. Springer, 1993.

[FLS96] H. Fargier, J. Lang, and T. Schiex. Mixed constraint satisfaction: a framework for decision problems under incomplete knowledge. In *Proceedings of AAAI'96*, 1996.

[FS03] H. Fargier and R. Sabbadin. Qualitative decision under uncertainty: back to expected utility. In *IJCAI'03*, pages 303–308, 2003.

[FW92] E. C. Freuder and R. J. Wallace. Partial constraint satisfaction. *Artificial Intelligence*, 58(1-3):21–70, 1992.

[GdBF03] M. Grabisch, B. de Baets, and J. Fodor. The quest for rings on bipolar scales. *Int. Journ. of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2003.

[Gea01] J. Geanakoplos. Three brief proofs of Arrow's impossibility theorem. *Economic Theory*, 2001.

[Gib73] A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41(3):587–601, 1973.

[GL05] M. Grabisch and Ch. Labreuche. Bi-capacities - parts i and ii. *Fuzzy Sets and Systems*, 151:211–260, 2005.

[HM66] F. Harary and L. Moser. The theory of round robin tournaments. *American Mathematical Monthly*, 73:231–246, 1966.

[Jun02] U. Junker. Preferences in AI and CP: Symbolic approaches. Technical Report WS-02-13, AAAI Technical Report, 2002.

[Kel78] J. S. Kelly. *Arrow Impossibility Theorems*. Academic Press, New York, 1978.

[Kel87] J. S. Kelly. *Social Choice Theory: An Introduction*. Springer Verlag, Berlin, 1987.

[Kie05] W. Kießling. Preference queries with sv-semantics. In *11th International Conference on Management of Data (COMMAD 2005)*, pages 15–26. Computer Society of India, 2005.

[KL05] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proc. IJCAI-05 Multidisciplinary Workshop on Advances in Preference Handling*, 2005.

[Lan02] J. Lang. From preference representation to combinatorial vote. In D. Fensel, F. Giunchiglia, D. L. McGuinness, and M. Williams, editors, *Proceedings of the Eight International Conference on Principles and Knowledge Representation and Reasoning (KR-02), Toulouse, France, April 22-25*, pages 277–290. Morgan Kaufmann, 2002.

[Lan04] J. Lang. Logical preference representation and combinatorial vote. *Annals of Mathematics and Artificial Intelligence*, 42(1):37–71, 2004.

[Las97] J.-F. Laslier. *Tournament solutions and majority voting*. Springer, 1997.

[Mac77] A. K. Mackworth. Consistency in networks of relations. *Artificial Intelligence*, 8(1):99–118, 1977.

[Mar95] G. De Marco. *Analisi Uno - Primo corso di analisi matematica*. Zanichelli, 1995.

[Mil80] N. Miller. A new solution set for tournaments and majority voting: further graph-theoretical approaches to the theory of voting. *American Journal of Political Science*, 24:68–9, 1980.

[Mou88] H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1988.

[MS77] E. Muller and M.A. Satterthwaite. The equivalence of strong positive association and strategy-proofness. *Economic Theory*, 14:412–418, 1977.

[NP05] K. Nehring and C. Puppe. On the possibility of strategy proof social choice: non dictatorship, anonimity and neutrality. Available at http://www.wior.uni-karlsruhe.de/LS_Puppe/Personal/Papers-Puppe/dict-sp-rev.pdf, 2005.

[OST57] C. E. Osgood, G. J. Suci, and P. H. Tannenbaum. *The Measurement of Meaning*. University of Illinois Press, Chicago, 1957.

[PY82] C. Papadimitriou and M. Yannakakis. The complexity of restricted spanning tree problems. *Journal of the ACM*, 29(2):285–309, 1982.

[Ren01] P. Reny. Arrow's theorem and the Gibbard Satterthwaite theorem: a unified approach. *Economics Letters*, pages 99–105, 2001.

[Rut94] Z. Ruttkay. Fuzzy constraint satisfaction. In *Proceedings 1st IEEE Conference on Evolutionary Computing*, pages 542–547, Orlando, 1994.

[RVW04] F. Rossi, K. B. Venable, and T. Walsh. mCP nets: representing and reasoning with preferences of multiple agents. In *AAAI-2004*, pages 322–327, 2004.

[Sat75] M.A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare function. *Economic Theory*, 10:187–217, 1975.

[Sch92] T. Schiex. Possibilistic Constraint Satisfaction problems or "How to handle soft constraints?". In D. Dubois and M. P. Wellman, editors, *UAI '92: Proceedings of the Eighth Annual Conference on Uncertainty in Artificial Intelligence, July 17-19, Stanford University, Stanford, CA, USA*, pages 268–275. Morgan Kaufmann, 1992.

[Sen70] A. Sen. *Collective Choice and Social Wellfare*. Holden-Day, 1970.

[SFPM02] P. Slovic, M. Finucane, E. Peters, and D. G. MagGregor. Rational actors or rational fools? Implications of the affect heuristic for behavioural economics. *The Journal of Socio-Economics*, 31:329–342, 2002.

[SFV95] T. Schiex, H. Fargier, and G. Verfaillie. Valued Constraint Satisfaction Problems: Hard and easy problems. In *Fourteenth International Joint Conference on Artificial Intelligence, (IJCAI 95)*, pages 631–639. Morgan Kaufmann, 1995.

[Str80] P. Straffin. *Topics in the Theory of Voting*. Birkhauser, 1980.

[TK92] A. Tversky and D. Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5:297–323, 1992.

[Ven05] K. B. Venable. *Reasoning with Preferences over temporal, uncertain , and conditional statements*. PhD in Computer Science, University of Padova and University of Bologna, 2005.

[VF99] T. Vidal and H. Fargier. Handling contingency in temporal constraint networks: from consistency to controllabilities. *J. Exp. Theor. Artif. Intell.*, 11(1):23–45, 1999.

[Vin82a] Ph. Vincke. Aggregation of preferences: a review. *European Journal of Operational Research*, 9:17–22, 1982.

[Vin82b] Ph. Vincke. Arrow's theorem is not a surprising result. *European Journal of Operational Research*, 10:22–25, 1982.

[Wal02] T. Walsh. Stochastic constraint programming. In *ECAI'02*, 2002.

[Wey84] J. A. Weymark. Arrow's theorem with social quasi-orderings. *Publich Choice*, 42:235–246, 1984.

[YSVR03] N. Yorke-Smith, K. B. Venable, and F. Rossi. Temporal reasoning with preferences and uncertainty. In G. Gottlob and T. Walsh, editors, *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 1385–1386. Morgan Kaufmann, August 2003.

[Zad78] L.A. Zadeh. Fuzzy sets as a basis for the theory of possibility. *Fuzzy sets and systems*, pages 13–28, 1978.