Game Theory at Work:

OR models and algorithms to solve multi-actor heterogeneous decision problems

Game Theory at Work:

OR models and algorithms to solve multi-actor heterogeneous decision problems

M. Elena Sáiz

Proefschrift
ter verkrijging van de graad van doctor
op gezag van de rector magnificus
van Wageningen Universiteit,
Prof. dr. M.J. Kropff,
in het openbaar te verdedigen
op woensdag 24 oktober 2007
des namiddags te half twee in de Aula

Game Theory at Work: OR models and algorithms to solve multi-actor heterogeneous decision problems
PhD thesis Wageningen Universiteit - with references - with summaries in English, Dutch and Spanish
Sáiz M. E., 2007
ISBN: 978-90-8504-769-8

Promotor:	Prof. ir. A.J.M. Beulens Hoogleraar Toegepaste Informatiekunde
	Prof. dr. ir. J.G.A.J. van der Vorst Hoogleraar Logistiek en Operationele Research
Co-promotor:	Dr. E.M.T. Hendrix Universitair Hoofddocent bij de Leerstoelgroep Operationele Research en Logistiek

Samenstelling promotiecommissie:

Prof. dr. F. Plastria	Vrije Universiteit Brussel
Prof. dr. J. Puerto Albandoz	Universidad de Sevilla
Prof. dr. A.M.A. van Deemen	Radboud Universiteit Nijmegen
Prof. dr. E.C. van Ierland	Wageningen Universiteit

Dit onderzoek is uitgevoerd binnen de Mansholt Graduate School of Social Sciences.

Abstract

The objective of this thesis is to explore the potential of combining Game Theory (GT) models with Operations Research (OR) modelling. This includes development of algorithms to solve these complex OR models for different empirical situations. The challenge is to get GT "at work" by applying such models and techniques to practical cases. Four different cases with a challenge on the development of algorithms are studied.

A first case illustrates a multiple coalition formation game in which membership rules and different transfer schemes are described. Given the GT model and the OR model, the goal is to develop methods for checking stability of coalition structures. A new mathematical programming formulation, crucial for the development of the algorithms, is elaborated. Available data is used to determine which stable coalitions appear and which procedures (transfer schemes) can be used to make coalitions stable. Also the influence of membership rules (whether actors are free to become a member) is investigated. Main conclusion is that transfer schemes are useful to be implemented to obtain stable coalitions. Moreover, different membership rules, e.g. veto or majority voting of current members, generate different results with and without transfer schemes.

A second case studies a model of coalition formation in politics with n parties trying to form a government. Given the number of parties n and policy dimension m (number of items), computational algorithms are developed to compute all possible majority coalitions and preferences of parties over those coalitions. Application to Dutch data and theoretical examples leads to testing of hypotheses with surprising results with respect to coalition formation such as: being a first mover is not necessarily advantageous, being less flexible is not necessarily advantageous, forming a minimal winning coalition is not necessarily advantageous.

A third case describes a two-stage location-quantity game where $n > 2$ firms are competing on $m > 2$ markets. The space where the firms can locate are nodes on a network. Analytical solutions for the supplying decisions and properties for determining the number of suppliers to each market are derived. In finding the equilibria, a complete enumeration algorithm and a local search algorithm are used. Two cases are elaborated to illustrate the procedures and the analytical results.

The last case deals with a competitive facility location problem in which the concept of Stackelberg leader-follower problem is applied. The follower problem and leader problem are global optimisation problems. Branch-and-Bound (B&B) algorithms that guarantee to find the optimum of both problems are designed.

Key words: Game theory, operations research, optimisation methods, algorithms.

Preface

By doing a PhD abroad, one is confronted with the culture etiquettes. As Spaniard, I tried to explain that we have sun but also snow; that we do not always take a "siesta" but a long lunch-break and keep working until 6, 7 or 8 p.m., we have central heating and we have the beautiful letter "ñ", without "her" what would happen with "mañana"...? We do not make appointments with friends, we just simply call and meet later the same day or "mañana", we do not have so many rules leading to "if-then" behaviour, we do not have a birthday calendar in the toilet, we just remember or forget the birthdays, but it is well accepted by friends. Yes, we are very good with jokes but we do not always speak so loud ... only when we are more than two.

Everything started in Sevilla. At the beginning of the summer of 2001 I moved to Sevilla to start my PhD studies at the department of Statistics and Operations Research. There, the Professor of the group assigned Rafael Blanquero as my supervisor. Thanks to Rafa I became enthousiastic about the PhD, and thanks to Rafa I came to Wageningen. Rafa, without you today would not have been possible.

I would like to thank my promoters Prof. ir. Adrie J.M. Beulens, Prof. dr. ir. Jack G.A.J. van der Vorst and my daily supervisor and co-promotor Dr. Eligius M.T. Hendrix. Adrie and Jack, first of all thanks for all the support, assistance, encouragement, criticism and philosophical questions during our meetings. I have learnt a lot from them, as you surely know. Adrie, thanks for all your active, positive, productive comments, and for always having a smile for everything. Jack, what can I say, since you arrived to our group everything changed ... to be better. Already in our first meeting I was impressed. At that moment I was already in my second year of PhD without any research question and any proposal, just going on here and there. You are the responsible on getting funds for the final two years of this adventure. You taught me lot of things during the meetings. For both, thanks for all the thinking on what is this thesis about, what are the research questions and for all the input on my "researcher" education. Eligius, thank you for your patient, guidance, thanks for all the thinkings, ideas, support, encouragement, jokes, we spent many many hours discussing the research. If I have to evaluate you as a supervisor, for me you are the best. I will be back to you later, as family.

I would like now to thank to the whole team of Operationele Research en Logistiek: Frits, thanks for asking now and then how I am doing; Theo, now Real Madrid is doing better with the Dutch players ehh; Joke and Karin, the girls, it was always funny to have some girls gossiping; Niels, we know how to respect and share the bad and good PhD mood, I hope you will have a successful career; the newcomer Jelena or Enna, she introduced a nice drink tradition on Fridays. And finally Ria, you were always so positive, kind, and a smile in the morning, thank you, I know that the bottles of wine were not the reason :-)

And during the last years I would also like to thank the team of Toegepaste Infor-

matekunde: Kees, Rob, Gert Jan, Ayalew, Dik, Mark, Yuan, Sebastiaan, Gerard, Jan, Kees, Sjoukje, Huub, Maarten and again ... Ria. I also want to express my gratitude to Paul van Beek for his helpful comments and interest in my research.

People from the Leeuwenborch: Ekko was responsible to bring me here to Wageningen. Then I met the whole STACO team. Thanks to all of you, you were always very supportive and new ideas were on the table at all of the meetings. Thanks to Juan Carlos, the Mexican part of the team. Thanks to Rob, Hans-Peter and Michael, clever and wise people.

I would like to express my gratitude to Agnieszka and Annelies, I really enjoyed to work together with you, it was a pleasure. I wish you all the best in your new positions. And many thanks for introducing me to Ad. I want to express my gratitude to José Fernández and Blas Pelegrín; research via e-mail is difficult but not impossible. Thanks for all the input-output of our common research.

Thanks are also extended to colleagues from the Social Sciences Department: to Amber and Janneke, I have always enjoyed a lot our coffee breaks outside; to Lan Ge, for sharing pre-conference *nervios* and *post-talk* fun, and for the years in here; to Morteza, my squash partner the first years. I would like to thank all people from Environmental Economics and Natural Resources Group and the Mansholt team. I should also mention the secretaries of the Leeuwenborch.

Specific gratitude goes to De Bongerd. I have all of the members in my heart. Thanks a lot for helping me to get rid of the stress or whatever I had in my head: Rob (2 Rob"os"), Henry, René, Wendy, John ... but special words go to Ellen and Ingi. Ellen, thanks for all the very nice moments/talks doing step, indoor biking or whatever out of sports. Sorry for missing a lot during my last PhD year ... Ingi, thanks for being the one to start an ongelofelijk-party-team with Ellen, you and me, thanks for the laughs together, the chats, the kinderen, the gezellige momentjes and thanks for being always there and for your support, enthusiasm and ideas about all the weird things I was doing. I must say thank to you for you being as you are. To both and all the team of the Bonderd, thanks for making many of my stuff easier.

I must also say thanks to Joanna. Last years were very gezellig, I had a lot of fun and I would never forget your 2006 birthday party. I would like to express my gratitude to the Rural corner of the Leeuwenborch. Jan Douwe, siempre has conseguido sacarme una sonrisa, gracias (todavía me debes una cerveza). Special thanks goes to Kei and Gustavo for sharing PhD up and downs, gracias por todo vuestro apoyo. A big word of thanks goes to Petra. I had a great time with you talking in Spanutch (siempre curiosa sobre el significado/raíz de muchas palabras), biking and discussing life issues. Ans and Cees, many thanks for taking care of the PhD team. I will never forget you and your gezellig parties. Ans, thanks giving us advice on the layout of our books. I would like to thank all the good friends I have met during my years in Wageningen, specially to Adolf, Olga, Toni, Marga, (Francesca)^2, Axel, Andre, Ada y la última en llegar, Noa "Ribera del Duero". I also wish to thank to Nelly for keeping my neck connecting my head and back.

I would like to thank Costanza, when she was living in Wageningen everything was different. I am glad that we met and we had/have many talks and discussions about life, thanks for sharing that with me. "Jenny" Co, "Il Sasha più adattato, zio felice, tranne nella pioggia. Un grande bacio e noi manchiamo a voi" :-O . Lola, Paul and Joek (het hondje), what can I say ... I do not have enough pages to say how important you are in my life, not only in Wageningen. Thanks for all your support and for the many many many good moments we have had. Thanks for sharing your friendship, for cheering me up when I needed. It maybe helped the jamón and chorizo for Paultje :-). Out of jokes, I love you and you will be always on my heart and my life. Special thanks goes to Lola. I had great time with you, thank you for your humor, your encourage, for sharing the good and the bad on our PhD

roken-punt-moments. And thanks for all the help. Gracias por todo, “Jenny” galleguiña, eres muy grande.

I would like to thank all my friends in Spain: Laurita, Clara “caminera”, Elvis, Jelen “gemela”, Corita, Marta “logroño”, Marta “mingo”, Ana, María, Fran, Avel y Marta,Valen, Yolanda, Almu y Sara, Mercedes y Elenita, Esperanza y Lina. Thank you for all the support and emails. Os quiero muchísimo, gracias por todo vuestro apoyo, por mantenerme al día de vuestras cositas. Un besazo enorme.

Gracias Eligius. I do not know how to express all my gratitude to you; thank you for being you, for taking me at your place when I came to Wageningen, for all the support, for all the mathematics and algorithmic ideas, for all the encouragement, for all the fun and the very nice talks while sharing the wine from Spain, for all the personal chats we had, for all the understanding, for all the gezellig during the travels to Spain by car, in summary, THANKS. As I always said, thanks for being my family, my father and ... mother in Wageningen. You are great.

Finally I would like to thank to my family. Gracias Mamá y Papá, Papá y Mamá, gracias a mis hermanitos (Javier-Javi y Antonio-Toño). Gracias a todos, sin vosotros habría sido imposible hacer esto. Ha sido muy duro estar lejos de vosotros, pero ha merecido la pena, ya sabéis, personalmente. Miles, millones, no existe el número para decir GRACIAS por todo vuestro apoyo en mis objetivos, por todas vuestras visitas, por vuestras llamadas telefónicas, POR ESTAR AHÍ. Gracias al resto de mi familia, pero también tengo que decir gracias a mi queridita tía y a mi prima favorita, gracias por todo vuestro apoyo, yes, lo he sentido muy dentro de mi corazón. Y a Sasha, mi perrito, gracias por compartir conmigo los últimos años en Wageningen. Esta tésis va dirigida a todos vosotros ... pero tengo que hacer una mención especial a alguien que estoy segura vais a respetar.

Esta tésis va dirigida especialmente a ti, abuelita, te conozco perfectamente y sé que habrías sido la primera en coger un avión y estar aquí conmigo hoy, disfrutando como nunca de esta parafernalia, hablando spanglish, aunque no tuvieras ni idea de inglés, tu sabes, hablando espanol bien alto, claro y, por supuesto, despacio ... que estos guiris no se enteran de otra forma ehh!? Y además habrías sido la reina de la fiesta, como siempre. Un beso al cielo desde la loca Tierra. Tu nieta.

(English translation: this thesis is specially dedicated to my grandma, I know you perfectly and I know that you would have been the first, taking a plane and being in here with me, enjoying as never this paraphernalia, talking Spanglish by you know, talking Spanish too loud, clear and, of course, slow ... these guiris do not understand in another way ehh!? Furthermore, you would have been the queen of the party, as always. One big kiss to the sky from the crazy Earth. Your granddaughter.)

CONTENTS

CHAPTER 1

General Introduction

1.1 Introduction

> "The work of managers, of scientists, of engineers, of lawyers – the work that steers the course of society and its economic and governmental organizations – is largely work of making decisions and solving problems. It is work of choosing issues that require attention, setting goals, finding or designing suitable courses of action, and evaluating and choosing among alternative actions. The first three of these activities – fixing agendas, setting goals, and designing actions – are usually called problem solving; the last, evaluating and choosing, is usually called decision making. Nothing is more important for the well-being of society than that this work be performed effectively, that we address successfully the many problems requiring attention at the national level (the budget and trade deficits, AIDS, national security, the mitigation of earthquake damage), at the level of business organizations (product improvement, efficiency of production, choice of investments), and at the level of our individual lives (choosing a career or a school, buying a house)."
> (*Simon et al. (1986), p.19*)

Decision making is present in daily life. Governments, organizations, workers are confronted everyday with different decision problems and with the need to address and resolve these problems. The moment we wake up everyday we start making decisions and try to do things the best way or as good as possible given the decision making situation. Often one tries to maximise a kind of utility in order to reach an objective. Some decisions can be made automatically, that is, the decision maker has been confronted with a similar problem many times and has learned to arrive at a solution using a predefined algorithm (solution method). These are called *programmed decisions*. Other decisions can be characterised as non-programmed and less structured, also called *unstructured decisions*. That means that the decision maker has to cope with unusual and less structured situations in which decisions are made based on information or intuition. Decision making takes place when a need for a decision is recognized. Based on it a problem is delineated and solution alternatives are generated. The alternatives are evaluated based on the objective to generate a ranking on preference order of alternatives. Based on the evaluation and identification of expected risks, consequences and values of each alternative, one can select the best-favourable alternative related to the objective of the decision maker.

Keen and Scott-Morton (1978) identified five different approaches of decision making:

- A *Rational* way of decision making, in which the manager is completely informed, knows all the alternatives and can make an optimal choice ('objective rationality');
- A *satisfycing* way of decision making, in which a decision alternative is sought that satisfies all participants ('bounded rationality');
- A *organisational procedures* way of decision making, which sees decisions as the output of standard operating procedures invoked by organisational subunits;
- A *political* way of decision making, in which a decision is seen as a result of negotiations between actors; power and influence determine the outcome of any given decision;
- A *individual differences* way of decision making, which presupposes a very important role for the character of the individual, and in which personality and style are of great importance.

In van der Vorst (2000) the satisfycing approach is emphasised as the most relevant:

> "The rational way of decision making could be used for the reordering of standard products; and the political way of decision making could be used when selecting another supplier. But in general, a mix of the mentioned approaches is applied when making a decision (Benders et al. (1983)). According to Simon (1976) 'most human decision making, whether individual or organisational, is concerned with the discovery and selection of satisfactory alternatives; only in exceptional cases is it concerned with the discovery and selection of optimal alternatives'. Because of this bounded rationality the choice in most decision making situations is for satisfycing decisions."
> (*van der Vorst (2000), p.58*)

In many decision situations the decision problem can be decomposed into components that are to be dealt with in parallel and/or subsequently. In other words, decision makers face the decision problem in a simultaneous or in a step-by-step way. Decision situations may also involve more than one decision maker, which leads to a competing or cooperating situation. For example, when a supermarket has to decide about where to locate and other competing supermarkets already exist or plan to locate at the same area. Or if a political party does not have a majority in the government, negotiations with other parties can take place in order to form a majority government. Decisions have in common that all are made in order to solve problems. Unfortunately, not all problems are easily solved without modelling approaches and the development of associated solvers (computational algorithms).

We focus on decision making problems following the rational way of decision. Within the context of decision making processes, it is well accepted (Simon (1976), Keen and Scott-Morton (1978) and van der Vorst (2000)) to use Model Based Decision Support Systems (MBDSS) with associated solvers for models incorporated. These models are then meant to help to generate decision alternatives and/or to evaluate consequences of decision alternatives. In this thesis we focus on a special class of optimisation models and algorithms (solvers) that can play an important role in such MBDSS. Models are made to represent the decision situation under study. Solvers are developed that allow the modeller to obtain efficiently and effectively optimal solutions for models developed. These "optimal" solutions can be of great value in practical decision situations. In the remainder of this thesis, we use the term "optimal" in the sense just described.

We hypothesise that Game Theory may have modelling merits to contribute to decision making situations where decision makers have competing or cooperating objectives, under the assumption that the decision makers are rational and they act in their best own interest. Game Theory is a key element for studying decision making problems involving two or more decision makers. It provides concepts and applies mathematical models for selecting an optimal strategy considering the strategy of the others. In daily life, one can observe cooperation

in the form of club membership, alliances between companies and countries (sky alliance, NATO, EU) on all levels. Part of the cooperation incentives is economic; the signatories of a coalition consider membership a win-win situation over non membership. Such incentives drive horizontal and vertical cooperation in supply chains and club formation between organisations and countries. One can also observe competitive situations, for example when two or more chains are trying to get more market share from a common area and they compete on prices. Competition not only takes place in business, also games like tennis or chess are competitive.

Many real world decision problems need more elaboration than just optimizing only one variable (e.g. time) or only one decision maker objective. Not only the number of decision makers plays a role, many problems involve decisions at many levels, over several periods or on a planning horizon. The objective is to guarantee to have an optimum. One could try to handle it with a hierarchical approach, solving each subproblem at each level, in some cases, using the information obtained from the decisions made in earlier subproblems. When more than one decision maker is involved, Game Theory gives us modelling techniques to do so. Game theory models are often applied to situations with few or symmetric multiple decision makers, which limits its modelling power. It is well accepted in literature that it is impossible to completely represent the real world in models. The real world and decision situations are too complex. Therefore, we aim at Game Theory models that are able to better represent or are able to represent more important features of an types of relationships of the real world. So we are trying to minimise the gap between our modelling requirements and models actually built. In addition, we aim to develop solvers for these enhanced Game Theory models.

Results and models in Game Theory can be found in literature. These results are often due to basic models with two actors leading to analytical solutions or alternatively to more actors that are all assumed to be the same; symmetric players. In this thesis we extend these models by studying cooperative or competitive decision situations in the observable world: building new models in Game Theory and reformulating into Operations Research models to study cases with more than two heterogeneous actors. In this case, one has to apply and develop new Operations Research techniques, that is, modelling optimisation problems and developing algorithms to solve them. Therefore, in order to compute the solutions it is required first to build new or to use existing Game Theory models for multi-actor cooperation; second to develop and reformulate these models using Operations Research techniques; and third to develop new algorithms for solving the models using sets of equalities, linear optimisation, nonlinear optimisation, combinatorial optimisation, etc. If we look at relevant literature and especially also at case descriptions in different areas such as environmental decisions, supply chain management decisions and political games, we find decision support requirements and decision situations that may benefit from a modelling approach using Game Theory models because of their modelling capabilities (e.g. to represent competing decision makers). As a consequence, it is of scientific and practical importance to research the value of Game Theory models and associated solvers.

The objective of this thesis is to explore the potential of combining Game Theory models with mathematical modelling and solve these complex Operations Research models by developing algorithms in different empirical situations. The idea is to contribute filling the gap between theory and practice developing Operations Research models and algorithms for Game Theory models describing Decision Making problems. The challenge is to get Game Theory "at work" by applying such models and techniques to practical cases. This provides insight into which concepts of Game Theory are practically applicable for which type of cases, and on the other hand, get insight in the use of mathematical programming models and algorithms to reach the target.

We first give a small overview on relevant Game Theory (GT) concepts and models in Section 1.2 and Operations Research (OR) techniques in Section 1.3. Section 1.4 specifies the objective of this thesis and the research questions. Section 1.5 elaborates the objective of the study and how the questions are investigated together with a brief outline of the thesis.

1.2 Game Theory Introduction

In von Neumann and Morgenstern (1944) a clear introduction to what constitutes a game is given:

> "First, one must distinguish between the abstract concept of *a game*, and the individual *plays* of that game. The *game* is simply the totality of the rules which describe it. Every particular instance at which the game is played – in a particular way – from beginning to end, is a *play*.
>
> Second, the corresponding distinction should be made for the moves, which are the component elements of the game. A *move* is the occasion of a choice between various alternatives, to be made either by one of the players, or by some device subject to chance, under conditions precisely prescribed by the rules of the game. The *move* is nothing but this abstract 'occasion', with the attendant details of description, – i.e. a component of the *game*. The specific alternative chosen in a concrete instance – i.e. in a concrete *play* – is the *choice*. Thus the moves are related to the choices in the same way as the game is to the play. The game consists of a sequence of moves, and the play of a sequence of choices.
>
> Finally, the *rules* of the game should not be confused with the *strategies* of the players. Exact definitions will be given consequently, but the distinction which we stress must be clear from the start. Each player selects his strategy – i.e. the general principles governing his choices – freely. While any particular strategy may be good or bad – provided that these concepts can be interpreted in an exact sense (...) – it is within the player's discretion to use or to reject it. The rules of the game, however, are absolute commands. If they are ever infringed, then the whole transaction by definition ceases to be the game described by those rules. In many cases it is even physically impossible to violate them."
>
> (*von Neumann and Morgenstern (1944)*, p. 49)

In this section, GT is briefly introduced and related to the focus of this thesis. For a more profound elaboration, interested readers are referred to other more specific texts on GT such as Myerson (1991) or Shubik (1984). GT Models are games, which are basically defined by: number of players, set of available strategies, outcomes and payoffs associated with the outcomes. GT concepts define the characteristics of the games. GT concepts give the details of the elements of the game (e.g. player, strategy, utility), details about the information available to the players (e.g. complete, perfect), or how players choose their strategies (e.g. simultaneous) and other characteristics of the game like communication rules. Next, these and some others GT concepts are introduced.

A *game* can be defined as a mathematical representation of a conflict situation. The outcome depends on mutual interaction between two or more *rational* individuals, called *players*. A player can be defined as a decision maker in a decision problem: a person, a group, an animal, or whatever entity. Depending on the number of players, games can be classified as a *two-person game*, *three-person game*, and in general a *N-person game*, $N > 2$.

Games have *rules* specifying the *actions* the player can take, the *information* the player has available and the *consequences* of the decisions. Consequences usually are not only for the player itself but also for the other players, which leads to a cooperative or to a competitive

situation. For example, in a tennis game, the objective is not only to hit the ball back to the other player but to hit it back in a way the other player cannot return it, which also depends on where the opponent is placed. A game must also include *payoffs or utilities* (profits of the players), i.e., the reward one gets. *Set of alternatives* is used to describe the different actions or strategies among which a given player can choose on the feasible space of the game, as quoted from von Neumann and Morgenstern (1944), a *move* is the occasion of a choice, the specific alternative chosen is the *choice*. A *strategy* defines a set of moves (choices) a player will follow in a game. The *strategy set* is the set of strategies all players have chosen.

There is a distinction between games with *complete* and games with *incomplete* information depending on the information the players have about the parameters of the game. In games with complete information, players have full information about the strategies and utility functions of the other players. In incomplete information games, players may or may not know some information (strategies, payoffs or preferences) about the other players. Games can also be divided into games with *perfect* and *imperfect* information depending on the information the players have about the past behaviour. In games with perfect information, players have full information about the actions that have already taken place.

According to how players choose their strategies, there is a distinction between *simultaneous-move games* and *sequential-move games*. In simultaneous games, players choose their strategies simultaneously, in ignorance of the strategies of the others. Players must predict what the opponents will do. A game can also be defined as a simultaneous game when decisions are not taken at the same moment but moves are in ignorance of what the rest of players are doing. In *sequential games*, players make decisions in sequential order.

Based on the gains or losses by the players, there are *zero-sum games* and *non-zero sum games*. The most common situation is to apply these games for two players. Zero-sum games are conflict/competitive situations; the gain of a player (set of players) is the loss of the other player (set of players), i.e. the total benefits for the players, for every combination of the available strategies, sum to zero. In non-zero sum games, aggregate payoffs are different from zero. Furthermore, non-zero sum games are not strictly competitive games.

Depending on the *communication rules*, distinction is made between *non-cooperative* and *cooperative games*. In non-cooperative games, players are unable to make enforceable agreements outside of those specified by the game. Any cooperation must be self-enforcing. In cooperative games, players can make enforceable agreements and form clubs before choosing their actions. Competition is between clubs and not between individuals.

Another distinction is between *symmetric games* or *asymmetric games*. In a *symmetric game*, independently of the player who is playing a particular strategy, a player earns the same payoff when she chooses the same strategy/action against her competitors. Otherwise, a game is called *asymmetric*. In games, players can be *homogeneous* or *heterogeneous*. The outcomes for heterogeneous players are not the same if they play the same strategy.

According to the *rules of coalition negotiation*, two different games can be outlined depending on whether or not the players are allowed to make utility side payments as an incentive for the members. If side-payments are allowed, a game is called *Transferable Utility Game*, otherwise it is called *Non-Transferable Utility Game*.

Another concept is that of *externalities*, positive or negative. A game with externalities occurs when a decision made by a player causes an effect in terms of benefits (positive externalities) or costs (negative externalities) to third parties (other players). Examples are pollution (negative externalities) or public goods (positive externalities). Positive externalities are related to the *free-rider problem*. A free-rider gets the benefits from positive externalities but does not contribute to its costs.

In this thesis we use two main solution concepts in GT. One is the *Nash equilibrium*. Nash (1951) introduced the concept in which no player can be better off by following another

strategy while the others remain unchanged. A Nash equilibrium is a set of strategies for the players such that no player has an incentive to unilaterally deviate from her strategy. At equilibrium, each player is using the strategy that is the best reply to the strategies chosen by the other players.

Definition 1. ***Nash Equilibrium:*** *A Nash equilibrium is a profile of strategy choices such that every strategy of the players is a best reply to the strategies chosen by the other players. In a Nash equilibrium, no player has any incentive to deviate unilaterally from it.*

The definition of Nash equilibrium using the concept of best responses is as follows:

Definition 2. *The strategy profile* $(s_1^*, \ldots, s_i^*, \ldots, s_n^*) \in S$ *is a Nash equilibrium if for each* $i \in N$*:*

$$s_i^* \in \arg\max_{s_i \in S_i} (\pi_i(s_1^*, \ldots, s_i, \ldots, s_n^*))$$

where S_i *denotes the set of all possible strategies for player* i *and* π_i *the utility function.*

We also consider the classical concept of *stability of coalitions* introduced for cartel coalition formation (one coalition) by d'Aspremont et al. (1983), which is in fact close to the Nash equilibrium concept. A coalition is stable if, on the one hand, for a member of the coalition there is no incentive to free-ride (internal stability) and on the other hand, for a non-member there is no incentive to take part of the coalition (external stability).

Two GT models are relevant in this study. The first is due to the French economist Augustin Cournot, the *Cournot competition model* (Cournot (1838)). It originally describes a competitive model between firms producing a homogeneous product. Firms compete simultaneously on quantities, choosing how much to deliver to markets. The objective of each firm is to maximise profit. The game belongs to the class of imperfect information. The second relevant model was proposed by the German economist Heinrich Freiherr von Stackelberg, the *Stackelberg competition model* (Stackelberg (1934)). It models behaviour of two competing players called the *leader-follower*. The leader player moves first and chooses a strategy, then the follower moves sequentially, knowing the decisions taken by the leader. This game belongs to the class of perfect information games.

In this thesis, we use most of the concepts outlined above in four different cases. In two of the cases, more than one decision needs to be made, that is, decisions are made in a so-called *two-level decision game*. At each stage or level of decision, all the players make their decisions simultaneously. In these cases, we use the GT concept *subgame perfect equilibrium*, which means that at every stage/step a Nash equilibrium is computed.

In Section 1.5.1, an outline of the relevant concepts we use through this thesis is given.

1.3 Operations Research

Most of the decisions that we make in daily life have a goal: optimising a set of variables, where variables mean characteristics defining a problem. Operations Research has been defined in many books and articles. In Hillier and Lieberman (1986) a complete description of what OR is given.

> "Operations Research is concerned with optimal decision making in, and modelling of, deterministic and probabilistic systems that originate from real life. These applications, which occur in government, business, engineering, economics, and the natural and social sciences, are largely characterized by the need to allocate limited resources. In these situations, considerable insight can be obtained from scientific analysis such as that provided by operations research. The contribution from the operations research approach stems primarily from:
>
> 1. Structuring the real life situation into a mathematical model, abstracting the essential elements so that a solution relevant to the decision maker's objectives can be sought. This involves looking at the problem in the context of the entire system.
> 2. Exploring the structure of such solutions and developing systematic procedures for obtaining them.
> 3. Developing a solution, including the mathematical theory, if necessary, that yields an optimal value of the system measure of desirability (or possibly comparing alternative courses of action by evaluating their measure)."
>
> (Hillier and Lieberman (1986), p. 6)

Thus, first step in modelling using OR is the formulation of the objective and specification of the decision variables of the problem. Based on the objective and decision variables identified, a model can be constructed as a mathematical representation of the problem. Conditions on the decision variables, like relationships between variables and bounds, define what it is called *constraints* of the problem. Constraints are defined by using mathematical equations and/or inequalities. All the points satisfying the constraints define what it is called the *feasible area*. The objective function is defined from the objective of the problem. The idea is to optimise the objective function, maximising or minimising, subject to the constraints (if they exist). The objective function together with the constraints define the mathematical programming model, also known as optimisation problem. The next step is to solve the model finding an optimal solution. A solution for the optimisation problem satisfies the constraints (so-called a feasible solution) and gives a numerical value to each of the decision variables. An optimal solution is feasible and represents the most desirable value for the objective of the problem. It can happen that a problem has no optimal solution, only one or an infinite number. Most of the cases, in order to solve an optimisation problem an algorithm is required. OR models (or optimisation models) and optimisation algorithms are described next.

1.3.1 Operations Research Models

In the process of decision making situations, mathematical techniques can be used. As introduced before, decision variables may be restricted by conditions like bounds or relations with other variables, this is called a *constrained optimisation problem*. When there are no constraints the problem is called *unconstrained optimisation problem*. Thus, a first basic way on modelling a decision problem is considering the restrictions on the decision variables. For example, in the tennis game the feasible area where one is allowed to play has limits.

When a problem has been defined as constrained or unconstrained, more characteristics can be used in modelling it. The decision variables of the problem may be *continuous* or *discrete*. In a *linear programming problem (LP)*, the objective and the constraints are linear functions and the decision variables are continuous. When at least one of the constraints or the objective function are non-linear, the problem is called a *nonlinear optimisation problem* (NLP). Within nonlinear optimisation problems, one can distinguish between *smooth* (when derivatives exists for all the functions in the model) and *non-smooth* (not all the functions have derivatives defined at all the points). More distinctions are: *unconstrained NLP optimisation*, a NLP problem without constraints; *quadratic programming* in which the objective

function have quadratic terms and the constraints are linear equalities and inequalities; *convex programming*, where the objective function is convex and constraints form a convex set; *separable programming* implies that the objective and the constraints are separable functions (a function where each term involves just a single variable, so that the function is separable into a sum of functions of individual variables. (Hillier and Lieberman (1986)); *non-convex programming* refers to nonlinear programming problems not satisfying the assumptions of convex programming; or *fractional programming* dealing with the optimisation of an objective function having the form of a fraction (or ratio). In general, several local minima (maxima if it is a maximisation problem) may exist. *Global Optimisation* aims at finding a solution which is guaranteed to be the global optimum within a predefined accuracy.

If decisions are hierarchically structured, the problem can be modelled by *multi-level optimisation programming*, which are hierarchical optimisation problems. If the number of levels is two, the problem is called a *Bilevel optimisation problem*. In bilevel optimisation problems, the mathematical programming problem contains an optimisation problem in the constraints. Bilevel optimisation problems are close to the Stackelberg problem from Game Theory (Colson et al. (2005)).

Now consider that the decision variables are discrete, that is, variables are restricted to values specified by a discrete set. If all the variables are required to take integer values, then the problem is modelled by *Integer Programming (IP)*. If in addition the objective and constraints are linear the problem belongs to the class of *Integer Linear Programming (ILP)* problems. When the problem has integer variables and continuous variables, the problem is called a *Mixed Integer Optimization Problem (MIP)*. A problem involving combinatorial choices is called a *Combinatorial Optimisation Problem*.

Linear or non-linear, continuous or discrete, decision making problems may also involve a time horizon space. When decisions need to be made over several periods, the problem is modelled by a *multi-period problem*. Otherwise, the problem is called a *static programming problem*. In Section 1.5.1 an outline of OR models relevant for the research in this thesis is given.

1.3.2 Operations Research Algorithms

An algorithm is a step-by-step procedure for solving a problem. It is usual that an algorithm repeats several steps over and over again until the desired result is obtained. In that sense, algorithms are iterative processes. Consider a group of friends. A member knows a gossip but she wants to keep it as exactly as possible, not running and inflating from mouth to mouth. Algorithm 1.1 shows a possible algorithm to solve the problem.

Algorithm 1.1 : Gossiping1

Funct Gossiping1(SET OF FRIENDS,*gossip*)

1. *List* := SET OF FRIENDS
2. **while** *List* is not empty
3. x := Take a friend from *List*
4. Tell the *gossip*
5. Get the promise to not say anything
6. Delete x from *List*
7. OUTPUT: Desired objective

When writing algorithms, two questions arise: do the procedures reach the desired solution and at what computational cost? As defined in Hendrix (2007), *efficiency* is the effort the algorithm needs to be successful. Several efficiency indicators appear in literature. Usual indicators are the number of function evaluations and memory requirements necessary to reach the optimum. Another question is related to *effectiveness*: does the algorithm find what we want? Algorithm 1.1 is effective (assuming that they will keep the promise) but is it efficient? Algorithm 1.2 shows another possibility for solving the problem.

Algorithm 1.2 : Gossiping2

Funct Gossiping2(SET OF FRIENDS,*gossip*)

1. Make an appointment with the SET OF FRIENDS
2. Tell the *gossip*
3. Get the promise to not say anything
4. *List* := SET OF FRIENDS - SET OF FRIENDS SHOWED UP
5. **while** *List* is not empty
6. x := Take a friend from *List*
7. Tell the *gossip*
8. Get the promise to not say anything
9. Delete x from *List*
10. OUTPUT: Desired objective

It can be seen that Algorithm 1.2 is not only effective but it is more efficient than Algorithm 1.1. In general, algorithms have an *initialization step* setting up the parameters to start iterations, *iterative step* performing the operations, and *optimality test or stopping rule* checking if the current solution is the optimal. Now the focus will be on algorithms solving the OR models outlined in Section 1.3.1.

The *simplex algorithm* (Dantzig (1963)) became a popular method for solving LP problems. There are variants of the simplex algorithms like the *upper bound technique*, the *dual simplex method* or *parametric programming*. Karmarkar (1984) introduced an interior point method for solving LPs. This algorithm is a polynomial time algorithm which is superior (in theory) to exponential time algorithms, like the simplex method.

Dynamic Programming is a tool that allows solving problems by using recursive calls to subproblems with the same structure and smaller size. Dynamic Programming models can be formulated when decisions have to be made in a sequence and are interrelated. Each stage influences the decision at the next stage. A recursive relation exists identifying the optimal solution of stage n given the optimal solution for stage $(n+1)$. Thus, the solution procedure consists of solving first the last stage and then recursively going backwards.

Combinatorial Optimisation Problems, Integer Programming (IP) and Integer Linear Programming (ILP) problems are not easy to solve. In many cases there exists an exponential explosion on the size of the problem increasing the computational difficulty. Dealing with these problems by enumeration can be computationally very expensive depending on the number of variables and structure of the problem. A possible algorithm for solving ILP problems uses LP methods by doing what is called LP-relaxation. However, this method does not always find the optimal solution (Schrijver (1998)). New algorithms are necessary. An often used technique for solving ILP problems is the Branch-and-Bound method. In contrast with a total enumeration method, the Branch-and-Bound approach uses a partial enumeration. Branch-and-Bound is also used to solve most of the Combinatorial Optimisation problems. An alternative to Branch-and-Bound methods are the Cutting Plane methods (Gomory (1958)). There also exists many heuristic algorithms for solving combinatorial op-

timisation problems where solutions may be not optimal but satisfactory.

In contrast with LP problems, for NLP Problems no single based-algorithm exists that will solve the problem. Sometimes one can derive explicit solutions based on analysis of first order optimality conditions. Traditionally this is called *analytically solving*. Algorithms have been developed to solve individual problems like the ones described in the former section. For these models, many different *Search Algorithms* have been developed during the last decades, procedures like Golden Section search, Penalty Approach, Steepest Ascent (descent if minimising) or Global Optimisation Methods. Branch-and-Bound methods are also used to solve GO models. For solving convex nonlinear models, available methods are for example, *interior point methods* or *ellipsoid methods*. One of the procedures applied in this thesis is based on *Sequential Quadratic Programming (SQP)* (Boggs and Tolle (1996)).

Multilevel programming optimisation problems can be solved by LP, ILP, Combinatorial, Dynamic or NLP techniques depending on the characteristics of the model. In Section 1.5.1, the methods used in this thesis are outlined.

1.4 Research Objective and Questions

GT not only gives analytical results (theorems) but can also be applied to numerical cases to derive tendencies for multi-actor cooperation in different circumstances. The challenge in this thesis is to build GT models based on GT concepts to describe economic incentives for multi-actor heterogeneous situations. First of all, one has to translate GT models into Mathematical Programming models. After that, for solving these models, algorithms have to be developed using OR techniques. Therefore, the focus of the study is on supporting the process of building models to represent decision situations and developing techniques to solve these models. Summarising and being more explicit, Figure 1.1 shows the conceptual model of the thesis research.

Information from the decision problem is used to select which GT concepts are necessary and applicable. From the GT concepts and the decision problem, one can build a GT model. Based on the GT model and from the objective(s) defined within the decision problem, a mathematical programming model (optimisation model) is formalised. One should try to code the problem in an appropriate way. With the mathematical definition of the model, algorithms are developed and used to interact with the decision problem. Interaction of algorithms and problem definition is made by testing and validating results according to the objectives. The challenge is to build GT and OR models and the development of algorithms for solving decision situations with multiple, heterogeneous and asymmetric actors.

With this conceptual model, the **research objective** is specified: to develop or contribute to the development of usable GT models and OR methods to solve them in practical situations. In other words, assess the capability of GT to describe economic incentives for complex decision problems involving multiple actors. In order to do this, there is a need to develop optimisation models and algorithms to implement analytical concepts and models from Cooperative/Non-Cooperative Game Theory and Competitive models such that they are applicable to numerical cases in economic literature.

With respect to the scientific relevance, it is necessary to distinguish between its methodological and theoretical gains. First of all, as explained above, the thesis aims to develop a set of methodological tools which allows us to do numerical analyses. Moreover, the thesis aims to construct a conceptual model that combines GT, OR and Social Sciences. This is of interest for all scientists involved in the field of research and, of course, specifically those involved in the study of GT models and OR techniques. The research aims to provide a

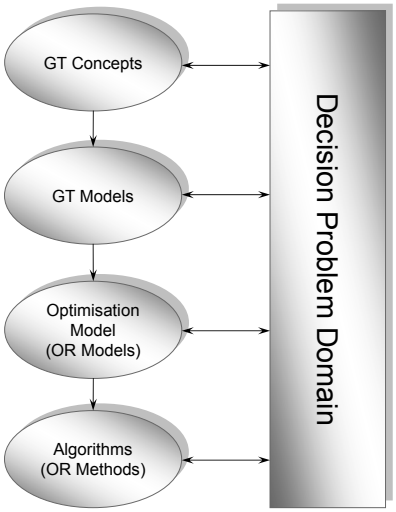

Figure 1.1. *Conceptual Model of Thesis Research.*

useful contribution to at least the following areas of related concern:

- Building optimisation (OR) models based on GT concepts/models
- Developing OR techniques as tools to solve optimisation (OR) models based on GT concepts/models with multiple heterogeneous actors
- Cases of cooperation and competition are used to show the applicability of the GT models and the OR techniques

We hypothesise that GT concepts and models might provide us with a valuable contribution for supporting decision makers provided that we can make OR models and algorithms to solve these models. We can translate these aims and hypothesis in the following **research questions**:

1. How do we formulate GT models for decision making situations with multiple heterogeneous actors using GT concepts? How can we contribute to that modelling?
2. If we have GT based models: What appropriate coding can be used to translate a specific GT model into a mathematical programming optimisation problem such that it can be solved?
3. Which solution methods can be developed to solve these models?
4. How do the outcomes aid to analyse the decision makers problem? or What is the contribution of the new outcomes to the decision makers problem?

The following describes how we deal with these questions.

1.5 Research Approach and Outline of the Thesis

Governments (countries), Companies, Markets (oligopoly, duopoly, bilateral monopoly, and so on) lend themselves to game-theoretic analyses because the fate of each participant depends on the actions taken by the other participant or participants. Methods are developed to deal with finite strategy approaches (enumeration) and continuous decision problems based on GT concepts.

1.5.1 Research Design: Selection of Cases

The conceptual model explained above connects GT concepts and models with OR models and algorithms. In illustrating the conceptual model, we use case study research. We are interested in cases covering some or all of the following characteristics:

1. Empirical studies involving cooperation and/or competition between the different players (actors)
2. Empirical studies in which more than two players are heterogeneous and asymmetric
3. Transferable Utility games and/or Non-Transferable Utility games
4. Multi-level decision problems. That is, best solutions are found by making several decisions at different stages, optimizing each of the sub-problems
5. Games with externalities (positive or negative), free-rider problems
6. Considering the possibility of different rules of coalition formation
7. Using concepts as Nash equilibria and stability analysis of coalitions
8. Different procedures on coalition formation

The starting points for each of the cases are from sources based on literature review. To illustrate cooperative/non-cooperative GT concepts/models we looked for models with a challenge on algorithmic development. In particular, for Chapter 2, we take the model from Eyckmans and Finus (2003a) where multiple coalition formation is considered. In the paper of Eyckmans and Finus (2003a), a case with six players was studied by doing calculation more or less by hand. However, larger games require to develop algorithms to solve it. In Chapter 3, we use the model from de Ridder and Rusinowska (2005) to apply two different procedures of coalition formation and to test several hypotheses. The model is multidimensional and with more than two heterogeneous players resulting into a computational challenge when using real data. One can illustrate the model in few dimensions and few players as was done in de Ridder and Rusinowska (2005). In order to apply the model to real data, it is necessary to develop OR methods.

To illustrate competition models, we looked for cases involving competition on location and competition on location and quantity (see e.g. Drezner and Hamacher (2004) and Nickel and Puerto (2005) for an overview on location theory). Chapter 4 is based on the model of Sarkar et al. (1997), a competitive game based on location-quantity decisions. They consider symmetric and fixed-entry at the markets, that is, all the firms (players) will enter the market. When considering asymmetric players and free-entry at the markets, algorithms for getting equilibria at quantities and locations are needed. Finally, Chapter 5 is based on the work of Drezner and Drezner (1998) with a follower and leader competing on facility location. The problem is a global optimisation problem and it has been solved for the follower problem in literature. However, as far as we know, the leader-follower problem has not been addressed with global optimisation methods giving a guarantee on finding the global optimum for the

leader. In summary, the objective is to contribute to these models extending the research either on the GT models and reformulation to OR models or the OR methods.

Solution methods are developed in order to find the most favourable solution to the objective defined by the players. OR involves the construction of mathematical models. Models are a collection of mathematical relationships representing aspects of the situation under study. We use concepts from GT to build models of the decision situation and apply techniques of OR to generate algorithms to obtain the optimal solution(s). Figure 1.2 represents how the conceptual model is combined with the case studies applied in this thesis.

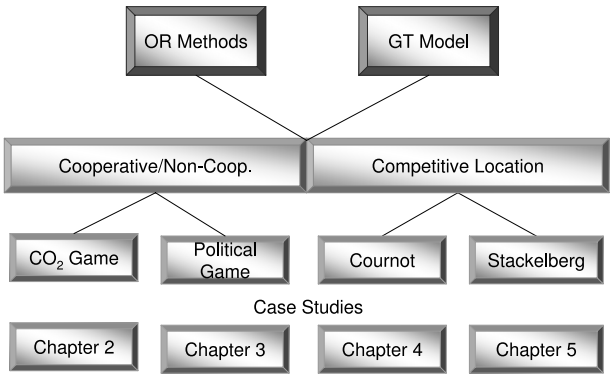

Figure 1.2. *Conceptual model and case studies*

From the conceptual model of the thesis research in Figure 1.1 and the characteristics of the research showed on Figure 1.2, an overview of the applied cases is shown in Tables 1.1, 1.2 and 1.3.

1.5.2 Outline of the Thesis

Chapter 2 describes and studies a multiple coalition formation game. Particular attention is paid to the analysis of the stability of coalitions under different membership rules. The idea is to analyse stability of multiple agreements based on the classical stability concept developed for cartel stability (see d'Aspremont et al. (1983)). Cartel stability restricts coalition formation to only one agreement. When multiple coalition formation is considered, the stability concept is more complex: for a signatory there should be no incentives to leave its coalition to join another one. Consequences of membership rules are modelled and studied.

Chapter 3 focuses on a model described in de Ridder and Rusinowska (2005) of multidimensional coalition formation in politics. In the model, a government consists of a majority coalition and a policy supported by this coalition. There are n parties trying to form a government. A formed government has a policy agreement represented in a m-multidimensional Euclidean policy space $\mathbb{R}^m$. The complexity increases with the number of parties n and

Table 1.1. *GT concepts used in this thesis*

	Chapter 2	Chapter 3	Chapter 4	Chapter 5
Complete information	X	X	X	X
Incomplete information				
Perfect information		X		X
Imperfect information	X		X	
Simultaneous moves	X	X	X	
Sequential moves		X		X
Zero sum				X
Non-zero sum	X	X	X	
Cooperative	X	X		
Non-cooperative	X	X	X	X
Asymmetric players	X	X	X	X
Symmetric players				
Heterogeneous players	X	X	X	X
Homogeneous players				
Transferable Utility	X			
Non-transferable Utility		X	X	X
Externalities	X			
Non externalities		X	X	X
Two-level decision making	X		X	
Stackelberg				X
Cournot			X	

Table 1.2. *OR models used in this thesis*

	Chapter 2	Chapter 3	Chapter 4	Chapter 5
Unconstrained				
Constrained	X	X	X	X
Continuous	X	X	X	X
Discrete	X	X	X	
NLP	X	X	X	X
Combinatorial Optimisation	X		X	
Static	X	X	X	X
Dynamic				
Convex Optimization	X	X	X	
GO				X
Bilevel				X

Table 1.3. *OR algorithms used in this thesis*

	Chapter 2	Chapter 3	Chapter 4	Chapter 5
Enumeration Algorithms	X		X	
Branch-and-Bound				X
Analytical Solutions	X		X	
Penalty approach		X		
NLP via SQP*		X		

* In Case 2 we use an external nonlinear optimisation algorithm using a *Sequential Quadratic Programming (SQP)* method.

the dimension of the policy spaces m. Given the number of parties n and policy dimension m, computational methods are necessary to compute all possible winning coalitions and preferences of parties over those (if many) coalitions. Furthermore, two ways of forming a government are considered: step-by-step and simultaneously.

Chapter 4 describes a competitive quantity-location "a la Cournot" two-level game between $n > 2$ heterogeneous firms and $m > 2$ markets. It extends the studies in Sarkar et al. (1997) and Rhim et al. (2003). The location space is a network, where the nodes are considered as possible locations for the firms. The number of firms entering the markets is not known in advance (as in Rhim et al. (2003)). A difference with Rhim et al. (2003) is that we consider asymmetric costs (firm-specific). Another difference is the procedure on how to find the equilibrium of the game. We consider not only the possibility the supplier to leave a market but also the possibility to move its facility to another location. Doing so, a firm has to re-think the quantity decision on how much to supply to which markets.

Chapter 5 describes a location decision of a new facility for two competing chains following the Stackelberg concept. It is a planar facility location problem where attractiveness to consumers depends on their distance to the location of the facility. The demand quantities are assumed to be known and fixed. There are two competitors (chains). First, the leader makes a decision on where to locate its facility in the plane. Second, the follower makes a decision with full knowledge of the decision of the leader. The objective of the leader is to maximize its market share after the entrance of the follower. The follower problem has been studied under deterministic consumer behaviour in Drezner (1994), Plastria (1997), using attraction functions of gravity type, and in Plastria and Carrizosa (2004) using different kinds of attraction functions.

Finally in Chapter 6, focus is on what the cases teach us with respect to the questions of applicability of methods; which methods can be applied in which situation. Results are reported between the existing theory and application of such a theory by developing OR methods and GT models to implement. The chapter includes answers to the research questions, discussions about the research, its limitations and further research.

CHAPTER 2

On the Computation of Stability in Multiple Coalition Formation Games

Chapter based on article: Sáiz, M.E., Hendrix, E.M.T. and Olieman, N. (2006), On the computation of stability in multiple coalition formation games, *Computational Economics*, 28 (3), 251-275.

Abstract

In non-cooperative models of coalition formation, players have to decide whether or not to participate in a coalition (alliance). Game theoretic analyses of the formation of alliances in games with externalities, stress the difficulties in designing self-enforcing treaties because of free-riding. The presence of a strong free-rider incentive prevents most alliances of being stable and/or effective. This paper focuses on computing stability in a game on multiple coalition formation with membership rules and different transfer schemes. A new mathematical programming notation for game theory concepts is outlined. To compute stability, the new notation is used for implementation into computer coding. Implementation and computation aspects are discussed. Numerical illustration of the algorithm shows that stability varies with the applied membership rules and transfer schemes. An application of coalition formation to International Environmental Agreements (IEAs) is provided.

2.1 Introduction

In daily life we observe cooperation in the form of club membership, alliances between companies and countries (sky alliance, NATO, EU) on all levels. Part of the incentives is economic; the signatories of the coalition consider membership a win-win situation over non membership. Recently in game theory the incentives for alliance or coalition formation is studied by modelling interaction of possible counterparts by coalition formation games (see e.g. Morasch (2000), Thoron (2000)). The research in this paper was inspired by negotiations at international level to cope with environmental problems, such as climate change.

Environmental problems have been the focus of negotiations at international level in the last decades. Using financial means to abate greenhouse gas emissions domestically by individual countries is inefficient from a cost perspective; some countries can abate more cheaply than others. This gives an economic drive to cooperate in the reduction of greenhouse gas emissions. The cooperation of countries and regions, such as observed in the Kyoto protocol, is based on many political aspects. Concepts from economy and game theory provide means for analysing whether *economic* incentives exist to form coalitions. The size of the coalition formed affects the size of the benefits that result from cooperation. The larger the number of participating countries the larger the benefits. Furthermore, countries outside of the agreement are going to benefit and it gives for some countries an strong incentive to free riding (externalities for nonmembers).

We analyse stability of multiple agreements based on the classical stability concept developed for cartel stability (see d'Aspremont et al. (1983)): A coalition is stable if, on the one hand, for a signatory there is no incentive to free-ride (internal stability) and on the other hand, for a non-signatory there is no incentive to take part in the coalition (external stability). Internal-External stability restricts coalition formation to only one agreement, i.e., countries have only one option to join a coalition. When multiple coalition formation is considered, the stability concept is more complex: for a signatory there should be no incentives to leave its coalition to join another one. Other features of coalition formation games are the membership rules. A difference between the rules of coalition formation lies in what can happen to the membership of a coalition once it has been formed: Can an existing coalition break apart, admit new members, or merge with other coalitions? (Topic is studied in Yi and Shin (1995)).

If countries are free to decide not only whether or not to be part of a coalition but also which coalition to join, there is generally more than one coalition at the equilibrium. This result is found in literature on coalition formation of economic coalitions, such as Ray and Vohra (1997) and (1999), Bloch (1995) and (1996), Yi and Shin (1995) and Yi (1997). These studies describe formation of multiple coalitions under different notions of stability and rules. Ray and Vohra (1997) and (1999) assume the "equilibrium binding agreement" rule which means that coalitions are allowed to break up into smaller sub-coalitions only. Bloch (1995) and (1996) consider an infinite horizon game and a coalition is formed only if all prospective members agree to form the coalition (coalition unanimity). Yi and Shin (1995) consider the "open membership" rule in which non signatories can join an existing coalition even without the consensus of the existing members. Yi (1997) examines endogenous coalition formation among symmetric players and analyses the stability of the grand coalition under different membership rules (open membership, coalition unanimity, equilibrium binding agreements). As a result, *different rules of coalition formation lead to different predictions about stable coalition structures* (Carraro and Marchiori (2003)). In our study, three different membership rules are used: open membership, exclusive membership with majority vote and exclusive membership with unanimity vote.

Finally, to enlarge the size of coalitions it is natural to propose economic transfers to

compensate those countries that may lose by signing the environmental agreement. Transfers can be paid from the gains of enlarging the coalition. Eyckmans and Finus (2003b) and (2003a) assume only transfers within coalitions where the surplus of a coalition from cooperation is allocated according to a proportional sharing rule applying to a numerical case with 6 regions. Botteon and Carraro (1997) analyse a cartel game (only one possible coalition) with 5 asymmetric world regions. In their study, payoffs, incentives to free-ride and incentives to broaden a stable coalition are computed for two different burden-sharing rules based on the Nash bargaining concept and the Shapley value concept. They use transfers for non-members to broaden a coalition. Transfers are only applied to "pre-transfers" stable coalitions. Bosello et al. (2003) and (2001) compare four burden-sharing criteria, the one implicit in the Kyoto protocol and three additional ones in a model with the world divided into 6 regions (203 coalition structures). This study shows that transfers can help broadening a given internally stable coalition.

The current paper describes the above developments and facilitates studying multiple coalition formation among players in a game. Particular attention is paid to the analysis of the stability of coalitions under different membership rules. The case of Eyckmans and Finus (2003a), is used for validation purposes to check the final implementation. Results of applying the suggested approach to the data described in Finus et al. (2005), gave new insight into economic incentives for coalition formation with respect to CO_2 emission reduction that are reported in Finus et al. (2004b). Our main contribution is the description of an algorithm to perform stability analysis of coalition structures in a multiple coalition formation game with different transfer schemes. An illustration of the algorithm is shown for a model with 12 players (more than four million of coalition structures). Furthermore, we do not only apply transfers to "pre-transfers" stable or internally stable coalition structures like Bosello et al. (2003) and (2001) do. The possibility to stabilize a coalition structure is studied.

This paper is organized as follows: in Section 2.2 a description of the definitions of stability in an open and exclusive membership game is given. In Section 2.3, a translation into mathematical programming notation of the concepts is elaborated. Moreover, the concept of neighbourhood is introduced for this formulation. In Section 2.4, the implementation used to compute stable coalitions in the game is discussed. In Section 2.5, results on stability analysis for a case with 12 players is shown. Conclusions can be found in Section 2.6.

2.2 Game Theoretic Background: Definitions and Concepts

Coalition games are part of *Cooperative Game Theory* where two or more players can cooperate to reach their goals. A *coalition structure* is a partition of the set of players N. Basic assumptions in situations where coalition games are applied are: the existence of binding agreements and the possibility of transferable payments between players. In this paper we focus on coalition formation in a two-level game. At the first level, players make a decision about cooperation with other players and form one or more coalitions; at the second level, a non-cooperative game is played given the coalition structure formed at the first level.

2.2.1 Definitions and Concepts

Let $N = \{1, \ldots, n\}$ be the set of players in the coalition game. Within coalition games one can distinguish between *Single Coalition Games* (*Cartel Games*), in which only one coalition can emerge, and *Multiple Coalition Games*, in which more than one coalition can be formed.

The *Cartel Game* (see e.g. Yi (1997)) assumes the existence of at most one coalition. Members of the coalition are called signatories and all other players non participating are called singletons or non-signatories. On the first level of this game, two membership strategies are available:

- strategy $\sigma_j = 0$: means player j is a non-signatory,
- strategy $\sigma_j = 1$: means player j is a signatory

In the *Multiple Coalition Game*, players decide on the first level one of the following strategies: to be member of a non-trivial coalition, (signatories); or can form a singleton coalition (non-signatories). Several coalitions may exist and players can choose which coalition to join. The Multiple Coalition Game has a larger membership strategy set than the Cartel Game.

We introduce formally the concept of a ***coalition structure*** $c = \{\kappa_1, \ldots, \kappa_i, \ldots, \kappa_m\}$ as a collection of coalitions κ_i of one or more players, with m the total number of coalitions in coalition structure c. In other words, a coalition structure $c = \{\kappa_1, \ldots, \kappa_m\}$ is a partition of the set of players, N, where a particular coalition is denoted by κ_i, $i \in \{1, \ldots, m\}$, $\kappa_i \cap \kappa_j = \emptyset$ for all $i \neq j$, $\cup \kappa_i = N$.

A coalition with only one player j, $|\kappa_j| = 1$, is called a **singleton** and the coalition where all players participate, $\kappa_1 = N$, is called the **Grand Coalition**. The two level game is defined as follows:

- Player $j \in N$ decides in which coalition κ_i to participate ($i \in \{1, \ldots, m\}$)
- At the second level, the coalitions are considered as a set of players who decide on their common strategy, such that the aggregate payoff (from now on called net benefit) of a coalition is maximized.

The input for level 2 is defined by coalition structure $c = \{\kappa_1, \ldots, \kappa_m\}$. The net benefit functions of level 2 for each coalition κ_i in the coalition structure c are defined by

$$NB_i^c(\underline{q}) = \sum_{j \in \kappa_i} \varphi_j(\underline{q})$$

where $\varphi_j(\underline{q})$ is the individual payoff for player j for strategy vector[1] $\underline{q}$. The Nash equilibrium profile strategy at the second level, $\underline{q}^*(c)$ depending on structure c, leads to an output of level 2 being a net benefit vector that can be translated into values of the payoffs of the individual players $j \in N$ possibly after applying transfers between the members of a coalition. When leaving out transfers, the payoff vector only depends on the output c of level 1 in the following way:

$$\underline{\Pi}(c) = \text{vector of payoff values following from level 2 under coalition structure } c \tag{2.1a}$$

[1] Some general remarks about syntax conventions in this paper: An underscore under a variable depicts a vector, such as $\underline{x}$. The i^{th} element of the vector $\underline{x}$ is denoted as x_i.

$$\Pi_j(c) = \varphi_j(\underline{q}^*(c)) \text{ payoff of player } j \text{ based on optimal strategy vector } \underline{q}^*(c) \tag{2.1b}$$

In the case of transfers, the individual payoff Π_j players receive is defined by a transfer scheme:

$$\Pi_j(c) = \varphi_j(\underline{q}^*(c)) + \xi_j, \quad \sum_{j\in\kappa} \xi_j = 0 \quad \forall \kappa \in c \tag{2.2}$$

where $\xi_j > 0$ means to receive a transfer and $\xi_j < 0$ means to pay a transfer. In the cases illustrated in Section 2.5, several alternative transfers schemes are used to establish values of $\Pi_j(c)$.

If the optimal strategy in the second stage is determined by a coalition structure c, then the entire coalition formation game reduces to one single stage.

2.2.2 Stability Concepts

In a single coalition game (Cartel Game), a distinction can be made between *internally* and *externally* stable (see Ray and Vohra (1997) and (1999), Finus et al. (2006) and (2005), Olieman and Hendrix (2005)). *Internally stable* means that there is no player *in* the coalition with the incentive to leave and become a singleton. *Externally stable* means, that there is no singleton player with the incentive to enter the coalition. In the Multiple Coalition Game we have more than one coalition and we have to be more careful defining stability.

We use the concept of *Inter-Coalition stability* (see Carraro (1999)). This concept means for a structure c^* that there is no player belonging to a coalition $\kappa_k \in c^*$ which would be better off by leaving the coalition to join another non-singleton coalition $\kappa_t \in c^*,\ t \neq k$.

Definition 3. ***Stability in a Multiple Coalition Game:***

- ***Internal Stability:*** *no cooperating player $j \in \kappa_k$ would be better off by leaving coalition $\kappa_k \in c^*$ to form a singleton κ_{m+1};*
- ***External Stability:*** *no singleton player j would be better off by joining any coalition $\kappa_t \in c^*$;*
- ***Inter-coalition Stability:*** *no player belonging to $\kappa_k \in c^*$, $|\kappa_k| > 1$, would be better off by leaving κ_k to join another coalition $\kappa_t \in c^*$.*

A coalition is called stable if it is as well internally stable, externally stable as inter-coalitionally stable. Note the following:

- A coalition structure without singletons is *externally stable.*
- In the definition of external stability we include the possibility that a singleton can join another singleton, that is, coalition $\kappa_t \in c^*$ in the definition above can be a coalition with only one player.

- In the definition of inter-coalition stability we assume that coalition $\kappa_k \in c^*$ has more than one player, that is, it is not a singleton.

In order to check stable coalition structures, we use stability concepts that capture the notion of open and exclusive membership (majority and unanimity voting):

Open Membership Game

The term *Open Membership* is used to indicate that for the current members of coalition κ_t, any other player is allowed to enter it.

Exclusive Membership Game

The exclusive membership in the single coalition game (Cartel Game) implies there is an additional condition to external stability: a non-coalition player is only allowed to join the existing coalition if the payoff for the existing coalition members will not decrease. In that case the members of the coalition will decide that the new member is allowed to join the coalition. If not, they will exclude the new member from the coalition.

Specifically one can consider two types of voting rules:

1. *Majority Voting*: if a player has an incentive to join coalition κ_t we have to check if the current members are better or worse off. If more than 50% are in favor of accession, then the candidate is accepted and the original coalition structure is not stable. We assume that if 1 is in favor and 1 against then accession is not accepted. The concept does not change when the 50% rule is changed to any other rule such as 66% or 75%.
2. *Unanimity Voting*: only if all coalition members are in favor of accession, then the candidate is allowed to enter. That is, if there is one player against (veto), then accession is denied.

The introduced concepts are elaborated further in the following section such that it is possible to compute whether coalition structures are stable for heterogeneous cases.

2.3 Mathematical Programming Notation and Stability Check

Implementation requires exact representation of concepts into computer coding. First the coding of the coalition structures is described. Then the idea of neighbourhoods is introduced to represent in an exact way the alternatives that players have on the first level. This is followed by a description of the stability checks.

2.3.1 Mathematical Programming Representation of Coalition Structures

For representing a coalition structure we now introduce the so-called **"Eyckmans notation"** (see Eyckmans and Finus (2003a)). The coalition structure in this notation is expressed as a string or vector of numbers ($\underline{c}$) indicating the number of the coalition to which a player belongs. A "1" indicates that a player is part of coalition "1"; a "2" indicates that a player is part of coalition "2"; and so on until the maximum number of possible coalitions "m = n". For example for six players:

$$\begin{aligned}\underline{c} = [1,2,2,3,4,2] \Rightarrow \quad & c = \{\{1\},\{2,3,6\},\{4\},\{5\}\} \text{ with} \\ & |\kappa_1| = 1, \kappa_1 = \{1\}, \\ & |\kappa_2| = 3, \kappa_2 = \{2,3,6\} \\ & |\kappa_3| = 1, \kappa_3 = \{4\} \\ & |\kappa_4| = 1, \kappa_4 = \{5\}\end{aligned}$$

That is: player $j = 1$ (called "1") is a singleton, κ_1 and $|\kappa_1| = 1$, players $j = 2,3,6$ form a coalition called "2", κ_2 and $|\kappa_2| = 3$, and players $j = 4$ and $j = 5$, called "3" and "4" respectively, κ_3, κ_4; $|\kappa_3| = 1$, $|\kappa_4| = 1$, are singletons. In this notation, the same coalition can be represented by:

$$\underline{c} = [2,1,1,3,4,1]$$

However, it is convenient if a coalition vector is represented uniquely. The idea is to **translate** alternative representations by re-assigning values to the elements of $\underline{c}$ in increasing order. The Grand Coalition, $c^{GC} = \{\kappa_1\}$, $|\kappa_1| = n$, in vector notation is represented by $\underline{c}^{GC} = [1,1,\ldots,1]$ and the coalition structure with only singleton members, $c^S = \{\kappa_1,\kappa_2,\ldots,\kappa_n\}$, $|\kappa_j| = 1$, for all j, in set notation, is given by $\underline{c}^S = [1,2,\ldots,n]$ in vector notation.

The total number L of all possible coalition structures is not known analytically, but can be derived numerically. A unique index function that maps $\underline{c}$ into $\{1,\ldots,l,\ldots,L\}$ to number the coalition structures is not straightforward either. However, the generated coalition structures in the (ordered) string notation defines an ordered list of numbers consisting of n digits. This gives the possibility to find a particular structure from the list by bisection in (at most) $^2\log L + 1$ steps. The number of multiple coalition structures derived numerically is[2]:

n	:	6	7	8	9	10	12
L	:	203	877	4140	21147	115975	4213597

One can determine all optimal payoff values $\Pi_j(\underline{c}_l)$ following (2.1) for all possible coalition structures c_l, $l = 1,\ldots,L$. As a result, we obtain a large $L \times n$ **payoff matrix $\mathbf{\Pi}$** of optimal payoff values for every individual player j when coalition structure l applies. The matrix can be used to study decisions on level 1.

2.3.2 Neighbourhood of Coalition Structures

Because of the different alternatives a player has in the multiple coalition game, we introduce some new concepts that are useful in the study of stability of coalition structures. In the Cartel Game (single coalition game) a player j has only one alternative action, either leave or stay in the coalition. Let m_l be the number of coalitions in coalition structure c_l. In the multiple coalition game, inside a coalition structure $c_l = \{\kappa_1,\ldots,\kappa_{m_l}\}$, each player j usually has more than one alternative:

[2]These numbers of partitions of a set of n elements are called Bell numbers B_n (Berge (1968)): $B_0 = B_1 = 1$, $B_{n+1} = \sum_{k=0}^{n} C_n^k B_k$

Case of a non-singleton coalition: Let player j be a member of a *non-singleton coalition* κ_i, $i \in \{1, \ldots, m_l\}$ in the coalition structure c_l. Player j has two possible alternatives to staying in the coalition κ_i: one alternative is to leave coalition κ_i to become a singleton forming a new coalition, κ_{m_l+1}, $|\kappa_{m_l+1}| = 1$; the other alternative is to leave coalition κ_i to join another coalition κ_t, $t \in \{1, \ldots, m_l\} \backslash \{i\}$.

Case of a singleton coalition: Now, suppose j forms a singleton coalition by itself κ_i, $|\kappa_i| = 1$. Player j has "only" one alternative: to join another coalition κ_t, $t \in \{1, \ldots, m_l\} \backslash \{i\}$.

The number of possible alternatives depends on the number of coalitions in the coalition structure. If there is more than one coalition κ_t, $t \in \{1, \ldots, m_l\}$ in the coalition structure c_l, then a player can join any other coalition except its current coalition, i.e. $m_l - 1$ coalitions and m_l possibilities if it is a singleton.

Given a coalition structure c_l, players can decide (strategy, level 1) on leaving a coalition and entering another. Here the stability definitions of the following section play a role. It is important to note that the deviation strategy of one player, while the others do not change their strategy, leads to a so-called **neighbour** coalition structure.

For the check of stability we need all the neighbours of c_l. The exact number of neighbours is not easy to identify in the multiple coalition game, in contrast to the cartel formation game. Let c_l be a coalition structure, $k(j)$ be the number of the coalition which j belongs to and let $\kappa_{k(j)}$ be that coalition. The neighbour coalition structures are represented by $c_l^{neig(j,t)}$, $j \in N$, $t \in \nu_{lj}$, with ν_{lj} the possible deviation index set of player j in structure c_l defined as follows:

- If coalition $\kappa_{k(j)}$ is formed by a singleton player j, $|\kappa_{k(j)}| = 1$, then the neighbours generated by player j are defined by the possible alternatives in the strategy set:

$$\nu_{lj} = \{1, 2, \ldots, m_l\} \backslash \{k(j)\} \quad \text{and} \quad |\nu_{lj}| = m_l - 1 \tag{2.3a}$$

- If j is non-singleton, $|\kappa_{k(j)}| > 1$, the possible alternatives in the strategy set are given by:

$$\nu_{lj} = \{1, 2, \ldots, m_l + 1\} \backslash \{k(j)\} \quad \text{and} \quad |\nu_{lj}| = m_l \tag{2.3b}$$

as j can also choose to proceed alone.

Note that each neighbour of the Grand Coalition corresponds to a Cartel Coalition Structure: a coalition formed by $n-1$ players and only one non-signatory player, a singleton.

Example 1. *Let* $\underline{c}_l = [1, 1, 2, 2, 3, 4, 3, 3, 2, 1, 1, 2]$ *be a coalition structure in string notation. For non-singleton player 1 the possible deviation index set is:*

$$k(1) = 1,\ \nu_{l1} = \{2, 3, 4, 5\}$$

and corresponding neighbour coalition structures are:

$$\begin{array}{lcl}
\underline{c}^{neig(1,2)} = [\mathbf{2}, 1, 2, 2, 3, 4, 3, 3, 2, 1, 1, 2] & = & [\mathbf{1}, 2, 1, 1, 3, 4, 3, 3, 1, 2, 2, 1] \\
\underline{c}^{neig(1,3)} = [\mathbf{3}, 1, 2, 2, 3, 4, 3, 3, 2, 1, 1, 2] & = & [\mathbf{1}, 2, 3, 3, 1, 4, 1, 1, 3, 2, 2, 3] \\
\underline{c}^{neig(1,4)} = [\mathbf{4}, 1, 2, 2, 3, 4, 3, 3, 2, 1, 1, 2] & = & [\mathbf{1}, 2, 3, 3, 4, 1, 4, 4, 3, 2, 2, 3] \\
\underline{c}^{neig(1,5)} = [\mathbf{5}, 1, 2, 2, 3, 4, 3, 3, 2, 1, 1, 2] & = & [\mathbf{1}, 2, 3, 3, 4, 5, 4, 4, 3, 2, 2, 3] \\
 & & \uparrow \\
 & & \textit{which after translating becomes}
\end{array}$$

For singleton player 6 the possible deviation index set is:

$$k(6) = 4,\ \nu_{l6} = \{1, 2, 3\}$$

and corresponding neighbour coalition structures are:

$$\begin{aligned}\underline{c}^{neig(6,1)} &= [1, 1, 2, 2, 3, \mathbf{1}, 3, 3, 2, 1, 1, 2]\\ \underline{c}^{neig(6,2)} &= [1, 1, 2, 2, 3, \mathbf{2}, 3, 3, 2, 1, 1, 2]\\ \underline{c}^{neig(6,3)} &= [1, 1, 2, 2, 3, \mathbf{3}, 3, 3, 2, 1, 1, 2]\end{aligned}$$

which do not require translation.

The **neighbourhood** of a coalition structure is defined by all the alternatives of all players. Due to overlap of (translated) neighbours, we only have an upper bound of the total number of neighbours of $\underline{c}_l$; it is not bigger than

$$\textstyle\sum_j |\nu_{lj}|$$

In the following section, we leave out the sub-index l and we only write c to denote a coalition structure, m for the total number of coalitions in c and ν_j for the deviation index set of a player j.

2.3.3 Mathematical Programming Notation for Stability Checks

Let us first summarise the notation introduced so far:

- $N = \{1, \ldots, n\}$: **set of players**;
- $c = \{\kappa_1, \ldots, \kappa_m\}$: **coalition structure** in set notation; $\underline{c} = [\kappa_1, \ldots, \kappa_m]$ in vector (string) notation;
- m: **number of coalitions in coalition structure** c.
- κ_i: **coalition** inside coalition structure c with one or more members, $i \in \{1, \ldots, m\}$, $m \leq n$;
- $k(j)$: **current coalition number of player** j;
- $\kappa_{k(j)}$: **coalition** inside coalition structure c to which player j belongs;
- ν_j: **set of alternative strategies of player** j in coalition structure c;

Example 2. *Consider coalition:*

$$\underline{c}^* = [1, 1, \underset{\substack{\uparrow\\ player\ 3}}{1}, 1, 2, 3, 2, 4, 1, 2, 3, 5]$$

A player, for instance player three, has six possible strategies and deviation set: $\nu_3 = \{2, 3, 4, 5, 6\}$ *. The first possible strategy is to stay in its current coalition 1; the strategies two to five are to leave the current coalition and join coalition number* i, $i \in \{2, 3, 4, 5\}$ *which is a subset of* ν_3*; the last possible strategy, number six, is to leave the current coalition and form a coalition number 6 to become a singleton; depending on what is the most favourable option. In the situation that none of the 12 players in* $\underline{c}^*$ *has the incentive to change the current situation, the coalition structure is defined stable.*

This implies the following criterion for determining the stability of a coalition: **Stability Criterion:** *if the payoff for each player in the current situation is higher than the payoff in all of the neighbour alternatives, then coalition structure* $\underline{c}^*$ *is stable.*

Example 3. *The* stability criterion *and the deviation index set* ν_3*, imply the coalition structure, with respect to player* 3*,*

$$\underline{c}^* = [1, 1, 1, 2, 3, 2, 4, 1, 2, 3, 5]$$

has to be compared with

- $\underline{c}^{neig(3,2)} = [1, 1, \mathbf{2}, 2, 3, 2, 4, 1, 2, 3, 5]$ *player 3 enters coalition 2*
- $\underline{c}^{neig(3,3)} = [1, 1, \mathbf{3}, 2, 3, 2, 4, 1, 2, 3, 5]$ *player 3 enters coalition 3*
- $\underline{c}^{neig(3,4)} = [1, 1, \mathbf{4}, 2, 3, 2, 4, 1, 2, 3, 5]$ *player 3 enters coalition 4*
- $\underline{c}^{neig(3,5)} = [1, 1, \mathbf{5}, 2, 3, 2, 4, 1, 2, 3, 5]$ *player 3 enters coalition 5(=m)*
- $\underline{c}^{neig(3,6)} = [1, 1, \mathbf{6}, 2, 3, 2, 4, 1, 2, 3, 5]$ *player 3 forms coalition 6(=m+1)*

The following can happen:

- *If option* $\underline{c}^{neig(3,t)}$ *does not result into a higher payoff for player 3, then player 3 will not enter coalition* t*, for all* $t \in v_3$*,*

There is no incentive to change anything as far as player 3 is concerned. The same applies to other players, where the player has possibly a different number of alternatives. If for each other player deviation will not result in a higher payoff, then it will not leave the coalition. Therefore: if in all of these cases the alternatives of the player does not give an expected improvement, no action will be taken and thus coalition structure $\underline{c}^*$ *is* stable*.*

Open Membership Stability:

In this section the Mathematical Programming Notation for checking stability of $c^* = \{\kappa_1, \ldots, \kappa_m\}$ is described. Let $t \in \nu_j$ be an index indicating the possible coalitions that j can join and $c^{neig(j,t)}$, the possible **neighbour coalition structure** arising from coalition structure c^* as a result of player j changing its status and forming coalition t (singleton) or joining another coalition t.

Consider player j changing its strategy from $k(j)$ to $t \in \nu_j$. This results into neighbour coalition structure:

1. If j is a singleton

$$c^{neig(j,t)} = \{\kappa_1, \kappa_2, \ldots, \kappa_t \cup \{j\}, \ldots, \kappa_{m-1}\}$$

2. If j is a non-singleton
   - either join other coalition κ_t

$$c^{neig(j,t)} = \{\kappa_1, \kappa_2, \ldots, \kappa_t \cup \{j\}, \ldots, \kappa_{k(j)} \backslash \{j\}, \ldots, \kappa_m\}$$

   - or form a singleton κ_{m+1}

$$c^{neig(j,t)} = \{\kappa_1, \kappa_2, \ldots, \kappa_t, \ldots, \kappa_{k(j)} \backslash \{j\}, \ldots, \kappa_{m+1}\}$$

   with

$$\kappa_{m+1} = \{j\}$$

Coalition structure c^* is defined:

internally stable in the open membership game if:

$$\begin{aligned} &\Pi_j(\underline{c}^*) \geq \Pi_j(\underline{c}^{neig(j,m+1)}) \\ &\text{for all } j \in N \text{ with } |\kappa_{k(j)}| > 1 \end{aligned} \tag{2.4}$$

externally stable in the open membership game if:

$$\begin{aligned} &\Pi_j(\underline{c}^*) \geq \Pi_j(\underline{c}^{neig(j,t)}) \\ &\text{for } t \in \nu_j \text{ and } j \in N \text{ with } |\kappa_{k(j)}| = 1 \end{aligned} \tag{2.5}$$

inter-coalitionally stable in the open membership game if:

$$\begin{aligned} &\Pi_j(\underline{c}^*) \geq \Pi_j(\underline{c}^{neig(j,t)}) \\ &\text{for } j \in N \text{ with } |\kappa_{k(j)}| > 1, \\ &t \in \nu_j \backslash \{m+1\} \text{ with } |\kappa_t| \geq 1 \end{aligned} \tag{2.6}$$

Exclusive Membership Stability:

In the multiple coalition game this rule depends on testing either external stability or inter-coalition stability. Consider player j wants to change its strategy from $k(j)$ to $t \in \nu_j$

- *Internal Stability*: in the exclusive membership game is still defined by equation (2.4).
- *External Stability*: Player j forms a singleton coalition by itself $|\kappa_{k(j)}| = 1$. If there is any member p in κ_t (unanimity) such that its payoff decreases, then player j is not allowed to join coalition κ_t.
- *Inter-coalition Stability*: Player j is a member of a non-trivial coalition, $|\kappa_{k(j)}| > 1$. If there is any member p in κ_t (unanimity) such that its payoff decreases, then player j is not allowed to join coalition κ_t.

The concept of majority voting in mathematical programming notation, requires introduction of a new binary variable yes/no symbol, δ_p, that tells us whether the current player in a neighbour coalition structure is against or in favour of another player entering:

$$\text{For } p \in \kappa_t : \begin{cases} \delta_p = 1 & \text{, if} \quad \Pi_p(\underline{c}^*) > \Pi_p(\underline{c}^{neig(j,t)}) \quad \text{(against)} \\ \delta_p = 0 & \text{otherwise.} \end{cases} \tag{2.7}$$

and the majority is against when:

$$\textstyle\sum_{p \in \kappa_t} \delta_p \geq |\kappa_t|/2$$

Table 2.1 defines more specifically the external and inter-coalitional stability of a coalition Structure c^* in the exclusive membership game.

Table 2.1. *External and Inter-coalition stability definition*

	External Stability	Inter-coalition stability
No voting rule	if for all j with $\|\kappa_{k(j)}\| = 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) \geq \Pi_j(\underline{c}^{neig(j,t)})$	if for all j with $\|\kappa_{k(j)}\| > 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) \geq \Pi_j(\underline{c}^{neig(j,t)})$
Majority Voting Rule	if for all j with $\|\kappa_{k(j)}\| = 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) < \Pi_j(\underline{c}^{neig(j,t)})$ and $\sum_{p\in\kappa_t} \delta_p \geq \|\kappa_t\|/2$	if for all j with $\|\kappa_{k(j)}\| > 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) < \Pi_j(\underline{c}^{neig(j,t)})$ and $\sum_{p\in\kappa_t} \delta_p \geq \|\kappa_t\|/2$
Unanimity Voting Rule	if for all j with $\|\kappa_{k(j)}\| = 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) < \Pi_j(\underline{c}^{neig(j,t)})$ and $\exists p \in \kappa_t$ such that $\delta_p = 1$	if for all j with $\|\kappa_{k(j)}\| > 1$, $t \in \nu_j$ $\Pi_j(\underline{c}^*) < \Pi_j(\underline{c}^{neig(j,t)})$ and $\exists p \in \kappa_t$ such that $\delta_p = 1$

The challenge is now to create a procedure to find for the first level game, where coalition formation takes place, stable coalition structures. Consequently, on the second strategy decisions level, for each structure the equilibrium of that level has to be determined. Such a procedure is outlined in the next section.

2.4 Description of the Algorithm

The mathematical programming formulation has been used to implement an algorithm for computing stability in a multiple coalition game. The larger the number of players is, the larger the number of coalition structures we must consider. As a consequence, the bigger the matrix with all necessary data to test stable coalition structures will be. For each coalition structure we determine:

- integer index for coalition structure.
- coalition structure in vector notation.
- payoff vector for each player in each coalition structure.
- three indicators 0-1 for internal, inter-coalition and external stability of the coalition structure (0 = non-stable; 1 = stable).

The algorithm consists of four steps. Steps 1 and 3 of the algorithm are case specific, the first step is the input data and the third step depends on the strategy at the second level of the game. To illustrate the algorithm, a *two-level* CO_2 emission game (as introduced in Finus et al. (2006)) is used. The players are world regions each with a particular benefit and cost structure on the abatement of CO_2 emissions. On the *first level* they decide simultaneously on their membership in a coalition; on the *second level* coalitions choose their abatement strategies. This choice depends on the coalition structure appearing on the first level.

The formulation starts by representing the abatement levels by a vector $\underline{q}$. Let N be the set of regions and $j \in N$ represent the region index. Let q_j be the individual abatement of region j, $B_j(\cdot)$ the benefit function from global abatement, $\sum_{j=1}^{n} q_j$, and $AC_j(\cdot)$ the abatement costs function from individual abatement q_j. The abatement decision on the

second level is based on the following payoff function:

$$\varphi_j(\underline{q}) = B_j(\sum_{j=1}^{n} q_j) - AC_j(q_j) \tag{2.8}$$

The complete algorithm consists of the following:

STEP 1 *Settings*: Region specific parameter values, used in calculation of the payoffs. Such parameters concern data about benefits, costs, abatement, payoff, emissions and concentration. We refer to Section 2.5.1 where values of this parameters are introduced.

STEP 2 *Generate*: In this step we generate all possible coalition structures in vector notation and save in matrix structure. To sort the structures, the string code is used as an integer index.

STEP 3 *Abatement*: calculate the optimal payoff for all possible coalition structures. The strategies are continuous decisions, $\underline{q}$, on abatement, reduction of emissions from a maximum level e_j^{max}, $0 \leq q_j \leq e_j^{max}$. The payoff for an individual region j in a coalition structure c is given by eq. (2.8), $\varphi_j(\underline{q}) = B_j(\sum_{r=1}^{n} q_r) - AC_j(q_j)$. Regions in a coalition decide on abatement to maximise

$$NB_i(\underline{q}) = \sum_{j \in \kappa_i} \varphi_j(\underline{q}) = \sum_{j \in \kappa_i} \left[B_j(\sum_{r=1}^{n} q_r) - AC_j(q_j) \right] \tag{2.9}$$

The payoff φ_j of region j and thus the aggregate net benefit NB_i^c of coalition i in coalition structure c, not only depends on its own strategy but also on those in other world regions. The game may now be seen as a *positive externality game*: payoff in region j increases with abatement in region p. Regions belonging to the same coalition maximize the aggregate net benefit of their coalition.

$$\max_{\substack{q_j \\ j \in \kappa_i}} NB_i(\underline{q}), \qquad i = 1, \ldots, m \tag{2.10}$$

The optimum is found by solving the set of equalities determined by the first order conditions:

$$\frac{\partial NB_i(\underline{q})}{\partial q_j} = 0 \qquad j \in \kappa_i \ , \quad i = 1, \ldots m$$

if the optimum is interior with respect to $0 \leq q_j \leq e_j^{max}$.
The equilibrium abatement strategy vector $\underline{q}^*$ for coalition structure $\underline{c}$ is derived as a Nash equilibrium between coalitions. The outcome of this step consists of a payoff matrix Π which contains all individual payoffs $\Pi_j(c_l)$ for all possible coalition structures, $\underline{c}_l$:

$$\Pi = \begin{bmatrix} \underline{\Pi}_1 \\ \underline{\Pi}_2 \\ \ldots \\ \underline{\Pi}_L \end{bmatrix} = \begin{bmatrix} \Pi_1(c_1) & \ldots & \Pi_j(c_1) & \ldots & \Pi_n(c_1) \\ \Pi_1(c_2) & \ldots & \Pi_j(c_2) & \ldots & \Pi_n(c_2) \\ \ldots & & & & \\ \Pi_1(c_L) & \ldots & \Pi_j(c_L) & \ldots & \Pi_n(c_L) \end{bmatrix} \tag{2.11}$$

STEP 4 *Stability*: main procedure. Now we proceed to look for those coalitions which are stable, that is, internally, inter-coalitionally and externally stable. The notation introduced in the last sections helps us in the implementation of this procedure. The

algorithm checks stability as follows: for each structure c_l and for each region j look to neighbourhood, $c_l^{neig(j,t)}$, $t \in \nu_{lj}$, and compare $\Pi_j(c_l)$ with $\Pi_j(c_l^{neig(j,t)})$ for all $t \in \nu_{lj}$.

2.5 Illustration of the Algorithm

Eyckmans and Finus (2003a) elaborate a case in a CO_2 emission game with multiple coalitions for 6 regions. This results in 203 possible coalition structures. For validation purposes of our algorithm the matrix with all possible payoffs (step 3, 2.11) has been used as input for checking stability in step 4. Cases applied are with and without transfers. Only in the exclusive membership game stable coalitions appear. With our implementation the number of stable, internally, inter-coalitionally and externally stable coalition structures are exactly the same as in their study either in the open, exclusive majority or exclusive unanimity.

A 12 region case based on data from a cartel game is shown in Olieman and Hendrix (2005). In this game a number of 4 thousand coalitions has been checked. This case is extended here to allow multiple coalition formation. This requires the determination of more than 4 million coalition structures. The source of the data can be found in Finus et al. (2005) and Dellink et al. (2003). They set up an empirical model, the so-called *STA*bility of *CO*alition model, STACO. For emissions in 2010 they choose the value of the DICE model (see Nordhaus (1994)), which amounts to 11.96 gigatons CO_2. In Sáiz et al. (2004) the data of that study are used to derive results for a 6, 8 and 10 region case. The regions distinguished in these studies are: USA (USA), Japan (JPN), European Union (EEC), other OECD countries (OOE), Eastern European countries (EET), former Soviet Union (FSU), energy exporting countries (EEX), China (CHN), India (IND), dynamic Asian economies (DAE), Brazil (BRA) and "rest of the world" (ROW) (see Finus et al. (2006) and (2005)). The algorithm checks the existence of stable coalition structures in cases without transfers and with four different transfer schemes.

2.5.1 Parameters of the Payoff Function

With respect to the parameter values of the payoff function, the analysis of Finus et al. (2005) is used. It starts in 2010 and covers a time period of 100 years in order to capture the long-run effects of the global warming problem. Benefits are expressed in the form of discounted reduced damages due to accumulated abatement over the entire period, $q = \sum_{t=2011}^{2110} q_t$. They come to $B(q) = 37.40q$ where allocation of global benefits from reduced environmental damages to the various world regions is based on the assumption:

$$B_j(q) = s_j B(q) \tag{2.12}$$

where s_j is the share of region j.

For the *Abatement Cost Function*, they assume an annual abatement cost function of the shape:

$$AC_{jt}(q_{jt}) = \frac{1}{3}\alpha_j(q_{jt})^3 + \frac{1}{2}\beta_j(q_{jt})^2$$

where simply q_{jt} is taken as $q_j/100$ assuming stationary strategies ($q_{j,2011} = \ldots = q_{j,2110}$). In the model, abatement means emission reduction with respect to (business-as-usual-scenario)-emissions. A total initial emission of 11.96 gigatons is allocated to the 12 regions. The total

abatement cost for region j is the discounted sum over $t = 2011, \ldots, 2110$ leading to

$$AC_j(q_j) = 43.1 AC_{jt}(q_{jt}) \tag{2.13}$$

Table 2.2. *Parameters of STACO model for eqs. AC_j and B_j*[a]

Regions j	Emissions in 2010 $E_{j,2010}$ *(Gigatons)*	Share of global benefits s_j	Abatement cost parameter α_j	Abatement cost parameter β_j
1 USA	2.416	0.2263	0.0005	0.00398
2 JPN	0.557	0.1725	0.0155	0.18160
3 EEC	1.399	0.236	0.0024	0.01503
4 OOE	0.621	0.0345	0.0083	0
5 EET	0.519	0.013	0.0079	0.00486
6 FSU	1.003	0.0675	0.0023	0.00042
7 EEX	1.219	0.030	0.0032	0.03029
8 CHN	2.356	0.062	0.00007	0.00239
9 IND	0.639	0.050	0.0015	0.00787
10 DAE	0.405	0.0249	0.0047	0.03774
11 BRA	0.128	0.0153	0.5612	0.84974
12 ROW	0.698	0.068	0.0021	0.00805
WORLD	$\sum$=11.96	$\sum s_j$=1		

[a] Eqs 2.12, 2.13

Table 2.2 shows parameter values about:

- Share of total of Emissions in 2010 in Gigatons, $E_{j,2010}$
- Share of Global Benefits, s_j. The sum is equal to 1.
- Abatement cost parameter α_j
- Abatement cost parameter β_j

that are used in the 12 region case.

2.5.2 STACO Analysis without Transfer Schemes

The payoff for the regions is defined as in eq. (2.1), $\Pi_j(c) = \varphi_j(\underline{q}^*(c))$. This payoff is used to compute the matrix (2.11). A Fortran implementation of the algorithm generates the 4.213.597 possible coalition structures and performs the stability checks as outlined on Section 2.3.3. Table 2.3 depicts the number of stable structures classified towards the different definitions of stability. Only in the exclusive membership game where members can apply a veto for other regions not to enter, stable structures appear.

The stable coalition structures and their corresponding monetary values are listed in Table 2.4[3] and are interpreted further in Finus et al. (2004b). The results on payoff, called net benefit here, can be used to analyse the economic incentive for coalitions to appear. One can observe for example that the gains from cooperation are large comparing the "grand coalition" with the "all singletons" coalition. However, neither the "old" and "new" Kyoto coalition structures are stable. The "largest" coalition structure in the multiple coalition game can achieve less in terms of Net Benefit and in terms of Global Emission Reduction than the "old" and "new" Kyoto coalition structure. USA and Japan are not members of any

[3]Coalition structures in tables are in a short notation, i.e, with only the regions in a coalition.

Table 2.3. *Number of stable coalition structures based on stability definitions. STACO Model without transfers*

	OM [a]	EM-MV [b]	EM-UV [c]
Stable	0	0	8
Internally Stable	98	98	98
Intercoalitionally Stable	7	1,834,950	3,922,082
Externally Stable	988,476	1,619,763	2,681,807

[a] Open Membership Game

[b] Exclusive Membership Game, Majority Voting Rule

[c] Exclusive Membership Game, Unanimity Voting Rule

stable coalition and for Brazil the incentive to cooperate is bigger. In reality there may be many other reasons for coalitions to be formed. The model only predicts economic viable possibilities for coalition formation.

Table 2.4. *Stable Coalition Structures STACO Model, case without transfers*

Coalition Structure	OM[1]	EM-MV[2]	EM-UV[3]	[a] Net Benefit	[b] Annual Emission Reduction
grand coalition	n	n	n	6,031	21.4
old Kyoto coalition: {USA,JPN,EEC,OOE,EET,FSU}	n	n	n	3,140	8.9
new Kyoto coalition : {JPN,EEC,OOE,EET,FSU}	n	n	n	2,692	6.9
{OOE,IND,BRA},{FSU,ROW},{EEX,CHN}	n	n	y	2,452	5.8
{FSU,BRA,ROW},{OOE,IND},{EEX,CHN}	n	n	y	2,451	5.8
{OOE,IND,BRA},{FSU,ROW},{CHN,DAE}	n	n	y	2,415	5.7
{FSU,BRA,ROW},{OOE,IND},{CHN,DAE}	n	n	y	2,414	5.7
{OOE,IND,BRA},{FSU,ROW},{EEX,DAE}	n	n	y	2,263	5.4
{FSU,BRA,ROW},{OOE,IND},{EEX,DAE}	n	n	y	2,262	5.4
{FSU,BRA,ROW},{OOE,DAE}{EEX,IND}	n	n	y	2,261	5.4
{FSU,BRA,ROW},{OOE,EEX},{IND,DAE}	n	n	y	2,255	5.3
all singletons	n	n	n	1,960	4.6

[1] Open Membership Game

[2] Exclusive Membership Game, Majority Voting Rule

[3] Exclusive Membership Game, Unanimity Voting Rule

[a] Net Benefit expressed in billion US dollar over 100 years

[b] Global Emission Reduction expressed in percentage from emissions as listed in Table 2.2 over 100 years

y = stable, n = not stable

2.5.3 STACO Analysis applying Transfer Schemes

The payoff function in case of transfers for a region j has already been introduced in Section 2.2.1, eq. (2.2). Transfers, ξ_j, are only paid between coalition members and $\sum_{j\in\kappa_{k(j)}} \xi_j = 0$. Let c^S be the singleton coalition structure, $\underline{c}^S = [1, 2, \ldots, 12]$ in vector notation or $c^S = \{\kappa_1, \kappa_2, \ldots, \kappa_{12}\}$ in set notation and c a coalition structure. Transfers for a region j in coalition $\kappa_{k(j)}$ in coalition structure c are assumed to be of the following form:

$$\xi_j = \varphi_j(\underline{q}^*(c^S)) - \varphi_j(\underline{q}^*(c)) + \lambda_j \sum_{p\in\kappa_{k(j)}} \Big[\varphi_p(\underline{q}^*(c)) - \varphi_p(\underline{q}^*(c^S))\Big] \tag{2.14}$$

where $0 \leq \lambda_j \leq 1$, $\sum_{j\in\kappa_{k(j)}} \lambda_j = 1$ are weights in the following sense: to each region a portion of the gain from cooperation is given. The gains from cooperation are measured as the difference between the Net Benefit in the coalition $\kappa_{k(j)}$ in coalition structure c and the Net Benefit of coalition $\kappa_{k(j)}$ in the all singletons coalition structure, c^S. The larger this weight, the higher the share is which a coalition member receives from the gains from cooperation. Using $\widehat{\Pi}_j(c)$ for the payoff with transfers and $\Pi_j(c)$ for the payoff without translates (2.14) into

$$\widehat{\Pi}_j(c) = \Pi_j(c^S) + \lambda_j(c) \sum_{p\in\kappa_{k(j)}} \Big[\Pi_p(c) - \Pi_p(c^S)\Big] \tag{2.15}$$

Table 2.5. *Weights of four transfers schemes* $\lambda_j(c^{GC})$ [a]

Regions	Equal Sharing	Population	Gross Domestic Products	Ability to Pay
	$\frac{1}{\lvert\kappa_i\rvert}$	$\frac{POP_j}{\sum_{p\in\kappa_{k(j)}} POP_p}$	$\frac{GDP_j}{\sum_{p\in\kappa_{k(j)}} GDP_p}$	$\frac{[GDP_j/POP_j]^{-1}}{\sum_{p\in\kappa_{k(j)}} [GDP_p/POP_p]^{-1}}$
	(1)	(2)	(3)	(4)
USA	8.3	4.8	27.0	0.5
JPN	8.3	1.9	17.0	0.3
EEC	8.3	5.8	29.2	0.6
OOE	8.3	2.2	5.8	1.1
EET	8.3	1.9	1.2	4.4
FSU	8.3	4.5	1.5	8.2
EEX	8.3	24.9	5.0	14.8
CHN	8.3	20.9	3.1	18.5
IND	8.3	17.8	1.4	37.0
DAE	8.3	3.2	3.0	3.2
BRA	8.3	3.0	2.4	3.6
ROW	8.3	9.1	3.4	7.8
Total	100	100	100	100

[a] All figures are expressed as a percentage and rounded to the first digit. Base data for computations are taken from Altamirano-Cabrera and Finus (2006).

Table 2.5 summarise the weights for the grand coalition, $c^{GC} = [1, 1, \ldots, 1]$, $\underline{c}^{GC} = \{\kappa_1\}$, for the four different transfer schemes we consider. For any other coalition c_l , weights $\lambda_j(c)$ are given by $\lambda_j(c) = \lambda_j(c^{GC}) / \sum_{t\in\kappa_{k(j)}} \lambda_j(c^{GC})$.

We only briefly comment on the four rules and refer the reader for a more comprehensive motivation to Finus et al. (2004b). "Equal Sharing" implies that each participant receives the same weight. Given the fact that regions are very heterogeneous in many respects, it is of course debatable whether equal sharing really implies equal and "fair" treatment. The transfer scheme "Population" acknowledges that all people should benefit equally from cooperation: "one man one vote". Evidently, energy exporting countries (EEX), China (CHN) and India (IND) receive high shares since many people live in these regions. The transfer scheme "Gross Domestic Product" belongs to the so-called sovereignty rules because they more or less preserve the current status of wealth. (GDP=The monetary value of all the goods and services produced by an economy over a specified period. This is an indicator of the economic health of a country). Hence, USA, Japan (JPN) and the European Union (EU) receive high shares. "Ability to Pay" allocates the gains from cooperation inversely to welfare (measured as GDP) per capita. Hence, this rule may be seen as a vehicle of development aid

through environmental policy. Again, those regions that receive high shares are also those that are the beneficiaries under the transfer scheme "Population".

Table 2.6 depicts the number of stable structures classified towards the different definitions of stability. With transfer schemes stable coalition structures appear when an exclusive membership game is applied with either a majority voting rule or unanimity voting rule.

Table 2.6. *Number of stable coalition structures based on stability rules. STACO Model with transfers*

Transfer Scheme	Stability	OM [a]	EM-MV [b]	EM-UV [c]
Gross Domestic Product	Stable	0	7	7
	Internally Stable	902	902	902
	Inter-coalitionally Stable	11	83,993	84,044
	Externally Stable	920,685	1,539,538	1,540,709
Per Capita	Stable	0	1	1
	Internally Stable	240	240	240
	Inter-coalitionally Stable	7	52,434	52,472
	Externally Stable	878,795	2,032,892	2,033,430
Equal Sharing	Stable	0	0	0
	Internally Stable	227	227	227
	Inter-coalitionally Stable	4	43,716	43,716
	Externally Stable	868,952	1,927,135	1,927,139
Ability to Pay	Stable	0	0	0
	Internally Stable	96	96	96
	Inter-coalitionally Stable	6	35,613	35,638
	Externally Stable	872,778	2,318,912	2,319,735

[a] Open Membership Game

[b] Exclusive Membership Game, Majority Voting Rule

[c] Exclusive Membership Game, Unanimity Voting Rule

Running the algorithms for different transfer schemes implies generating the corresponding $\widehat{\Pi}$ matrices (as in 2.11). Table 2.6 summarises the results of the STACO model with the transfer schemes used. Table 2.7 shows the stable coalition structures. Stable coalition structures are "better" in the sense of net benefit and global emission reduction. Moreover, transfer schemes make possible coalitions structures stable with coalition members as USA and Japan which were not coalition members in the no transfer case. It also appears that the transfer scheme "Gross Domestic Product" is more profitable than the others. More economic interpretation can be found in Finus et al. (2004b).

Results show that transfer schemes are useful to be implemented. Computation with membership rules generate different results with and without transfer schemes. In the cases without transfers, the algorithm only found stability when an exclusive membership game with unanimity voting rule is applied. This result is different when transfers are applied; either majority and unanimity voting are sufficient to stabilise coalition structures. With respect to the number of stable coalition structures, there are some transfer schemes which do not generate stable coalition structures. Transfer schemes are anyway interesting in terms of net benefit and global emission reduction. Furthermore, some of the regions considered that are "not interested" in cooperation without transfer schemes, became interested in forming a coalition when transfers appear.

2.5.4 Implementation Aspects

All results are obtained using a simple processor Pentium IV. The implementation procedures are written in Matlab. The main program, *run.m*, calls sequentially the other subprograms,

Table 2.7. *Stable Coalition Structures STACO Model, case with transfers*

Coalition Structure	OM[1]	EM-MV[2]	EM-UV[3]	[a] Net Benefit	[b] Annual Emission Reduction
grand coalition	n	n	n	6,031	21.4
old Kyoto coalition: {USA,JPN,EEC,OOE,EET,FSU}	n	n	n	3,140	8.9
new Kyoto coalition: {JPN,EEC,OOE,EET,FSU}	n	n	n	2,692	6.9
Gross Domestic Product					
{USA,DAE,BRA},{JPN,EET},{EEC,CHN},{OOE,EEX}	n	y	y	3,278	8.1
{USA,CHN},{JPN,EET},{EEC,DAE},{OOE,EEX}	n	y	y	3,249	8.0
{USA,DAE},{JPN,EET},{EEC,CHN},{OOE,EEX}	n	y	y	3,249	8.0
{USA,OOE,EEX},{EEC,DAE,BRA},{JPN,EET},{CHN,ROW}	n	y	y	2,992	7.3
{USA,DAE,BRA},{EEC,OOE,EEX},{JPN,EET},{CHN,ROW}	n	y	y	2,980	7.2
{USA,OOE,EEX},{EEC,DAE},{JPN,EET},{CHN,ROW}	n	y	y	2,973	7.2
{USA,DAE},{JPN,EET},{EEC,OOE},{CHN,ROW}	n	y	y	2,951	7.2
Per Capita					
{USA,EEC},{OOE,FSU},{EET,ROW},{CHN,IND}	n	y	y	2,795	6.9
Equal Sharing					
There are no coalition structures stable					
Ability to Pay					
There are no coalition structures stable					
all singletons	-	-	-	1,960	4.6

[1] Open Membership Game

[2] Exclusive Membership Game, Majority Voting Rule

[3] Exclusive Membership Game, Unanimity Voting Rule

[a] Net Benefit expressed in billion US dollar over 100 years

[b] Global Emission Reduction expressed in percentage from emissions as listed in Table 2.2 over 100 years

y = stable, n = not stable

representing each one part of the algorithm. The subprograms are:

- step 1 of the algorithm: *settings*
- step 2 of the algorithm: *generate*
- step 3 of the algorithm: *abatement*
- step 4 of the algorithm: *stability*

The original Matlab code was translated to Fortran code with the aim to speed-up calculation and improve memory use. The total CPU times obtained with the Fortran code are displayed in Table 2.8. The table shows the development of the total number of coalition structures and the time to process them when the number of regions increases from 6 to 12.

Table 2.8. *CPU Times of data processing with increasing size of the case*

Cases StaCo Model [a]		**6 regions**	**8 regions**	**10 regions**	**12 regions**
Total Number of Coalition Structures		203	4.140	115.975	4.213.597
CPU Times Per Case (Seconds)					
OM [1]		0,001	0,453	27,594	1.894,641
	MV [2a]	0,016	0,500	29,312	1.987,078
EM [2]					
	UV [2b]	0,016	0,484	28,516	1.953,750

[a] (Finus et al. 2005)

[1] Open Membership [2] Exclusive Membership

[2a] EM Majority Voting [2b] EM Unanimity Voting

2.6 Conclusions

The concept of the multiple coalition game as outlined in among others Carraro (1999), Eyckmans and Finus (2003a), Ray and Vohra (1997) and (1999), has been reformulated in a mathematical programming language. The new vector notation and neighbourhood notation allows implementation in an algorithmic context.

We showed that the implementation provides the feasibility to study a multiple coalition game with 12 asymmetric regions. We compute more than 4 million coalition structures and compared the data handling for stability purposes within 1.950 seconds (approx. 1/2 hour). To validate the algorithm, the case of Eyckmans and Finus (2003a) has been implemented and the algorithm generates the same result as found in their study with 6 regions.

The research in this paper shows that the mathematical re-definition of existing concepts and efficient Fortran coding, generates relevant results for a relative large case. To our knowledge, empirical results for comparable cases were not reported before in the theory of coalition formation.

Finally, based on earlier studies and results applying the STACO model (Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Olieman and Hendrix (2005)), this paper shows in a numerical study that the number of stable coalition structures in a multiple coalition game is bigger than in a cartel game and preferable if we are measuring in terms of either Net Benefit or Global Emission Reduction.

CHAPTER 3

On Computing Negotiation Positions and Preferences in a Spatial Coalition Model

Chapter based on: de Ridder A., Rusinowska A., Sáiz M.E. and Hendrix E.M.T. (2007), Coalition formation: the role of procedure and policy flexibility. *Submitted* and Sáiz, M.E., Hendrix, E.M.T., de Ridder, A. and Rusinowska,A. (2007), On the computation of Negotiation Positions and Preferences in a Spatial Coalition Model, *Mansholt Working Paper WMP*

Abstract

Negotiations to form a coalition in politics appear in any parliamentary democracy. Many studies on literature deal with coalition formation games. Starting point of this paper is based on a model on political coalition formation. In this model, two different procedures of coalition formation between political parties are considered. In the first approach, a step-by-step procedure is used and new members are added one-by-one. In the second approach, a simultaneous procedure is applied in which members in a coalition decide and negotiate simultaneously. Furthermore, when the players are political parties, many different decision variables play a role in the game. A government is defined as consisting of a majority coalition and a policy supported by this coalition. Because of the different party positions on different topics, a multidimensional decision space is considered in which each party has an ideal position and the coalition policy is formed. When considering multidimensional space and a large number of parties, computational methods become an important tool to find which stable government(s) is(are) in equilibrium. We analyse and develop computational algorithms for both procedures. Different cases in political games are used to illustrate the methods and data is used to test hypotheses on coalition formation.

3.1 Introduction

The topic of coalition formation is widely studied in literature. This paper focuses on a model described in de Ridder and Rusinowska (2005) of multidimensional coalition formation in politics. In the model, a government consists of a majority coalition and a policy supported by this coalition. There are n parties trying to form a government. A formed government has a policy agreement represented in a m-multidimensional Euclidean policy space $\mathbb{R}^m$. The complexity increases with the number of parties n and the dimension of the policy spaces m. Given the number of parties n and policy dimension m, computational methods are necessary to compute all possible winning coalitions and preferences of parties over those (if many) coalitions. Furthermore, two ways of forming a government are considered: step-by-step and simultaneously. Each procedure requires a specific algorithm.

In de Ridder et al. (2007) the model is used to show that procedure plays an important role in reaching a coalition agreement and that political parties do not necessarily benefit from being a first-mover. Moreover, that study shows that a decrease in a party's flexibility can be beneficial in coalition negotiations. Hypotheses on power sharing tactics are also investigated. In the current paper, we develop methods to study the two dynamical aspects of coalition formation (procedure and policy flexibility) and report on the findings for testing hypotheses by analysing the formal model and deducing implications from this model based on real-life data. The computational aspects of both the model and the empirical test are discussed.

In Section 3.1.1, the model is embedded in literature on application of Game Theory in political coalition formation. In Section 3.1.2, theoretical backgrounds about the multidimensional spatial coalition formation model are outlined. In Section 3.1.3, the procedures to form a coalition and complexity of both are introduced. In Section 3.2, the algorithms for two different procedures of coalition formation are described. In Section 3.3, hypotheses are formulated and checked based on Dutch political data. Finally, conclusions are drawn in Section 3.4.

3.1.1 Game Theory in Political Coalition Formation

In multi-party democracies, political parties have to form coalitions to achieve majority governments. As a part of coalition negotiations, coalition members bargain and agree on a package of policy agreements, the coalition agreement (see Timmermans (2003) for an extensive discussion on coalition agreements). In 63% of the coalition formations in Western-Europe studied by Müller and Strøm (2003), such coalition agreements were reached (in e.g. Austria, Ireland, Belgium, and The Netherlands). In order to reach such a coalition agreement, parties in the coalition will have to make compromises as each party has its own ideal policy. Only by adjusting their policy positions, parties can reach the compromise needed for the coalition agreement.

An important subject is the procedure used to reach a coalition. Roughly speaking, two different ways of coalition formation can be discerned: a step-by-step or hierarchical procedure versus a simultaneous or non-hierarchical procedure (Laver and Schofield (1990)). The step-by-step approach sees coalition formation as a process in which the group incrementally forms: new members are added gradually. An alternative approach is to negotiate immediately with *all* the members of the coalition, as in a simultaneous procedure. In spite of these two different procedures which are recognized in the literature and which both occur in real life coalition formation, little attention has been paid to the consequences of these procedures for the result of coalition formation. Some earlier theoretical results show that procedure plays an important role in coalition formation and that, except for some special

situation, different procedures lead to different results (de Ridder and Rusinowska (2005)). The special conditions require that the ideal positions of the players are really close, which is unrealistic in a political setting. The model introduced in de Ridder and Rusinowska (2005) is positioned among spatial coalition models (based on Downs (1957), see e.g. Grofman (1982); Laver and Shepsle (1996)) and has been applied to alliance formation between firms.

The field of research of formal coalition models is large and extensive, see, amongst others, Axelrod (1970), Vries (1999), Martin and Stevenson (2001), Grofman (1982), Laver and Shepsle (1996), van Deemen (1989), von Neumann and Morgenstern (1944), and Warwick (1998). So far, most of those studies have focused on why coalition form and, based on that, which parties will cooperate. Arguments for coalition formation were found in power, policy, or institutional arguments. However, the strategy and process of coalition formation have been ignored in the literature (Laver and Schofield (1990): how will coalitions be formed, and, what is the best strategy for a party during the process of coalition formation? Also, from a more formal theoretical point of view, several authors have pointed at this lack of dynamics in the models (Arnold and Schwalbe (2002); Tohmé and Sandholm (1999); van Deemen (1997)). It seems unnatural to analyse coalition formation with a static approach, since coalition formation is clearly dynamic in nature: for example, parties need a few weeks, sometimes months, to reach a coalition agreement, different procedures are used to form a coalition, and parties move their positions to be able to compromise. The suggestion that process plays a role in coalition formation - and should thus be included as an explanatory variable - is strengthened by earlier research (Austen-Smith and Banks (1988); Baron (1993); Bloch (1996); Brams et al. (2005); de Ridder and Rusinowska (2005)). This earlier research has not evolved towards a coherent and empirically verified stream of research, and, moreover, the role of procedure has been ignored.

3.1.2 The Model

We deal with the following model of spatial coalition formation, considered in de Ridder and Rusinowska (2005). There are n players, here political parties, which try to form a majority coalition S and to decide about a policy of the coalition x_S hereafter called the coalition position. This coalition position is the formal representation of the policy agreement of a coalition. Party $i \in N$, where N denotes the set of all parties, has a weight $w_i > 0$, which is based on the number of seats in parliament party i possesses.

Each party i may choose a policy position x_i from an m-multidimensional Euclidean policy space $\mathbb{R}^m$, $m \geq 1$. A distance between two positions $x_i = (x_{i1}, ..., x_{im})$ and $x_j = (x_{j1}, ..., x_{jm})$ is given by

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{m} (x_{ik} - x_{jk})^2}. \tag{3.1}$$

Parties have a certain amount of flexibility on the policy positions, i.e., they have their preferences defined in $\mathbb{R}^m$. Each player $i \in N$ is assumed to have an ideal position $x_i^* \in \mathbb{R}^m$, which is the most preferred position of party i, and a maneuvering space, an equivalent of the policy horizon by Warwick (2000), which consists of all positions acceptable to party i. The model assumes the maneuvering space to be a ball in $\mathbb{R}^m$. M_i denotes the maneuvering space of party i with middle point x_i^* and radius r_i, i.e.,

$$M_i = \{y \in \mathbb{R}^m \mid d(x_i^*, y) \leq r_i\}. \tag{3.2}$$

The maneuvering space of a party is then the set of policy positions with distances from the ideal position of the party not greater than the radius. Of course, some positions are more preferred to a party than others. Preferences of a party on positions are expressed by the following rule: the closer a position is to the ideal position of a party, the more preferred this position is to the party.

Given coalition $S \subseteq N$ and the ideal positions x_i^* for $i \in S$, all parties of the potential coalition S have to agree on a coalition position for S. We consider two alternative procedures for forming a coalition and choosing a coalition position for that coalition. Although the procedures differ from each other, there are two common assumptions for these procedures. First of all, it is assumed that no party will agree on a position which does not belong to its maneuvering space as these positions are unacceptable for a party. In other words, the necessary condition for a coalition S to be formed is a non-empty intersection of the maneuvering spaces of all members of S (we call this a feasible coalition), i.e.,

$$\bigcap_{i \in S} M_i \neq \emptyset,$$

and of course, the position x_S of the formed coalition S must belong to this intersection as there has to be commonality in positions, i.e.,

$$x_S \in \bigcap_{i \in S} M_i.$$

A similar assumption is adopted in the policy-horizon model: 'With horizons, there are definite limits to the willingness of parties to compromise on policy in order to participate in government; beyond those limits, parties would prefer to remain in opposition' (Warwick, 2000, 39).

An illustration of the model in a three-party, two dimensional example is given in Figure 3.1. Based on the preferences rule, the valuation (loss) of a party i when a winning coalition

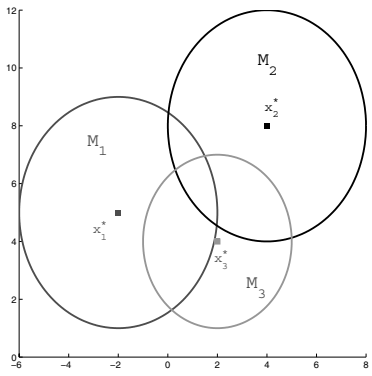

Figure 3.1. *Illustration of the model.*

S is formed, denoted by Π_i^S, is defined as follows:

$$\Pi_i^S(x_S) = d(x_i^*, x_S) \tag{3.3}$$

In Section 3.1.3, both procedures are outlined.

3.1.3 Coalition Formation Process

Now, our approach takes a different course from the one adopted by Warwick. To find a solution to the basic coalition formation model, we consider and compare two procedures: a step-by-step procedure and a simultaneous procedure. These two procedures coincide with the distinction in political science literature between hierarchical and non-hierarchical coalition formation (Laver and Schofield (1990)). So far, spatial coalition theories have most often neglected the different procedure of forming a coalition (as in Grofman (1982) who studies one procedure, but see Brams et al. (2005), and Bloch (1996) who do consider the consequences of different procedures). In de Ridder and Rusinowska (2005), it has formally been proven that it matters which procedure is adopted, and also that there is no procedure which is always better.

The first kind of procedure, the hierarchical view, sees ' ... coalition building as a process in which actors with similar policy preferences first get together in some sort of provisional alliance and, only after this has been done ..., do they cast around for other coalition partners, adding these until the formation criterion is satisfied' (Laver and Schofield (1990), p. 140). The proto-coalition model of Grofman (1982) is such a hierarchical model. In the model we present here, the step-by-step procedure is a hierarchical procedure. Although it is difficult to look behind the often closed doors of coalition negotiations, e.g. Ireland, Belgium, and Denmark have known instances of this step-by-step approach (Müller and Strøm (2003)).

In the step-by-step procedure, the first step is that two parties (e.g. party 1 and 2) negotiate. These two will reach an agreement if their maneuvering spaces overlap and hence a first coalition position $x_{\{1,2\}}$ is agreed on. This coalition position is determined by choosing a position in the intersection of their maneuvering spaces and taking the weights of the players into account. That is, a big party can pull the coalition position more towards its ideal. To be more precise, when determining $x_{\{1,2\}}$, first, parties 1 and 2 each choose a position (called the negotiation position) in the intersection of the maneuvering spaces such that the distance of that position to the ideal point of the party is minimal. These negotiation positions are denoted with $\widetilde{x}_1$ and $\widetilde{x}_2$. The coalition position $x_{\{1,2\}}$ is the gravity center (a weighted average) of the negotiation positions.

Now, a third party (3) joins the negotiations. Players 1 and 2 operate as proto-coalition {1,2}, and an agreement with 3 is only reached if the maneuvering spaces of 1, 2, and 3 overlap. If so, coalition $\{1,2,3\}$ with position $x_{\{\{1,2\},3\}}$ is formed, which is the gravity center of the negotiation positions of the proto-coalition $\{1,2\}$ and party 3. This process continues with adding new parties until a majority coalition S with position $x_{\overline{S}}$ has been reached, where $\overline{S}$ denotes an order, a set of parties, that indicates the sequence that leads to coalition S. In de Ridder and Rusinowska (2005), it has been proven that this step-by-step procedure leads to a unique and Pareto efficient solution. Hence, one coalition position is reached such that there is no other position in the intersection of the maneuvering spaces that is more preferred by all members of the coalition. An illustration of the step-by-step procedure of forming a three-party coalition is given in Figure 3.2.

Second, we also find a non-hierarchial approach which considers coalition formation as a one-step procedure. Laver and Shepsle (1996) generalize political coalition formation as

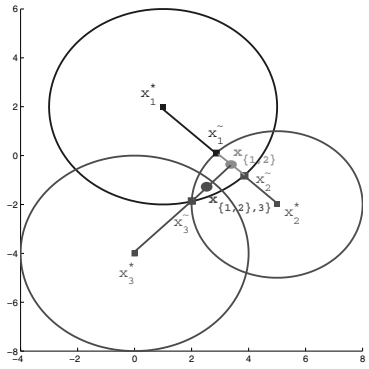

Figure 3.2. *The step-by-step procedure.*

a process in which one party proposes a particular cabinet, which can be vetoed by all its members. In such a case, there are no proto-coalitions which form intermediate steps before a definitive coalition is reached. Non-hierarchical coalition formation is a process in which all the parties of a coalition sit round the table to negotiate simultaneously. In the overview of coalition formation in Western-Europe, Müller and Strøm (2003) report many instances of such a way of bargaining.

In our model, the simultaneous procedure looks as follows. If parties $1, 2$, and 3 form coalition $\{1, 2, 3\}$, their coalition position is $x_{\{1,2,3\}}$. A coalition forms if maneuvering spaces of all three parties overlap. The coalition position will be in the intersection of their three maneuvering spaces and will depend on the weights of the players. The position $x_{\{1,2,3\}}$ is the gravity center of the negotiation positions of all parties in question. More general, the simultaneous procedure of forming a majority coalition S results in a position x_S of the coalition. Again, it has also been proven that this procedure leads to a unique and Pareto optimal solution (de Ridder and Rusinowska (2005)). An illustration of the simultaneous procedure of forming a three-party coalition is given in Figure 3.3.

Beware that although both the step-by-step procedure and the simultaneous procedure can study a coalition with for instance parties $1, 2$, and 3, their respective outcomes are usually different [1]. According to the step-by-step procedure, coalition $\{1, 2, 3\}$ can form in three different ways: first a bilateral agreement with two parties and then the third party $1, 2$ or 3 respectively joins. The simultaneous procedure predicts just one way of forming the coalition: all negotiate together. Hence, in spite of a cooperation between the same three parties, four different paths to form a coalition and four different coalition positions are discerned: $x_{\{\{1,2\},3\}}$, $x_{\{\{1,3\},2\}}$, $x_{\{\{2,3\},1\}}$, and $x_{\{1,2,3\}}$.

Calculations have shown that the number of different paths and coalition positions can increase dramatically. In a coalition game with ten parties, $2^{10} - 11 = 1013$ different 10-party coalitions are possible. However, when taking different procedures into account, 4932045 different step-by-step coalitions can be discerned plus 1013 simultaneously formed coalitions.

[1] When the ideal positions of two parties starting the coalition formation process belong to the intersection of the maneuvering spaces of the three parties, the step-by-step procedure with the given parties' order of forming a coalition, and the simultaneous procedure lead to the same position for the coalition.

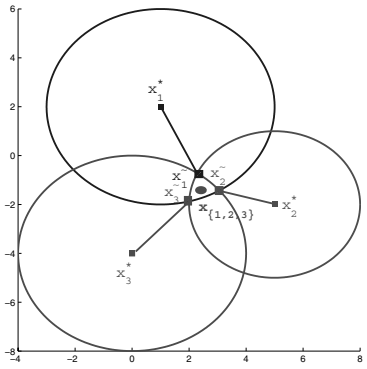

Figure 3.3. *The simultaneous procedure.*

In sum, if ten parties play a coalition game, there are 4933058 different ways of forming a coalition. Figure 3.4 shows the different paths and coalitions to analyse in a case with only 3 parties.

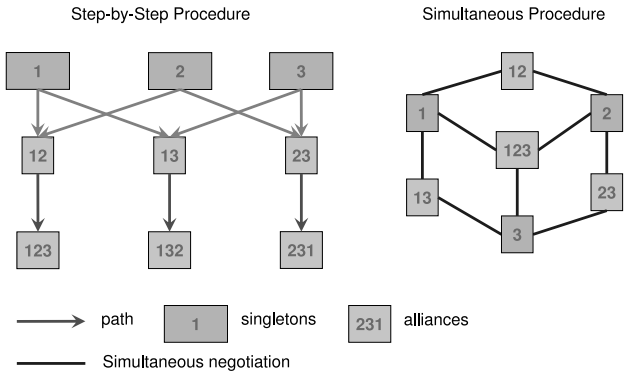

Figure 3.4. *Number of paths and coalitions.*

In this way there are $\frac{n!}{2}$ possible paths each represented by a permutation of n parties. The number of different paths of forming a coalition S is $\frac{|S|!}{2}$. The number L of coalitions in a step-by-step procedure can be calculated:

$$\begin{aligned} L &= C_n^2 + C_n^2 * (n-2) + \ldots + C_n^2 * (n-2) * \ldots * (n-(n-2)) * (n-(n-1)) \\ &= \frac{n!}{2} \times \sum_{k=2}^{n} \frac{1}{(n-k)!} \end{aligned} \tag{3.4}$$

Compared to the step-by-step procedure, the simultaneous procedure has only one path to be followed. It models the situation where $|S|$ parties are sitting together to come to an agreement. Besides the grand coalition, there are many possible partial-coalitions. The more parties, the more possible coalitions can be formed. The number K of possible coalitions is given by:

$$K = C_n^2 + C_n^3 + \ldots + C_n^n = \sum_{k=2}^{n} C_n^k = 2^n - (n+1) \tag{3.5}$$

Table 3.1 shows how complexity increases with the number of players (parties) in both procedures.

Table 3.1. *Number of coalitions following different procedures.*

	step-by-step		simultaneous
Number of parties	Possible paths	Number of Coalitions	Number of Coalitions
2	1	1	1
3	3	6	4
4	12	30	11
5	60	160	26
6	360	975	57
7	2520	6846	120
8	20160	54796	247
9	181440	493200	502
10	1814400	4932045	1013

Disregarding some special conditions, the two procedures usually lead to different positions for the coalition and consequently different appreciations by the coalition members. Given the distance between the ideal position of a party and the coalition position, parties will have a preference ranking over the different positions of the coalitions, over the different coalitions, and hence over the procedures to reach them. The closer a coalition agreement is to the ideal position of a party, the more this party will prefer this coalition agreement. In this way, we show that parties should not only form preferences over coalitions, but should also take the procedure into consideration. In conclusion, the procedure of coalition formation should be a strategic resource in coalition formation and should play a role in coalition negotiations similar to the composition of the coalition.

3.2 Algorithms for the Different Procedures

The coalition compromise differs for each different path in the step-by-step procedure. According to the all-coalition-path configuration, one can calculate the agreement points of all coalitions and corresponding valuations by following the procedure described in the next section. An index l is used to distinguish coalition. A coalition S in a step-by-step formation is an ordered subset of N. A coalition S in a simultaneous formation is a subset of N. Table 3.2 summarises the notation used.

Table 3.2. *Notation.*

N	Set of parties
i, j	Index of parties
L	Total number of coalitions
l	coalition index
x_i^*	Ideal position for party i
S_l	A coalition, $\|S_l\| \geq 2, S_l \subseteq N$
x_i^S	Negotiation position of party i when coalition S is formed
x_S	Compromise coalition position of S
$x_S^{S\cup\{i\}}$	Negotiation position of a coalition S when party i joins

3.2.1 Forming a Coalition Step-by-Step

Consider two parties, i and j, forming a coalition $S = \{i, j\}$. By proposition 3.1 in de Ridder and Rusinowska (2005), the negotiation positions for the two parties are calculated as follows. The negotiation position for party i is: if $r_j < d(x_i, x_j)$,

$$x_i^S = x_j^* + r_j \times \frac{x_i^* - x_j^*}{d(x_i^*, x_j^*)} \tag{3.6}$$

otherwise,

$$x_i^S = x_i^* \tag{3.7}$$

where the negotiation position of j is given by switching i and j in (3.6) and (3.7). Once the parties have defined their negotiation positions, the compromise position is calculated by

$$x_S = \frac{w_i \times x_i^S + w_j \times x_j^S}{w_i + w_j} \tag{3.8}$$

Let S be a coalition with p members, $p \geq 1$. The compromise position of the coalition is x_S. If party i joins the coalition, both, the coalition S and the party i have to choose new negotiation positions: $X_S^{S\cup\{i\}}$ and $x_i^{S\cup\{i\}}$ respectively. Next step is to agree on a compromise coalition position $X^{S\cup\{i\}}$. To choose the new negotiation positions, the problem to solve is:

$$x_i^{S\cup\{i\}} = \arg \min_{z \in \bigcap_{j\in S\cup\{i\}} M_j} d(x_i^*, z) \tag{3.9}$$

$$x_S^{S\cup\{i\}} = \arg \min_{z \in \bigcap_{j\in S\cup\{i\}} M_j} d(X^S, z) \tag{3.10}$$

The compromise position for the new coalition $S \cup \{i\}$ is calculated as follows:

$$x_{S\cup\{i\}} = \frac{w_i \times x_i^{S\cup\{i\}} + X_S^{S\cup\{i\}} \times \sum_{j\in S} w_j}{\sum_{j\in S} w_j + w_i} \tag{3.11}$$

Based on the model by de Ridder and Rusinowska (2005) we introduce a procedure to determine the compromise (agreement) points and valuations of all coalitions at each possible path. In this procedure, (see Algorithm 3.1) first the negotiation positions and

Algorithm 3.1 Step-by-Step algorithm.

Funct Step-by-Step(Ideal positions of parties, X; Radius for each party, R; weights, voting power, W; quota q; number of parties, n and dimension, m)

1. $l := 0$
2. $L := n!/2 * \sum_{k=2}^{n} 1/(n-k)!$ ▷ *number of possible coalitions*
3. **for** each two-party coalition ▷ *Compute new positions and negotiation points for all the possible two-party coalitions*
4. **if** $M_i \cap M_j \neq \emptyset$
5. $l := l + 1$
6. $S_l := \{i, j\}$
7. $NS_l := N \backslash S_l$
8. $[x_i^{\{i,j\}}, x_j^{\{i,j\}}] := \underline{\textbf{Neg-Pos2}}(x_i^*, x_j^*)$
9. $X_l^G = \frac{w_i \times x_i^{\{i,j\}} + w_j \times x_j^{\{i,j\}}}{w_i + w_j}$
10. **if** $\sum_{i \in S_l} w_i > q$
11. **for** $i \in S_l$
12. $\Pi_{li} = d(x_i^*, X_l^G)$
13. **for** $k = 1$ to $L - n$
14. **while** $j \in NS_k$ AND $\bigcap_{i \in S_l} M_i \cap M_j \neq \emptyset$
15. $l := l + 1$
16. $S_l := S_k \cup \{j\}$
17. $NS_l := NS_k \backslash \{j\}$
18. $[x_j^{S_l}, X_k^G] := \underline{\textbf{Negotiation}}(x_j^*, X_k^G)$
19. $X_l^G = \frac{w_j \times x_j^{S_l} + \sum_{i \in S_k} w_i \times X_k^G}{\sum_{i \in S_l} w_i}$
20. **if** $\sum_{i \in S_l} w_i > q$
21. **for** $i \in S_l$
22. $\Pi_{li} = d(x_i^*, X_l^G)$
23. OUTPUT: {Coalitions, compromise positions and valuations of winning coalitions}

compromise points for all the possible two-party coalitions are computed. Procedure 3.2 is used to compute the negotiation positions. For each two-party coalition S, the procedure builds up the coalition adding one-by-one new members. If the maneuvering spaces of the new member i and the members of S overlap, the negotiation positions (for the new member and the coalition) are computed (Procedure 3.3). If the new coalition $S \cup \{i\}$ is a winning coalition, then valuations for each member are calculated. For the computation of (3.9) and (3.10), Procedure 3.3 uses an external non-linear programming algorithm, $fmincon$. Additionally, a penalty approach is used to check whether or not an intersection (feasible area) exists between the maneuvering spaces of the negotiating parties. Given potential coalition S, we minimise over x the penalty function

$$F(x) = \max_{j \in S}(d(x_i^*, x) - r_i) \tag{3.12}$$

If the result is negative the intersection is nonempty.

3.2.2 Forming a Coalition Simultaneously

Let $S \subseteq N$ be a coalition and $M_i(x_i, r_i)$ for $i \in S$ be maneuvering spaces in $\mathbb{R}^m$ such that $\cap_{i \in S} M_i \neq \emptyset$.

Algorithm 3.2 Computes negotiation positions for parties and coalition position.

Funct Neg-Pos2(Ideal positions for each of the two parties: x_i^*, x_j^*)

1. *New position for party i*
2. **if** $r_j < d(x_i, x_j)$
3. $\quad x_i^{\{i,j\}} = x_j^* + r_j \times \frac{x_i^* - x_j^*}{d(x_i^*, x_j^*)}$
4. **else**
5. $\quad x_i^{\{i,j\}} = x_i^*$
6. *New position for party j*
7. **if** $r_i < d(x_i, x_j)$
8. $\quad x_j^{\{i,j\}} = x_i^* + r_i \times \frac{x_j^* - x_i^*}{d(x_i^*, x_j^*)}$ **else**
9. $\quad x_j^{\{i,j\}} = x_j^*$
10. OUTPUT: {New positions for the parties and coalition position: $x_i^{\{i,j\}}, x_j^{\{i,j\}}$}

Algorithm 3.3 Computes new negotiation positions for coalition and new member (party).

Funct Negotiation(Ideal position for the new member i, x_i^*, of the new coalition $S \cup \{i\}$ and compromise position of coalition S, X^S)

1. *objfun=objective function; objcon=objective constraints;x_0=starting point*
2. **if** $M_i \cap \bigcap_{j \in S} M_j \neq \emptyset$
3. $\quad x_i^{S \cup \{i\}} =$ **Fmincon**$(objfun, objcon, x_0, x_i^*)$
4. $\quad X_S^{S \cup \{i\}} =$ **Fmincon**$(objfun, objcon, x_0, X^S)$
5. OUTPUT: {New negotiation positions for party i and coalition S}

1. Each party $i \in S$ chooses the negotiation position x_i^S:

$$x_i^S = \arg \min_{z \in \bigcap_{j \in S} M_j} d(x_i^*, z) \tag{3.13}$$

2. Coalition position x_S is chosen as gravity center of positions x_i^S with weights w_i:

$$x_S = \frac{\sum_{j \in S} w_j x_j^S}{\sum_{j \in S} w_j} \tag{3.14}$$

Algorithm 3.4 is the main procedure when coalitions are formed simultaneously. It computes the winning coalitions, its coalition positions and valuation-preferences for the parties. First, it generates all possible coalitions based on $0-1$ notation: 0 means the party is not a member; 1 means the party is a member of the coalition. If the coalition is winning and the maneuvering spaces of the members overlap, the algorithm calls a second procedure (Algorithm 3.5) to compute the negotiation positions of the members. Algorithm 3.5 uses an external non-linear programming algorithm to calculate the positions. Back in the main algorithm, the coalition position and valuations are computed in order to generate a preference order.

Algorithm 3.4 Simultaneous algorithm.

Funct Simult(Ideal positions of parties, x^*; Radius for each party, r; weights, voting power, w; quota q; number of parties, n and dimension, m)

1. *number of possible coalitions:* $L := 2^n - (n+1)$
2. **for** $k = 1$ to L ▷ *Compute new positions and negotiation points for all the feasible coalitions*
3. *Generate coalition* S_k
4. **if** $\sum_{i \in S_k} w_k > q$ and $\bigcap_{i \in S_k} M_i \neq \emptyset$
5. **for** $i \in S_k$
6. $x_i^{S_k} :=$ **Neg-Sim**(x^*)
7. $X_k^G = \frac{\sum_{i \in S_k} w_i \times x_i^{S_k}}{\sum_{i \in S_k} w_i}$
8. $\Pi_{ki} = d(x_i^*, X_k^G)$
9. OUTPUT: {Coalitions, new party and negotiation positions for the simultaneous procedure}

Algorithm 3.5 Procedure to compute new negotiation positions for members in coalition S.

Funct Neg-Sim(Ideal position for the party i, x_i^*)

1. *objfun=objective function; objcon=objective constraints;x0=starting point*
2. $x_i^S =$ **Fmincon**$(objfun, objcon, x0)$
3. OUTPUT: {New negotiation positions for party i}

3.2.3 Numerical Illustration

We provide an example here to illustrate how the algorithms and model work. This example uses the Dutch election result of 2003 (Klingemann et al. (2006)). As input for the model, we need ideal policy positions of Dutch parties, and a weight and a radius for each political party. The ideal policy positions are derived from a data set with policy positions of Dutch political parties on 56 dimensions from 1998 and 2003 (Klingemann et al. (2006)). Because the model is working with spherical maneuvering spaces based on distance calculations, the data is all scaled between 0 and 10. The weight of the parties is determined by the amount of seats each party had in parliament (total of 150 seats). The radii that model the flexibility of the parties is relatively arbitrary for illustrative purposes and leave a degree of freedom for our analysis. In reality, each party has its own radius which is dependent on the specific situation and which might be subject to change. In this case we have used similar radii for all parties. The names of the parties are the following:

CDA - Christian Democrats (Christen Democratisch Appel)
CU - Christian Union (Christen Unie)
D66 - Democrats 66 (Democraten '66)
GRL - Green Left (Groen Links)
LPF - List Pim Fortuyn (Lijst Pim Fortuyn)
PvdA - Labor Party (Partij van de Arbeid)
SP - Socialist Party (Socialistische Partij)
VVD - People's Party for Freedom and Democracy (Volkspartij voor Vrijheid en Democratie)

Note that the SGP (Political Reformed Party) is not included in this table, as it was not included in the dataset from Klingemann et al. (2006) (in Klingemann et al. (2006), Appendix IV, is explained that the election program for the collection of data was missing).

As output of the model, we only consider coalition positions of majority coalitions of parties that have an overlap of their maneuvering spaces given their ideal policy positions,

Table 3.3. *Data for 2003.*

	Parties							
	CDA	CU	D66	GRL	LPF	PvdA	SP	VVD
Radius	30	30	30	30	30	30	30	30
Weight	44	3	6	8	8	42	9	28

i.e. of feasible winning coalitions. As said earlier, the biggest party gets the initiative for coalition formation in The Netherlands. In 2003, this was the CDA. The majority coalitions with overlapping maneuvering spaces containing CDA are included in Table 3.4. For each coalition reached with a certain procedure, the distance between the coalition position and the ideal position of the party are calculated. The {PvdA, CDA} coalition leads to the same coalition position with both procedures as no third party joins here. However, for a coalition between CDA, PvdA, and LPF (e.g. {{CDA, PvdA}, LPF} and {CDA, PvdA, LPF}) procedure plays a role as different procedures lead to different distances. More generally, we see in all the calculations done for this paper that procedure really makes a difference: different procedures lead to different results.

Table 3.4. *Distances from ideal points for 2003 example.*

Step-by-Step Procedure									
Coalition	Seats	Distance							
		CU	D66	GRL	PvdA	SP	VVD	LPF	CDA
$\{CDA, PvdA\}$	86	-	-	-	20.52	-	-	-	20.07
$\{\{CDA, PvdA\}, SP\}$	95	-	-	-	24.92	29.39	-	-	26.92
{{CDA,PvdA},LPF}	94	-	-	-	25.45	-	-	29.24	23.69
$\{\{CDA, SP\}, PvdA\}$	95	-	-	-	26.04	29.42	-	-	25.97
{{CDA,LPF},PvdA}	94	-	-	-	25.38	-	-	29.15	23.86
Simultaneous Procedure									
		CU	D66	GRL	PvdA	SP	VVD	LPF	CDA
$\{CDA, PvdA\}$	86	-	-	-	20.52	-	-	-	20.07
$\{CDA, PvdA, SP\}$	95	-	-	-	26.01	29.00	-	-	26.47
{CDA,PvdA,LPF}	94	-	-	-	25.31	-	-	28.62	24.59

Based on these distances, the preferences of the players can be calculated. The closer the coalition position to the ideal position of a party, the more the party will prefer this coalition and the procedure. Table 3.5 reports this. As an example, CDA's most favorite option is to cooperate with PvdA. If CDA would cooperate with PvdA and SP, then the best procedure for CDA would be to negotiate first with SP alone. The step-by-step procedure with SP joining as last is CDA's least preferred procedure for this coalition. Note that we do not consider preferences of the parties not participating in the coalition.

In reality, the coalition that formed was {CDA, VVD, D66}. Although it is not the aim of this paper to predict which coalitions have occurred, we can explain why this coalition did not appear in the results. According to the model and, in particular, the adopted input, this coalition would not be viable. That means that the adopted radii did not lead to an overlap of the parties' maneuvering spaces; the {CDA, VVD, D66} coalition is less acceptable than the coalitions that appear in the table.

Table 3.5. *Preference order for 2003.*

Step-by-Step Procedure									
Coalition	Seats	Preference order							
		CU	D66	GRL	PvdA	SP	VVD	LPF	CDA
$\{CDA, PvdA\}$	86	-	-	-	1	-	-	-	**1**
$\{\{CDA, PvdA\}, SP\}$	95	-	-	-	2	2	-	-	**7**
$\{\{CDA, PvdA\}, LPF\}$	94	-	-	-	5	-	-	3	2
$\{\{CDA, SP\}, PvdA\}$	95	-	-	-	7	3	-	-	5
$\{\{CDA, LPF\}, PvdA\}$	94	-	-	-	4	-	-	2	3
Simultaneous Procedure									
		CU	D66	GRL	PvdA	SP	VVD	LPF	CDA
$\{CDA, PvdA\}$	86	-	-	-	1	-	-	-	1
$\{CDA, PvdA, SP\}$	95	-	-	-	6	1	-	-	6
$\{CDA, PvdA, LPF\}$	94	-	-	-	3	-	-	1	4

3.3 Hypothesis Testing

The described model and introduced computational method was used in de Ridder et al. (2007) to do an extensive study to test hypotheses derived from intuition with the aid of Dutch data. The rest of this paper reports on the findings. First we formulate the hypotheses and the Dutch situation as a platform of analysis. After that we point wise discuss the results that can be found in de Ridder et al. (2007).

3.3.1 Procedure: Hypothesis on First Mover

Empirical observations of how coalitions form show that procedures are in some countries standard and formalized in laws (e.g. Belgium, Finland, Luxembourg, and The Netherlands, Müller and Strøm (2003)). That diminishes the opportunity for parties to use procedure as a strategic means during the coalition process. An important observation is that many multi-party democracies have the (unwritten) law that the party that came out of the elections as the largest gets the initiative (from a head of state) for forming a coalition. Examples of countries in which this (more or less frequently) happens are The Netherlands, Sweden, Finland, Austria, Belgium, and Luxembourg (Müller and Strøm (2003)). The idea behind this is that these initiative taking parties are supposed to lead the negotiations and to have an advantage in the bargaining situation. The earlier a party is involved in coalition negotiations, the more this party is able to pull the negotiations towards its own ideas. In this way, this party can determine and influence the negotiations more and can get advantage out of it. This brings us to the first hypothesis: *Being a first-mover in coalition negotiations is advantageous.*

3.3.2 Flexibility Hypothesis

The second innovation of our model, is the flexibility during negotiations we attribute parties via maneuvering spaces. In the literature of coalition formation models, it has most often been assumed that political parties have a fixed position in policy space (Grofman (1982); Vries (1999)). However, more scholars begin to acknowledge the importance of studying the dynamics of party competition (Laver (2005); Timmermans (2003), van der Brug (1999)): '...positions are not frozen or fixed; parties move in the policy space in different directions over time' (Timmermans (2003), p. 9). Here, we concentrate on dynamics of policy positions

not with vote maximizing as goal (as e.g. Budge (1994); Enelow and Hinich (1984); Laver (2005)), but dynamics due to coalition formation.

The idea is that in order to form a coalition, political parties will move their policy position, but only *to a certain limit* (Warwick (2000)) as formalized by the maneuvering space. Coalition formation implies making a coalition agreement: a compromise between the members of a coalition on the ideological course of the coalition, consisting of a position for the coalition. As a consequence, parties participating in a coalition need to adjust their position in order to reach such an agreement (also see Martin and Vanberg (2004)). It is not likely that parties will cooperate with a party which has opposing policy ideals. We therefore assume parties will only be willing to compromise if they can stay within their maneuvering space of acceptable positions.

The question now rises what is mostly in a party's interest: a big or small maneuvering space? When forming a two-party coalition, the answer is straightforward: being less flexible is never disadvantageous. If a coalition consists of only two parties, the more flexible party of the two will be forced to move its position more than the other. One can speak of a zero-sum situation: what one wins, is lost by the other.

Nonetheless, when forming a k-party coalition, for $k \geq 3$, the answer is less easy. Intuitively, one would consider that staying closer to a party's ideal position is also better in multi-party coalitions. Hence, a decrease in flexibility would always be in a party's advantage. However, this is less easy to analyse due to the amount of players involved. Therefore, we use the data and theoretical results to study whether the following (second) hypothesis holds: *Being less flexible in coalition negotiations is more advantageous.*

3.3.3 Sharing Power: Hypothesis on Minimal Winning

As a final point, we study the role of sharing power. Coalition formation has long been considered as a combination of achieving power, and simultaneously sharing this power with coalition partners. Coalition formation is therefore a delicate balance between on the one hand getting this power by compromising into the coalition, and on the other hand, forming a coalition which gives a party relatively the best power. In this tradition, the minimal winning (von Neumann and Morgenstern (1944)) and minimum size theory (Riker (1962)) have been formulated. Minimal winning coalitions are coalitions that contain enough members to be winning, but are not oversized. Minimal winning coalitions cannot miss any member without becoming losing. Minimum size coalitions contain enough weight to be winning, but not more than that.

In line with this, one could reason that oversized coalitions imply sharing power with more partners and hence compromising with more partners than necessary. The chance is bigger that a coalition position will be reached which is farther from a party's ideal position. Less members in a coalition make it easier to reach an agreement which is closer to a party's ideal point. Hence hypothesis 3a: *Being in a smaller (winning) coalition is more advantageous than being in an oversized coalition.*

In a similar way, we can argue that forming a coalition with a stronger partner is not advantageous, since the stronger party may 'pull' the position of a formed coalition more towards it's own ideal position. Hence we propose hypothesis 3b: *Increase of a party's weight is disadvantageous for its coalition partners.*

3.3.4 Dutch Situation and Data

In The Netherlands, coalition governments are the standard, considering that the Dutch multi-party democracy only has had coalition governments since 1945 (Müller and Strøm (2003)). Also, The Netherlands has a tradition of majority coalitions. Furthermore, two of the issues we highlight - procedures and flexibility - are important. Concerning procedures, the process of coalition formation is by far the longest in Western Europe with an average of 70.6 days. This could denote an important role for procedures. The first mover issue is relevant as it is characteristic for the Dutch coalition practice that the biggest party gets the initiative to form a coalition. Concerning flexibility, coalition agreements play an important role in coalition negotiations: each cabinet agrees on such a document as the course of action during their period of government. Data however show different ideal policy positions of Dutch parties (e.g. Vries (1999); van der Brug (1999)) which implies compromises and hence flexibility of parties. As explained in Section 3.2, the radii that model the flexibility of the parties is relatively arbitrary for illustrative purposes and leave a degree of freedom for our analysis. Due to the lack of empirical data on this aspect, we have taken two different ways to determine the radius: a radius similar for each party and a radius different for each party, randomly generated. In the case (as in the case of Section 3.2) in which we have used similar radii for all parties, the radii have been determined by optimizing the case such that enough, but not too many, instances were found which could help us investigate the hypotheses.

To run the model with real-life multidimensional data, one needs computational algorithms. We have performed calculations with the model using data from Dutch politics, and, moreover, we present some theoretical results. Both the empirical and theoretical calculations provide some counter-intuitive situations which show that certain expectations do not always hold. Also, we illustrate that certain traditions in real-life coalition formation are not necessarily advantageous.

During the paper, we study which strategic moves are advantageous for a potential coalition member. *Advantageous* is defined in terms of preference of a party over a coalition and the path to reach this coalition. This is measured by taking the distance from the ideal position of the party to the position of the coalition compromise. The closer the coalition position, the better. The policy-distance effect on government composition, meaning that the incentive of a party to join a parliamentary coalition government decreases with the distance between the policy position and the position of the government, was elaborated and tested in particular by Warwick (1998).

3.3.5 Results on Procedure

Two different procedures of coalition formation, leading to different coalition positions, are under study. The research question is whether being a first-mover is always advantageous for a party in coalition negotiations, as in real-life the biggest party, after elections, is most often rewarded with the initiative for coalition negotiations.

In the 2003 case presented in Section 3.2, we indeed saw that for the LPF being the first mover was advantageous. When comparing the LPF's preference on the two step-by-step procedures it is involved in, it prefers {{CDA, LPF}, PvdA} over {{CDA, PvdA}, LPF}. So, it prefers being a first mover over being a late mover. A small counter example can be found due to the PvdA that in the same coalition prefers to step in later. The data of 1998 and Table 3.6 show a stronger counter example, as can be observed from Table 3.7.

In the 1998 case, PvdA was the biggest party and had to take the initiative in coalition negotiations. For the three party coalition {PvdA, SP, CDA}, two step-by-step and one simultaneous procedures were considered as PvdA always had to be a first mover. In the two

Table 3.6. *Weights and radius 45 for 1998 data*

	Parties					
	GRL	SP	PvdA	D66	VVD	CDA
Radius	45	45	45	45	45	45
Seats	11	5	45	14	38	29

Table 3.7. *Preference order for 1998 data*

Step-by-Step Procedure							
Coalition	Seats	Preference order					
		GRL	SP	PvdA	D66	VVD	CDA
$\{\{PvdA, SP\}, CDA\}$	79	-	8	3	-	-	2
$\{\{PvdA, D66\}, CDA\}$	88	-	-	5	7	-	4
$\{\{PvdA, CDA\}, SP\}$	79	-	4	1	-	-	7
$\{\{PvdA, CDA\}, D66\}$	88	-	-	2	4	-	8
$\{\{\{PvdA, SP\}, D66\}, CDA\}$	93	-	5	10	9	-	6
$\{\{\{PvdA, SP\}, CDA\}, D66\}$	93	-	1	7	8	-	9
$\{\{\{PvdA, D66\}, SP\}, CDA\}$	93	-	9	11	5	-	6
$\{\{\{PvdA, D66\}, CDA\}, SP\}$	93	-	7	8	1	-	10
$\{\{\{PvdA, CDA\}, SP\}, D66\}$	93	-	2	6	8	-	9
$\{\{\{PvdA, CDA\}, D66\}, SP\}$	93	-	7	5	2	-	10
Simultaneous Procedure							
		GRL	SP	PvdA	D66	VVD	CDA
$\{PvdA, SP, CDA\}$	79	-	3	4	-	-	1
$\{PvdA, CDA, D66\}$	88	-	-	9	3	-	3
$\{PvdA, D66, SP, CDA\}$	93	-	6	12	6	-	5

step-by-step procedures, CDA would be better off being a late instead of a first mover. Let $\succ_i$ denote the preference relation of party i. For this coalition, the preference order of CDA is as follows: {PvdA, SP, CDA} $\succ_{CDA}$ {{PvdA, SP}, CDA} $\succ_{CDA}$ {{ PvdA, CDA}, SP}. This also holds for SP, which in case of step-by-step formation rather joins as last member in the negotiations.

We can therefore conclude that hypothesis 1 does not hold:

Result 1: *Being a first mover is not always advantageous.*

3.3.6 Results on Flexibility

As a second major point, focus is on policy flexibility of parties. One of the central assumptions of the model is that parties have maneuvering spaces which reflect their flexibility to deviate from their ideal positions. No party will accept a coalition position which lies outside its maneuvering space. This assumption is similar to the one made in a policy-horizon model by Warwick (2000), (2005a), (2005b)). We study the hypothesis *Being less flexible in coalition negotiations is more advantageous.*

A search in the data did not provide a counter example to this hypothesis. It was found that a decrease in a party's flexibility always seems to be in the party's advantage. In other words, the intuition which was provided earlier holds. As seen more easy in two-party coalitions, less flexibility always leads to a more advantageous coalition agreement for

a party. Although we did not find a counter-example in the Dutch data, we did come up with a one-dimensional theoretical example which shows that being less flexible can be a *disadvantage*.

Example 1. *We consider a three-party example, in which parties 1 and 2 have the same weight, while the weight of party 3 is twice as big as the weight of party 1 and 2, i.e.*

$$N = \{1, 2, 3\}, \quad w_1 = w_2, \quad w_3 = 2w_2$$

The situation is illustrated in Figure 3.5. Since this is a one-dimensional example, the ideal positions x_1^*, x_2^* and x_3^* are points (denoted in Figure 3.5 by squares) on a line, while the maneuvering spaces M_1, M_2 and M_3 are intervals (denoted in Figure 3.5 by two-headed arrows). We have

$$x_1^* = 0, \quad x_2^* = 4, \quad x_3^* = -2$$

All parties are assumed to be equally flexible and their radii are equal to

$$r_1 = r_2 = r_3 = 6$$

Hence, the maneuvering spaces are

$$M_1 = [-6, 6], \quad M_2 = [-2, 10], \quad M_3 = [-8, 4]$$

and their intersections (also two-headed arrows)

$$M_1 \cap M_3 = [-6, 4], \quad M_1 \cap M_2 = [-2, 6]$$

$$M_2 \cap M_3 = M_1 \cap M_2 \cap M_3 = [-2, 4] \neq \emptyset.$$

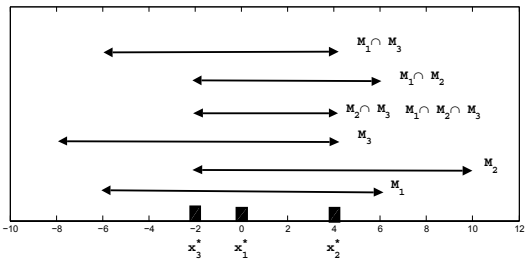

Figure 3.5. *Counter-example "being less flexible can be a* disadvantage*". Ideal points (squares) and maneuvering spaces (two-headed arrows)*

Since $M_1 \cap M_2 \cap M_3 \neq \emptyset$, the necessary condition for coalition $\{1, 2, 3\}$ to be formed is satisfied. Let us consider the step-by-step procedure of forming coalition $\{1, 2, 3\}$, in which first parties 1 and 2 form a coalition $\{1, 2\}$, and then party 3 joins. The steps of the procedure are explained in Section 3.2. The negotiation positions $x_1^{\{1,2\}}$ and $x_2^{\{1,2\}}$ of parties 1 and 2 are equal to their ideal positions, because the ideal points lie in the intersection of the maneuvering spaces, i.e.

$$x_1^{\{1,2\}} = 0 = x_1^*, \quad x_2^{\{1,2\}} = 4 = x_2^*$$

Since the weights of parties 1 and 2 are the same and the coalition position is the gravity

center of the negotiation positions, we get

$$x_{\{1,2\}} = 2 \in M_3$$

Next, party 3 joins proto-coalition $\{1,2\}$*. Because* x_3^* *and* $x_{\{1,2\}}$ *lie in the intersection of the maneuvering spaces, the negotiation positions of party 3 and proto-coalition* $\{1,2\}$ *are equal to* $x_3^* = -2$ *and* $x_{\{1,2\}} = 2$*, respectively. Since the weight of party 3 is equal to the weight of* $\{1,2\}$*, we get*

$$x_{\{\{1,2\},3\}} = 0 = x_1^*$$

Hence, the step-by-step procedure of forming $\{\{1,2\},3\}$*, in which first parties 1 and 2 form a coalition, and then party 3 joins, leads to the coalition position* $x_{\{\{1,2\},3\}}$ *which is the best possible position for party 1.*

Next, let us assume that party 1 becomes less flexible, that is, its new radius decreases to $r_1' = 3$*. All remaining components of the example are unchanged. Then,*

$$M_1' = [-3,3], \quad M_1' \cap M_2 = M_1' \cap M_2 \cap M_3 = [-2,3]$$

We consider the same step-by-step procedure of forming $\{1,2,3\}$ *with the new radius* $r_1' = 3$*. The new negotiation position* $y_1^{\{1,2\}}$ *of party 1 is the same as before (equals* $x_1^{\{1,2\}}$*), since its ideal point lies in the intersection of the maneuvering spaces. However, the new negotiation position* $y_2^{\{1,2\}}$ *of party 2 is different, i.e.*

$$y_1^{\{1,2\}} = x_1^* = 0, \quad y_2^{\{1,2\}} = 3.$$

The new position $y_{\{1,2\}}$*, as the gravity center of* $y_1^{\{1,2\}}$ *and* $y_2^{\{1,2\}}$ *with equal weights* $w_1 = w_2$*, is now*

$$y_{\{1,2\}} = \frac{3}{2} \in M_3$$

The new coalition position $y_{\{\{1,2\},3\}}$*, as the gravity center of the negotiation positions* $y_{\{1,2\}}$ *and* $x_3^* = -2$*, with equal weights for* $\{1,2\}$ *and party 3, is now*

$$y_{\{\{1,2\},3\}} = -\frac{1}{4}$$

Hence, the step-by-step procedure of forming $\{\{1,2\},3\}$*, in which first parties 1 and 2 form a coalition, and then party 3 joins, results now in the coalition position* $y_{\{\{1,2\},3\}}$ *which is worse for party 1 than the coalition position* $x_{\{\{1,2\},3\}}$*, for the case where party 1 is more flexible, i.e.*

$$x_{\{\{1,2\},3\}} \succ_1 y_{\{\{1,2\},3\}}$$

This means that becoming less flexible made party 1 worse off.

To conclude, although the data have shown that less flexibility always seems to be advantageous to a party, a theoretical counter example has illustrated how a decrease in flexibility can be a disadvantage for a party. Hence:

Result 2: *When forming a k-party coalition, for* $k \geq 3$*, being less flexible is usually advantageous, but can theoretically be a disadvantage.*

3.3.7 Results on Sharing Power

Additionally, we study a minor point: the role of sharing power. The question here is whether striving for a coalition in which a party gets the best relative power position is always ad-

vantageous. Earlier empirical results confirm the role of power-sharing motives of parties (Martin and Stevenson (2001)), but do not show that oversized can be an advantage for coalition members (cf. Volden and Carrubba (2004) who explain when oversized coalitions occur). Sub-issues here are the minimal winning argument (von Neumann and Morgenstern (1944)) and the influence of weight. The minimal winning argument states that only coalitions will form that have enough members to be winning, but not more than that. But is a minimal winning coalition necessarily advantageous for a party? Or, more general, is a smaller coalition necessarily more advantageous than an oversized coalition?

We have found many counter-examples in Dutch data which show that the hypothesis does not always hold. We consider Dutch data after the 1998 elections (see Table 3.6). Here, we change the radii for the parties and let the radius be different for different parties. We get an instance as shown in Table 3.8. Table 3.9 shows the preference order for this case. Note that under the step by step as well as simultaneous procedure, PvdA finds the non-minimal winning coalition formed by PvdA, VVD and D66 more attractive than the minimal winning coalition {PvdA, VVD}.

Table 3.8. *Weights and different radii for 1998*

	Parties					
	GRL	SP	PvdA	D66	VVD	CDA
Radius	45	55	25	65	85	45
Seats	11	5	45	14	38	29

Concluding, we get the following result.

Result 3a: *Forming a minimal winning coalition is not always advantageous.*

Table 3.9. *Preference order with different radii for 1998*

Step-by-Step Procedure							
Coalition	Seats	Preference order					
		GRL	SP	PvdA	D66	VVD	CDA
$\{PvdA, VVD\}$	83	-	-	**3**	-	1	-
$\{\{PvdA, SP\}, VVD\}$	88	-	4	12	-	5	-
$\{\{PvdA, D66\}, VVD\}$	97	-	10	**1**	11	6	-
$\{\{PvdA, VVD\}, SP\}$	88	-	8	7	-	4	-
$\{\{PvdA, VVD\}, D66\}$	97	-	11	**2**	10	2	-
$\{\{\{PvdA, SP\}, D66\}, VVD\}$	102	-	1	8	1	12	-
$\{\{\{PvdA, SP\}, VVD\}, D66\}$	102	-	2	11	2	10	-
$\{\{\{PvdA, D66\}, SP\}, VVD\}$	102	-	6	5	5	12	-
$\{\{\{PvdA, D66\}, VVD\}, SP\}$	102	-	9	4	7	11	-
$\{\{\{PvdA, VVD\}, SP\}, D66\}$	102	-	5	7	4	9	-
$\{\{\{PvdA, VVD\}, D66\}, SP\}$	102	-	9	6	6	7	-
Simultaneous Procedure							
		GRL	SP	PvdA	D66	VVD	CDA
$\{PvdA, VVD\}$	83	-	-	**3**	-	1	-
$\{PvdA, SP, VVD\}$	88	-	7	10	-	3	-
$\{PvdA, D66, VVD\}$	97	-	11	**2**	8	2	-
$\{PvdA, SP, D66, VVD\}$	102	-	3	9	3	8	-

Concerning weight, we like to consider the consequence the weight of a party (number of seats in parliament) has for its coalitional partners. The last research question is then: Does an increase of a party's weight imply a disadvantage for its coalition partners?

In a similar way, we can argue that forming a coalition with a stronger partner is not advantageous, since the stronger party may 'pull' the position of a formed coalition more towards it's own ideal position. Hence we propose hypothesis 3b: *Increase of a party's weight is disadvantageous for its coalition partners.*

One can show that forming a two-party coalition with a stronger party is never advantageous to the coalition partner. The intuition is that in such a 'zero-sum' situation, the larger party will always be able to pull the coalition position to its own position, further away from its partner. Nevertheless, it does not necessarily hold when forming a larger coalition. We can illustrate this with the following theoretical example.

Example 2. *We consider the same situation as in Example 1 with party 1 being less flexible, i.e.,*

$$N = \{1,2,3\}, \quad x_1^* = 0, \quad x_2^* = 4, \quad x_3^* = -2$$
$$r_1' = 3, \quad r_2 = r_3 = 6, \quad w_1 = w_2, \quad w_3 = 2w_2$$
$$M_1' = [-3,3], \quad M_2 = [-2,10], \quad M_3 = [-8,4]$$
$$M_1' \cap M_2 = M_1' \cap M_2 \cap M_3 = [-2,3]$$

As calculated in Example 1, the coalition position $y_{\{\{1,2\},3\}}$ results from the step-by-step procedure of forming $\{\{1,2\},3\}$, in which first parties 1 and 2 form a coalition, and then party 3 joins, is equal to $y_{\{\{1,2\},3\}} = -\frac{1}{4}$. Next, let us assume that the weight of party 1 increases: it is twice as big as the weight of party 2 and the same as the weight of party 3, i.e.,

$$w_1' = 2w_2 = w_3$$

The remaining components of the model remain unchanged. We consider the same step-by-step procedure of forming $\{\{1,2\},3\}$. The new negotiation positions $z_1^{\{1,2\}}$, $z_2^{\{1,2\}}$, and coalition positions $z_{\{1,2\}}$, $z_{\{\{1,2\},3\}}$ are now the following:

$$z_1^{\{1,2\}} = x_1^* = 0, \quad z_2^{\{1,2\}} = 3, \quad z_{\{1,2\}} = 1 \in M_3$$

$$z_{\{\{1,2\},3\}} = -\frac{1}{5}$$

Comparing the distance between coalition position $y_{\{\{1,2\},3\}}$ and the ideal point x_2^* of party 2 and the distance between the new coalition position $z_{\{\{1,2\},3\}}$ and x_2^*, one can conclude that

$$z_{\{\{1,2\},3\}} \succ_2 y_{\{\{1,2\},3\}}$$

It means that an increase of the weight of party 1 makes party 2 better off.

This gives the following result.

Result 3b: *When forming a k-party coalition, for $k \geq 3$, an increase of a party's weight may be an advantage for its coalition partner.*

In order to show a pure effect of an increase of a party's weight in Example 2, somewhat artificially we have increased the weight of party 1, keeping all remaining elements unchanged. This is of course not what happens in a parliament, since elections (usually) preceding coalition formation fix the weights of the parties. However, it can be used by parties defining a coalition formation strategy before elections. For example, in its campaign a party may be less negative with respect to another party whose bigger size might be beneficial. Nevertheless, although we believe that this result is mainly of a theoretical nature, we have also constructed an instance using the data. Consider the case of Table 3.8 that

presents the 1998 data with varying flexibility for the parties taking the real number of seats. The distance of the ideal of D66 to the compromise of coalition $\{\{\{PvdA, SP\}, D66\}, VVD\}$ is 52.25. Let us now hypothetically assume that SP increases its weight by 30, while the other parties keep their original weights. Now the distance of the ideal of D66 to the coalition position becomes 51.53. This means its position improves due to an increase of another party.

3.4 Conclusions

In spite of the many unwritten laws and traditions during coalition formation in countries as Italy, Luxembourg, The Netherlands, Belgium, and Ireland, political parties should be aware of the important role of the process of coalition formation. In this paper, we have shown how several aspects of this coalition process play an important role for the result of the coalition negotiations. We use a formal model of coalition formation which considers political parties as players with ideal policy positions and maneuvering spaces denoting their flexibility to deviate from their ideal points. The output of the model is a set of feasible coalitions, which have a majority and whose members' maneuvering spaces overlap. The model describes which coalition position will be reached by the members given the procedure adopted. The complexity of the model increases with the number of players (parties) and policy dimension.

To generate coalitions from political data, algorithms have been presented. We have introduced computational algorithms for the different procedures. The algorithms compute all winning coalitions and preferences of parties over those coalitions. Furthermore, the algorithms are used to test different hypotheses.

The analysis in de Ridder et al. (2007) focused on three aspects of coalition formation and formulated hypotheses: procedure, flexibility, and power sharing. The following questions which political parties may (and should) take into account when forming a coalition were under study: Does procedure of coalition negotiations matter? Is it more advantageous to be a first-mover in the coalition process? Is it better to be more or less flexible in coalition formation? Should we invite more parties to join to a (minimal) winning coalition or is it better to stay with the existing one(s)? Is it better to form a coalition with a stronger party or rather with a smaller one? Applying the algorithms to Dutch data and using theoretical results, we have arrived at several (counter-)examples. These counter-examples have shown the importance of the process and give important implications for political parties involved in coalition formation. Also, these results have implications for future coalition research.

From the output of the applied methods the following can be observed. First, procedure matters. When forming a coalition, political parties should be aware of the important role procedure plays in determining the result of the coalition. The calculations have shown that procedure partly determines which coalition point is agreed on. However, earlier research has analyzed that there is not one procedure which is always best (de Ridder and Rusinowska (2005)).

Related to procedure, being a first mover is not necessarily advantageous. This result is also surprising in the sense that in many countries (e.g. The Netherlands, Belgium, Luxembourg, and Austria) the tradition is that the largest party can start the negotiations and determines who will negotiate first. Being involved early in the process is considered an advantage. However, from the model it appeared that this is not always the case. The rationale here is that, by studying coalition compromises the other coalition partners will reach without a party (assuming complete information), this party can estimate whether this compromise is close to its ideal position. If it is, it may pay to join later. If the compromise

is not close, it may be better for the party to join earlier in the process.

With respect to flexibility, being less flexible is not necessarily advantageous. In the data, we have found that being less flexible results in a (pre-)coalition compromise which is closer to a party's position. So, being less flexible pays off. Nevertheless, we have presented a theoretical three-party counter-example in which being less flexible is a disadvantage. In this example, the first mover's ideal position was somewhere between the ideal position of the remaining two parties. Although being less flexible gave a better pre-coalition outcome, the final coalition position was worse for the party than the coalition position with the party being more flexible.

Related to power sharing theories (as minimal winning theory), computations show that forming a minimal winning coalition is not necessarily advantageous. Moreover, forming a coalition with a stronger party is not necessarily disadvantageous. So, it might pay off to share power with more and stronger parties than predicted by power sharing theory. To explain this counter-intuitive finding, for the minimal winning case it holds that new parties may determine a final coalition outcome closer to a party's ideal position, although this depends on the ideal positions of the new parties. For the stronger partner case, a stronger party joining usually moves the pre-coalition compromise further from a party's own ideal position. However, a strong party may determine a final coalition position which is closer to a party's position. In that case, a strong partner may be beneficial to cooperate with.

Game theoretic models like the coalition formation model allows analysis only for few player situations. The developed computational methods allow empirical testing of hypotheses using huge data sets with many players. We have provided theoretical examples and empirical cases which confirm the thesis that the coalition process matters. We aim to reach the agenda of coalition research with this message. Due to the focus on making and illustrating this message, we have neglected other aspects of the research. We suggest for future research to investigate how to empirically determine a party's flexibility, development of more dynamic coalition models, and empirical analyses of more countries.

CHAPTER 4

Methods for Computing Nash Equilibria of a Location-Quantity Game

Chapter based on article: Sáiz, M.E., Hendrix, E.M.T.(2007), Methods for computing Nash equilibria of a location-quantity game, *Accepted by Computers & Operations Research* in Press: doi:10.1016/j.cor.2007.02.022

Abstract

A two stage model is described where firms take decisions on where to locate their facility and on how much to supply to which market. In such models in literature, typically the market price reacts linearly on supply. Often two competing suppliers are assumed or several that are homogeneous, i.e. their cost structure is assumed to be identical. The focus of this paper is on developing methods to compute equilibria of the model where more than two suppliers are competing that each have their own cost structure, i.e. they are heterogeneous. Analytical results are presented with respect to optimality conditions for the Nash equilibria in the two stages. Based on these analytical results, an enumeration algorithm and a local search algorithm are developed to find equilibria. Numerical cases are used to illustrate the results and the viability of the algorithms. The methods find an improvement of a result reported in literature.

4.1 Introduction

Many studies in literature describe a so-called non-cooperative game where competing firms decide on production locations and supply quantities to markets. To make a game theoretic analysis tractable, often a limited number of suppliers are considered, or alternatively homogeneous firms and markets are assumed. We focus on situations where companies can be as well similar as not similar. In supply chains, farm cooperatives, etc., many decisions appear in which preferences cannot be assumed to be homogeneous. Also symmetric behaviour, finite strategy set or a two or few actors setting are strong assumptions in literature. Decisions are influenced by differences on prices or cost ("player" depending) between actors and between the location of the facilities. Our focus is on constructing solution methods for games in which players are: asymmetric, heterogeneous and facing multiple decisions in several stages.

Different competitive location models are available in the literature, see for instance the survey papers Eiselt and Laporte (1996), Eiselt et al. (1993), Plastria (2001) and the references therein. They vary in the ingredients which form the model. For instance, the *location space* may be the plane, a network or a discrete set.

In Cournot (1838) the idea of a Cournot oligopoly equilibrium was introduced, where two firms compete on the same market. Due to price reaction of the market on the total quantity offered, a price equilibrium appears. Hotelling (1929), added the idea of having a freedom in choice of location, where the possible location area is a simple line in between the markets. A generally applicable concept is that of a Nash equilibrium, Nash (1951), which is defined by the situation where none of the firms (players) is better off by changing its current (equilibrium) strategy. Because choice of location is usually prior to decision on quantities, in the model under consideration, this concept is applied to a supply chain study where two nested levels of decisions are at stake: that of supply quantity and location choice. The corresponding solution concept adopted is that of a *subgame perfect Nash equilibrium*, (Selten (1975)).

The basis of the model has been introduced by Bulow et al. (1985) who consider a game with 2 markets and 2 firms. Later, Farrell and Shapiro (1990) studied a game on quantity decisions with one market and n firms where decisions are simultaneous and products are homogeneous. Labbé and Hakimi (1991) consider a two-stage location-quantity simultaneous game with m markets and 2 firms with linear demand. Sarkar et al. (1997) extend these results considering a 2-stage static and simultaneous game with m markets and n firms in a network. They only consider a case with a fixed number of firms entering in the market, i.e., the quantities offered by each firm in all markets are strictly positive. Rhim et al. (2003) extend the work in Sarkar et al. (1997) by considering free entry (simultaneous and sequential) with symmetric cost (site specific) and capacity limitations. Their setting is a 3-stage game with m markets and n firms with production capacity and quantity decisions, and final stage is the location choice in a network. Recently, Dorta-González et al. (2004) apply the Stackelberg equilibrium in a two-stage non-cooperative Cournot game with location and quantity choice with n markets located at the vertices of a network and r firms. They use the Nash equilibrium concept in the location stage. In all of these studies, cases applied are small, most are symmetric, and no computational experience is reported.

This paper extends the studies in Sarkar et al. (1997) and Rhim et al. (2003). A two-stage location-quantity game with m markets and n firms is described. The location space is a network, where the nodes are considered as possible locations for the firms. Free entry is possible as in Rhim et al. (2003), i.e., the number of firms entering the markets is not known in advance, but in our case costs are asymmetric (firm-specific). We provide conditions for the supplying decisions (second-stage of the game). Moreover, as firms will be affected by

the timing and level of entry on the market, properties on how to determine the size of the market are derived. Another difference with the study of Rhim et al. (2003) is the procedure on how to find the equilibrium of the game. We consider not only the possibility of leaving a market but also the possibility of that the supplier moves its facility to another location. Doing so, a firm has to re-think the quantity decision on how much to supply to which markets. By applying the method in the cases of Sarkar et al. (1997), a mistake is found in the outcome given in their study. Their reported possible equilibrium appears to be wrong as is shown in Section 4.4. Moreover, a sequential analysis is followed in this paper. It appears that starting with the cheapest firm, one can successively arrive at the size of each of the markets. When market sizes are determined, the optimum quantities each supplier delivers to each markets they enter can be computed.

In Section 4.2, a model is outlined consisting of a non-cooperative game where quantity decisions and location decisions take place. Furthermore, theoretical results concerning the optimum decisions in these models are derived. In Section 4.3, methods for computing the Nash equilibria on quantity-location decisions and for computing the size of a market are described. A complete enumeration algorithm and a local search procedure are outlined. Numerical illustrations of model and methods can be found in Section 4.4. Finally, Section 4.5 discusses the conclusions.

4.2 Location-Quantity Game: Problem Formulation

The model describes a two-stage non-cooperative game. In the first stage of the game, firms take a simultaneous decision about where to locate a supplying facility in a network, i.e., each firm chooses a location-strategy without knowledge of the strategy chosen by the other firms. In the second stage of the game, firms decide about the quantity to be produced at these facilities and how much to supply to each market.

The model on quantity decisions and location choice is described by the following notation. Firms are denoted by an index $i \in N = \{1, \ldots, n\}$ and markets are denoted by an index $h \in M = \{1, \ldots, m\}$ each demanding a quantity of a good, depending on its price. In game theory, usually a linear price reaction model is assumed. We will follow this tradition. The demand is fulfilled by the supply of a quantity Q_{ih} from the facility of firm i to market h. The location x_i of the facility of firm i determines its marginal production cost $c_i(x_i)$. The regional dispersion effect comes in when every market appears to be situated at one location and, an important assumption, each supply firm can open a facility at only one of the locations. The relations are formalised as follows.

Let $G = (V, E)$ be an undirected graph with V and E as its sets of nodes and edges respectively, $|V| = m$. Given two nodes $v_i, v_j \in V$, $d(v_i, v_j)$ is the length of a shortest (with respect to the sum of edge lengths) path on G connecting v_i and v_j. There are m markets located each at one node on the network ; there are n firms that open a facility each at one node with $n \leq m$. Let $x_i \in V = \{v_1, \ldots, v_m\}$ be the location decision by firm i on the network. The cost of establishing a facility by firm i at x_i is $w(x_i) \geq 0$. The quantity decision matrix $\underline{Q}$ for all firms and all markets is given by:

$$\underline{Q} = \begin{pmatrix} Q_{11} & \ldots & Q_{1h} & \ldots & Q_{1m} \\ \ldots & \ldots & \ldots & \ldots & \\ Q_{i1} & \ldots & Q_{ih} & \ldots & Q_{im} \\ \ldots & \ldots & \ldots & \ldots & \\ Q_{n1} & \ldots & Q_{nh} & \ldots & Q_{nm} \end{pmatrix}$$

where the sum of a row indicates the quantity supply by firm i over all markets $h \in \{1,\ldots,m\}$, $s_i = \sum_{h=1}^{m} Q_{ih}$ and the sum of a column indicates the quantity supplied by all firms $i \in \{1,\ldots,n\}$ to market h, $q_h = \sum_{i=1}^{n} Q_{ih}$. The price $p_h(q_h)$ at market h is assumed to depend on the quantity according to the relation:

$$p_h(q_h) = \max\{0, \alpha_h - \beta_h q_h\}, \quad q_h \geq 0 \tag{4.1}$$

with price parameters $\alpha_h \geq 0$, $\beta_h > 0$. Notice that α_h is the price when quantity offered is zero, and β_h is the price reaction parameter of the inverse demand function. The price at market h depends on the quantity decision of all firms that supply to market h.

The n firms interact over two stages. In the first stage, firms simultaneously choose the locations of their facilities, x_i, $i = 1,\ldots,n$, vector $X = (x_1,\ldots,x_n)$ gives the location of the firms. In the second stage, depending on the location decisions x_i, firms choose quantities Q_{ih} to be supplied to markets, which results in the quantity decision matrix $\underline{Q}$. The profit firm i wants to maximise is denoted by $\pi_i(x_i, \underline{Q})$. A strategy for firm i at market h, $[x_i, Q_{ih}]$, comprises a choice of x_i for stage 1 and a choice of Q_{ih} for stage 2; a strategy $[x_i, Q_{i\cdot}]$, for all markets, where $Q_{i\cdot}$ denotes the row vector $(Q_{i1},\ldots,Q_{im})$.

The game is solved backwards. First the second stage is solved. Firm i chooses optimally the vector of quantities $Q_{i\cdot} = (Q_{i1},\ldots,Q_{im})$, based on what the others deliver and depending on the chosen location x_i:

$$Q^*_{i\cdot} = \arg\max_{Q_{i\cdot}} \pi_i(x_i, \underline{Q^*}(X)) \tag{4.2}$$

The game can be considered a one stage problem when matrix $\underline{Q^*}$ is defined for each location vector X. Now firm i chooses a location strategy x_i^* such that:

$$x_i^* = \arg\max_{x_i} \pi_i(x_i, \underline{Q}^*(X))$$

The unit transportation cost between the location x_i of the facility of firm i and location v_h of market h, is represented by $t_{ih} = T(d(x_i, v_h))$, where T is concave and increasing in the distance[1]. The total cost of the location and supply decision of firm i is given by:

$$\begin{aligned} TC_i(x_i, Q_{i\cdot}) &= \sum_{h=1}^{m} t_{ih}Q_{ih} + c_i(x_i)s_i + w(x_i) \\ &= \sum_{h=1}^{m} t_{ih}Q_{ih} + c_i(x_i)\sum_{h=1}^{m} Q_{ih} + w(x_i) \\ &= \sum_{h=1}^{m} (t_{ih} + c_i(x_i))Q_{ih} + w(x_i) \end{aligned}$$

For the convenience of notation we represent the total unit cost of firm i at market h by

$$TCu_{ih} = t_{ih} + c_i(x_i).$$

[1] This assumption also appears on the studies of Lederer and Thisse (1990), Labbé and Hakimi (1991), Sarkar et al. (1997) among others

Profit is denoted by π_i and defined as:

$$\pi_i(x_i, \underline{Q}) = \sum_{h=1}^{m} p_h(q_h) Q_{ih} - TC_i(x_i, Q_{i\cdot}) \tag{4.3}$$

Price at market h is given by equation (4.1). Firms determine quantities for the markets to maximize profit. Substituting the price relation of the markets into (4.3) gives

$$\pi_i(x_i, \underline{Q}) = \sum_{h=1}^{m} \max\left[\alpha_h - \beta_h \sum_{j=1}^{n} Q_{jh}, 0\right] Q_{ih} - TC_i(x_i, Q_{i\cdot}) \tag{4.4}$$

Table 4.1 summarises the notation used.

Table 4.1. *Notation*

N, M	Set of firms and markets, respectively
x_i	Location of firm i
v_h	Location of market h
Q_{ih}	Quantity supply by firm i at market h
$Q_{i\cdot} = (Q_{i1}, \ldots, Q_{im})$	Quantity decision vector for firm i
$s_i = \sum_{h \in M} Q_{ih}$	Total quantity supplied by firm i
$q_h = \sum_{i \in N} Q_{ih}$	Total quantity supply at market h
α_h, β_h	Price parameters
$p_h(q_h) = \max\{0, \alpha_h - \beta_h q_h\}, \quad q_h \geq 0$	Price at market h
$t_{ih} = T(d(x_i, v_h))$	Unit transportation cost
$w(x_i)$	Cost of establishing a centre at x_i
$c_i(x_i)$	Marginal production cost
$TCu_{ih} = t_{ih} + c_i(x_i)$	Total unit cost
$TC_i(x_i, Q_{i\cdot})$	Total cost of location and supply
$\pi_i(x_i, \underline{Q})$	Profit for firm i depending on location and quantities

In Section 4.2.1, properties are given of the equilibrium prices and quantities depending on the location decision of the firms. Section 4.2.2 describes the criterion for selecting optimal location decisions, X^*, based on the optimal quantity decisions, $\underline{Q}^*(X)$.

4.2.1 Quantity Decision

The Nash equilibrium is the solution concept used in the quantity-stage of the game. From (4.2), the Nash elements of the $\underline{Q}$ matrix can be determined by an iterative process. Nash equilibrium quantities shipped by firm i to market h follow from the first order condition optimising (4.4) over Q_{ih}:

$$Q^*_{ih} = \max\left\{0, \frac{\alpha_h - \beta_h \sum_{j=1, j\neq i}^{n} Q^*_{jh} - t_{ih} - c_i(x_i)}{2\beta_h}\right\} \tag{4.5}$$

This means that the equilibrium quantity Q^*_{ih} can be either 0 or positive. In the remaining we will study for which firms the quantity Q^*_{ih} is positive and derive the exact quantity. First, we distinguish for each market h between two groups: A_h with firms delivering to h,

$Q^*_{ih} > 0$; and $\overline{A}_h = N \backslash A_h$ with firms not delivering to h, $Q^*_{ih} = 0$.

$$\begin{cases} Q^*_{ih} > 0 & \text{for} \quad i \in A_h \\ Q^*_{ih} = 0 & \text{for} \quad i \in \overline{A}_h \end{cases}$$

Proposition 1 provides the equilibrium quantity for each firm $i \in A_h$.

Proposition 1. *Let A_h be the set of firms which supply market h, $|A_h| = k_h$. The positive equilibrium quantities are given by*

$$Q^*_{ih} = \frac{\alpha_h - k_h(c_i(x_i) + t_{ih}) + \sum_{j \in A_h \backslash \{i\}} (c_j(x_j) + t_j)}{(k_h + 1)\beta_h} \tag{4.6}$$

with $Q^*_{ih} > 0 \quad \forall i \in A_h$. Q^*_{ih} depends on production and transportation cost of the active suppliers.

Proof. From equation (4.5) follows for $i \in A_h$

$$Q^*_{ih} = \frac{\alpha_h - t_{ih} - c_i(x_i)}{2\beta_h} - \frac{1}{2} \sum_{j \in A_h \backslash \{i\}} Q^*_{jh} \tag{4.7}$$

Let $a_{ih} = (\alpha_h - t_{ih} - c_i(x_i)) / (2\beta_h)$, then (4.7) can be written as

$$Q^*_{ih} = a_{ih} - \frac{1}{2} \sum_{j \in A_h \backslash \{i\}} Q^*_{jh}$$

In vector notation

$$\begin{pmatrix} Q^*_{1h} \\ \cdots \\ Q^*_{ih} \\ \cdots \\ Q^*_{k_h h} \end{pmatrix} = \begin{pmatrix} a_{1h} \\ \cdots \\ a_{ih} \\ \cdots \\ a_{k_h h} \end{pmatrix} - \frac{1}{2} \left[\underline{1_{k_h}} \underline{1'_{k_h}} - \underline{I} \right] \begin{pmatrix} Q^*_{1h} \\ \cdots \\ Q^*_{ih} \\ \cdots \\ Q^*_{k_h h} \end{pmatrix}$$

$$\underline{Q^*}_h = \underline{a}_h - \frac{1}{2} \left[\underline{1_{k_h}} \underline{1'_{k_h}} - \underline{I} \right] \underline{Q^*}_h$$

where $\underline{1_{k_h}}$ is the all ones vector and $\underline{I}$ is the $k_h \times k_h$ unit matrix. By linear algebra,

$$\underline{I} \underline{Q^*}_h = \underline{a}_h - \frac{1}{2} \left(\underline{1_{k_h}} \underline{1_{k_h}}' - \underline{I} \right) \underline{Q^*}_h$$

$$\underline{a}_h = \frac{1}{2}\left[\underline{1}_{k_h}\underline{1}'_{k_h} + \underline{I}\right]\underline{Q^*}_h$$

then, $\underline{Q^*}_h$ can be written as

$$\underline{Q^*}_h = \underline{B}^{-1}\underline{a}_h \tag{4.8}$$

where $\underline{B}$ is the $k_h \times k_h$ matrix

$$\underline{B} = \frac{1}{2}\left[\underline{1}_{k_h}\underline{1}'_{k_h} + \underline{I}\right]$$

having the following form,

$$\underline{B} = \begin{pmatrix} 1 & \dots & 1/2 & \dots & 1/2 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & 1/2 & \dots \\ 1/2 & \dots & 1 & \dots & 1/2 \\ \dots & 1/2 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 1/2 & \dots & 1/2 & \dots & 1 \end{pmatrix}$$

The inverse matrix can be derived to be,

$$\underline{B}^{-1} = 2\left(\underline{I} - \frac{1}{k_h+1}\underline{1}_{k_h}\underline{1}'_{k_h}\right)$$

$$\underline{B}^{-1} = 2\begin{pmatrix} k_h/k_h+1 & \dots & -1/k_h+1 & \dots & -1/k_h+1 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & -1/k_h+1 & \dots \\ -1/k_h+1 & \dots & k_h/k_h+1 & \dots & -1/k_h+1 \\ \dots & -1/k_h+1 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ -1/k_h+1 & \dots & -1/k_h+1 & \dots & k_h/k_h+1 \end{pmatrix}$$

The equivalence of equations (4.5) and (4.6) for each market h now follows from (4.8):

$$\underline{Q^*}_h = \underline{B}^{-1}\underline{a}_h = 2\left[\underline{I} - \frac{1}{k_h+1}\underline{1_{k_h}}\underline{1'_{k_h}}\right]\underline{a_h} =$$

$$= 2\begin{pmatrix} k_h/k_h+1 & \dots & -1/k_h+1 & \dots & -1/k_h+1 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & -1/k_h+1 & \dots \\ -1/k_h+1 & \dots & k_h/k_h+1 & \dots & -1/k_h+1 \\ \dots & -1/k_h+1 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ -1/k_h+1 & \dots & -1/k_h+1 & \dots & k_h/k_h+1 \end{pmatrix}\begin{pmatrix} a_{1h} \\ \dots \\ a_{ih} \\ \dots \\ a_{k_h h} \end{pmatrix}$$

and for each firm i we obtain:

$$Q^*_{ih} = \frac{2k_h}{k_h+1}a_{ih} - \frac{2}{k_h+1}\sum_{j\in A_h\setminus\{i\}} a_{jh}$$

$$= \frac{2k_h}{k_h+1}\frac{\alpha_h - t_{ih} - c_i(x_i)}{2\beta_h} - \frac{2}{k_h+1}\sum_{j\in A_h\setminus\{i\}}\frac{\alpha_h - (t_{jh}+c_j(x_j))}{2\beta_h}$$

$$= \frac{2k_h\alpha_h - 2k_h(t_{ih}+c_i(x_i))}{2(k_h+1)\beta_h} - \frac{(k_h-1)\alpha_h}{(k_h+1)\beta_h} + \sum_{j\in A_h\setminus\{i\}}\frac{t_{jh}+c_j(x_j)}{(k_h+1)\beta_h}$$

$$= \frac{\alpha_h - k_h(t_{ih}+c_i(x_i)) + \sum_{j\in A_h\setminus\{i\}}(t_{jh}+c_j(x_j))}{(k_h+1)\beta_h}$$

which corresponds to equation (4.6). □

Consequently, the total quantity supplied to market h is:

$$q^*_h = \sum_{j\in A_h} Q^*_{jh} = \frac{1}{(k_h+1)\beta_h}\left(k_h\alpha_h - \sum_{j\in A_h}(c_j(x_j)+t_{jh})\right) \tag{4.9}$$

which means that higher average marginal cost and transportation costs decrease the total quantity supplied. The optimal price at each market can now be derived by substituting (4.9) into (4.1):

$$p^*_h = \frac{1}{k_h+1}\left(\alpha_h + \sum_{j\in A_h}(c_j(x_j)+t_{jh})\right) \tag{4.10}$$

Optimal prices at each market proportionally rise with average marginal cost and transporta-

tion cost over the firms supplying the market. Higher costs leads to a higher equilibrium price and lower costs leads to higher quantity supplied.

In order to have any delivery at market h in (4.9) a necessary condition is that $\exists j \in N$ such that $TCu_{jh} < \alpha_h$. From Proposition 1 also follows the result for the symmetric case.

Theorem 1. *Let unitary costs be symmetric (the same) for all the suppliers at market h. If unitary costs are lower than α_h, all the suppliers will enter the market.*

Proof. Let the n firms entering the market have the same cost, $TCu_{1h} = TCu_{2h} = \ldots = TCu_{nh} = Cu_h$ and $Cu_h < \alpha_h$, the optimal quantity and price can be derived from equation (4.6) and (4.10),

$$Q^*_{ih} = \frac{\alpha_h - nTCu_{ih} + \sum_{j=1, j\neq i}^{n} TCu_{jh}}{(n+1)\beta_h} = \frac{\alpha_h - Cu_h}{(n+1)\beta_h} > 0$$

and

$$p^*_h = \alpha_h - \beta_h q_h = \alpha_h - \beta_h n \frac{\alpha_h - Cu_h}{(n+1)\beta_h} = \frac{\alpha_h + nCu_h}{n+1}$$

□

Corollary 2. *Let unitary costs be symmetric (the same) for all the suppliers at market h. If unitary costs are higher than α_h, no suppliers will enter the market.*

From Proposition 1 can also be derived when a firm would be interested to enter market h, given that a set of firms A_h is already delivering.

Proposition 2. *Let A_h be a set of firms supplying market h. A firm i is interested in supplying market h if $TCu_{ih} < p^*_h$.*

Proof. Follows from the partial derivative of π_i with respect to Q^*_{ih} for $Q^*_{ih} = 0$. □

Proposition 3. *In the optimum $\underline{Q}^*$, $\forall i \in A_h$, $TCu_{ih} < p^*_h$.*

Proof. From equation (4.10) the equilibrium price is

$$p^*_h = \frac{1}{k_h + 1}\left(\alpha_h + \sum_{j \in A_h} (c_j(x_j) + t_{jh})\right) = \frac{1}{k_h + 1}\left(\alpha_h + \sum_{j \in A_h} TCu_{jh}\right)$$

From equation (4.6) equilibrium quantities are given by

$$\begin{aligned}
Q^*_{ih} &= \frac{\alpha_h - k_h(c_i(x_i) + t_{ih}) + \sum_{j\in A_h\setminus\{i\}}(c_j(x_j) + t_j)}{(k_h+1)\beta_h} \\
&= \frac{\alpha_h - k_h TCu_{ih} + \sum_{j\in A_h\setminus\{i\}} TCu_{jh} + TCu_{ih} - TCu_{ih}}{(k_h+1)\beta_h} \\
&= \frac{\alpha_h - (k_h+1)TCu_{ih} + \sum_{j\in A_h\setminus\{i\}} TCu_{jh} + TCu_{ih}}{(k_h+1)\beta_h} \\
&= \frac{\alpha_h - (k_h+1)TCu_{ih} + \sum_{j\in A_h} TCu_{jh}}{(k_h+1)\beta_h} \\
&= \frac{\alpha_h + \sum_{j\in A_h} TCu_{jh}}{(k_h+1)\beta_h} - \frac{(k_h+1)TCu_{ih}}{(k_h+1)\beta_h} \\
&= \frac{p^*_h}{\beta_h} - \frac{TCu_{ih}}{\beta_h} = \frac{p^*_h - TCu_{ih}}{\beta_h}
\end{aligned}$$

From equation (4.6), at equilibrium $Q^*_{ih} > 0 \quad \forall i \in A_h$, $(p^*_h - TCu_{ih})/\beta_h > 0$ such that $p^*_h > TCu_{ih}$. □

Consequently, for $i \in A_h$

$$c_i(x_i) + t_{ih} < \frac{1}{|A_h|+1}\left(\alpha_h + \sum_{j\in A_h}[c_j(x_j) + t_{jh}]\right)$$

For all $j \notin A_h$, $Q^*_{jh} = 0$ and

$$c_j(x_j) + t_{jh} \geq \frac{1}{|A_h|+1}\left(\alpha_h + \sum_{i\in A_h}[c_i(x_i) + t_{ih}]\right)$$

Proposition 4. *The relation between the firm with the highest total unit costs in the active set, $i \in A_h$, with any firm $j \in \overline{A}_h$ which is not entering the market is*

$$TCu_{ih} < \frac{\alpha_h + \sum_{r\in A_h} TCu_{rh}}{|A_h|+1} \leq TCu_{jh}$$

Proof. The first inequality follows from $TCu_{ih} < p_h$ and is satisfied by any firm in the active set A_h. The last inequality is satisfied by any firm $j \in \overline{A}_h$ following from $TCu_{jh} \geq p_h$. □

Proposition 4 shows that

$$\max_{i\in A_h} TCu_{ih} < p_h \leq \min_{j\in\overline{A}_h} TCu_{jh}$$

This is used in the algorithms in Section 4.3 to determine the number of active firms $|A_h|$.

Firms are ordered on the basis of total unit costs, such that $TCu_{(1)h} \leq TCu_{(2)h} \leq \ldots \leq TCu_{(n)h}$. The rule that is used is the following

1. Initialise $p = \alpha$, $|A| = 0$.
2. while $TCu_{(k)} < p$, (k) enters the market and the price is updated.

More details of the algorithms are given in Section 4.3.

Let M_i be the set of markets in which firm i is active, $M_i = \{h \in M | i \in A_h\}$. The total quantity supplied by each firm is

$$s_i = \sum_{h \in M_i} Q^*_{ih} = \sum_{h \in M_i} \frac{\alpha_h - k_h(c_i(x_i) + t_{ih}) + \sum_{j \in A_h \setminus \{i\}}(c_j(x_j) + t_{jh})}{(k_h + 1)\beta_h}$$

Total cost for each firm is

$$TC_i = \sum_{h \in M_i} (c_i(x_i) + t_{ih}) \frac{\alpha_h + \sum_{j \in A_h \setminus \{i\}}(c_j(x_j) + t_{jh}) - k_h(c_i(x_i) + t_{ih})}{(k_h + 1)\beta_h} + w(x_i)$$

Using (4.6) and (4.9), the final payoff for each firm given location vector X is:

$$\begin{aligned}
\pi_i(X) &= \sum_{h \in M_i} (p^*_h - (c_i(x_i) + t_{ih})Q^*_{ih} - w(x_i) \\
&= \sum_{h \in M_i} \frac{\left[\alpha_h + \sum_{j \in A_h}(c_j(x_j) + t_{jh}) - (k_h + 1)(c_i(x_i) + t_{ih})\right]^2}{(k_h + 1)^2\beta_h} - w(x_i) \\
&= \sum_{h \in M_i} \frac{\left[\alpha_h + \sum_{j \in A_h \setminus \{i\}}(c_j(x_j) + t_{jh}) - n(c_i(x_i) + t_{ih})\right]^2}{(k_h + 1)^2\beta_h} - w(x_i) \\
&= \sum_{h \in M_i} \beta_h (Q^*_{ih})^2 - w(x_i)
\end{aligned}$$

Concluding, the optimum Q^*_{ih}, q^*_h and p^*_h in equation (4.6), (4.9) and (4.10), respectively, is a Nash Equilibrium for the competitive second stage of the game given location vector X.

4.2.2 Location Decision

Given the optima of the second stage, focus is on the first stage of the game. Considering the equilibrium supply quantity choice in the second stage, $Q^*(X)$, each firm i maximizes the profit function π_i by selecting a location on the network. Firms locate at one of the nodes of the network. We assume that several firms can be located at the same site. At equilibrium, no other location decision is better off for each firm.

The strategy $X^* = (x^*_1, \ldots, x^*_n)$ is a **Nash Equilibrium** if for each firm i, x^*_i is the best response to the strategies specified by the $n - 1$ other firms:

$$\pi_i(x^*_i, \underline{Q}^*(X^*)) \geq \pi_i(x_i, \underline{Q}^*(\hat{X})) \quad \text{with} \quad \hat{X} = (x^*_1, \ldots, x_i, \ldots, x^*_n) \quad \forall \ x_i$$

for every feasible strategy x_i. That is, x_i^* solves

$$\max_{x_i} \pi_i(x_i, \underline{Q}^*(\hat{X}))$$

The method and algorithms used to select optimal locations and quantities for the firms are described in Section 4.3.

4.3 Methods for Computing Nash Quantities and Nash Locations

The results of Section 4.2 can be used to find equilibria of the two level game. A procedure is described that solves the quantity game given a location configuration. This procedure can be used in algorithms to find equilibria for the location decision. The idea is to do a global search to find all Nash equilibria.

Global search methods include complete (enumerative) search strategies and stochastic search algorithms (e.g., pure random search and multistart, see Hendrix (2007)).

First, we describe an algorithm that systematically enumerates all location possibilities, for which equilibrium quantities are computed. After that, it tries to detect which location vectors correspond to a Nash equilibrium by checking whether it is better for a firm to relocate its facility. Notice that all m^n location configurations are generated. This is called a full enumeration.

Second, a multistart algorithm based on an application of Teitz and Bart location-allocation heuristic is described (see Teitz and Bart (1968))[2]. *Efficiency* and *effectiveness* are discussed in comparison to the complete enumeration algorithm.

Algorithm 4.1 Algorithm for searching Nash equilibria.

Funct MAIN(Number of firms n; number of markets m; parameters α and β ; distance matrix $d(v_i, v_j)$; marginal costs $c_i(x_i)$, opening costs $w(x_i)$)

1. *all possible locations:* $L \leftarrow m^n$
2. Generate location matrix with rows X_l iteratively
3. **for** each location l
4. $\quad Q_l^* :=$ **Quantity**(X_l)
5. $\quad \Pi_l^* :=$ **Profit**(X_l, Q_l^*)
6. $E^* :=$ **Equilibria**(Π^*)
7. OUTPUT: {Nash equilibria of the non-cooperative game}

The enumeration algorithm is sketched in "Algorithm for searching equilibria" (Algorithm 4.1) which calls iteratively to a subroutine called `Quantity`. This procedure computes the Nash equilibrium quantities for each location vector based on the size of the market and equilibrium price (line 4 in Algorithm 4.1). Once the optimal quantities have been determined, the subroutine called `Profit` computes the profit for the firms at all the possible location vectors based on Nash quantities (line 5). The output is the profit (payoff) matrix Π. Finally, a subroutine determining the Nash equilibria on location decisions, called `Equilibria` (line 6), is described.

Procedure `Quantity` for computing Nash equilibrium on Quantities
Procedure `Quantity` is called by Algorithm 4.1 for each of the possible location vectors for

[2]We thank an anonymous referee for inviting us to consider this algorithm

the suppliers. Every time the procedure is called, total unit costs are computed for each firm at each market and ordered, $TCu_{(1)h}, \ldots, TCu_{(n)h}$ (`Sort` in line 3). Results derived in Section 4.2.1 are applied to compute optimal quantities. The computation generates these by the following two procedures:

Algorithm 4.2 Quantity(X) : Procedure to compute Nash equilibrium quantities, Q^*.

Funct Quantity(location vector X and global variables)

1. **for** $h \in M$
2. $\quad TCu_{.h} := t_{.h} + c(X)$ ▷ *Total unit cost*
3. $\quad STCu_{.h} := \mathbf{Sort}(TCu_{.h})$ ▷ *Order the firms on total unit costs*
4. $\quad [k_h, p_h] := \mathbf{SizeMarket}(STCu_{.h}, \alpha_h, k_h, p_h, A_h)$
5. $\quad Q^*_{.h} := \mathbf{OptQ}(STCu_{.h}, p_h, \alpha_h, \beta_h, k_h, A_h)$
6. OUTPUT: {Nash equilibrium Quantity decisions: Q^*}

1. Procedure `SizeMarket` (Algorithm 4.3): this procedure determines the size of the active set A_h (Proposition 4) and the equilibrium price p^*_h;
2. Procedure `OptQ` (Algorithm 4.4): this procedure computes optimal quantities for firms entering the market h. From equation (4.6) (Proposition 1) and depending on the asymmetry of the firms, the method finds the optimal quantities for the active firms.

Algorithm 4.3 SizeMarket(TC, α, k, p, A): Procedure to determine the size k of market h and equilibrium price p.

Funct SizeMarket(Total unitary costs TC, α)

1. $p := \alpha$ ▷ *Initial price at the market*
2. $k := 0$ ▷ *Initial size of the market*
3. **while** $k \leq n$ and $TC_{k+1} < p$
4. $\quad k := k + 1$
5. $\quad p := \frac{\alpha + \sum_{j=1}^{k} TC_j}{(k+1)}$
6. OUTPUT: {Size and price of market h: k, p; and active set, A}

Algorithm 4.4 OptQ($TC, p, \alpha, \beta, k, A$): Procedure to compute the optimal quantities for firms at market h.

Funct OptQ(Total unitary costs TC,price at market, p, parameters α, β and size of the market, k, active set, A)

1. **if** $k == 0$
2. $\quad Q := 0 \ \ \forall i$
3. **else**
4. $\quad$ **for** $i \in A$
5. $\quad\quad Q_i := \frac{\alpha - (k+1)TC_i + \sum_{j=1}^{k} TC_j}{(k+1)\beta}$
6. $\quad$ **for** $i \in \overline{A} = N \backslash A$
7. $\quad\quad Q_i := 0$
8. OUTPUT: {Optimal quantities: Q^*}

Algorithm Location Stage

In the location-stage, the problem is to maximize profit by selecting a node where to locate

the facility in the network. Given n supplier firms and m markets, the feasible set $\underline{X}$ has $L = m^n$ elements. A Nash equilibrium can be identified for the first stage game by testing each element of $\underline{X}$ in the following way. Consider location vector $X_l = (x_{l1}, \ldots, x_{li}, \ldots, x_{ln})$. Note that it is possible that $x_{li} = x_{lj}$ for $i \neq j \in N$. For each firm, one should test whether firm i located at x_{li} is better off leaving its current location choosing another, $x_{ki} \in V$ with $k \neq l$ and $k \leq L$. For this, one should check the profit of the firm at another location configuration which is in the so-called *neighbourhood* of X_l. The concept of neighbourhood in this context has been introduced in Sáiz et al. (2006). This step generates a set of possible configurations where only firm i has a different location with respect to X_l (step 4 in Algorithm 4.5). The firm knows that its decision could generate changes in the quantity choice by the other supplier firms. If none of the other locations is profitable for firm i, this firm is so-called "at equilibrium" with respect to its current location decision, x_{li}. As values of n and m are given in the input of the algorithm, the set of feasible locations is finite and the search process stops with an optimum, equilibrium, if it exists.

If firm i finds a location $x_{ki} \in V, x_{ki} \neq x_{li}$, in which it is better off than at x_{li}, then location vector $X_l = (x_{l1}, \ldots, x_{li}, \ldots, x_{ln})$ is not at equilibrium. Firm i will prefer to move and to locate its facility at x_{ki}. The procedure follows with another possible vector location with all the possible deviations, i.e. from its neighbourhood. If no firm j improves by moving to another site from x_{lj} $(\forall j \in N)$, then X_l is at equilibrium. The procedure proceeds checking all possible location vectors one by one whether or not it is a Nash equilibrium. In this way, all Nash equilibria (if many) are found by the complete enumeration algorithm.

Algorithm 4.5 Equilibria(Π) : Procedure to compute all the equilibria in the game.

Funct Equilibria(Profit Matrix, $\Pi(1 \ldots L, 1 \ldots n)$, for all location configurations in $X(1 \ldots L, 1 \ldots n)$ and all firms)

1. **for** each location configuration l
2. *whether location vector X_l is at equilibrium:* $Eq := TRUE$
3. **while** $i \leq n$ and Eq is $TRUE$
4. $\varepsilon :=$ set of neighbour configurations w.r.t. X_l for firm i
5. $s := 1$
6. **while** $s <= m - 1$ and Eq is $TRUE$
7. $alt := \varepsilon(s)$ *alt is an alternative location*
8. $a :=$ index for neighbour configuration, $a \in \{1, \ldots, L\}$
9. **if** $\Pi(l, i) \geq \Pi(a, i)$
10. $s := s + 1$;
11. **else**
12. $Eq := FALSE$;
13. OUTPUT: {Set of Equilibria}

Algorithm Multistart

The basic idea of the multistart methods is to do a local search procedure given a random starting point. Algorithm 4.6 is based on Teitz and Bart location-allocation heuristic in Teitz and Bart (1968). A random location vector is generated and it is used as the input in the local search of Algorithm 4.7. Algorithm 4.7 starts with the randomly generated location and "walks" to an equilibrium if it exists. The idea of the local search algorithm is as follows. From the original random location, the algorithm selects from its neighbourhood the most motivated firm to move, that is, the algorithm compares for all firms the payoff at the original location with the payoff at the neighbour. Notice that each firm has $m - 1$ alternatives, such that each location vector has $n \times (m - 1)$ neighbours. The algorithm

Algorithm 4.6 MultiStart($max, maxiter$) : Multistart computing equilibria.

Funct MultiStart($max, maxiter$, maximum number of random configurations to generate and maximum number of iterations in the Local Search procedure (in case of no equilibrium))

1. **for** $numiter = 1$ to max
2. Generate random location vector configuration X_r
3. $X^* :=$ **LocalSearch**$(X_r, maxiter)$
4. OUTPUT: {Set of Equilibria}

moves to that location vector from its neighbourhood, where one of the firms has the biggest gain. It proceeds from the "original" location vector until an equilibrium is found. At each iteration (location vector) quantities and payoffs have to be computed, which means that the number of (quantity) functions evaluations is $n \times (m-1)$. In contrast with the complete enumeration algorithm, it is not known whether or not all equilibria (if many) are found.

Algorithm 4.7 LocalSearch($X_r, maxiter$) : Local Search to compute equilibria in the game given a starting location vector X_r.

Funct LocalSearch(Location vector X_r)

1. $iter := 0$
2. **while** not Eq AND $iter \leq maxiter$
3. $iter := iter + 1$
4. $Q^*(X_r) :=$ QUANTITY(X_r) ▷ *Compute* Q^* *for* X_r
5. $\Pi_{best} :=$ PROFIT$(X_r, Q^*(X_r))$ ▷ *Compute* Π^* *for* X_r
6. **for** $i = 1$ to n
7. **for** $j = 1$ to $m - 1$
8. $X_a :=$ neighbour w.r.t alternative j for firm i;
9. *Compute* Q^* *for neighbour*
10. $Q^*(X_a) :=$ QUANTITY(X_a)
11. *Compute* Π^* *for neighbour*
12. $\Pi_a :=$ PROFIT$(X_a, Q^*(X_a))$
13. $Pm_j := [\Pi_a(i) - \Pi_{best}]$
14. $IMP_i := \max_j\{Pm_j\}$
15. $gain := \max_i\{IMP_i\}$
16. **if** $gain < 0$
17. Equilibrium found. Exit.
18. **else**
19. $X_r :=$ neighbour location vector X_a corresponding to $gain$
20. OUTPUT: {Equilibrium}

Effectiveness and Efficiency

Two questions: do the procedures reach the Nash equilibria and at what computational cost? As defined in Hendrix (2007), *efficiency* is the effort the algorithm needs to be successful. Several efficiency indicators appear in literature. Usual indicators are the number of function evaluations necessary to reach the optimum and memory requirements. *Effectiveness*: does the algorithm find what we want? Two targets can be distinguished in this paper:

- To discover all Nash equilibria
- To detect at least one (or a number k) of equilibria

The enumeration aims at finding all equilibria. A local search aims at finding one equilibrium given a starting point. The enumeration algorithm generates $L = m^n$ location vectors for which the quantity (and profit) are evaluated and stored. The number of comparisons within

the $\underline{\Pi}$ matrix has an upper bound of $L \times n \times (m-1)$. The multistart also attempts to find all equilibria. The number of equilibria that are generated (possibly many times) is max, a parameter set by the user, and coincides with the number of local searches. The number of iterations for each local search has to be measured empirically. At each iteration $n \times (m-1)$ evaluations are performed of the quantity routine.

In Section 4.4, we will measure for some illustrative examples which equilibria are found against what computational costs.

4.4 Numerical Illustration

Two cases are elaborated to illustrate the procedures and the analytical results. The first case is taken from in Sarkar et al. (1997) with $n = 3$ firms and $m = 6$ markets. A mistake in the output in Sarkar et al. (1997) is found when the algorithms outlined in the last section are applied. In their study an extra location vector is obtained as equilibrium. In Section 4.4.1 we show why this location vector can not be an equilibrium. The second numerical example consists of 4 different cases. It is used to show the viability of the algorithm when bigger and more sophisticated cases are applied. This is illustrated in Section 4.4.2.

4.4.1 Network with 6 Markets and 3 Firms

The location decisions are represented by the node of the market in which firms are located (for example, $(1, 2, 1)$ means firms one and three are located at the same node of market 1 and firm two at market 2, see Figure 4.1). Market locations are denoted by $v_h, h = \{1, \ldots, 6\}$.

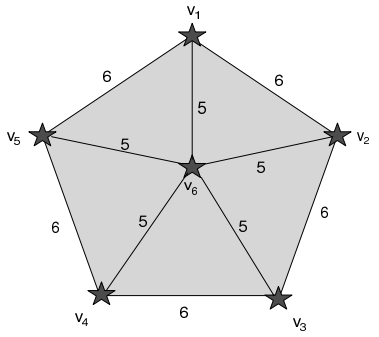

Figure 4.1. *Network for Example 1.*

In this example, $T(d(x_i, v_h)) = d(x_i, v_h)$ for all $x_i, v_h \in V$. At each of the vertices h, marginal cost are $c_i(x_i) = 10$ and $\beta_h = 1$. Sarkar et al. (1997) describe four different configurations for parameter $\alpha = (\alpha_1, \ldots, \alpha_6)$. The algorithms described in Section 4.3 are coded in Fortran and applied to each of the cases.

At equilibrium, for each configuration α a location vector and all its permutations are Nash equilibria, because all c_i have the same value. Table 4.2 shows the location vectors at equilibrium for each configuration of α and Table 4.3 shows the number of entrants at each of the markets (size of the markets) and the corresponding payoffs.

Only in the case of the last α-configuration there is no firm supplying market number 6. In all other cases, all firms supply all markets. This shows that Sarkar et al. (1997) choose

Table 4.2. *Location equilibria for example 1.*

Configuration α	Corresponding Nash location X^*
$(50, 50, 400, 400, 50, 250)$	$(3,4,6)$,$(3,6,4)$,$(4,3,6)$,$(4,6,3)$,$(6,3,4)$,$(6,4,3)$
$(50, 50, 500, 500, 40, 50)$	$(3,3,4)$,$(3,4,3)$,$(4,3,3)$
$(50, 500, 50, 50, 500, 50)$	$(6,6,6)$
$(1000, 1000, 1000, 1000, 1000, 0)$	$(6,6,6)$

Table 4.3. *Size of markets and profits of firms for the equilibria.*

Configuration	Location equilibria	No. of entrants $\|A_h\|$						Profit		
α	x_l^*	v_1	v_2	v_3	v_4	v_5	v_6	f_1	f_2	f_3
1	$(3,4,6)$	3	3	3	3	3	3	22392	22392	22388
2	$(3,3,4)$	3	3	3	3	3	3	29895	29895	29901
3	$(6,6,6)$	3	3	3	3	3	3	29733	29733	29733
4	$(6,6,6)$	3	3	3	3	3	0	303200	303200	303200

relatively easy configurations; they did not have to determine the number of active firms. For the first α-configuration and all the equilibria, the maximum total unit costs for a firm is $\max_{i\in N} TCu_{ih} = \max_{i\in N}[c_i(x_i) + t_{ih}] = 20$ ($c_i(x_i) = 10\ \forall i \in N$, $\max_{i\in N} t_{ih} = 10$) and $\min_{h\in M}[(\alpha_h + \sum_{i\in N} TCu_{ih})/(n+1)] = 22.5$, then from Proposition 4 all firms will supply all markets:

$$\max_{h\in M}\max_{i\in N} TCu_{ih} = 20 < 22.5 = \min_{h\in M} \frac{\alpha_h + \sum_{i\in N} TCu_{ih}}{n+1}$$

The same applies for the second and third α-configurations,

$$\max_{h\in M}\max_{i\in N} TCu_{ih} = 20 < 23.75 = \min_{h\in M} \frac{\alpha_h + \sum_{i\in N} TCu_{ih}}{n+1}$$

$$\max_{h\in M}\max_{i\in N} TCu_{ih} = 15 < 20 = \min_{h\in M} \frac{\alpha_h + \sum_{i\in N} TCu_{ih}}{n+1}$$

For the last α-configuration and markets 1 to 5,

$$\max_{h\in M}\max_{i\in N} TCu_{ih} = 15 < 261.25 = \min_{h\in\{1\ldots,5\}} \frac{\alpha_h + \sum_{i\in N} TCu_{ih}}{n+1}$$

In case of market 6, no firm will supply since from Proposition 2, firm i ($\forall i$) is not interested in supplying market 6 since $TCu_{i6} = 10\ \forall i$, and the initial price at market 6 is $p_6 = \alpha_6 = 0$, such that $TCu_{i6} > p_6$.

For the second α-configuration, Sarkar et al. (1997) describe an additional stable location for the firms, namely $(3,4,4)$ and its corresponding permutations. Our algorithm does not find that location vector as equilibrium. This can be seen as follows. Consider the profits for the firms locating at $(3,4,4)$:

$$\begin{aligned}\text{Firm } 1 &\longmapsto 29926\\ \text{Firm } 2 &\longmapsto 29880\\ \text{Firm } 3 &\longmapsto 29880\end{aligned}$$

A firm is in Nash equilibrium if it does not have an incentive to move to another location. Consider firm 2 and suppose the others do not change location. Five possible strategies

should be evaluated to determine a possible improvement of the profit. Evaluation of π_2 for $x_2 \in \{v_1, \ldots, v_6\}$ results in Table 4.4.

Table 4.4. *Firm 2 profits for each location while firm 1 and 3 are fixed.*

	Equilibrium in Sarkar et al. (1997)	Alternatives for firm 2				
Location	$(3,4,4)$	$(3,1,4)$	$(3,2,4)$	$(3,3,4)$	$(3,5,4)$	$(3,6,4)$
Profit firm 2	29880	27585	28251	29895	28214	29368

One can observe that firm 2 is better off changing strategy by moving to market 3. This means that $(3,4,4)$ is not an equilibrium as wrongly concluded in Sarkar et al. (1997).

A check was made running the local search algorithm with this "wrong" equilibria as starting point. After two iterations equilibrium location vector $(3,3,4)$ was reached. Two iterations lead to $2 \times n \times (m-1) = 36$ (quantity) function evaluations.

The complete enumeration calls the quantity routine $6^3 = 216$ times and stores all payoffs in a matrix of 216×3. Furthermore, many comparisons are made in the global $do - loop$.

The multistart algorithm has been run for 60 random generated starting location configurations for the cases. Depending of the starting location, the algorithm "walks" to one of the equilibria in table 4.2. Table 4.5 shows average, standard deviation and the best number of iterations of the local search procedure for the different α values. The last column of table 4.5 shows the average and standard deviation for the number of (quantity) function evaluations.

Table 4.5. *Multistart algorithm. Number of iterations and function evaluations for the different α-configurations (Cases from Sarkar et al. (1997))*

Configuration α	Avg (SD)	Best Nr of iterations	Function evaluations Avg (SD)
$(50, 50, 400, 400, 50, 250)$	3.83(1.10)	1	57.50(16.47)
$(50, 50, 500, 500, 40, 50)$	3.45(0.74)	2	51.75(11.10)
$(50, 500, 50, 50, 500, 50)$	3.63(0.58)	2	54.50(8.65)
$(1000, 1000, 1000, 1000, 1000, 0)$	3.47(0.67)	1	52.00(10.05)

The number of function evaluations depends on the number of neighbours. For each neighbour a function evaluation is computed. Notice that different location configurations lead to the same neighbour. For example, location configurations $(2,6,4)$ and $(1,5,4)$ have as a neighbour $(1,6,4)$ for firm 1 and 2 respectively, which means that if these vectors are starting location configurations, the (quantity) function evaluation for $(1,6,4)$ is called double. This means that the same location vector is repeatedly evaluated several times. In theory, one does not know whether all equilibria have been detected. For this example, the algorithm did find all of them. Table 4.6 shows the number of times each equilibrium is found for the different α-configurations. The unique equilibrium $(6,6,6)$ for the last two α-configurations was found at each of the 60 repetitions.

4.4.2 Network with 15 Markets and 5 Firms

For the illustration of the viability of the algorithms, data are generated for 15 markets and four cases have been studied with 5, 4, 3 and 2 firms, respectively. Figure 4.2 shows the network and table 4.7 shows the location points of the markets.

Table 4.6. *Multistart algorithm. Times equilibria are found for the different α-configurations (Cases from Sarkar et al. (1997))*

	Equilibria and number of times is found					
$\alpha = (50, 50, 400, 400, 50, 250)$	$(3, 4, 6)$	$(3, 6, 4)$	$(4, 3, 6)$	$(4, 6, 3)$	$(6, 3, 4)$	$(6, 4, 3)$
Nr. of repetitions	8	10	3	10	13	16
$\alpha = (50, 50, 500, 500, 40, 50)$	$(3, 3, 4)$	$(3, 4, 3)$	$(4, 3, 3)$			
Nr. of repetitions	21	18	21			

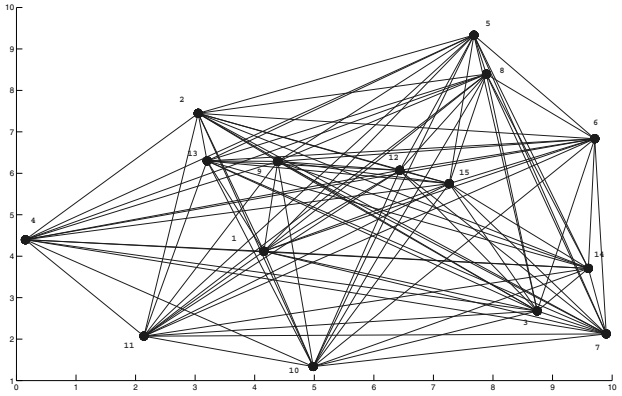

Figure 4.2. *Network for Example 2.*

Table 4.7. *Location of 15 markets randomly generated.*

v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
$\begin{pmatrix} 4.1537 \\ 4.1195 \end{pmatrix}$	$\begin{pmatrix} 3.0500 \\ 7.4457 \end{pmatrix}$	$\begin{pmatrix} 8.7437 \\ 2.6795 \end{pmatrix}$	$\begin{pmatrix} 0.1501 \\ 4.3992 \end{pmatrix}$	$\begin{pmatrix} 7.6795 \\ 9.3338 \end{pmatrix}$	$\begin{pmatrix} 9.7084 \\ 6.8333 \end{pmatrix}$	$\begin{pmatrix} 9.9008 \\ 2.1256 \end{pmatrix}$	$\begin{pmatrix} 7.8886 \\ 8.3924 \end{pmatrix}$
v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	
$\begin{pmatrix} 4.3866 \\ 6.2878 \end{pmatrix}$	$\begin{pmatrix} 4.9831 \\ 1.3377 \end{pmatrix}$	$\begin{pmatrix} 2.1396 \\ 2.0713 \end{pmatrix}$	$\begin{pmatrix} 6.4349 \\ 6.0720 \end{pmatrix}$	$\begin{pmatrix} 3.2004 \\ 6.2989 \end{pmatrix}$	$\begin{pmatrix} 9.6010 \\ 3.7048 \end{pmatrix}$	$\begin{pmatrix} 7.2663 \\ 5.7515 \end{pmatrix}$	

The input parameters α, β, w at each market are given in Table 4.8. Table 4.17 (Appendix A) shows the distance matrix $d(v_i, v_j)$). Marginal costs, $c_i(v_j)$, are detailed in Table 4.18 (Appendix A).

Table 4.8. *Parameters α, β, w.*

	Values for each of the markets														
Parameter	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
α	976	615	804	743	946	881	728	509	911	722	808	896	961	869	588
β	3	5	5	3	5	1	2	5	1	1	2	1	4	2	1
w	145	216	252	206	228	142	176	257	236	192	214	259	111	221	110

When the case with 5 firms is considered, the enumeration algorithm found one equilibrium in location: $(1, 10, 9, 10, 2)$. Table 4.9 shows the total unit costs when firms are located at equilibrium, table 4.10 shows the Nash quantity matrix, table 4.11 shows the number of

entrants at each market and table 4.12 the corresponding payoffs for the firms. The enumeration requires the quantity evaluation for $L = 15^5 = 759375$ possible location vectors.

Table 4.9. *Total Unit Costs for the firms at each market (TCU_{ih}). Case 15 markets, 5 firms - Nash location* $(1, 10, 9, 10, 2)$.

	Markets														
	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
x_1^*	113	116.5	117.8	117.0	119.3	119.2	119.1	118.7	115.2	115.9	115.9	116.01	115.4	118.5	116.5
x_2^*	237.9	241.4	239	240.7	243.4	242.3	240	242.6	240	235	237.9	240	240.3	240.2	240
x_3^*	309.2	308.8	312.7	311.6	311.5	312.4	313.9	311.1	307	312	311.8	309.06	308.2	312.8	309.9
x_4^*	257.9	261.4	259	260.7	263.4	262.3	260	262.6	260	255	257.9	260	260.3	260.2	260
x_5^*	113.5	110	117.4	114.2	115	116.7	118.7	114.9	111.8	116.4	115.5	113.7	111.2	117.5	114.5

Table 4.10. *Quantity Matrix - Supply from each firm to each market. Case 15 markets, 5 firms - Nash location* $(1, 10, 9, 10, 2)$.

	Markets														
	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
f_1	73.9	30.5	37.9	59.4	42.8	203.1	87.0	25.5	209.0	173.0	95.6	206.4	54.3	100.6	147.3
f_2	32.3	5.5	13.7	18.1	17.9	80.0	26.6	0.7	84.2	53.9	34.6	82.5	23.1	39.8	23.8
f_3	8.5	0	0	0	4.3	9.9	0	0	17.2	0	0	13.4	6.1	3.4	0
f_4	25.6	1.5	9.7	11.5	13.9	60.0	16.6	0	64.2	33.9	25.6	62.5	18.1	29.8	3.8
f_5	73.7	31.8	38.0	60.3	43.6	205.6	87.2	26.3	212.4	172.5	95.8	208.8	55.4	101.1	149.3

Table 4.11. *Number of entrants for each market. Case 15 markets, 5 firms - Nash location* $(1, 10, 9, 10, 2)$.

v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
5	4	4	4	5	5	4	3	5	4	4	5	5	5	4

Table 4.12. *Profit for each firm. Case 15 markets, 5 firms - Nash location* $(1, 10, 9, 10, 2)$.

f_1	f_2	f_3	f_4	f_5
295653.69	39470.23	818.54	21239.80	301487.76

Following Proposition 4, markets $v_1, v_5, v_6, v_9, v_{12}, v_{13}$ and v_{14} satisfy

$$\max_{i \in N} TCu_{ih} < \frac{\alpha_h + \sum_{i \in N} TCu_{ih}}{n+1} \quad (h = 1, 3, 5, 6, 9, 12, 13, 14)$$

For markets v_2, v_3, v_4, v_7, v_{10}, v_{11} and v_{15}:

$$\max_{i \in N} TCu_{ih} < \frac{\alpha_h + \sum_{i \in N} TCu_{ih} - \max(TCu_{ih})}{n} \quad (h = 2, 3, 4, 7, 10, 11, 15)$$

And for market v_8:

$$\max_{i \in N} TCu_{ih} < \frac{\alpha_h + \sum_{i \in N} TCu_{ih} - Max_1 - Max_2}{n-1} \quad (h = 8)$$

where Max_1 and Max_2 are given by

$$Max_1 = \max_{i \in N}(TCu_{ih})$$
$$Max_2 = \max_{i \in N \setminus \{Max_1\}}(TCu_{ih})$$

Tables 4.13 and 4.14 show the equilibrium in location, number of entrants and profit for cases with 4, 3 and 2 firms. Finally, table 4.15 shows the computational CPU times for the four cases and the increase in complexity when the number of firms goes from 2 to 5. Algorithms have been implemented in Fortran and run on a core-duo Pentium IV processor.

Table 4.13. *Equilibrium location and profit for each firm. Case 15 markets and 4,3,2 (respectively) supplier firms.*

Supplier firms				Equilibrium				Profit			
f_1	f_2	f_4	f_5	1	3	7	8	311846.92	319044.00	12580.02	45953.31
	f_1	f_2	f_4		4	14	9		379668.31	362520.90	30005.05
		f_3	f_5			8	4			663895.26	62272.21

Table 4.14. *Number of entrants at each market. Case 15 markets and 4,3,2 (respectively) supplier firms.*

Case	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
4 suppliers	4	3	4	4	4	4	4	3	4	4	4	4	4	4	3
3 suppliers	3	3	3	3	3	3	3	2	3	3	3	3	3	3	2
2 suppliers	2	2	2	2	2	2	2	1	2	2	2	2	2	2	2

Table 4.15. *Complexity. CPU times in seconds.*

Number of markets	Number of firms	Cases evaluated	CPU Time
15	5	759375	58.10938
15	4	50625	2.703125
15	3	3375	0.1718750
15	2	225	1.5625000E-02

The local search algorithm has been run with 60 random generated starting locations for the cases with 15 markets and $5, 4, 3$ and 2 supplier firms. The unique equilibrium at all cases is found by the local search algorithm. Table 4.16 shows the average and standard deviation of the number of iterations and function evaluations.

Table 4.16. *Multi-start algorithm. Number of iterations and function evaluations. Cases with 15 markets and 5,4,3,2 (respectively) supplier firms.)*

Nr. of firms	Avg (SD)	Best Nr of iterations	Nr of function evaluations. Avg (SD)
5 $\{f_1, f_2, f_3, f_4, f_5\}$	5.67(0.57)	4	396.67(39.74)
4 $\{f_1, f_2, f_4, f_5\}$	4.75(0.50)	3	266.00(28.23)
3 $\{f_1, f_2, f_4\}$	3.80(0.40)	3	159.60(16.80)
2 $\{f_3, f_5\}$	2.87(0.34)	2	80.27(9.52)

The number of iterations increases with the number of firms considered.

4.5 Conclusions

A competitive location and quantity "a la Cournot" game has been described in this paper to study the oligopolistic competition between $n > 2$ heterogeneous firms. Firms have to decide where to locate a facility and then decide on how much to supply to all or some of $m > 2$ spatially separated markets from these facilities. The following results were derived with respect to the optimal supply decisions where we are dealing with possibly heterogeneous firms:

- A necessary condition to have any delivery to a market
- Analytic expression of the equilibrium quantities of the firms that supply to a market
- Necessary condition for a firm to supply to a market
- Based on the former, a new procedure has been developed to identify those firms that are supplying to a market, the active set, which determines the size of the market

Based on these results algorithms are designed to find Nash equilibria of the game. The results and algorithms are illustrated numerically. By using the algorithms as a systematic computation instrument to cases reported in literature, a mistake was detected in Sarkar et al. (1997). In that paper a solution is given that appears not to be an equilibrium of the model. Furthermore, tests on larger generated instances show the viability of the approach.

Appendix A: Input Data for Example Two

Distance matrix $d(v_i, v_j)$ and marginal costs $c_i(v_j)$ are given in Table 4.17 and Table 4.18, respectively.

Table 4.17. *Distance Matrix. Case 15 markets, 5 firms.*

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
v_1	0														
v_2	3.50	0													
v_3	4.81	7.43	0												
v_4	4.01	4.21	8.76	0											
v_5	6.29	5.00	6.74	9.01	0										
v_6	6.18	6.69	4.26	9.86	3.22	0									
v_7	6.08	8.67	1.28	10.01	7.54	4.71	0								
v_8	5.68	4.93	5.78	8.71	0.96	2.40	6.58	0							
v_9	2.18	1.77	5.66	4.64	4.49	5.35	6.91	4.09	0						
v_{10}	2.90	6.41	3.99	5.72	8.44	7.25	4.98	7.63	4.99	0					
v_{11}	2.87	5.45	6.63	3.06	9.13	8.94	7.76	8.54	4.78	2.94	0				
v_{12}	3.00	3.65	4.10	6.50	3.49	3.36	5.25	2.74	2.06	4.95	5.87	0			
v_{13}	2.38	1.16	6.62	3.59	5.41	6.53	7.89	5.13	1.19	5.27	4.36	3.24	0		
v_{14}	5.46	7.54	1.34	9.48	5.95	3.13	1.61	4.99	5.82	5.19	7.64	3.95	6.91	0	
v_{15}	3.51	4.54	3.41	7.24	3.61	2.67	4.48	2.71	2.93	4.97	6.31	0.89	4.10	3.10	0

Table 4.18. *Marginal Production Costs for each firm (depending on location) $c_i(v_j)$. Case 15 markets, 5 firms.*

	Possible locations for firms														
	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
f_1	113	772	501	939	519	477	862	573	282	705	855	117	713	441	849
f_2	553	739	486	374	270	274	714	372	488	235	728	440	874	869	634
f_3	547	910	840	681	836	694	408	361	307	581	755	378	855	611	433
f_4	733	592	500	725	659	816	462	570	693	255	982	344	327	889	764
f_5	223	110	905	279	369	695	356	522	158	990	625	481	564	400	490

CHAPTER 5

On a Branch-and-Bound Approach for a Huff-like Stackelberg Location Problem

Chapter based on article: Sáiz, M.E., Fernández, J., Hendrix, E.M.T. and Pelegrín, B. (2007), On a Branch-and-Bound approach for a Huff-like Stackelberg location problem, *submitted*. Mansholt Working Paper MWP-37, Wageningen

Abstract

Modelling the location decision of two competing firms that intend to build a new facility in a planar market can be done by a Huff-like Stackelberg location problem. In a Huff-like model, the market share captured by a firm is given by a gravity model determined by distance calculations to facilities. In a Stackelberg model, the *leader* is the firm that locates first and takes into account the actions of the competing chain (*follower*) locating a new facility after the leader. The follower problem is known to be a hard global optimisation problem. The leader problem is even harder, since the leader has to decide on location given the optimal action of the follower. So far, in literature only heuristic approaches have been tested to solve the leader problem.

Our research question is to solve the leader problem rigorously in the sense of having a guarantee on the reached accuracy. To answer this question, we develop a Branch-and-Bound approach. Essentially, the bounding is based on the zero sum concept: what is gain for one chain is loss for the other. We also discuss several ways of creating bounds for the underlying (follower) sub-problems, and show their performance for numerical cases.

5.1 Introduction

Many factors must be taken into account when locating a new facility which provides goods or a service to the customers of a given area. One of the most important points is the existence of competitors in the market providing the same goods or service. When no other competitor exists, the facility to be located will have the monopoly of the market in that area. However, if in the area there already exist other facilities offering the same goods, then the new facility will have to compete for the market.

Many competitive location models are available in the literature, see for instance the survey papers (Eiselt and Laporte 1996, Eiselt et al. 1993, Plastria 2001) and the references therein. They vary in the ingredients which form the model. For instance, the *location space* may be the plane, a network or a discrete set. We may want to locate just one or more than one new facility. The competition may be *static*, which means that the competitors are already in the market and the owner of the new facility knows their characteristics, or *with foresight*, in which the competitors are not in the market yet but they will be soon after the new facility enters. In this case it is necessary to make decisions with foresight about this competition, leading to a Stackelberg-type model (competition model in which a leader firm moves first and then the follower firm moves sequentially). Demand is usually supposed to be concentrated in a discrete set of points, called *demand points*.

The *patronising behaviour* of the customers must also be taken into account, since the *market captured* by the facilities depends on it. In some models customers select among the facilities in a *deterministic* way, i.e, the full demand of the customer is served by the facility to which he/she is *attracted* most. In other cases, the customer splits his/her demand among more than one facility, leading to *probabilistic* patronising behaviour. On the other hand, it is also necessary to specify what the *attraction (or utility) function* of a customer towards a given facility is. Usually, the attraction function depends on the distance between the customer and the facility, as well as on other characteristics of the facility which determine its *quality*.

In this paper, we consider a planar facility location problem with foresight, having probabilistic consumer behaviour, based on an attraction function depending on both the locations and the qualities of the facilities to be located. The demand quantities are assumed to be known and fixed. For the current study, also the quality values of the new facilities to be located are assumed to be given. There are two competitors (chains). First, the leader makes a decision on where to locate its facility in the plane (the location of the facility is considered the variable of the problem). Second, the follower makes a decision with full knowledge of the decision of the leader. The objective of the leader is to maximize its market share after the entrance of the follower.

The follower problem has been studied under deterministic customer behaviour in (Drezner 1994) and (Plastria 1997), using attraction functions of gravity type, and in (Plastria and Carrizosa 2004) using different kinds of attraction functions. For probabilistic customer behaviour, the problem has been studied in (Drezner and Drezner 1994), where the location problem is solved for a wide range of quality values (see also (Drezner and Drezner 2004)).

However, due to its difficulty, the literature on the leader problem is rather scarce. To our knowledge, the leader problem with deterministic behaviour on the plane has only been addressed in (Drezner and Drezner 1982) and (Bhadury et al. 2003), and with probabilistic behaviour only in (Drezner and Drezner 1998), where three heuristics are described for a variant of the model considered in this paper. The question addressed in this paper is whether the leader problem can be solved up to a guaranteed accuracy. We will show in the current paper that one can make use of the zero-sum perspective to construct a Branch-and-Bound method that achieves that aim.

In Section 5.2, the notation is introduced and both the leader and the follower problem are formulated. In Section 5.3 and 5.4, a detailed description of the Branch-and-Bound algorithms to solve the follower and leader problem (respectively) is given. The algorithms are illustrated by instances in Section 5.5 and the efficiency is investigated for different parameter settings. Conclusions and future work are discussed in Section 5.6.

5.2 Description of the Problem

The following notation will be used throughout:

Indices

- i index of demand points, $i = 1, \ldots, n$
- j index of existing facilities, $j = 1, \ldots, m$ (the first k of those m facilities, $0 \leq k \leq m$, belong to the leader chain, and the rest to the follower)
- l index for the new facilities, $l = 1, 2$

Variables

- $x_l = (x_{l1}, x_{l2})$ location of the leader ($l = 1$) and follower ($l = 2$)

Data

α_l	quality of the leader ($l = 1$) and follower ($l = 2$)
p_i	location of the i-th demand point
w_i	demand (or buying power) at p_i
q_j	location of the j-th existing facility
d_{ij}	distance between p_i and q_j
a_j	quality of facility j
$g(\cdot)$	a positive non-decreasing function
$a_j/g(d_{ij})$	attraction that i feels for facility j
S	location space where the leader and the follower will locate the new facility

Miscellaneous

δ_{il}	distance between p_i and $x_l, l = 1, 2$
$\alpha_l/g(\delta_{il})$	attraction that i feels for new facility l
$M_l(x_1, x_2)$	market capture by the leader ($l = 1$) and follower ($l = 2$)

The best location in attraction models is usually situated in the convex hull of the demand points. In this paper we consider as the feasible location space S a rectangle enclosing that convex hull. Notice that $M_1(x_1, x_2) + M_2(x_1, x_2) = \sum_{i=1}^{n} w_i$. This 'zero-sum' character of the model is essential in the method used to solve it. In the model, the market share captured by the leader chain after the leader locates in x_1 and the follower in x_2 is

$$M_1(x_1, x_2) = \sum_{i=1}^{n} \omega_i \frac{\dfrac{\alpha_1}{g(\delta_{i1})} + \displaystyle\sum_{j=1}^{k} \dfrac{a_j}{g(d_{ij})}}{\dfrac{\alpha_1}{g(\delta_{i1})} + \dfrac{\alpha_2}{g(\delta_{i2})} + \displaystyle\sum_{j=1}^{m} \dfrac{a_j}{g(d_{ij})}}$$

and the corresponding market share captured by the follower chain is

$$M_2(x_1, x_2) = \sum_{i=1}^{n} \omega_i \frac{\frac{\alpha_2}{g(\delta_{i2})} + \sum_{j=k+1}^{m} \frac{a_j}{g(d_{ij})}}{\frac{\alpha_1}{g(\delta_{i1})} + \frac{\alpha_2}{g(\delta_{i2})} + \sum_{j=1}^{m} \frac{a_j}{g(d_{ij})}} \tag{5.1}$$

Given x_1, problem $(FP(x_1))$ of the follower is the so-called $(1|x_1)$-medianoid problem introduced by (Hakimi 1983)

$$\max_{x_2 \in S} \{G(x_2) = M_2(x_1, x_2)\} \tag{5.2}$$

Since $M_1(x_1, x_2) + M_2(x_1, x_2) = \sum_{i=1}^{n} w_i$, $(FP(x_1))$ in (5.2) is equivalent to

$$\min_{x_2 \in S} M_1(x_1, x_2) \tag{5.3}$$

Let $x_2^*(x_1)$ represent an optimal solution of $(FP(x_1))$. Problem (LP) for the leader is the $(1|1)$-centroid problem (see (Hakimi 1983))

$$\max_{x_1 \in S} \{F(x_1) = M_1(x_1, x_2^*(x_1))\} \tag{5.4}$$

In (Drezner and Drezner 2004) and (Fernández et al. 2007), procedures are given to maximize the market share captured by a given chain when the facility locations of the competitors are fixed as in problem $(FP(x_1))$. As studied by (Fernández et al. 2007), we are dealing with a Global Optimization problem; see Figure 5.1, which shows the multimodal behaviour of problem $(FP(x_1))$.

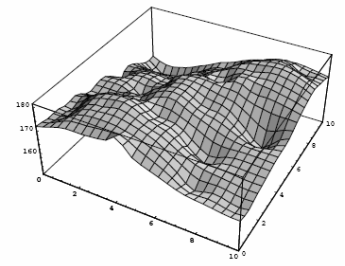

Figure 5.1. *Plot of the objective function of a follower problem.*

In the solution procedure that we have designed to cope with the leader problem, we are also interested in solving a similar problem to that of the follower, in which the leader wants to locate a new facility at x_1, given the location and the quality of all the facilities of the competitor (the follower). In this case, the leader has to solve a medianoid problem in which the roles of leader and follower are interchanged. We will call this problem a *reverse medianoid* problem.

The leader problem (LP) is much more difficult to solve than the follower problem. To the extent of our knowledge, the leader problem with probabilistic behaviour on the plane

has only been addressed in (Drezner and Drezner 1998), where heuristic procedures were presented for a similar version of the problem considered here. Among others, they applied variants of multistart and grid search to generate solutions of the leader and follower problems. In Section 5.3, a Branch-and-Bound algorithm for the medianoid (follower) and reverse medianoid problems with four different ways of obtaining an upper bound are introduced. In Section 5.4, a Branch-and-Bound algorithm for the (1|1)-centroid problem (leader) is described.

5.3 A Branch-and-Bound Algorithm for the Medianoid (follower) Problem

In the medianoid problem ($FP(x_1)$), the follower wants to locate a new facility, knowing the location and the quality of all the facilities of the competitor (the leader). Next we describe the details of the algorithm for the follower problem. For the reverse medianoid problem of the leader, the algorithm is similar.

The basic idea in B&B methods consists of a recursive decomposition of the original problem into smaller disjoint subproblems until the solution is found. The method avoids visiting those subproblems which are known not to contain a solution. B&B methods can be characterized by four rules: *Branching, Selection, Bounding*, and *Elimination* (see (Ibaraki 1976, Mitten 1970)). For problems where the solution is determined with a desired accuracy, a *Termination rule* has to be incorporated. The method works as follows. The initial set $C_1 = S$ is subsequently partitioned in more and more refined subsets (branching) over which upper and lower bounds of the objective function are determined (bounding). In a maximization problem, subsets with upper bounds lower than the best lower bound are eliminated for subsequent partitions (pruning), since these subsets cannot contain the maximum. At every iteration, the B&B method has a list Λ of subsets C_k of C_1. The method stops when the list is empty. For every subset C_k in Λ, upper bounds z^{kU} of the objective function on C_k are determined. Moreover, a global lower bound z^L is updated. Next, we give a more detailed description of the steps of the algorithm.

5.3.1 The Algorithm

To take both the medianoid and the reverse medianoid problems into account, we will denote by M the objective function of the problem at hand and by C its feasible set. The B&B method is described in Algorithm 5.1. Its output is the best point found during the process and its corresponding function value. The best point is guaranteed to differ less than ε_f in function value from the optimal solution of the problem (by considering the difference between lower and upper bounds).

5.3.2 Branching Rule

The branching rule applied uses rectangles and new rectangles are generated by bisecting a subset C over its longest edge. Two variants are implemented. Either we start with the initial rectangle S, or we start with an initial partition of it into rectangles such that none of the demand points is interior with respect to a rectangle. As will be outlined, this may improve the upper bounding applied, but on the other hand may generate more partition sets than strictly necessary.

Algorithm 5.1 : Branch-and-Bound algorithm for the *(reverse) medianoid problem.*

Funct B&B(M, x, C, ε_f)

1. $\Lambda := \emptyset$
2. $C_1 := C$
3. Determine an upper bound z^{1U} on C_1
4. Compute y^1:=midpoint(C_1), $BestPoint := y^1$
5. Determine lower bound: $z^1 := M(y^1)$, $z^L := z^1$
6. Put C_1 on list Λ, $r := 1$
7. **while** ($\Lambda \neq \emptyset$)
8. Take a subset C (selection rule) from list Λ and bisect into C_{r+1} and C_{r+2}
9. **for** $t := r + 1$ to $r + 2$
10. Determine upper bound z^{tU}
11. **if** $z^{tU} > z^L + \varepsilon_f$
12. Compute y^t:=midpoint(C_t) and $z^t := M(y^t)$
13. **if** $z^t > z^L$
14. $z^L := z^t$, $BestPoint := y^t$ and remove all C_r from Λ with $z^{rU} < z^L$
15. **if** $z^{tU} > z^L + \varepsilon_f$
16. save C_t in Λ
17. $r := r + 2$
18. **endwhile**
19. OUTPUT: $\{BestPoint, z^L\}$

5.3.3 Selection Rule

The selection rule is important in the sense of efficiency measured by computational time and memory requirements. Within selection rules, one can find: depth-first-search, breadth-first-search and best-bound-search. In Section 5.5.1 the effect on efficiency of those rules is measured.

5.3.4 Lower Bound

The classical lower bound is obtained as the best objective value at a finite set of feasible solutions $\{x_2^1, \ldots, x_2^r\}$

$$z^L = \max\{G(x_2^1), \ldots, G(x_2^r)\}.$$

A good initial lower bound can be obtained by applying the (local search) Weiszfeld-like algorithm described in (Drezner and Drezner 1994) from 20 or 50 starting random points. We simply use the best objective function value found at the evaluated points.

5.3.5 Upper Bounds for the Follower Problem $(FP(x_1))$

The idea of the upper bound is to overestimate M_2 over a rectangle C. The market share captured by the follower (eq. 5.1) can be rewritten as

$$M_2(x_1, x_2) = \sum_{i=1}^{n} \omega_i \frac{1 + \frac{1}{\alpha_2}\left(\sum_{j=k+1}^{m} \frac{a_j}{g(d_{ij})}\right) g(\delta_{i2})}{1 + \frac{1}{\alpha_2}\left(\frac{\alpha_1}{g(\delta_{i1})} + \sum_{j=1}^{m} \frac{a_j}{g(d_{ij})}\right) g(\delta_{i2})}. \tag{5.5}$$

Introducing

$$h_i = \frac{1}{\alpha_2} \sum_{j=k+1}^{m} \frac{a_j}{g(d_{ij})}$$
$$k_i = \frac{1}{\alpha_2} \left(\frac{\alpha_1}{g(\delta_{i1})} + \sum_{j=1}^{m} \frac{a_j}{g(d_{ij})} \right)$$

and defining

$$f_i(g(\delta_{i2})) = \frac{1 + h_i g(\delta_{i2})}{1 + k_i g(\delta_{i2})} \tag{5.6}$$

equation (5.5) becomes

$$M_2(x_1, x_2) = \sum_{i=1}^{n} \omega_i f_i(g(\delta_{i2})).$$

An upper bound for M_2 is

$$\overline{M}_2(x_1, x_2) = \sum_{i=1}^{n} \omega_i UB_i(C)$$

where $UB_i(C)$ is an overestimation of $f_i(g(\delta_{i2}))$ over rectangle C. Notice that $h_i < k_i$ and f_i is monotonously decreasing in $g(\delta_{i2})$ with a limit of $\frac{h_i}{k_i}$.

We now describe various possible variants of the upper bounding. We will also evaluate numerically which bound is sharper than the others. The first upper bound is simply based on underestimating distance. The second and third upper bounds exploit the D.C. structure of the objective function. The fourth upper bound builds a convex overestimating function based on the third one.

Upper Bound 1

A first upper bound for $f_i(g(\delta_{i2}))$ over a rectangle C is calculated in the following way. For demand point p_i, the distance to the follower x_2 when $x_2 \in C$ is underestimated by assuming that x_2 delivers from the complete rectangle C. In this way the market share of the demand point for the follower is overestimated. The demand points within rectangle C have a distance $\Delta_i(C) = 0$ from C. For demand points out of rectangle C, $p_i \notin C$, the shortest distance $\Delta_i(C)$ of p_i to the rectangle is calculated. An upper bound $UB_i^1(C)$ for $f_i(g(\delta_{i2}))$ over rectangle C for demand point p_i is given by

$$UB_i^1(C) = \frac{1 + h_i g(\Delta_i(C))}{1 + k_i g(\Delta_i(C))} \tag{5.7}$$

where $\Delta_i(C)$ is the distance from demand point p_i to rectangle C, $\Delta_i(C) = \min_{x \in C} d(x, p_i)$. The distance $\Delta_i(C)$ can be determined as follows. Rectangle C is defined by two points: lower-left point $L = (L_1, L_2)$ and upper-right point $U = (U_1, U_2)$. The shortest distance from demand point p_i to the rectangle $C = [L, U]$ can be computed by:

$$\begin{aligned} \Delta_{i1} &= \max\{L_1 - p_{i1}, p_{i1} - U_1, 0\} \\ \Delta_{i2} &= \max\{L_2 - p_{i2}, p_{i2} - U_2, 0\} \\ \Delta_i &= \sqrt{\Delta_{i1}^2 + \Delta_{i2}^2} \end{aligned} \tag{5.8}$$

Summarising,

$$\Delta_i(C) = \begin{cases} 0 & \text{if } p_i \in C \\ \sqrt{\Delta_{i1}^2 + \Delta_{i2}^2} & \text{if } p_i \notin C \end{cases} \tag{5.9}$$

This distance calculation is easily extendible to higher dimensions. A similar description is used in (Plastria 1992). Equation (5.9) underestimates the distance from demand point p_i to facilities in C. Since the new facility is only located at one point within the rectangle, we obtain an overestimation (upper bound) of the market capture of the new facility ($f_i(g(\delta_{i2}))$ is decreasing in δ_{i2}).

Upper Bound 2

The second upper bound is more sophisticated and it is based on convexity of the functions f_i and g. From now on, we will use the convex function $g(\delta_{i2}) = \sqrt{\delta_{i2}^2 + K_i^2}$ that was suggested in (Drezner and Drezner 1997), where K_i is a constant representing demand agglomeration. Equation (5.6) can be seen as a composition of functions f_i and g. We will define an upper bound by using D.C. decomposition. A d.c. decomposition of a function s defined on a convex $C \subset \mathbb{R}^n$ can be expressed, for all $x \in C$, in the form

$$s(x) = s_1(x) - s_2(x)$$

where s_1 and s_2 are convex functions on C. The following lemma is adapted from Lemma 1 in (Tuy et al. 1995). Let $f'_+(x)$ be the right derivative of $f(x), x \in \mathbb{R}$.

Lemma 3. *Let $g(\delta(x))$ be a convex function on a convex and compact subset $C \subset \mathbb{R}^2$ such that $g(\delta(x)) \geq 0$ for all $x \in C$. If $f : \mathbb{R}_+ \mapsto \mathbb{R}$ is a convex nonincreasing function such that $f'_+(0) > -\infty$, then $f(g(\delta(x)))$ is a d.c. function in C and can be expressed as:*

$$f(g(\delta(x))) = b(x) - Rg(\delta(x)) \tag{5.10}$$

where $b(x) = f(g(\delta(x))) + Rg(\delta(x))$ is a convex function for each positive constant R satisfying $R \geq |f'_+(0)|$.

By using Lemma 3 we can obtain a d.c. decomposition for each f_i. In particular, if $g(\delta_{i2}) = \sqrt{\delta_{i2}^2 + K_i^2}$, a d.c. decomposition for $f_i(g(\delta_{i2}))$ is defined by

$$f_i(g(\delta_{i2})) = b_i(x) - R_i g(\delta_{i2}) = b_i(x) - R_i\sqrt{\delta_{i2}^2 + K_i^2} \tag{5.11}$$

where $b_i(x) = f_i(g(\delta_{i2})) + R_i\sqrt{\delta_{i2}^2 + K_i^2}$ and $R_i = k_i - h_i$. Market capture for the follower can be expressed by

$$G(x) = M_2(x_1, x) = \sum_{i=1}^{n} \omega_i f_i(g(\delta_{i2})) = \sum_{i=1}^{n} \omega_i \left[b_i(x) - R_i \sqrt{\delta_{i2}^2 + K_i^2} \right]$$
$$= \sum_{i=1}^{n} \omega_i \left\{ \frac{1 + h_i \sqrt{\delta_{i2}^2 + K_i^2}}{1 + k_i \sqrt{\delta_{i2}^2 + K_i^2}} + (k_i - h_i) \sqrt{\delta_{i2}^2 + K_i^2} \right\}$$
$$- \sum_{i=1}^{n} \omega_i (k_i - h_i) \sqrt{\delta_{i2}^2 + K_i^2}.$$

Let $\delta_i^2(x) = (\|x - p_i\|_2)^2$ be the squared Euclidean distance between x and demand point p_i and $V(C)$ be the set of vertices v of rectangle C. An upper bound is defined as

$$UB = \max_{v \in V(C)} \left\{ \sum_{i=1}^{n} \omega_i \left\{ \frac{1 + h_i \sqrt{\delta_i^2(v) + K_i^2}}{1 + k_i \sqrt{\delta_i^2(v) + K_i^2}} + (k_i - h_i) \sqrt{\delta_i^2(v) + K_i^2} \right\} \right\}$$
$$- \min_{x \in C} \left\{ \sum_{i=1}^{n} \omega_i (k_i - h_i) \sqrt{\delta_{i2}^2 + K_i^2} \right\} \tag{5.12}$$

UB is a valid upper bound of M_2 over C. To facilitate computation, one can underestimate $\min_{x \in C} \left\{ \sum_{i=1}^{n} \omega_i (k_i - h_i) \sqrt{\delta_{i2}^2 + K_i^2} \right\}$ by $\sum_{i=1}^{n} \omega_i (k_i - h_i) \sqrt{\Delta_i^2(C) + K_i^2}$. Then, UB^2 is defined as

$$UB^2(C) = \max_{v \in V(C)} \left\{ \sum_{i=1}^{n} \omega_i \left\{ \frac{1 + h_i \sqrt{\delta_i^2(v) + K_i^2}}{1 + k_i \sqrt{\delta_i^2(v) + K_i^2}} + (k_i - h_i) \sqrt{\delta_i^2(v) + K_i^2} \right\} \right\}$$
$$- \sum_{i=1}^{n} \omega_i (k_i - h_i) \sqrt{\Delta_i^2(C) + K_i^2} \tag{5.13}$$

Upper Bound 3

For the ease of notation, let $z_i(x) = g(\delta_{i2})$. In this way, $G(x) = M_2(x_1, x)$ can be written as

$$G(x) = M_2(x_1, x) = \sum_{i=1}^{n} \omega_i f_i(z_i(x)) = \sum_{i=1}^{n} \omega_i \frac{1 + h_i z_i(x)}{1 + k_i z_i(x)}$$

Let x^0 be the centre of rectangle C and $z_i^0 = z_i(x^0)$. According to Taylor's theorem there exist $g(\Delta_i) \leq \tilde{z}_i$ such that

$$G(x) = G(x^0) + \sum_{i=1}^{n} \omega_i \left[\frac{h_i - k_i}{(1 + k_i z_i^0)^2} (z_i(x) - z_i^0) + \frac{k_i (k_i - h_i)}{(1 + k_i \tilde{z}_i)^3} (z_i(x) - z_i^0)^2 \right]$$

The first bounding operation is based on replacing $\tilde{z}_i$ by $g(\Delta_i)$,

$$G(x) \leq G(x^0) + \sum_{i=1}^{n} \omega_i \left[\frac{h_i - k_i}{(1 + k_i z_i^0)^2}(z_i(x) - z_i^0) + \frac{k_i(k_i - h_i)}{(1 + k_i g(\Delta_i))^3}(z_i(x) - z_i^0)^2 \right]$$

By introducing

$$r_i = w_i \frac{k_i - h_i}{(1 + k_i z_i^0)^2}$$
$$s_i = w_i \frac{k_i(k_i - h_i)}{(1 + k_i g(\Delta_i))^3}$$
$$t_i = r_i + 2 s_i z_i^0$$

and rearranging terms we obtain

$$G(x) \leq G(x^0) + \sum_{i=1}^{n} (r_i z_i^0 + s_i (z_i^0)^2) - \sum_{i=1}^{n} t_i z_i(x) + \sum_{i=1}^{n} s_i z_i(x)^2 \tag{5.14}$$

Although z_i is convex, the function in the right part of (5.14) is not. However, it is clearly a D.C. function. Let $V(C)$ be the set of vertices v of rectangle C. Then, one can overestimate (5.14) by taking

$$UB = Const1 - \min_{x \in C} \sum_{i=1}^{n} t_i z_i(x) + \max_{v \in V(C)} \sum_{i=1}^{n} s_i z_i(v)^2$$

where $Const1 = G(x^0) + \sum_{i=1}^{n} (r_i z_i^0 + s_i (z_i^0)^2)$. As with upper bound UB^2, one can underestimate $\min_{x \in C} \sum_{i=1}^{n} t_i z_i(x)$ by $\sum_{i=1}^{n} t_i g(\Delta_i(C))$. Then, UB^3 is defined as

$$UB^3(C) = Const1 - \sum_{i=1}^{n} t_i g(\Delta_i(C)) + \max_{v \in V(C)} \sum_{i=1}^{n} s_i z_i(v)^2 \tag{5.15}$$

Upper Bound 4

In this section, a convex overestimation $\Gamma_C(x)$ of $G(x)$ over a rectangle C is derived starting from (5.14). One can linearly overestimate the term $-t_i z_i(x)$ due to convexity of function $z_i(x)$ as follows

$$z_i(x) \geq z_i^0 + \nabla z_i^0 (x - x^0)$$

Substitution gives

$$\begin{aligned} G(x) &\leq G(x^0) + \sum_{i=1}^{n} (r_i z_i^0 + s_i (z_i^0)^2) - \sum_{i=1}^{n} t_i z_i^0 - \sum_{i=1}^{n} t_i \nabla z_i^0 (x - x^0) + \sum_{i=1}^{n} s_i z_i(x)^2 \\ &= G(x^0) - \sum_{i=1}^{n} s_i (z_i^0)^2 - \sum_{i=1}^{n} t_i \nabla z_i^0 (x - x^0) + \sum_{i=1}^{n} s_i z_i(x)^2 = \Gamma_C(x) \end{aligned}$$

Function $\Gamma_C(x)$ is convex. An upper bound over rectangle C, $UB^4(C)$, can be expressed by

$$UB^4(C) = Const2 + \max_{v \in V(C)} \left\{ \sum_{i=1}^{n} s_i z_i(v)^2 - \sum_{i=1}^{n} t_i \nabla z_i^0 (v - x^0) \right\} \tag{5.16}$$

where $Const2 = G(x^0) - \sum_{i=1}^{n} s_i (z_i^0)^2$.

5.4 A Branch-and-Bound Algorithm for the Leader Problem

In this section, a new method based on Branch-and-Bound is formulated to generate a solution of the (1|1)-centroid problem. The final outcome is guaranteed to differ less in function value than a preset accuracy ε_l from the optimum solution. Next, we introduce the algorithm and its ingredients.

5.4.1 The Algorithm

The branching and selection rules used were the same as in Algorithm 5.1. The output of the B&B method (see Algorithm 5.2) is again the best point found during the process and its corresponding function value, which differs less than ε_l from the optimum value of the problem.

Algorithm 5.2 : Branch-and-Bound algorithm for *Leader problem.*

Funct B&BLeader($\varepsilon_l, \varepsilon_f$)

1. $\Lambda := \emptyset$
2. $C_1 = S$
3. Compute x_1^1:=midpoint(C_1), $BestPoint := x_1^1$
4. Solve the problem for the follower: $\{x_2^1, z\} :=$ **B&B**$(M_2, x_1^1, C_1, \varepsilon_f)$
5. Determine an upper bound z_1^{1U} on C_1 solving a reverse medianoid problem: $\{y, z_1^{1U}\} :=$ **B&B**$(M_1, x_2^1, C_1, \varepsilon_l)$
6. Determine lower bound: $z_1 := F(x_1^1) = M_1(x_1^1, x_2^1)$, $z^L := z_1$
7. Put C_1 on list Λ , $r := 1$
8. **while** ($\Lambda \neq \emptyset$)
9. Take a subset C (selection rule) from list Λ and bisect into C_{r+1} and C_{r+2}
10. **for** $t := r + 1$ to $r + 2$
11. Compute x_1^t:=midpoint(C_t)
12. Solve the problem for the follower: $\{x_2^t, z\} :=$ **B&B**$(M_2, x_1^t, C_1, \varepsilon_f)$
13. Determine upper bound z_1^{tU} solving a reverse medianoid problem: $\{y, z_1^{tU}\} :=$ **B&B**$(M_1, x_2^t, C_t, \varepsilon_l)$
14. **if** $z_1^{tU} > z^L + \varepsilon_l$
15. Determine $z_t := F(x_1^t) = M_1(x_1^t, x_2^t)$
16. **if** $z_t > z^L$
17. $z^L := z_t$, $BestPoint := x_1^t$, and remove all C_r from Λ with $z_1^{rU} < z^L$
18. **if** $z_1^{tU} > z^L + \varepsilon_l$
19. save C_t in Λ
20. $r := r + 2$
21. **endwhile**
22. OUTPUT: $\{BestPoint, z^L\}$

5.4.2 Lower Bound

The classical lower bound is obtained as the best objective value at a finite set of feasible solutions $\{x_1^1, \ldots, x_1^r\}$ for the leader problem,

$$z^L = \max\{F(x_1^1), \ldots, F(x_1^r)\}$$

One can follow the objective function value $F(x_1^p)$ of the iterates, or alternatively define an initial lower bound z^L based on running another algorithm that generates a good approximate solution.

5.4.3 Upper Bounds

Let $C \subseteq \mathbb{R}^2$ denote a subset of the search region of (LP), and assume that x_2 is given. An upper bound of $F(x_1)$ over C can be obtained by having the leader solve the reverse medianoid problem.

Lemma 4. *$UB(C, x_2) = \max_{x_1 \in C} M_1(x_1, x_2)$ is an upper bound of $F(x_1)$ over C.*

Proof. According to (5.3), $F(x_1) = M_1(x_1, x_2^*(x_1)) \leq M_1(x_1, x_2)$ such that

$$\max_{x_1 \in C} F(x_1) \leq \max_{x_1 \in C} M_1(x_1, x_2) = UB(C, x_2).$$

□

Given a finite set $\{x_2^1, \ldots, x_2^r\}$ of feasible solutions for the follower, then

$$\min\{UB(C, x_2^1), \ldots, UB(C, x_2^r)\}$$

is an upper bound of $F(x_1)$ over C.

For a specific rectangle C, the choice of x_2 for the upper bound calculation is done as follows. We take x^C=midpont(C) as the midpoint of the rectangle. Now one solves $(FP(x^C))$ obtaining $\hat{x}_2$. An upper bound is determined by solving the problem

$$ub_1(C) = UB(C, \hat{x}_2) = \max_{x_1 \in C}\{M_1(x_1, \hat{x}_2)\} \tag{5.17}$$

Another easy possibility is to set x_2 equal to x_1 (that is, to assume co-location). In that way, one obtains the following upper bound.

Lemma 5. *$ub_2(C) = UB(C, x_1) = \max_{x_1 \in C} M_1(x_1, x_1)$ is an upper bound of $F(x_1)$ over C.*

In the next two sections, we use numerical cases to illustrate the outcomes and efficiency of the algorithm.

5.5 Numerical Examples

The effectiveness and efficiency of the algorithms are investigated with the aid of numerical cases. In a first case, we experiment with algorithm settings (variants of the algorithm) and study the performance. In the following cases, the performance is studied with a good algorithm setting. The effectiveness question concerns the algorithms and several ways of upper bounding. Performance indicators of the efficiency are the number of iterations used by the algorithms and the memory requirement. In general, Branch-and-Bound algorithms deliver a guarantee of detecting the global optimum up to a pre-set accuracy, but the cost of the memory requirement may be high if the dimension is going up or the accuracy is increasing, see e.g. (Casado et al. 2007). In the first study, we will vary carefully the selection rule and the accuracy and inspect values of the performance indicators and effectiveness of the different bounds. Moreover, we evaluate a variant where an initial partition is generated to improve bound number 4. The second case is an illustration from literature. In the last case, we generate many instances at random where the size of the problem is varied to validate the viability of the approach with increasing number of demand points and existing facilities.

5.5.1 Case I, Varying Algorithm Setting

This case has been generated randomly with $n = 10$ demand points, $m = 4$ existing facilities and a varying number k of those facilities belonging to the leader's chain, $k = 0, \ldots, 4$. The generated demand points can be found in Appendix 5.6. The other parameters are chosen as follows:

- buying power: $w_i = 100,\ i = 1, \ldots, 10$
- quality of existing facilities: $a_j = 5.5,\ j = 1, \ldots, 4$
- quality of new facilities: $\alpha_l = 5,\ l = 1, 2$
- $g(d_{ij}) = \sqrt{(q_{j1} - p_{i1})^2 + (q_{j2} - p_{i2})^2 + (10^{-5})^2},\ i = 1, \ldots, 10,\ j = 1, \ldots, 4$
- $g(\delta_{il}) = \sqrt{(x_{l1} - p_{i1})^2 + (x_{l2} - p_{i2})^2 + (10^{-5})^2},\ l = 1, 2$
- accuracy for leader and follower: $\varepsilon_l = \varepsilon_f = 10^{-2}$

The resulting optimal locations are shown in Table 5.1, which also gives the market capture of both chains, when the number k of existing facilities of the leader chain is increasing. One can observe a characteristic of the model, where leader and follower tend to co-locate when the number of existing facilities of the leader is low. In fact, the follower by locating at the same position, mitigates the effect of the relatively newcomer in the market who is going to compete for market capture. Notice also that when the leader is dominant in the market (it owns $k = 3$ of the $m = 4$ existing facilities, or all of them, $k = 4$) then the leader suffers a decrease in market share after the location of the two new facilities (see the negative values in the last line of Table 5.1). This is because in those cases the follower increases its market share more than the leader.

Figure 5.2 illustrates how the algorithm proceeds. It gives: location of the demand points (squares); location of the existing facilities (triangle up, belongs to the follower, triangle down, belongs to the leader); the optimum for the locations of leader (diamond) and the follower (circle) and the final partition of the search space for the leader for the cases when the number of existing facilities of the leader are $k = 1$ and $k = 3$. Each of the boxes has been evaluated and it has been proven by bounding that the optimum location of the leader cannot be there.

Table 5.1. *Optimal locations and market capture for different number of leader facilities, $k = 0, \ldots, 4$. Parameter z_l^* = market capture for the leader after locating facility, Mb_l before; locations and market captures are rounded to two decimals.*

		$k=0$	$k=1$	$k=2$	$k=3$	$k=4$
Optimal Location	Leader	$\begin{pmatrix}2.44\\3.97\end{pmatrix}$	$\begin{pmatrix}5.03\\0.69\end{pmatrix}$	$\begin{pmatrix}5.33\\4.34\end{pmatrix}$	$\begin{pmatrix}5.33\\4.34\end{pmatrix}$	$\begin{pmatrix}5.03\\0.69\end{pmatrix}$
	Follower	$\begin{pmatrix}2.44\\3.97\end{pmatrix}$	$\begin{pmatrix}5.03\\0.69\end{pmatrix}$	$\begin{pmatrix}1.41\\4.65\end{pmatrix}$	$\begin{pmatrix}1.75\\3.79\end{pmatrix}$	$\begin{pmatrix}1.75\\3.79\end{pmatrix}$
Market Capture	Leader	186.29	367.87	497.70	611.07	773.44
	Follower	813.71	632.13	502.30	388.93	226.56
$z_l^* - Mb_l$ (gain or loss for the leader)		186.29	100.67	14.17	-72.46	-226.56

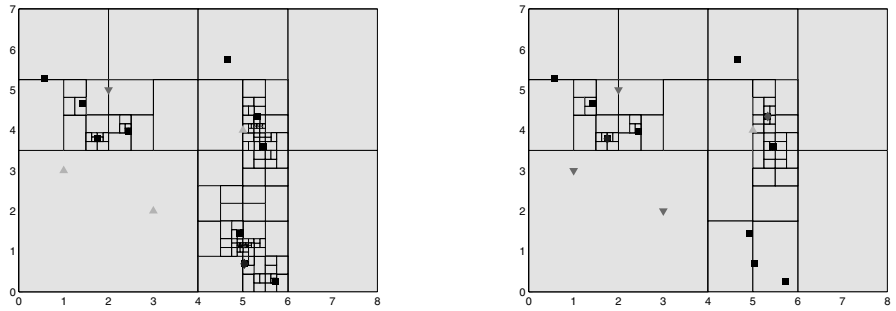

Figure 5.2. *Generated partition by the algorithm. Cases with $k = 1$ (left) and $k = 3$ (right)*

Table 5.2. *Efficiency base case algorithm. Iterations. Upper bound UB^1 in Algorithm 5.1, selection rule: breadth-first-search in both algorithms.*

		Medianoid problems			
	Leader problem	Follower M. problems		Reverse M. problems	
k		Max	Avg	Max	Avg
0	1325	503	308.62	3645	215.48
1	1017	427	313.98	3107	248.09
2	1161	545	439.71	2709	166.13
3	209	501	447.42	2421	296.95
4	131	675	515.11	1009	190.15

In Tables 5.2 and 5.3 we focus on the efficiency of the algorithm and the different ways of bounding. Table 5.2 concerns the base case, where only UB^1 is used as upper bound in Algorithm 1, and breadth-first-search is used as selection rule in both Algorithms 1 and 2. It shows the number of iterations for the leader problem and the maximum and average number of iterations for Algorithm 5.1 when it is called at each iteration of Algorithm 5.2 to solve the corresponding (reverse) medianoid problems. First of all, one can observe from the number of iterations, that it is relatively easier for the algorithm to detect what is the

global optimum for the leader when it has already many existing facilities. The intuition is as follows. When the leader is a newcomer, it has many options to gain market capture by going close to existing facilities of the competitor; there are many local optima. The result is that it is harder for the algorithm (requires more splitting) to verify that an already found location is the best one. Typically, this is easier when the leader has already several facilities. The global optimum is far more pronounced and defined by staying away from its own facilities. Accordingly, the number of iterations required for solving the follower medianoid problems increases with k.

In Table 5.3, we focus on the effectiveness of the upper bounds of Algorithm 5.1. At each iteration, it computes the four upper bounds described in Section 5.3.5 and chooses the minimum of the upper bounds. In all the cases, upper bounds UB^1 and UB^4 were used. Upper bounds UB^2 and UB^3 which are based on the d.c. concept appeared not to be efficient since they were never lower than UB^1 or UB^4. Observing the computations during the process, we found that UB^4 mainly improves the bounding of UB^1 when the partition sets get small. In this way, it contributes to speeding up the algorithm compared to only using UB^1. As in the previous table, the first two columns of Table 5.3 give the maximum and average number of iterations for Algorithm 5.1 when it is called at each iteration of Algorithm 5.2 to solve the corresponding (reverse) medianoid problems . The next four columns show the maximum and average number of iterations that the bounds UB^1 and UB^4 were the ones giving the minimum upper bound when solving the medianoid problems, whereas the last four columns give similar values when solving the reverse medianoid problems. Comparing Tables 5.2 and 5.3 we can see that the use of the both bounds reduces the number of iterations required for solving the corresponding (reverse) medianoid problems.

Table 5.3. *Number of iterations and upper bounds used. Selection rule: breadth-first-search in both algorithms.*

	Iterations				Upper bounds used							
	Follower med. problems		Reverse med. problems		Follower medianoid problems				Reverse medianoid problems			
					UB^1		UB^4		UB^1		UB^4	
k	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg
0	497	295.70	3645	218.32	479	278.62	49	17.08	3645	208.39	695	9.93
1	411	302.16	3107	241.31	392	280.92	40	21.24	3107	222.89	1471	18.42
2	527	414.59	2709	164.11	496	390.28	58	24.31	2709	160.59	241	3.52
3	467	410.79	2421	291.36	418	367.99	60	42.80	2398	275.37	328	15.99
4	571	471.90	1009	190.91	495	412.98	91	58.92	1009	184.93	172	5.98

In a next computational analysis we vary two rules of the algorithm. First of all, we compare the efficiency of the selection rule changing from breadth-first-search to best-bound-search, i.e., the rectangle with the lowest value of z^L is selected to be split next in Step 8 of Algorithm 5.1 and Step 9 of Algorithm 5.2. Secondly, we evaluate the performance when initially a partition is generated such that none of the demand points is interior as illustrated in Figure 5.3. The idea is that the upper bounds UB^4 get sharper.

Comparing Tables 5.2 and 5.4, one can observe that Algorithm 5.1 clearly improves over the thousands of problems solved with the selection rule best-bound-search. Algorithm 5.2 for the leader problem does not always improve for this particular case. For the algorithm variant where the best upper bound is used, comparison of Tables 5.3 and 5.5 confirms that best-bound-search is better for Algorithm 5.1 than breadth-first-search.

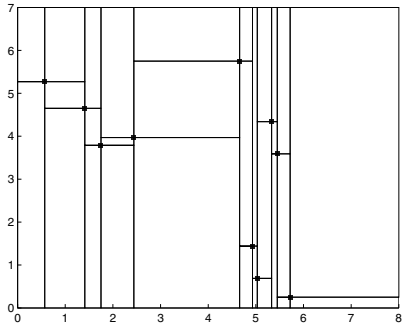

Figure 5.3. *Initial partition generated for the follower medianoid.*

Table 5.4. *Efficiency changing to best bound selection. Iterations. Upper bound UB^1 in Algorithm 5.1, selection rule: best-bound-search in both algorithms*

		Medianoid problems			
	Leader problem	Follower M. problems		Reverse M. problems	
k		Max	Avg	Max	Avg
0	689	613	184.25	2945	115.70
1	675	497	241.24	2893	71.21
2	1739	539	299.91	2519	58.59
3	463	401	362.57	8363	120.87
4	85	561	434.12	3871	140.64

Table 5.5. *Efficiency, best upper bound used, selection rule: best-bound-search.*

	Iterations				Upper bounds used							
	Follower med. problems		Reverse med. problems		Follower medianoid problems				Reverse medianoid problems			
					UB^1		UB^4		UB^1		UB^4	
k	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg
No initial partition												
0	589	184.13	2943	116.72	537	163.97	81	20.16	2943	105.41	234	11.31
1	479	209.67	2891	70.07	466	192.50	54	17.17	2891	64.05	80	6.02
2	389	249.43	2517	50.95	325	226.04	76	23.39	2517	49.36	106	1.59
3	277	236.35	8363	116.47	233	214.83	44	21.52	8363	112.45	221	4.02
4	471	282.69	3871	141.90	390	249.48	84	33.20	3871	138.23	29	3.67
With initial partition												
0	495	308.14	2856	146.22	473	269.90	101	38.24	2856	134.96	233	11.26
1	517	356.47	2938	76.81	415	297.37	115	59.10	2938	70.88	80	5.93
2	707	492.82	2578	53.54	617	407.77	148	85.05	2578	51.94	77	1.60
3	525	443.36	8363	126.74	480	392.15	79	51.21	8363	123.50	221	3.24
4	647	455.16	3871	143.81	525	391.40	142	63.76	3871	137.94	30	5.87

Comparing efficiency between generating an initial partition or not, Table 5.5 shows that the case "No initial partition" is better for the medianoid problems. This effect is less for the reverse medianoid problems, because for this problem Algorithm 5.1 is applied to smaller rectangles.

We now focus on the memory requirement as performance indicator. As said, Branch-and-Bound algorithms are usually hindered by huge search trees that need to be stored in memory. This part of complexity usually increases rapidly with dimension and with accuracy. Table 5.6 shows the memory requirements when the best of the four upper bounds is used. Selection rule applied is best-bound-search for both algorithms and the accuracies are $\varepsilon_l = 0.01$ and $\varepsilon_f = 0.01$. The second column shows the number of rectangles required by Algorithm 5.2 as the maximum number stored during the iterations. In the columns 3 to 6 the maximum and average number (over the solved problems) are given of memory requirement for the medianoid and reverse medianoid problems, respectively.

Table 5.6. *Memory requirement. The best of upper bounds is used, selection rule: best-bound-search and* $\varepsilon_l = 0.01$ *and* $\varepsilon_f = 0.01$.

	Leader problem	Follower med. problems		Reverse med. problems	
k	No. Rec.	Max	Avg	Max	Avg
0	15	22	9.92	26	7.43
1	20	15	11.84	24	6.23
2	23	30	13.04	27	5.08
3	17	15	14.00	26	9.10
4	5	22	14.56	22	8.38

Table 5.7. *Efficiency when accuracy is increasing. Case with* $k = 4$. *Selection rule: best-bound-search. For the follower (Follower M.) and reverse medianoid (Reverse M.) problems, numbers are on average.*

		Accuracy of the leader. ε_l					
		0.01			0.001		0.0001
		Accuracy of the medianoid and reverse medianoid problems. ε_f					
		0.01	0.001	0.0001	0.001	0.0001	0.0001
Iterations	Leader	85	95	95	143	151	219
	Follower M.	282.69	314.6	416.54	305.10	397.19	386.20
	Reverse M.	141.90	433.55	1186.64	296.20	784.34	549.65
Memory Req.	Leader	5	6	6	8	9	9
	Follower M.	14.56	15.54	18.6	15.36	18.38	18.26
	Reverse M.	8.38	11.21	14.44	9.02	12.01	9.71

One can observe that the memory requirement of the Branch-and-Bound approach for these continuous location problems is not dramatic for the used accuracy; there are never more than 30 subsets in the storage tree. Is this still the case if we increase accuracy? Notice that to have valid upper and lower bounds of the leader problem, the follower problem (giving lower bounds) and reverse medianoid (giving upper bounds) should be solved with

an accuracy that is at least as tight as that of the leader problem. We evaluate the number of iterations as well as the memory requirement if the accuracy is tightened for the case where the number of existing facilities is taken as $k = 4$. The results in Table 5.7 show that the number of iterations of the algorithms increases less than linear with the used accuracy in terms of $1/\varepsilon$. The memory requirement hardly goes up, showing that the best bound selection rule is efficient.

Given the evaluations of different variants of the algorithm on this case, in the next cases we apply a best-bound selection rule, the best upper bound at each iteration and no initial partitioning of the domain is generated.

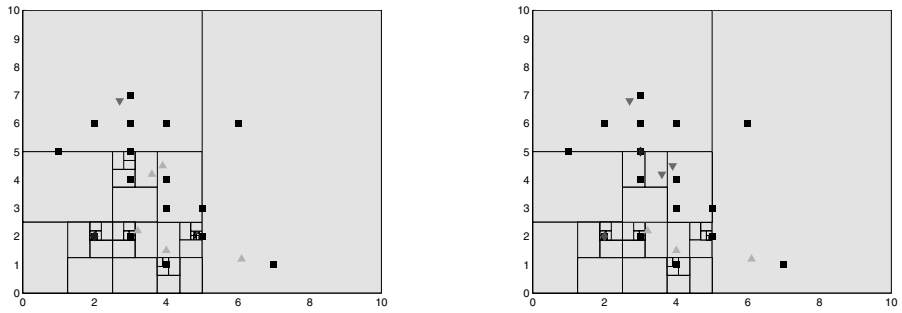

Figure 5.4. *Generated partition by the algorithm. Case from Drezner and Drezner (1998):* $k = 1$ *(left),* $k = 3$ *(right)*

5.5.2 Case II From Literature

In the second case where $n = 16$ and $m = 6$, data have been taken from (Drezner and Drezner 1998). In that paper, the existing facilities all belong to other chains different from the leader or follower. Thus, to adjust the data to our model, we have assigned the first k existing facilities to the leader and the rest to the follower. The data is different from randomly generated examples, as many points are situated along co-ordinate lines as can be observed from Figure 5.4. The exact location of demand points and other facilities can be found in Appendix 5.6. Table 5.8 shows the results of the algorithm for $k = 0, \ldots, m$. The optimal locations and resulting market capture for both chains are given.

One can observe the co-location effect when the number of existing facilities of the leader is low. Notice that this effect can also be observed when the leader is a newcomer with less facilities than the follower. Co-location of the new facilities does not occur when the follower is a newcomer, albeit co-location occurs with an existing facility of the competitor. Figure 5.4 gives an impression of the final partition generated by the Branch-and-Bound algorithm for the leader (cases with $k = 1$ and $k = 3$), together with the locations of demand points, existing facilities and new facilities.

Table 5.9 shows the number of iterations and the use of the 4 upper bounds. As in Case I, only upper bounds UB^1 and UB^4 were used.

Finally, Table 5.10 shows the memory requirements for Case II. The second column shows the maximum number of rectangles stored during the iterations by Algorithm 5.2. Columns 3 to 6 show the maximum and average number of rectangles stored for the follower medianoid and reverse medianoid, respectively.

Table 5.8. *Optimal locations Case II, market capture and number of iterations for both chains. Parameter z_l^* = market capture after locating facility, Mb_l before; locations and market captures are rounded to two decimals.*

		$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
Optimal location	Leader	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}2.00\\2.00\end{pmatrix}$	$\begin{pmatrix}2.00\\2.00\end{pmatrix}$	$\begin{pmatrix}2.00\\2.00\end{pmatrix}$
	Follower	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}1.99\\1.99\end{pmatrix}$	$\begin{pmatrix}3.00\\5.00\end{pmatrix}$	$\begin{pmatrix}3.00\\5.00\end{pmatrix}$	$\begin{pmatrix}3.00\\5.00\end{pmatrix}$	$\begin{pmatrix}3.00\\4.99\end{pmatrix}$
Market Capture	Leader	203.36	368.82	455.09	661.24	872.68	1037.21	1087.25
	Follower	1143.14	977.68	891.41	685.26	473.82	309.29	259.25
$z_l^* - Mb_l$ (gain or loss)		203.36	157.31	129.67	48.31	-140.26	-234.34	-259.25

Table 5.9. *Number of iterations when the best of the 4 upper bounds are considered. Selection rule: best-bound-search in Algorithm 5.1 and Algorithm 5.2.*

						Upper bounds used							
		Iterations				Follower medianoid problems				Reverse medianoid problems			
	Iterations	Follower med. problems		Reverse med. problems		UB^1		UB^4		UB^1		UB^4	
k	Leader	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg	Max	Avg
0	1417	913	450.32	4633	165.23	839	413.93	119	36.39	4633	128.21	2107	37.02
1	1127	297	232.14	1517	54.40	288	222.85	25	9.29	1517	48.00	121	6.40
2	715	277	217.93	2001	82.97	269	209.05	19	8.88	2001	81.62	117	1.35
3	249	261	174.36	1513	118.04	243	160.58	20	13.78	1513	107.06	315	10.98
4	177	239	183.17	573	83.25	214	153.96	33	29.21	573	75.65	103	7.60
5	181	249	190.83	405	63.19	219	155.67	38	35.16	405	59.58	37	3.61
6	125	389	248.33	557	61.77	345	215.78	44	32.55	557	56.76	29	5.01

Table 5.10. *Memory requirement Case II. Max number of stored rectangles.*

	Leader	Follower M.		Reverse M.	
k	Problem	Max.	Avg.	Max.	Avg.
0	22	29	18.32	27	9.80
1	24	12	11.15	26	6.11
2	16	11	10.92	28	6.18
3	10	12	11.16	28	7.15
4	10	12	11.77	17	6.58
5	10	12	12.00	21	6.14
6	10	15	12.73	22	6.45

5.5.3 Case III, Varying Problem Size

In this section, numerical results of the evaluation of the Algorithms 5.1 and 5.2 are discussed. The wider question is whether the algorithms are able to solve larger problems in reasonable time. To study the performance of the algorithms, we have generated different types of problems, varying the number n of demand points, the number m of existing facilities and the number k of facilities belonging to the leader chain. For every type of setting, ten problems were randomly generated. The settings are defined by choosing:

- $n = 20, 30, \ldots, 110$
- $m = 5, 10, 15$
- $k = [m/2]$

For each n, m-combination parameter values of ten problems were uniformly chosen within the following intervals:

- $p_i, q_j \in ([0, 10], [0, 10]), i = 1, \ldots, n, j = 1, \ldots, m$
- $w_i \in [1, 10], i = 1, \ldots, n$
- $a_j \in [0.5, 5], j = 1, \ldots, m$

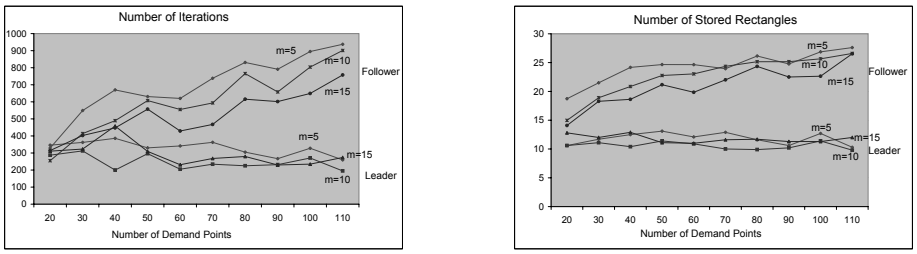

Figure 5.5. *Average number of iterations and memory requirement (rectangles) over 10 random cases varying number of demand points* $n = 20, \ldots, 110$*, existing facilities* $m = 5, 10, 15$ *and* $k = m/2$*. Selection rule: best-bound-search and* $\varepsilon_l = \varepsilon_f = 0.01$*.*

From Figure 5.5, one can observe that an increasing number of demand points does not make the problem more complex in terms of the memory requirement for the Branch-and-Bound. The leader problem neither needs more iterations. The follower problem however, needs more iterations on average to reach the predefined accuracy. The experiment suggests that no exponential effort is required to solve the problems with increasing number of demand points. This confirms the viability of the approach.

5.6 Conclusions and Future Work

In this paper, we described a competitive Huff-like Stackelberg location model for market share maximization. There are two competitors (chains); first the leader locates and then the follower makes a decision with full knowledge of choices of the leader. We consider competition with foresight and probabilistic behaviour. Attraction of a customer is depending on the location and the quality of the facility. The location of the leader facility is the variable of the problem. The problem is known to be a Global Optimization problem. In order to solve it, we have constructed a Branch-and-Bound algorithm for the follower problem and for the leader problem. The Branch-and-Bound algorithms guarantee a global optimum within a given accuracy (gap between lower and upper bound). The introduced bound of the leader problem is based on the zero sum concept where gain of one chain is loss for its competitor. We have developed and compared four different upper bounds for the algorithm of the (reverse) medianoid problem.

The algorithms were illustrated with several cases. In a first case, the algorithm settings and performance were studied. The selection rule and accuracy were varied to study the performance and effectiveness of the different bounds. A variant where an initial partition is generated was also studied. In a second case taken from literature, good algorithm settings

from the first case were used. In the last case, many instances were generated at random where the size and the number of existing facilities is varied to validate the viability of the approach.

Looking at effectiveness, one can observe the co-location behaviour of the optimum strategy as one can expect. Also the difficulty on multimodal behaviour is reflected when measuring the efficiency as the number of iterations to solve the problem up to desired accuracy ε. Efficiency has been measured computationally. Comparing bounds and several variants with respect to selection rule and generating an initial partition to improve bounds, we found the following. More sophisticated bounds are not necessarily more effective than simple bounds based on distance comparison over the complete run of the algorithm. One can best focus on measuring the quality of the bound during the run and take the sharpest one. For the selection rule, the focus on the best bound (most promising) selection of the next subset to be split has the tendency to result in minimum effort on number of function evaluations. However, one always has to keep in mind that a depth first search may lead to less memory requirement of a Branch-and-Bound algorithm. Where memory requirement is usually a problem for higher dimensions, it is not necessarily a focus point for the location problem in two dimensional space.

Future research will include the quality of the leader and follower as variables of the problem.

Appendix A: Test Problems

Table 5.11. *Locations and distances from demand points to facilities*

	Demand points		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
	X axis		2,44	5,33	0,57	5,03	4,66	5,72	5,41	1,75	4,93	5,45
Facility		Y axis	3,97	4,34	5,27	0,69	5,75	0,25	1,65	3,79	1,44	3,59
1	2	5	1,12	3,40	1,45	5,27	2,76	6,04	4,78	1,24	4,61	3,72
2	3	2	2,05	3,30	4,07	2,42	4,10	3,24	2,43	2,18	2,01	2,92
3	1	3	1,73	4,53	2,31	4,65	4,58	5,47	4,61	1,09	4,23	4,49
4	5	4	2,56	0,47	4,61	3,31	1,79	3,82	2,39	3,25	2,56	0,61

Appendix B: Input Data for Example from (Drezner and Drezner (1998))

Table 5.12. *Distances from demand points to facilities*

	Demand Points															
Fac.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	1.82	0.36	1.06	4.81	2.48	0.85	2.82	4.85	5.32	7.22	5.94	3.09	1.53	4.02	4.44	3.40
2	1.03	2.66	2.42	2.66	2.94	1.75	1.03	3.14	2.73	4.68	3.50	0.51	1.50	1.50	1.86	2.58
3	1.00	2.86	2.41	2.28	2.72	1.90	0.63	2.72	2.61	4.67	3.22	0.45	1.84	1.26	1.84	3.00
4	2.81	4.80	3.98	0.28	3.56	3.81	1.81	1.22	1.81	3.98	1.44	1.97	3.88	1.13	1.97	4.72
5	3.64	5.59	4.92	1.12	4.61	4.61	2.69	2.06	1.12	3.04	0.50	2.50	4.50	1.50	1.80	4.92
6	4.90	6.58	6.31	3.20	6.36	5.71	4.18	4.18	1.36	0.92	2.11	3.50	5.24	2.77	2.11	4.80

Table 5.13. *Location and buying power for demand points and location and attractiveness for existing facilities*

	Facility points				Demand points		
Number	q_1	q_2	a_j	Number	p_1	p_2	w_i.
1	2.7	6.8	7	**1**	3	5	163.8
2	3.9	4.5	3	**2**	3	7	28.8
3	3.6	4.2	7	**3**	2	6	39.0
4	3.2	2.2	10	**4**	3	2	77.4
5	4.0	1.5	7	**5**	1	5	42.0
6	6.1	1.2	3	**6**	3	6	107.0
				7	3	4	64.5
				8	2	2	250.6
				9	5	2	101.4
				10	7	1	57.6
				11	4	1	132.0
				12	4	4	77.6
				13	4	6	29.6
				14	4	3	67.5
				15	5	3	50.7
				16	6	6	57.0

CHAPTER 6

General Discussion and Conclusions

6.1 Introduction

This chapter evaluates and discusses the research done in this thesis. In Section 6.2 a brief outline of the thesis is given. Section 6.3 presents the main conclusions by answering the research questions outlined on Chapter 1. The discussion is presented in Section 6.4, where contributions of this study, its limitations and suggestions for future research are discussed.

6.2 Brief Outline of the Research

The objective of this thesis was to develop or contribute to the development of usable GT models and OR methods to solve them in practical situations. For different cases, GT models were built from GT concepts and translated into mathematical programming models. The research aimed to provide a useful contribution to:

- Building optimisation models from GT concepts/models
- Developing OR techniques as tools to solve optimisation models based on GT concepts/models with multiple heterogeneous actors
- To apply the developed techniques on cases of cooperation and competition to show the applicability of the GT concepts/models, OR models/techniques.

We hypothesised that GT concepts and models might provide us with a valuable contribution for supporting decision makers provided that we can make OR models and algorithms to solve these models. In order to achieve the objective, aims and hypothesis of this thesis four research questions were addressed:

1. How do we formulate GT models for decision making situations with multiple heterogeneous actors using GT concepts? How can we contribute to that modelling?
2. If we have GT based models: What appropriate coding can be used to translate a specific GT model into a mathematical programming optimisation problem such that it can be solved?
3. Which solution methods can be developed to solve these models?
4. How do the outcomes aid to analyse the decision makers problem? or What is the contribution of the new outcomes to the decision makers problem?

In daily life, situations described by cooperative games can be observed in the form of alliances between companies, coalition formation in the parliament or club membership, among others. In this thesis we have studied two types of coalition formation. Chapter 2 describes and studies a multiple coalition formation game with membership rules and different transfer schemes. Furthermore, externalities were also an ingredient of the game. The chapter focuses on the computation of stability when the players are heterogeneous and asymmetric. GT concepts were translated into a new mathematical programming notation which allowed the implementation in an algorithmic context. Application of the method has been done in a case on International Environmental Agreements. The second type of coalition formation was studied in Chapter 3 which describes a cartel formation game, where only one coalition can be formed. The model was introduced in de Ridder and Rusinowska (2005). There are n parties trying to form a majority government and to agree on a policy agreement represented in a m-multidimensional Euclidean policy space $\mathbb{R}^m$. It was shown how the complexity increases with the number of parties n and the dimension of the policy spaces m. Computational methods were developed for two different procedures of coalition formation: sequential and simultaneous. The methods compute all possible winning coalitions and preferences of parties over those (if many) coalitions. The developed OR methods provided the ability to study several hypotheses.

In daily life not only cooperation can be observed; competition takes place between companies, in the parliament and between clubs. In this thesis we studied competition in two chapters. Chapter 4 describes competition on quantity and location. The game describes decision on two levels and the interaction between $n > 2$ heterogeneous firms. There are $m > 2$ markets to be served. The game studied in Sarkar et al. (1997) and Rhim et al. (2003) was extended by considering asymmetric costs (firm-specific). Another difference with the study of Rhim et al. (2003) is the procedure on how to find the equilibrium of the game. Algorithms were designed and illustrated numerically. Chapter 5 studies a Stackelberg (leader-follower) competitive facility location situation. There are two competitors (chains). First the leader makes a decision on where to locate its facility in the plane. Second, the follower makes a decision with full knowledge of the decision of the leader. The objective of the leader is to maximize its market share considering that the follower will react to her decision. Thus, the leader must anticipate the decision of the follower. Table 6.1 shows an overview of the cases applied.

6.3 Main Conclusions and Research Questions

In this section, the main conclusions are presented by answering the research questions formulated in Section 6.2 (see also Chapter 1). How the cases contribute to insight in the research questions is described.

Answer to the Research Question 1

- How do we formulate GT models for decision making situations with multiple heterogeneous actors using GT concepts? How can we contribute to that modelling?

It is interesting for science and, in particular for algorithm developers, to try to handle large datasets and make GT more applied and bring it closer to applications. In Chapter 1, Section 1.5.1, the main characteristics of the cases were described. Within these characteristics, ingredients for modelling decision making situations with multiple heterogeneous actors were included. The contribution of this study on the modelling aspect is due to reformulating

Table 6.1. *Overview of the cases applied.*

Conceptual Model Phases	Chapter 2	Chapter 3	Chapter 4	Chapter 5
GT Concept	Nash, stability, multiple coalition formation, transferable utility	Single coalition formation, preference orders	Cournot, Nash, competition	Stackelberg, competition
GT Model	Eyckmans and Finus (2003a), two-level game	de Ridder and Rusinowska (2005), political game	Sarkar et al. (1997), location-quantity, two-level game	Drezner and Drezner (1998), competitive location
OR Model	Continuous optimisation at one level, discrete location	NLP with inequalities	Generalised existing model, NLP	Global optimisation, continuous location
Algorithm	FOC quadratic function, enumeration, coding large matrices	Penalty approach, iterative algorithm, standard NLP	Analysis, properties of continuous equilibria, enumeration, local search, multistart	Continuous optimisation, Branch-and-Bound

models in order to be able to handle large datasets. In the following, the main contributions of this study on the modelling are outlined.

In Chapter 2, we used a GT model for a multiple coalition formation game from Eyckmans and Finus (2003a). Contribution in the modelling was done by introducing a large number of heterogeneous players and modelling different side payments. The GT model is based on the GT concepts of Nash equilibrium, stability and transferable utility.

In Chapter 3, a single coalition formation model from de Ridder and Rusinowska (2005) was used. There are n players (political parties), which try to form a majority coalition and to decide about a policy of the coalition. This coalition position is the formal representation of the policy agreement of a coalition. Each player has a weight which is based on the number of seats in parliament. The model implies two procedures on coalition formation: sequential and simultaneous. Players are heterogeneous and asymmetric. For both procedures, all winning coalitions and preference orders over these coalitions were generated.

In Chapter 4, a two-stage location-quantity game with m markets and n firms was described. The location space is a network, where the nodes are considered as possible locations for the firms. We extend the study of Sarkar et al. (1997) and Rhim et al. (2003). Free entry is possible as in Rhim et al. (2003). Moreover, we allow asymmetric costs (firm-specific). Another difference with the study of Rhim et al. (2003) is the procedure on how to find the equilibrium of the game. We consider not only the possibility for a suplier to leave a market, but also the possibility to move its facility to another location. Doing so, a firm has to re-think the quantity decision on how much to supply to which markets.

In Chapter 5, the model is based on a Stackelberg zero-sum game, which models the case discussed in Drezner and Drezner (1998) as a zero-sum leader-follower competitive location problem. The problem considered is a planar facility location with foresight, having probabilistic consumer behaviour, based on an attraction function depending on the locations of the facilities to be located. We reformulated the model in an exact way.

In summary, the different decision making problems studied were modelled with the aid of the above mentioned GT concepts/models. Contribution was most significant in Chapter 2 and Chapter 4 adding additional characteristics to cases taken from literature. In general, asymmetric, heterogeneous and multi-actor situations on the decision making problems are the basis for all the cases applied.

Answer to the Research Question 2

- If we have GT based models: What appropriate coding can be used to translate a specific GT model into a mathematical programming optimisation problem such that it can be solved?

As explained above, we consider GT concepts/models as a challenge to be applied on large, asymmetric, heterogeneous, multi-actor cases. In order to develop OR methods/algorithms for solving decision making problems, one needs to translate the model formulation from GT into mathematical programming optimisation models. Contribution on the reformulation of GT models into OR models is described below.

In Chapter 2, concepts from GT related to the case were translated into a new mathematical programming notation. This new mathematical programming notation was used for developing a constrained, continuous (in abatement) *NLP* model. In modelling the equilibria, stability of coalitions, a combinatorial optimisation is used. The research in Chapter 2 shows how by mathematically redefining concepts, computer coding has been facilitated that made it possible to generate relevant results for huge cases. The resulting model was used in further studies (Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005)).

In Chapter 3, a mathematical programming model from de Ridder and Rusinowska (2005) was re-written and used for developing a continuous and constrained *NLP* model. The model involves also a discrete space when considering the fixed number of possible coalitions. The model considers both static and dynamic aspects with respect to the two different procedures under consideration: simultaneous and dynamic processes of coalition formation. The formulated model and coding made the application of the model possible using real and multidimensional data.

In Chapter 4, a two-stage model on location and quantity decisions was described. Variants from the original model were introduced, and therefore an exact formulation was necessary to introduce a constrained NLP model. The model is continuous on quantity decisions, and discrete on location decisions. Conditions for the supplying decisions (second-stage of the game) were provided. Moreover, as firms will be affected by the timing and level of entry on the market, properties on how to determine the size of the market were derived.

In Chapter 5, a new mathematical programming re-formulation for a Stackelberg (leader-follower) problem is derived. The leader-follower problem described belongs to a constrained, global optimisation (GO) problem. The formulation using OR models designed the way on how to solve the problem with global optimisation methods on two levels.

Answering this question, the "coding into mathematical programming notation" becomes an important factor and results in a crucial contribution for the next steps of the conceptual model outlined in Chapter 1 (see also Figure 6.1). It helped to get insight on how to solve the decision problems faced in the different cases.

Answer to the Research Question 3

- Which solution methods can be developed to solve these models?

A significant contribution of this thesis is due to the development of algorithms for solving decision making problems modelled by GT concepts and reformulated as OR models. The first two questions answered the modelling and reformulation of the models. The reformulation step is crucial for the development of algorithms.

Chapter 2 shows that the implementation provides the feasibility to study a multiple coalition game with a large number of asymmetric and heterogeneous players. Challenge was the development of algorithms for getting stable coalition structures. The mathematical programming coding described above was used to build an OR method for implementation into computer coding. Analytical solutions were derived for optimal abatement levels for each coalition structure. A complete enumeration algorithm was developed. The algorithm was able to compute stable coalition structures in a multiple coalition game for a model with 12 players (regions), leading to more than four million of coalition structures to be checked for stability. Implementation and computational aspects were outlined and results were shown.

In Chapter 3, two different procedures were developed. The first approach is a dynamic procedure, a step-by-step process of coalition formation. The second approach is a static procedure, a simultaneous process on negotiations. The challenge is to apply the methods in multidimensional decision space, multi-actor cases. In doing so, algorithms became a necessary tool in order to apply the model to cases with real data. Algorithms were developed to generate all winning coalitions and preference orders. The algorithms used an external procedure for the determination of the output based on sequential quadratic programming (SQP) methods. Moreover, a penalty approach was used to determine whether or not feasible negotiation spaces intersect between the negotiating parties (the players of the game).

In Chapter 4, based on the continuous model on quantity decisions, analytical results were used to develop an algorithm to get the equilibrium on quantities. The model is discrete when considering location decisions, two different methods were developed to find the equilibria in locations: a complete enumeration algorithm and a local search procedure. The complete enumeration algorithm aims to find all Nash equilibria of the game. In contrast to the complete enumeration algorithm, for the local search algorithm it is not known whether or not all equilibria (if many) are found. The results and algorithms were illustrated numerically.

The problem in Chapter 5, to our knowledge, was only addressed in Drezner and Drezner (1998), where three heuristics are described for a variant of the model considered in this chapter. The question was whether the leader problem can be solved up to a guaranteed accuracy. The branch-and-bound algorithms guarantee a global optimum within a given accuracy (gap between lower and upper bound). Thus, branch-and-bound algorithms for the follower problem and for the leader problem were developed. Moreover, four different upper bounds for the algorithm of the medianoid problem (follower) were developed and compared.

In answering this question, an important contribution was done with the help of the coding step. Algorithms were developed for each of the problems covered in this research. Algorithms were based on enumeration methods, iterative approaches, analytical solutions, SQP methods, penalty approach and GO methods. From the cases applied, different challenges inspired the development of the algorithms: application to huge cases, asymmetry, heterogeneity or global optimisation methods.

Answer to the Research Question 4

- How do the outcomes aid to analyse the decision makers problem? or What is the contribution of the new outcomes to the decision makers problem?

The research questions followed the conceptual model of the thesis. GT concepts/models were reformulated into OR models to help in the development of OR algorithms. The last research question is answered by looking at the output of the algorithms.

Numerical results from Chapter 2 showed that, in the environmental game applied, exclusive membership with a unanimity voting rule leads to more stability than open membership. By applying transfer schemes the number of stable coalition structures was bigger but not as much as one could expect. Comparing with earlier studies and results applying the STACO model (Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005), Finus et al. (2005), Olieman and Hendrix (2005)), Chapter 2 showed in a numerical study that the number of stable coalition structures in a multiple coalition game is bigger than in a cartel game and preferable if we are measuring in terms of either Net Benefit or Global Emission Reduction.

In Chapter 3, methods were developed to study two dynamical aspects of coalition formation (procedure and policy flexibility) and to report on the findings for testing hypotheses by analysing the formal model and deducing implications from this model based on real data. The analysis in de Ridder et al. (2007) focused on three aspects of coalition formation and formulated hypotheses: procedure, flexibility, and power sharing. By applying the developed algorithms to Dutch data and theoretical examples, several questions/hypotheses were tested and gave implications for political parties involved in coalition formation.

Algorithms developed in Chapter 4 were compared and both effectiveness and efficiency were studied. We found analytically a necessary condition to have delivery to a market, an analytic expression of the equilibrium quantities and necessary condition for the firms that supply to a market. Furthermore, by using the algorithms as a systematic computation instrument to cases reported in literature, a mistake was detected in Sarkar et al. (1997). In that paper a solution is given that appears not to be an equilibrium of the model. Furthermore, tests on larger generated instances showed the viability of the approach.

In Chapter 5, GO techniques for solving the leader and the follower problem were illustrated with several cases. In a first case, the algorithm performance and effectiveness of the different bounds were studied by varying the selection rule and accuracy. A variant where an initial partition is generated was also studied. A good algorithm setting from the first case was used to get computational results. With the optimal settings, many instances were generated at random where the size and the number of existing facilities is varied to validate the viability of the approach. The difficulty on multimodal behaviour is reflected when measuring the efficiency as the number of iterations to solve the problem up to desired accuracy ε. Efficiency has been measured computationally.

In summary, the idea is that outcomes of the developed algorithms aim to help decision makers and analysts to get insight into a decision situation: what are optimal or better decisions. Two of the chapters contributed on aiding researches when real data are used (Chapters 2 and 3). Chapters 4 and 5 were illustrated with randomly generated data. From the outcomes and different approaches, different conclusions and discussions on the decision problems can be made. Moreover, conclusions and discussions about the results help to get more insight on further research and new questions.

6.4 Discussion of the Research

This section is organised as follows. Section 6.4.1 discusses the contribution of the study to the existing literature. Section 6.4.2 indicates the limitations of the research.

6.4.1 Contributions of this Study

The research done in this thesis studies cooperative and competitive games. This section makes also a contribution to the answers on the research questions, where the main findings of the study were outlined. The first two cases, Chapter 2 and Chapter 3, were based on coalition formation. We will first discuss the contribution to coalition formation games and the contribution made on OR techniques. To remind to the reader, Chapter 2 applies a game on simultaneous multiple coalition formation, with heterogeneous and asymmetric players, four different transfers schemes and externalities; Chapter 3 applies a game on cartel (single) coalition formation, with heterogeneous and asymmetric players, multidimensional decision space and two different procedures on forming a coalition (sequential and simultaneously).

In Chapter 2, a significant contribution has been made by representing a multiple coalition formation game with a mathematical programming notation, which helped to introduce the concept of neighbourhood of coalition structures in an exact way. This new mathematical notation was very useful for doing stability checks when heterogeneous, asymmetric and more than 2 actors are interacting in a cooperative game in which more than one coalition is allowed to form. For stability checks, an enumeration algorithm was developed to check the equilibrium or all equilibria (if many) of the game based on analytical solutions for an NLP problem on abatement. An application on an International Environmental Agreement problem was studied. This application implies more than 4 million coalition structures to be analysed. Validation of the techniques was made by applying a small case from literature. The approach helped other researches in developing new ideas and publications within the scientific-research world. An important and valuable contribution to the STACO project was made. Many papers were developed using the coding and methods for getting economic incentives on environmental issues (see Sáiz et al. (2006), Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005)).

An extensive literature can be found on political coalition models. Most of the studies focus on which parties will form a coalition. However, little research has been done on the procedures to form a coalition (Laver and Schofield (1990): how will coalitions be formed, and, what is the best strategy for a party during the process of coalition formation? In Chapter 3, two different procedures were considered and methods for each procedure were developed. The methods allow the applicability to real data on high dimensions with heterogeneous players. Real-data and theoretical cases were used not only to obtain winning coalitions and preferences of parties over these coalitions and procedures, they also contributed on testing hypotheses. From the hypotheses, we found: that being a first mover is not always advantageous; when forming a k-party coalition, for $k \geq 3$, being less flexible is usually advantageous, but can theoretically be a disadvantage; forming a minimal winning coalition is not always advantageous; and when forming a k-party coalition, for $k \geq 3$, an increase of a party's weight may be an advantage for its coalition partner

Next we discuss the contribution done to competitive games and OR techniques. Competitive games were studied on Chapter 4 and Chapter 5. Competitive location is an ingredient for both chapters, in a discrete and continuous setting (respectively). In particular, quantity competition "a la Cournot", with asymmetric costs, $n > 2$ firms, $m > 2$ markets and free-entry was applied in Chapter 4. In Chapter 5, a Stackelberg leader-follower game with probabilistic behaviour of the customers was studied. We first dicuss Chapter 4 and then Chapter 5.

Many studies can be found in literature describing a so-called location-quantity game. Most of the studies assume symmetric behaviour or a finite strategy set or two or few players or there is no free-entry, that is, all the firms will enter the market. In Chapter 4, a game from literature was extended. Free-entry is possible and costs are asymmetric (firm-specific). Two

different methods were developed: a complete enumeration method discovering all equilibria on the game; and a local search method in which given a starting point, the algorithm walks to one of the equilibria (if many). The first method guarantees that all equilibria are found. With the second method it is not known whether or not all equilibria are found.

In Chapter 5, GO methods were developed to solve the leader and the follower problems. The follower problem has been solved under deterministic and probabilistic behaviour in several studies. To our knowledge, the leader problem has only been addressed in Drezner and Drezner (1982) and Bhadury et al. (2003) (with deterministic behaviour), and in Drezner and Drezner (1998) (with probabilistic behaviour), where three heuristics are described for a variant of the model considered. In this chapter, by making use of the zero-sum perspective of the game, the Stackelberg leader-follower problem with probabilistic behaviour was solved up to a guaranteed accuracy. Moreover, we found that more sophisticated bounds are not necessarily more effective than simple bounds based on distance comparison over the complete run of the algorithm. For the selection rule, the focus on the best bound (most promising) selection of the next subset to be split has the tendency to result in minimum effort on number of function evaluations. However, one always has to keep in mind that a depth first search may lead to less memory requirement of a Branch-and-Bound algorithm.

In brief, the main contributions of this thesis can be summarised as follows. In this thesis we have studied GT models and reformulated the models into OR optimisation problems. The reformulation into OR optimisation problems was a worthwhile step for the development of algorithms. For all the cases applied, algorithms were designed following the conceptual model introduced in Chapter 1. For the easy of reading and presentation, we show again this conceptual model in Figure 6.1. We coded the GT concepts and models into mathematical

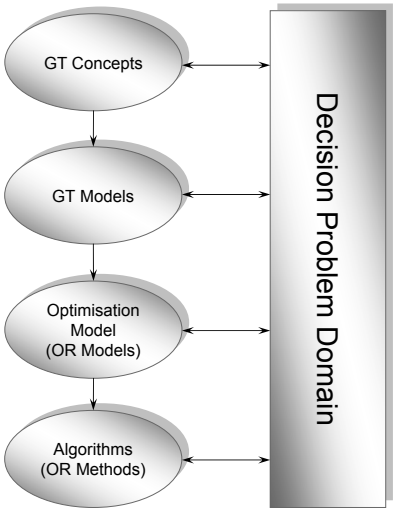

Figure 6.1. *Conceptual Model of Thesis Research.*

programming notation. This coding step makes the design of the methods for solving the decision making problems easier. In addition, real data was used in two cases, which makes the development of methods for solving the models more relevant (Chapters 2 and 3). Fur-

thermore, in one case we extended the GT model incorporating new ingredients (Chapter 4). In another case, we have solved the model using GO techniques (Chapter 5).

6.4.2 Limitations of the Research and Suggestions for Further Research

In the scientific world, every study has its limitations and findings must be considered within boundaries. However, looking at the optimistic part of limitations, boundaries are the next step on research. Of course, this research also had limitations and boundaries. Main recommendation is that other cases could be studied in which other GT concepts, models, OR models and OR algorithms are applied. This study should be seen as a first step in making GT more applicable to larger, more complex decision problems. The cases applied have specific characteristics which guided us in the selection of specific GT models, OR models and algorithms. First of all, Tables 1.1, 1.2 and 1.3 (Chapter 1) show the GT concepts/models and OR models/algorithms illustrated in this thesis. These are the first boundaries of the research. In the outline given, within decision making problems, less attention is given to bounded rationality. Further research can be directed to consider bounded rationality in decision making problems. With respect to GT, we did not study incomplete information games. Research on cases with large number of players on incomplete information games are of interest. Further research could be done on the development of GT models, on the coding of these GT models into OR models and on the development of new OR methods/algorithms. It will contribute to further insight on GT concepts/models and OR models/techniques. When considering the OR models and algorithms used in this research, the models and algorithms developed in this study were based on the characteristic of the cases. Further research includes the study of the performance of the developed algorithms in comparison to other algorithms from OR. Next, limitations of the research are described by chapters.

In Chapter 2, further research could be done by studying a case in which multiple deviation is considered. In this chapter, stability is checked when only one player can deviate from her strategy. If we consider multiple deviation, more than one player can deviate from their strategies. It will lead to more insights on GT as well as on OR formulation of models and algorithms. Another limitation of the study is related to the algorithms. The development of new algorithms for this specific case could improve the performance of the enumeration method. If multiple deviation is considered, future research could study either to update the algorithm developed or to develop new algorithms.

With respect to Chapter 3, one of the main limitations is the flexibility of the players (parties). To remain, the flexibility of a party (player) was defined by the radii of the maneuvering space of the party, which consists of all acceptable positions. The maneuvering spaces are defined as a m-multidimensional Euclidean policy space. The flexibility of the players was selected for testing the formulated hypotheses. More research and theory development could be done in order to derive a good setting for the flexibility of players in political games. A further step in selecting the flexibility data for the parties would make the output more realistic. It is also interesting for further research to study the application of existing or new stability concepts of hedonic games.

Chapter 4 considers a game on location and quantity in a discrete location model. The methods and cases consider the location on a node of a network. More research could be done by considering that location is also possible at the edges. The question is if the multi-actor heterogeneous, free-entry and simultaneous location-quantity game can be reformulated and coded. Challenge is the development of new algorithms for the characteristics of the game. Moreover, the enumeration algorithm guarantees to find all the equilibria of the game, as in Chapter 1. Future research could also investigate the development of more efficient

algorithms with the guarantee of finding all the equilibria.

In Chapter 5, the quality (or size) of the facilities were assumed to be input data for the model. An interesting way of further research is the development of new GO methods when considering the quality as another variable to be optimised. Within the settings of the problem of this research, it has been proven that more sophisticated bounds are not more efficient. The question is if this still holds when applying the algorithms with other ways of branching. Another interesting way of looking at the problem is related to the bounding. Further research could focus on the development of more efficient bounds. Within Stackelberg (leader-follower) facility location games, another challenge for further research is to investigate new GO methods when more than two chains are competing (leader-followers).

BIBLIOGRAPHY

Altamirano-Cabrera, J. C. and Finus, M.: 2006, Permit trading and stability of international climate agreements, *Journal of Applied Economics* **IX (1)**, 19–48.

Altamirano-Cabrera, J., Finus, M. and Dellink, R.: 2005, Do abatement quotas lead to more successful climate coalitions, *Technical report*, Working Paper in Economics 383. (Hagen University).

Arnold, T. and Schwalbe, U.: 2002, Dynamic coalition formation and the core, *Journal of Economic Behaviour and Organization* **40**, 363–380.

Austen-Smith, D. and Banks, J.: 1988, Elections, coalitions and legislative outcomes, *The American Political Science Review* **82**, 405–422.

Axelrod, R.: 1970, *Conflict of interest, a theory of divergent goals with applications to politics*, Chicago: Marham.

Baron, D.: 1993, Government formation and endogenous parties, *The American Political Science Review* **87(1)**, 34–47.

Benders, J., Beulens, A. and van Nunen, J.: 1983, *Groot Vademecum voor Bezuiniging en Besparing*, De Toorts-Haarlem/Van der Wolk Publicaties-Laren NH„ chapter Operations Research optimaliseert activiteiten, pp. 5.01–5.18.

Berge, C.: 1968, *Principes de combinatoire*.

Bhadury, J., Eiselt, H. and Jamarillo, J.: 2003, An alternating heuristic for medianoid and centroid problems in the plane, *Computers and Operations Research* **30**, 553–565.

Bloch, F.: 1995, Endogenous structures of association in oligopolies, *Rand J. Econ.* **26**, 537–556.

Bloch, F.: 1996, Sequential formation of coalitions in games with externalities and fixed payoff division, *Games and Economic Behavior* **14(1)**, 90–123.

Boggs, P. T. and Tolle, J. W.: 1996, Sequential quadratic programming, *Acta Numerica* **4**, 1–51.

Bosello, F., Carraro, C. and Buchner, B.: 2003, Equity, development, and climate change control, *Journal of the European Economic Association* **1(2-3)**, 601–611(11).

Botteon, M. and Carraro, C.: 1997, *International Environmental Negotiations: Strategic Policy Issues*, E. Elgar: Cheltenham, chapter Burden-Sharing and Coalition Stability in Environmental Negotiations with Asymmetric Countries, pp. ch. 3:26–55.

Brams, S., Jones, M. and Kilgour, M.: 2005, Forming stable coalitions: The process matters., *Public Choice* **125(1-2)**, 67–94.

Budge, I.: 1994, A new spatial theory of party competition: Uncertainty, ideology and policy equilibria viewed comparatively and temporally, *British Journal of Political Science* **24(4)**, 443–467.

Bulow, J. I., Geanakoplos, J. D. and Klemperer, P. D.: 1985, Multimarket oligopoly: Strategic substitutes and complements, *Journal of Political Economy* **93**(3), 488–511.

Carraro, C.: 1999, *International Environmental Agreements on Climate Change*, Kluwer Academic Publishers, Dordrecht, chapter The Structure of International Agreements on Climate Change.

Carraro, C., Bosello, F., Buchner, B. and Raggi, D.: 2001, Can equity enhance efficiency? lessons from the kyoto protocol, *Technical report*, FEEM Working Paper No. 49.2001.

Carraro, C. and Marchiori, C.: 2003, *The Endogenous Formation of Economic Coalitions*, E. Elgar: Cheltenham, chapter Stable Coalitions.

Casado, L. G., Hendrix, E. M. T. and García, I.: 2007, Infeasibility spheres for finding robust solutions of blending problems with quadratic constraints, *Journal of Global Optimization, doi: 10.1007/s10898-007-9157-x* .

Colson, B., Marcotte, P. and Savard, G.: 2005, Bilevel programming: A survey, *A Quarterly Journal of Operations Research* **3 (2)**, 87–107.

Cournot, A.: 1838, *Recherches sur les Principes Mathématiques de la Théorie des Richesses (Englishedition, Researches into the Mathematical Principles of the Theory of Wealth,N. Bacon (ed.), 1897)*, Macmillan, New York.

Dantzig, G.: 1963, *Linear Programming and Extensions*, Princeton University Press, Princeton, New Jersey.

d'Aspremont, C., Jacquemin, A., Gabszewicz, J. J. and Weymark, J. A.: 1983, On the stability of collusive price leadership, *Canadian Journal of Economics* **16**(1), 17–25.

de Ridder, A. and Rusinowska, A.: 2005, Different sequences of alliance formation, *Technical report*, Working paper series on Research in Relationship Management. RRM-2005-13-STR, Radboud University Nijmegen.

de Ridder, A., Rusinowska, A., Sáiz, M. E. and Hendrix, E. M. T.: 2007, Coalition formation: the role of procedure and policy flexibility, *Submitted* .

Dellink, R. B., Altamirano-Cabrera, J. C., Finus, M., Ekko C. van Ierland, A. R. and Weikard, H. P.: 2003, Empirical background paper of the staco model, *Technical report*, mimeo, Wageningen University.

Dorta-González, P., Santos-Peñate, D. R. and Suárez-Vega, R.: 2004, Cournot oligopolistic competition in spatially separated markets: The stackelberg equilibrium, *Ann. Reg. Sci.* **38 (3)**, 499 – 511.

Downs, A.: 1957, *An economic theory of democracy*, New York: Harper and Row Publishers.

Drezner, T.: 1994, Locating a single new facility among existing unequally attractive facilities., *Journal Regional Science* **34(2)**, 237–252.

Drezner, T. and Drezner, Z.: 1982, Competitive location strategies for two facilities, *Regional Science and Urban Economics* **12**, 485–493.

Drezner, T. and Drezner, Z.: 1994, Optimal continuous location of a retail facility, facility attractiveness and market share: an interactive model, *Journal of Retailing* **70**, 49–64.

Drezner, T. and Drezner, Z.: 1997, Replacing continuous demand with discrete demand in a competitive location model, *Naval Research Logistics* **44**, 81–95.

Drezner, T. and Drezner, Z.: 1998, Facility location in anticipation of future competition, *Location Science* **6**, 155–173.

Drezner, T. and Drezner, Z.: 2004, Finding the optimal solution to the huff based competitive location model, *Computational Management Science* **1**, 193–208.

Drezner, Z. and Hamacher, H. W.: 2004, *Facility Location: Applications and Theory*, Springer Verlag.

Eiselt, H. and Laporte, G.: 1996, Sequential location problems, *European Journal of Operational Research* **96**, 217–231.

Eiselt, H., Laporte, G. and Thisse, J.-F.: 1993, Competitive location models: a framework and bibliography, *Transportation Science* **27**, 44–54.

Enelow, J. and Hinich, M.: 1984, *The spatial theory of voting. An introduction*, Cambridge: Cambridge University Press.

Eyckmans, J. and Finus, M.: 2003a, Coalition formation in a global warming game: How the design of protocols affects the success of environmental treaty-making, *Technical report*, CLIMNEG WP 56.

Eyckmans, J. and Finus, M.: 2003b, New roads to international environmental agreements: The case of global warming, *Technical report*, FEEM Working Paper No. 88.2003.

Farrell, J. and Shapiro, C.: 1990, Horizontal mergers: An equilibrium analysis, *Amer. Econom. Rev.* **80**(1), 107–26.

Fernández, J., Pelegrín, B., Plastria, F. and Tóth, B.: 2007, Solving a huff-like competitive location and design model for profit maximization in the plane, *European Journal of Operational Research* **179**, 1274–1287.

Finus, M., Altamirano-Cabrera, J. C. and van Ierland, E. C.: 2005, The effect of membership rules and voting schemes on the success of international climate agreements, *Public Choice* **125**, 97–127.

Finus, M., Sáiz, M. E. and Hendrix, E. M.: 2004b, An empirical test of new develpments in coalition theory for the design of international environmental agreements, *Technical report*, Mansholt Working Paper (Wageningen Universiteit). submitted to Environmental & Resource Economics.

Finus, M., van Ierland, E. and Dellink, R.: 2006, Stability of climate coalitions in a cartel formation game, *Economics of Governance* **7**, 271–291.

Gomory, R. E.: 1958, Outline of an algorithm for integer solutions to linear programs, *Bulletin of the American Mathematical Society* **64**, 275–278.

Grofman, B.: 1982, A dynamic model of protocoalition formation in ideological n-space, *Behavioral Science* **27(1)**, 77–90.

Hakimi, S.: 1983, On locating new facilities in a competitive environment, *European Journal of Operational Research* **12**, 29–35.

Hendrix, E. M. T.: 2007, On the goodness of global optimisation algorithms, an introduction into investigating algorithms, *in* A. Törn and J.Žilinskas (eds), *Models and Algorithms for Global Optimization*, Springer, New York, pp. 225–248.

Hillier, F. S. and Lieberman, G. J.: 1986, *Introduction to Operations Research*, McGraw-Hill Science.

Hotelling, H.: 1929, Stability on competition, *The Economic Journal* **39**, 41–57.

Ibaraki, T.: 1976, Theoretical comparisons of search strategies in branch and bound algorithms, *Int. J. Comput. Inform. Sci.* **5**, 315–344.

Karmarkar, N.: 1984, A new polynomial time algorithm for linear programming, *Combinatorica* **4**, 373–395.

Keen, P. and Scott-Morton, M.: 1978, *Decision Support Systems, an Organisational Perspective*, Addison Westley.

Klingemann, H.-D., Volkens, A., Bara, J. and Budge, I. (eds): 2006, *Mapping Policy Preferences II: Estimates for Parties, Electors and Governments in Central and Eastern Europe, European Union and OECD 1990-2003*, Oxford: Oxford University Press.

Labbé, M. and Hakimi, S. L.: 1991, Market and locational equilibrium for two competitors, *Operations Research* **39**(5).

Laver, M.: 2005, Policy and the dynamics of political competition, *American Political Science Review* **99(2)**, 263–281.

Laver, M. and Schofield, N.: 1990, *Multiparty government: The politics of coalition in Europe*, Oxford: Oxford University Press.

Laver, M. and Shepsle, K.: 1996, Coalitions and cabinet government, *American Political Science Review* **84(3)**, 873–890.

Lederer, P. and Thisse, J.: 1990, Competitive location on networks under delivered pricing, *Operations Research Letters* **9**, 140–153.

Martin, L. and Stevenson, R.: 2001, Government formation in parliamentary democracies, *American Journal of Political Science* **45(1)**, 33–50.

Martin, L. and Vanberg, G.: 2004, Policing the bargain: coalition government and parliamentary scrutiny, *American Journal of Political Science* **48(1)**, 13–27.

Mitten, L. G.: 1970, Branch and bound methods: general formulation and properties, *Operations Research* **18**, 24–34.

Morasch, K.: 2000, Strategic alliances as stackelberg cartels - concept and equilibrium alliance structure., *International Journal of Industrial Organization* **18(2)**, 257–282.

Müller, W. and Strøm, K.: 2003, *Coalition governments in Western Europe*, Oxford: Oxford University Press.

Myerson, R.: 1991, *Game Theory: Analysis of Conflict*, Harvard University Press, Cambridge.

Nash, J.: 1951, Non-cooperative games, *Annals of Mathematics* **54**, 298–95.

Nickel, S. and Puerto, J.: 2005, *Location Theory. A Unified Approach*, Springer.

Nordhaus, W. D.: 1994, *Managing the Global Commons*, Ed. MIT Press, Cambridge, MA.

Olieman, N. J. and Hendrix, E. M.: 2005, Stability likelihood of coalitions in a two-stage cartel game: An estimation method, *forthcoming in European Journal of Operational Research* .

Plastria, F.: 1992, Gbsss, the generalized big square small square method for planar single facility location, *European Journal of Operational Research* **62**, 163–74.

Plastria, F.: 1997, Profit maximising single competitive facility location in the plane, *Studies in Locational Analysis* **11**, 115–126.

Plastria, F.: 2001, Static competitive facility location: an overview of optimisation approaches, *European Journal of Operational Research* **129**, 461–470.

Plastria, F. and Carrizosa, E.: 2004, Optimal location and design of a competitive facility, *Mathematical Programming* **100**, 247–265.

Ray, D. and Vohra, R.: 1997, Equilibrium binding agreements, *Journal of Economic Theory* **73**, 30–78.

Ray, D. and Vohra, R.: 1999, A theory of endogenous coalition structures, *Games and Economic Behavior* **26**, 286–336.

Rhim, H., Ho, T. H. and Karmarkar, U. S.: 2003, Competitive location, production, and market selection, *European Journal of Operational Research* **149 (1)**, 211–228.

Riker, W.: 1962, *The theory of political coalitions*, Yale University Press.

Sarkar, J., Gupta, B. and Pal, D.: 1997, Location equilibrium for cournot oligopoly in spatially separated markets, *Journal of Regional Science* **37 (2)**, 195–212.

Schrijver, A.: 1998, *Theory of Linear and Integer Programming*, John Wiley & Sons, New York.

Selten, R.: 1975, Reexamination of the perfectness concept for equilibrium points in extensivegames, *International Journal of Game Theory* **4**, 25–55.

Shubik, M.: 1984, *Game Theory in the Social Sciences: Concepts and Solutions*, MIT Press. Cambridge, MA.

Simon, H.: 1976, *Administrative behaviour, a study of decision making processes in administrative organisations*, New York/London.

Simon, H. A., Dantzig, G. B., Hogarth, R., Piott, C. R., Raiffa, H., Schelling, T. C., Shepsle, K. A., Thaier, R., Tversky, A. and Winter, S.: 1986, *Research Briefings 1986*, National Academy Press, Washington, DC., chapter Report of the Research Briefing Panel on Decision Making and Problem Solving, pp. 17–36.

Sáiz, M. E., Hendrix, E. M. T. and Olieman, N. J.: 2006, On the computation of stability in multiple coalition formation games, *Computational Economics* **28 (3)**, 251–275.

Sáiz, M. E., Olieman, N. J., Hendrix, E. M. and Finus, M.: 2004, Determination of stable structures in a multiple coalition game, *Technical report*, Mansholt Working Paper (Wageningen Universiteit).

Stackelberg, H. V.: 1934, *Marktform und Gleichgewicht*, Vienna: Julius Springer.

Teitz, M. and Bart, P.: 1968, Heuristic methods for estimating the generalized vertex median of a weighted graph, *Operations Research* **16**, 955–961.

Thoron, S.: 2000, *Interaction and Market Structure*, Springer Verlag, chapter Market organization: Noncooperative models of coalition formation, pp. 207 – 223.

Timmermans, A.: 2003, *High politics in the low countries. An empirical study of coalition agreements in Belgium and The Netherlands*, Aldershot: Ashgate.

Tohmé, F. and Sandholm, T.: 1999, Coalition formation processes with relief revision among bounded rational self-interested agents, *Journal of Logic and Computation* **9**, 793–816.

Tuy, H., Al-Khayyal, F. and Zhou, F.: 1995, A d.c. optimization method for single facility location problems, *Journal of Global Optimization* **7**, 209–227.

van Deemen, A.: 1989, Dominant players and minimum size coalitions, *European Journal of Political Research* **17**, 313–332.

van Deemen, A.: 1997, *Coalition formation and social choice*, Boston: Kluwer Academic Publisher.

van der Brug, W.: 1999, Voters' perceptions and party dynamics, *Party Politics* **5(2)**, 147–169.

van der Vorst, J. G.: 2000, *Effective Food Supply Chains; generating, modelling and evaluating supply chain scenarios*, PhD thesis, Wageningen Universiteit.

Volden, C. and Carrubba, C.: 2004, The formation of oversized coalitions in parliamentary democracies, *American Journal of Political Science* **48(3)**, 521–537.

von Neumann, J. and Morgenstern, O.: 1944, *Theory of Games and Economic Behavior*, Princeton: Princeton University Press.

Vries, M. D.: 1999, *Governing with your closest neighbour: An assessment of spatial coalition formation theories*, Enschede: Print Partners Ipskamp.

Warwick, P.: 1998, Policy distance and parliamentary government, *Legislative Studies Quarterly* **23(3)**, 319–345.

Warwick, P.: 2000, Policy horizons in west european parliamentary systems, *European Journal of Political Research* **38**, 37–61.

Warwick, P.: 2005a, When far apart becomes too far apart: Evidence for a threshold effect in coalition formation, *British Journal of Political Science* **35**, 383–401.

Warwick, P.: 2005b, Do policy horizons structure the formation of parliamentary governments?: The evidence from an expert survey, *American Journal of Political Science* **49(2)**, 373–387.

Weikard, H., Finus, M. and Altamirano-Cabrera, J.: 2006, The impact of surplus sharing on the stability of international climate agreements, *Oxford Economic Papers* **58**, 209–232.

Yi, S.-S.: 1997, Stable coalition structures with externalities, *Games and Economic Behavior* **20**, 201–237.

Yi, S.-S. and Shin, H.: 1995, Endogenous formation of coalitions in oligopoly, *Dartmouth College Department of Economics WP No.* **95-2**, 537–556.

Summary

Decision making is present in daily life. A decision making problem appears when there is a need to take a decision. Most of the decisions have a goal; a set of variables, characteristics defining a problem, that need to be optimised. A decision problem can be formulated and modelled using Game Theory (GT) and Operations Research (OR) methods. Based on the model, solution alternatives are generated and evaluated in order to obtain a preference order of the alternatives based on the objective of the decision maker. Decision situations may also involve multiple decision makers, leading to cooperative and/or competitive decision making situations.

GT may have modelling merits to contribute to decision making situations where decision makers have competing or cooperating objectives, under the assumption that the decision makers are rational and they act in their best own interest. GT models are often applied to situations with few or symmetric multiple decision makers, which limits its modelling power. Results in existing literature are often attributable to basic models with only two actors leading to analytical solutions or alternatively to more actors that are all assumed to be the same; symmetric players. In this thesis we contribute to cooperative or competitive decision situations by building new or existing GT models and by reformulating those into OR models to study cases with more than two heterogeneous actors. To solve those decision making problems, one has to apply and develop new OR techniques by developing models and algorithms. Looking at relevant literature, decision situations like environmental decisions, supply chains decisions and political games are of interest for research.

The research objective of this thesis is to develop or contribute to the development of usable GT models and OR methods to solve decision making problems in practical situations. We modelled those problems with new or existing GT models for multiple, asymmetric and heterogeneous actors in cooperative/non-cooperative and competitive decision situations. The GT models are reformulated using OR techniques and new OR algorithms are developed for obtaining the optimal solutions for the models.

We hypothesised that GT concepts and models might provide us with a valuable contribution for supporting decision makers provided that we can make OR models and algorithms to solve these models. In order to achieve the objective of this thesis four research questions are addressed:

1. How do we formulate GT models for decision making situations with multiple heterogeneous actors using GT concepts? How can we contribute to that modelling?
2. If we have GT based models: What appropriate coding can be used to translate a specific GT model into a mathematical programming optimisation problem such that it can be solved?
3. Which solution methods can be developed to solve these models?

4. How do the outcomes aid to analyse the decision makers problem? or What is the contribution of the new outcomes to the decision makers problem?

In answering the research questions, four different cases are used. In all the cases, new or existing GT models are developed and reformulated into OR optimisation models and OR algorithms for solving these models. The starting points for each of the cases are from sources based on literature review. To illustrate cooperative/non-cooperative GT concepts/models we looked for models with a challenge on algorithmic development. Two types of coalition formation are considered: multiple coalition formation and single (cartel) coalition formation (Chapter 2 and Chapter 3). To illustrate competition models, we looked for cases involving competition on location and competition on location and quantity (Chapter 4 and Chapter 5). The objective is to contribute to these models extending the research either on the GT models and reformulation to OR models or the OR methods.

Chapter 2 describes a multiple coalition formation game in which membership rules and different transfer schemes are described. *Open membership rule* is used to indicate that for the current members of a coalition, any other player is allowed to enter it. *Exclusive membership rule* implies that a non-member is only allowed to join the existing coalition if the payoff (outcome) for the existing coalition members will not decrease, only in that case the members of the existing coalition will allow the new member to join them. If not, they will exclude her. The study also considers two types of voting rules: *Majority Voting*, where the majority of the members of the coalition have the decision on yes/no to the new coalition potential partner and *Unanimity Voting*, when one coalition member can disapprove when a new member wants to join. Moreover, four different transfer schemes are used, namely: "Equal Sharing", "Population", "Gross Domestic Product" and "Ability to Pay". Given the GT model and the OR model, the goal is to develop methods for checking stability of coalition structures. For the check of stability we need all the neighbours of each coalition structure. Neighbours in a coalition structure for each player are determined by the single deviation from players to other coalitions or players deviating and being alone, that means becoming a singleton player. The exact number of neighbours is not easy to identify. The larger the number of players is, the larger the number of coalition structures and as a consequence, the bigger the matrix with all necessary data to test stable coalition structures. A new mathematical programming formulation, crucial for the development of the algorithms, is elaborated. Available data is used to determine which stable coalitions appear and which procedures can be used to make coalitions stable. A case study on the Kyoto negotiation process is applied to check the algorithms. Main conclusions are that transfer schemes are useful to be implemented and computation with membership rules generate different results with and without transfer schemes. In the cases without transfers, the algorithm only found stability when an exclusive membership game with unanimity voting rule is applied. This result is different when transfers are applied; either majority and unanimity voting are sufficient to stabilise coalition structures. With respect to the number of stable coalition structures, there are some transfer schemes which do not generate stable coalition structures. Transfer schemes are anyway interesting in terms of net benefit and global emission reduction. Furthermore, some of the regions considered that are "not interested" in cooperation without transfer schemes, became interested in forming a coalition when transfers appear. Furthermore, the resulting model was used in further studies (Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005)).

Chapter 3 studies a model of multidimensional coalition formation in politics. There are n parties trying to form a government. A formed government has a policy agreement represented in a m-multidimensional Euclidean policy space $\mathbb{R}^m$. The complexity increases with the number of parties n and the dimension of the policy spaces m. Given the number

of parties n and policy dimension m, computational methods are necessary to compute all possible winning coalitions and preferences of parties over those coalitions. Furthermore, two ways of forming a government are considered: step-by-step and simultaneously. Procedures differ from each other, but they have in common that no party will agree on a position which does not belong to its maneuvering space as these positions are unacceptable for a party. In other words, the position of the formed coalition must belong to the intersection of the maneuvering spaces of the potential coalition. Calculations show that the number of different paths and coalition positions in the step-by-step procedure can increase dramatically. For computing the output of the model, computational methods are necessary and algorithms for the different procedures are introduced. The algorithms determine the compromise (coalition position) points of all winning coalitions as well as preference order based on distances to ideal points. The developed algorithms use an external procedure for the determination of the output based on sequential quadratic programming (SQP) methods. Moreover, a penalty approach is developed to determine whether or not feasible negotiation spaces intersect between the negotiating parties (the players of the game). The analysis in de Ridder et al. (2007) focused on three aspects of coalition formation and formulated hypotheses: procedure, flexibility, and power sharing. By applying the developed algorithms to Dutch data and theoretical examples, several hypotheses are tested and implications for political parties involved in coalition formation are given.

Chapter 4 describes a competitive two-stage location-quantity game with $m > 2$ markets and $n > 2$ firms. In the first stage of the game location decisions by firms are taken; in the second stage, firms decide about quantities to supply at each of the markets. At each step, decisions are simultaneous. The location space is discrete, in particular is a network where the nodes are considered as the possible locations. Free entry is possible, the number of firms entering the markets is not known in advance; and costs are asymmetric, firm-specific. Analytical solutions for the supplying decisions and properties for determining the size of the market are derived. The computation of the optimum quantities to supply (equilibrium) is possible when market sizes are determined. In finding the equilibria on location, a complete enumeration algorithm and a local search algorithm are used. The complete enumeration algorithm guarantees that all the equilibria of the game are found. The local search algorithm aims at finding one equilibrium given a starting point. Two cases are elaborated to illustrate the procedures and the analytical results. The first case is taken from literature with $n = 3$ firms and $m = 6$ markets. The second numerical example consists of 4 different cases. It is used to show the viability of the algorithms when bigger and more sophisticated cases are applied. Furthermore, by using the algorithms as a systematic computation instrument to cases reported in literature, a mistake is detected in Sarkar et al. (1997). In that paper a solution is given that appears not to be an equilibrium of the model.

Chapter 5 deals with a competitive facility location problem in which the concept of Stackelberg leader-follower problem is applied. There is a leader chain who wants to locate a facility and a follower chain taking the location decision after the leader. The market share captured by a firm is given by a gravity model determined by distance calculations to facilities. The follower problem, is known to be a hard global optimisation problem. The leader problem is even harder, since the leader has to decide on location given the optimal action of the follower. So far, in literature only heuristic approaches have been tested to solve the leader problem. Branch-and-Bound (B&B) algorithms for the leader and the follower are designed. The B&B algorithm for the leader has to solve two follower problems: one for solving (and anticipating) the optimal location of the follower (already a GO problem) and the second one for obtaining an upper bound for her own B&B algorithm (GO problem). For the B&B algorithm for the follower problem, four different upper bounds are derived. The algorithm evaluates and chooses the best of the four upper bounds. The branching rule uses

rectangles and new rectangles are generated by bisecting a rectangle over its longest edge. Moreover, two variants are implemented and compared: starting with an initial rectangle containing the feasible area, and starting with an initial partition into rectangles. It is outlined that, although generating an initial partition improves the upper bounding applied, it generates more partition sets than strictly necessary. Furthermore, two selection rules are compared: breadth-first-search and best-bound-search. The effect on efficiency of those rules is measured. The effectiveness and efficiency of the algorithms are investigated with the aid of numerical cases. As main conclusion, it is found that more sophisticated bounds are not necessarily more effective than simple bounds based on distance comparison over the complete run of the algorithm. One can best focus on measuring the quality of the bound during the run and take the sharpest one. For the selection rule, the focus on the best bound (most promising) selection of the next subset to be split has the tendency to result in minimum effort on number of function evaluations. As memory requirement is usually a problem for higher dimensions, it is not necessarily a focus point for the location problem in two dimensional space.

By answering the research questions, the main conclusions of this study are outlined. Within the research of this study, GT has been a useful tool for modelling decision making situations when multiple, asymmetric and heterogeneous decision makers are involved. Contribution to GT models is most significant in Chapter 2 and Chapter 4 by adding additional characteristics to cases taken from literature. An important factor in this thesis is what is called "coding into mathematical programming notation"; it has a relevant contribution on the development of algorithms to solve the decision making problems. From the cases studied, different challenges inspired the development of the algorithms: application to huge cases, asymmetry, heterogeneity and multiple actors. Algorithms are developed for each of the problems covered in this research; algorithms are based on enumeration methods, iterative approaches, analytical solutions, SQP methods, penalty approach and GO methods.

In brief, the main contributions of this thesis can be summarised as follows. In this thesis we have studied GT models and reformulated the models into OR optimisation problems. The reformulation into OR optimisation problems is a worthwhile step for the development of algorithms. Furthermore, in this research, both real-life data and randomly generated data are used. For all the cases applied, algorithms are designed following the conceptual model introduced in Chapter 1. The idea is that outcomes of the developed algorithms aim to help decision makers and analysts to get insight into a decision situation. Two of the cases contributed on aiding researches when real data are used (Chapters 2 and 3). Models and methods in Chapters 4 and 5 are illustrated with randomly generated data.

This study should be seen as a first step in making GT more applicable to larger, more complex decision problems. The cases applied have specific characteristics which guided us in the selection of specific GT models, OR models and algorithms. Within decision making problems, we have focused on rational behaviour of decision makers. Further research could be done by considering bounded rationality in decision making problems. With respect to GT, incomplete information games have not been studied. It is of interest to do more research on cases with large number of players on incomplete information games and how to contribute on the development of GT models, on the coding of these GT models into OR models and on the development of new OR methods/algorithms. When considering the OR models and algorithms, the models and algorithms developed in this study are focused on the characteristic of the cases. Further research includes the study of the performance of the developed algorithms in comparison to other algorithms from OR. In summary, there exist new challenges on developing Operations Research models and algorithms to get Game Theory at work.

Samenvatting

Het nemen van beslissingen is een dagelijkse bezigheid. Vaak is er sprake van een te optimaliseren doelstelling en diverse variabelen die het probleem karakteriseren. Een beslissingsprobleem kan worden geformuleerd en gemodelleerd met behulp van speltheorie en Operations Research methoden. Beslissingsalternatieven worden gegenereerd en geëvalueerd, zodat een preferentievolgorde van de alternatieven ontstaat, gebaseerd op de doelstelling van de besluitvormer. Beslissingsituaties kunnen ook meerdere beslissers bevatten die samenwerking of competitie nastreven.

Speltheorie heeft de potentie om bij te dragen aan beslissingssituaties waarbij de belanghebbenden samenvallende of conflicterende doelstellingen hebben, zolang wordt aangenomen dat de besluitvormers rationeel handelen voor hun eigen bestwil. Gewoonlijk wordt speltheorie toegepast om situaties te beschrijven met een beperkt aantal identieke besluitvormers. Dit beperkt de kracht van de analyse. In de bestaande literatuur zien we vaak basale modellen met slechts twee actoren, zodat analytische oplossingen kunnen worde afgeleid. Als alternatief zien we situaties met meerder identieke actoren beschreven. Dit proefschrift draagt bij aan de modellering van samenwerkende en conflicterende beslissingsituaties door het bouwen van bestaande of nieuwe speltheoretische modellen en door het omzetten daarvan naar OR modellen om cases met meer dan twee heterogene actoren te bestuderen. Om deze problemen op te lossen worden nieuwe OR technieken ontwikkeld door het uitwerken van modellen en algoritmen. Gebaseerd op de literatuur, zijn beslissingsituaties met milieubeslissingen, met betrekking to "supply chainsën in politieke situaties interessant.

De doelstelling van dit proefschrift is om bij te dragen aan de ontwikkeling van bruikbare speltheoretische modellen en OR methoden om praktische beslissingssituaties op te lossen. Daartoe werden nieuwe en bestaande speltheoretische modellen ontwikkeld voor het beschrijven van beslissingssituaties met asymmetrische, heterogene actoren die wel of niet samenwerken en conflicterende doelstellingen hebben. De modellen zijn omgezet met behulp van OR methoden en nieuwe algoritmen zijn ontwikkeld om optimale resultaten voor de modellen af te leiden. De hypothese is dat speltheoretische concepten en modellen zouden kunnen bijdragen aan het ondersteunen van besluitvormers indien we in staat zijn om OR modellen en algoritmen te maken die bijdragen aan de oplossing ervan. Om de doelstelling te bewerkstelligen worden vier onderzoeksvragen geformuleerd en uitgewerkt:

1. Hoe kunnen we speltheoretische modellen ontwikkelen voor beslissingsituaties met meerdere heterogene actoren vanuit speltheoretische concepten? Hoe kunnen we bijdragen aan de modellering?
2. Als we eenmaal modellen hebben gebaseerd op speltheorie, welke codering kan worden gebruikt om een speltheoretisch model om te zetten naar een wiskundig optimaliser-

ingprobleem zodat het kan worden opgelost?

3. Welke technieken kunnen worden toegepast om dit soort problemen op te lossen?
4. Hoe dragen de uitkomsten bij aan het analyseren van het beslissingprobleem? Wat is de bijdrage van de nieuwe uitkomsten aan het begrijpen van het beslissingprobleem?

Voor het beantwoorden van de onderzoeksvragen worden vier cases uitgewerkt. In elke case wordt een nieuw of bestaand speltheoretisch model uitgewerkt en omgezet naar een OR optimaliseringprobleem. Algoritmen worden ontwikkeld om het probleem op te lossen. Elke case is gebaseerd op een literatuuronderzoek. Om coöperatieve versus non-coöperatieve speltheorie te illustreren zochten we naar modellen met een uitdaging op algoritmisch gebied. Twee typen coalitievorming zijn onderzocht: vorming van meervoudige coalities en vorming van een enkelvoudige (kartel) coalitie (hoofdstukken 2 en 3). Voor de illustratie van competitieve modellen richtte de aandacht zich op concurrentie tussen aanbieders op het gebied van locatie en concurrentie met betrekking tot aangeboden hoeveelheid en locatie (hoofdstukken 4 en 5). De doelstelling is om bij te dragen aan het onderzoek door de ontwikkeling van speltheoretische modellen, de vertaling richting OR modellen en het bijdragen aan de oplossingmethode.

Hoofdstuk 2 beschrijft een spel van coalitievorming met verschillende lidmaatschapregels en verschillende herverdelingregels. Een öpen lidmaatschap"betekent dat de huidige leden het goed vinden dat een andere actor lid wordt. Een ëxclusief lidmaatschap"regel betekent dat een aspirant lid alleen lid kan worden van de coalitie als het voordeel (baten) voor de huidige leden er niet op achteruit gaat. Binnen dit kader bekijkt de studie twee stemprocedures: meerderheid stemming, waarbij de meerderheid van de huidige leden beslist om een nieuw lid wel of niet toe te laten en unanimiteit, waarbij één lid voldoende is om lidmaatschap van een aspirant lid te blokkeren door middel van een veto. Verder worden vier verschillende schema's gebruikt om de baten te (her)verdelen, namelijk: "gelijk delen". "bevolking", "GDPën "koopkracht". Gegeven het speltheoretisch model en de vertaling daarvan naar een OR probleem, is de uitdaging om methoden te ontwikkelen die de stabiliteit van een coalitie testen. Voor deze test is het nodig om van een coalitiestructuur alle aanpalende of buur-coalitiestructuren te kennen. Zo'n buur wordt bepaald doordat een van de spelers haar strategie verandert; of een lid van een coalitie gaat naar een andere coalitie of gaat alleen verder; of een speler die nog geen lid is wordt lid van een coalitie. Het precieze aantal buren is niet eenvoudig te bepalen. Hoe meer spelers, hoe groter het aantal coalitiestructuren en hoe groter de matrix die alle nodige informatie bevat om stabiliteit te testen. Een nieuwe mathematisch besliskundige formulering is uitgewerkt voor het ontwikkelen van algoritmen. Beschikbare gegevens zijn gebruikt om na te gaan welke coalitiestructuren stabiel zijn en welke procedures de stabiliteit kunnen bevorderen. Een case over het Kyoto onderhandelingproces is gebruikt om algoritmen te testen. Belangrijkste conclusies zijn dat het nuttig kan zijn om herverdelingschema's te implementeren en dat berekeningen met verschillende lidmaatschapregels verschillende resultaten geven voor verschillende herverdelingschema's. In gevallen zonder herverdeling, vond het algoritme alleen stabiele coalities als een exclusief lidmaatschapregel bestaat met veto stemming. Als herverdeling wordt geïntroduceerd kunnen coalities ook stabiel worden bij meerderheid stemming. Als wordt gekeken naar het aantal stabiele coalities kan worden opgemerkt dat er herverdelingschema's zijn die geen enkele stabiele coalitie geven. Herverdelingschema's zijn altijd van belang als we kijken naar de totale baten en globale emissiereductie. Bovendien blijken sommige regio's die zonder herverdeling niet geïnteresseerd zijn in samenwerking wel geïnteresseerd om tot een coalitie toe te treden wanneer er wel herverdelingschema's bestaan. De ontwikkelde methodiek is toegepast in verdere studies, zoals (Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005).

Hoofdstuk 3 beschrijft een model over meerdimensionale coalitievorming in de politiek. Er zijn n partijen die proberen te komen tot een coalitieregering. De gevormde regering heeft een politieke agenda (overeenkomst) die gerepresenteerd wordt door een punt in een m-meerdimensionale Euclidische politieke ruimte. De complexiteit van de analyse neemt toe met het aantal partijen n en de dimensie m van de politieke ruimte. Rekenmethoden zijn nodig om alle mogelijke meerderheidcoalities te bepalen en de preferenties van de partijen over de coalities. Twee mogelijke procedures voor het vormen van een coalitie worden bestudeerd: een stap-voor-stap procedure en een simultane procedure. Onafhankelijk van de procedure zal een partij niet tot een overeenkomst komen als die niet ligt binnen haar spelingruimte. Met andere woorden, een overeenkomst van een gevormde coalitie moet liggen in de doorsnede van de spelingruimten van de partijen die behoren tot de coalitie. Berekeningen hebben laten zien dat het aantal mogelijke paden om tot coalities te komen en de bijbehorende overeenkomsten drastisch kan toenemen met het aantal partijen. Om de uitkomst van het model te bepalen zijn rekenmethoden nodig; algoritmen voor de verschillende procedures zijn uitgewerkt. De algoritmen bepalen de compromis overeenkomsten van alle meerderheidcoalities zowel als de preferentie van de verschillende partijen gebaseerd op de afstand van de overeenkomst tot hun ideale politieke agenda. De ontwikkelde algoritmen gebruiken een externe routine gebaseerd op sequentiële kwadratische programmering (SQP). Verder is een boetefunctie aanpak uitgewerkt om te bepalen of de onderhandelende partijen wel onderhandelingruimte hebben bestaande uit de doorsnede van de spelingruimten. De analyse in de Ridder et al. (2007) concentreert zich op drie aspecten t.a.v. coalitievorming en test daar hypothesen over: de procedure, de flexibiliteit en het delen van macht. Door de ontwikkelde algoritmen toe te passen op gegevens uit de Nederlandse politiek en op theoretische voorbeelden zijn de hypothesen getest en consequenties voor partijen betrokken bij een onderhandeling worden gegeven.

Hoofdstuk 4 beschrijft een competitief spel bestaande uit twee fasen besluitvorming: locatie beslissing en hoeveelheid beslissing, waarin n>2 bedrijven concurreren op m>2 markten. In de eerste fase van het spel nemen de bedrijven een beslissing t.a.v vestiging op een locatie. In de tweede fase besluiten ze de hoeveelheid die ze aanbieden op de markten. In beide fasen worden de beslissingen simultaan genomen. De vestigingruimte is discreet, waarbij de potentiële locaties bestaan uit knopen in een netwerk. De bedrijven zijn vrij om een markt te bevoorraden, zodat men niet van te voren weet hoeveel bedrijven een markt zullen benaderen. De kosten zijn bedrijfafhankelijk, dus asymmetrisch. Voor het vinden van een evenwicht t.a.v. de vestigingbeslissing zijn een aftelalgoritme en een lokale zoekprocedure ontwikkeld. Het aftelalgoritme vindt alle mogelijke evenwichten. De lokale zoekprocedure vindt een evenwicht vanuit een startconfiguratie. Twee cases zijn uitgewerkt om de procedures en analytische resultaten te illustreren. De eerste case uit de literatuur beschrijft een instantie met n = 3 bedrijven en m = 6 markten. Het tweede numerieke voorbeeld bevat 4 instanties met toenemende grootte. Het laat zien dat problemen van toenemende grootte kunnen worden aangepakt met de ontwikkelde procedures. Gebruik van de algoritmen op testcases uit de literatuur leidde tot de ontdekking van een fout in Sarkar et al. (1997). Een gepresenteerde oplossing van een model in het artikel blijkt geen evenwicht te zijn.

Hoofdstuk 5 behandelt een competitief vestigingprobleem waarin het Stackelberg concept van de leider-volger wordt toegepast. Een leider winkelketen vestigt eerst een filiaal en een volgketen opent vervolgens een vestiging. Het marktaandeel dat een keten verovert wordt beschreven door een zwaartepunt model gebaseerd op de afstanden tussen vestigingen en klanten. Het is bekend uit de literatuur dat het vinden van de beste locatie voor de volgketen een moeilijk globaal optimaliseringprobleem (GO) is. Het probleem van de leider is echter nog moeilijker, daar die rekening moet houden met de optimale locatie van de volger. Voor zover we weten werd het leiderprobleem in de literatuur alleen met heuristische methoden

aangepakt. In dit proefschrift worden Branch-and-Bound (B&B) algoritmen voor het leider-en volgprobleem ontworpen. Het algoritme voor de leider vereist in elke iteratie dat twee volgproblemen worden opgelost: één voor het bepalen van de optimale locatie voor de volger en één voor het bepalen van een bovengrens voor de leider, beide GO problemen. Voor het B&B algoritme voor het volgprobleem zijn vier verschillende bovengrenzen (bounds) ontworpen. Het algoritme berekent de waarden ervan in elke iteratie en gebruikt de strakste grens. Het vertakken (branching) naar subproblemen werkt met rechthoeken die gesplitst worden over de langste zijde. Twee strategieën zijn uitgeprobeerd. Eén waarbij de rechthoek van het toegelaten gebied als startgebied wordt genomen en één waarbij deze rechthoek al is gepartitioneerd naar sub-rechthoeken. De studie laat zien dat hoewel de initiële partitie de waarden van de grenzen verbetert, het algoritme niet in efficiëntie wint, omdat teveel sub-rechthoeken worden geprobeerd. Twee keuzeregels zijn vergeleken voor de keuze van de rechthoek die wordt bekeken: zoeken in de breedte en prioriteit voor de beste grens. De efficiëntie is gemeten voor beide regels. Effectiviteit en efficiency is nagegaan met behulp van numerieke instanties. Een belangrijke conclusie is dat verfijnde grenzen niet noodzakelijk betere resultaten geven dan een eenvoudige grens die gebaseerd is op het onderschatten van afstand tussen klant en filiaal. Het beste is om de scherpste grens te nemen gedurende het verloop van het algoritme. Voor de keuzeregel van de volgende rechthoek die wordt gesplitst heeft focus op de beste grens de neiging om het minst aantal iteraties te vereisen. Gewoonlijk is voor dit soort algoritmen het geheugenbeslag een probleem wanneer toegepast in hogere dimensies. Voor het locatieprobleem is dit minder het geval omdat het zich in de tweedimensionale ruimte afspeelt.

Voor de belangrijkste conclusies richten we ons op de onderzoeksvragen. De studie laat zien dat speltheorie een handig hulpmiddel kan zijn voor het modelleren van beslissingsituaties met meerdere asymmetrische en heterogene belanghebbenden. De bijdrage aan speltheoretische modellen is belangrijk in hoofdstukken 2 en 4, waarbij nieuwe aspecten aan bestaande modellen zijn toegevoegd. Een belangrijke factor in dit proefschrift is wat genoemd wordt de "codering van modellen in mathematische programmering taalï.e. het vertalen naar wiskundige optimaliseringproblemen. Dit is met name relevant voor het ontwikkelen van algoritmen om beslissingproblemen op te lossen. De bestudeerde cases bevatten diverse interessante aspecten voor de ontwikkeling van algoritmen: toepassing van grote instanties, asymmetrie, heterogeniteit en meerdere actoren. Algoritmen zijn ontwikkeld voor elk probleem dat in deze studie is aangepakt. De algoritmen zijn gebaseerd op volledige aftelling, iteratieve methoden, analytische oplossingen, SQP methoden, boetefunctie aanpakken en globale optimalisering.

De belangrijkste bijdragen van dit proefschrift kunnen als volgt worden samengevat. In dit proefschrift zijn diverse speltheoretische modellen bestudeerd en vertaald naar OR optimaliseringmodellen. Het herformuleren van modellen naar optimaliseringproblemen is een nuttige stap voor het ontwikkelen van algoritmen. Gegevens in dit onderzoek komen uit de praktijk of zijn random gegenereerd. Voor elke casus zijn algoritmen ontwikkeld volgens het conceptuele model in hoofdstuk 1. Het doel van de uitkomsten van de algoritmen is dat ze de besluitvormers helpen om inzicht te krijgen in een beslissingsituatie. Twee cases hielpen onderzoekers voor het uitwerken van gegevens uit de praktijk (hoofdstukken 2 en 3). De modellen en methoden in hoofdstukken 4 en 5 worden met name geïllustreerd door random gegenereerde gegevens.

Deze studie kan gezien worden als een eerste stap om speltheorie toepasbaar te maken voor grotere en complexere beslissingproblemen. De bestudeerde cases gaven aanleiding tot selectie van specifieke speltheoretische modellen, OR modellen en algoritmen. Binnen de theorie van de beslissingproblemen hebben we ons gericht op het rationeel handelen van actoren. Verder onderzoek kan worden gedaan op het gebied van begrensde rationaliteit

in beslissingsituaties. Met betrekking tot speltheorie zijn modellen met onvolledige informatie niet bestudeerd. Het kan interessant zijn om onderzoek te doen naar gevallen met een groot aantal spelers (actoren) die geen volledige informatie hebben en te bestuderen hoe dit soort modellen gecodeerd kunnen worden in OR modellen die nieuwe typen OR algoritmen/methoden vereisen. De modellen en methoden die ontwikkeld zijn in dit proefschrift zijn toegespitst op de karakteristieken van de bestudeerde cases. Verder onderzoek bevat de bestudering van de prestatie van de ontwikkelde algoritmen in vergelijking tot andere algoritmen uit de OR. Samengevat bestaan er verdere uitdagingen in de ontwikkeling van Operations Research modellen en methoden om speltheorie äan het werk"te krijgen.

Resumen

El proceso de la toma de decisiones (teoría de la decisión) está presente en la vida diaria. Inherente a la toma de decisiones es la necesidad de elegir entre diferentes alternativas. La mayoría de las decisiones tienen una meta u objetivo, y se toman en base a un conjunto de variables que definen un problema y que necesitan ser optimizadas. Los problemas de toma de decisiones pueden ser formulados y modelados usando la Teoría de Juegos (TJ) y métodos de Investigación Operativa (IO). En base al modelo, se generan soluciones alternativas que son evaluadas para obtener un orden de preferencia dependiendo del objetivo del agente responsable de tomar la decisión (agente o jugador más adelante). La toma de decisiones puede involucrar a más de un agente, lo cuál genera problemas cooperativos y/o competitivos.

Por lo tanto, la TJ es una herramienta muy valiosa a la hora de modelar problemas de decisiones en los cuales los objetivos de los jugadores son de tipo cooperativo o competitivo, y donde los jugadores actúan de forma racional y en su propio interés. En ocasiones la TJ es usada para modelar juegos en los que el número de jugadores es reducido o los juegos son simétricos, lo que limita su potencial de modelado. En la literatura existente se pueden encontrar resultados en modelos básicos con sólo 2 jugadores en los que se obtienen soluciones analíticas, y/o modelos con más jugadores asumiendo simetría. Esta tesis contribuye a la TJ mediante la construcción de nuevos modelos o el uso de los ya existentes con más de dos jugadores heterogéneos y mediante la construcción o reformulación en modelos de IO. Para obtener una solución a dichos modelos es necesaria la aplicación y el desarrollo de nuevas técnicas de IO mediante la construcción de modelos y algoritmos. Con base a una revisión de la literatura existente, se pueden encontrar diferentes e interesantes áreas de investigación como: medio-ambiente, cadenas de suministro y juegos políticos (formación de coaliciones, gobiernos).

Por lo tanto, El objetivo de esta tesis es desarrollar o contribuir al desarrollo de modelos de TJ y métodos de IO para resolver casos aplicados dentro de la teoría de la decisión. Por ello, se han modelado diferentes problemas con nuevos o existentes modelos de TJ con múltiples jugadores, asimétricos y heterogéneos en juegos cooperativos/no-cooperativos y competitivos. Por otro lado, se reformulan o traducen dichos modelos de TJ usando técnicas de IO, y a su vez se desarrollan nuevos algoritmos de IO para poder obtener las soluciones óptimas a los modelos.

Se parte de la hipótesis de que los conceptos y modelos de la TJ tienen un gran potencial a la hora de contribuir a la teoría de decisión una vez que es posible desarrollar modelos y algoritmos de IO. Para alcanzar el objetivo de esta Tesis, se han planteado cuatro preguntas de investigación:

1. Cómo se pueden formular modelos de TJ para problemas de toma de decisiones con múltiples jugadores heterogéneos usando conceptos de TJ? Cómo se puede contribuir a esos modelos?

2. Si esos modelos de TJ están disponibles: Qué método de codificación se puede usar para traducir/trasladar esos modelos de TJ a problemas de optimización de programación matemática de manera que se pueda obtener una o varias soluciones?
3. Qué métodos de solución se pueden construir para solucionar esos problemas?
4. De qué manera las soluciones pueden ayudar al análisis de esos problemas? o Cómo contribuyen las soluciones al análisis de los problemas de toma de decisiones?

Para responder a estas preguntas, se utilizan cuatro casos diferentes. En todos los casos, se han desarrollado nuevos o existentes modelos de TJ, dichos modelos son traducidos/formulados en modelos de optimización de IO y se desarrollan nuevos algoritmos para obtener las soluciones. El punto de arranque para cada uno de los casos está basado en una revisión de la literatura existente. Los casos/capítulos se han elegido para plantear un reto al desarrollo de nuevos algoritmos en los que se usan conceptos/modelos de la TJ cooperativa/no-cooperativa. Por una parte, se consideran dos casos diferentes de formación de coaliciones: formación de coaliciones múltiple y simple (cartel) (Capítulos 2 y 3, respectivamente). Por otra parte, también se consideran dos modelos de localización competitiva (Capítulos 4 y 5), y uno de ellos además considera competición en cantidad a ofrecer al mercado (Capítulo 4). El objetivo es contribuir a esos modelos extendiendo la investigación ya sea en los modelos de la TJ y traduciéndolos en modelos de IO o en los métodos de IO.

El capítulo 2 describe un juego cooperativo "multi-actor" en el que se permite la coexistencia de varias coaliciones y se describen diferentes reglas de admisión y transferencias de utilidad. Se usa el término *Open membership* para indicar que los miembros de una coalición permiten que cualquier otro jugador no miembro decida unirse a ella. El término *Exclusive membership* implica que sólo se permite a un jugador que no es parte de la coalición unirse a ella y sólo si la ganancia (beneficio) no disminuye para los actuales miembros de la coalición. En este capítulo también se consideran dos diferentes reglas de votación: *Majority voting rule* (regla de votación por mayoría), en cuyo caso es necesario el voto de la mayoría de los socios para tomar una decisión "si/no" a la entrada de un nuevo miembro; y *Unanimity voting rule* (regla de votación por unanimidad) en la que un sólo socio de la coalición puede vetar la entrada de un nuevo miembro. Además, se consideran cuatro esquemas diferentes de transferencia de utilidad: "Equal Sharing" ("división equitativa"), "Population" ("Población"), "Gross Domestic Product" ("Producto Interior Bruto") y "Ability to Pay" ("Disposición a Pagar"). Basado en el modelo de TJ e IO, el objetivo es desarrollar métodos para poder obtener que estructuras de coalición son estables. Para poder comprobar si una estructura de coalición es estable o no es necesario compararla con, lo que se denomina, sus "vecinos". Los "vecinos" de una estructura de coalición están representados por las posibles desviaciones de la estructura de coalición por parte de cada uno de los jugadores a otras coaliciones. También se considera como "vecino" el caso en el que el jugador decide ser independiente y formar su propia coalición. Es decir, vecinos son aquellas estructuras de coalición definidas por las posibilidades que tiene cada jugador de "mudarse" de coalición. El número exacto de "vecinos" no es fácil de identificar. Cuanto mayor sea el número de jugadores, mayor es el número de posibles estructuras de coalición y como consecuencia mayor es la matriz con todos los datos necesarios para obtener y comprobar qué coaliciones son estables. Un paso importante en este estudio fue el desarrollo de una nueva formulación del modelo de programación matemática. Por otra parte, se usa la disponibilidad de datos numéricos para determinar que coaliciones son estables y qué procedimientos pueden ser usados para estabilizar coaliciones. Para comprobar el funcionamiento de los algoritmos desarrollados se usa como caso de estudio el protocolo de Kyoto. Las principales conclusiones extraídas de este trabajo son: que el uso de esquemas de transferencia de utilidad tiene un gran potencial para estabilizar coaliciones; que la aplicación de diferentes reglas de admisión genera diferentes resultados, se tengan en

cuenta o no transferencias de utilidad. En los casos en los que no existen transferencias, el algoritmo únicamente detecta estabilidad si el juego usa como reglas la "admisión con excepciones" y la "votación por unanimidad". Este resultado es diferente cuando se usan transferencias de utilidad; en este caso, tanto la "votación por mayoría" como la "votación por unanimidad" son suficientes para encontrar coaliciones estables. Con respecto al número de estructuras de coalición estables, existen algunos esquemas de transferencia de utilidad que no estabilizan o generan coaliciones estables. En cualquier caso, se puede considerar que la aplicación de esquemas de transferencia de utilidad son de gran utilidad si se considera el resultado obtenido en términos de beneficio y reducción de emisiones. Además, se puede ver como algunas de las "regiones" (jugadores) no interesados en cooperar cambian de opinión en el momento que las transferencias forman parte del juego. El modelo desarrollado en este estudio sirve como base para otros artículos dentro del mismo proyecto (véase Finus et al. (2006), Altamirano-Cabrera and Finus (2006), Finus et al. (2005), Weikard et al. (2006) and Altamirano-Cabrera et al. (2005)).

En el capítulo 3 se estudia un modelo de formación de coaliciones en política. Se consideran n partidos políticos en la mesa de negociaciones para intentar formar gobierno. En el momento que se forma un gobierno, se considera que se ha llegado a un acuerdo político que es representado en un espacio Euclídeo m-multidimensional $\mathbb{R}^m$. La complejidad es mayor cuanto mayor es el número n de partidos políticos y mayor es la dimensión m de los espacios euclídeos que contienen el área de negociación de cada partido. Dado el número de partidos n y la dimensión política m, se necesitan métodos computacionales para calcular todas las posibles coaliciones ganadoras y preferencias de los partidos sobre esas coaliciones. Aparte de eso, en el capítulo se consideran dos formas posibles a la hora de negociar para formar gobierno: "paso-a-paso" (secuencial) y "simultáneamente". Estos dos procedimientos difieren uno del otro pero tienen en común que ningún partido llegará a un acuerdo político que no pertenece a su área de negociación ya que dicho acuerdo es inaceptable para el partido. En otras palabras, el acuerdo político de un posible gobierno (posición en el espacio euclídeo) debe pertenecer a la intersección de las áreas de negociación de los partidos pertenecientes a dicha coalición potencial. Los cálculos demuestran que en el procedimiento secuencial el número de posibles rutas para alcanzar una coalición ganadora y por tanto, el número de posibles gobiernos, puede incrementar de forma drástica. Para calcular la solución del modelo son necesarios métodos computacionales y, por lo tanto, el desarrollo de nuevos algoritmos para cada uno de los procedimientos. Los algoritmos determinan el compromiso (posición del acuerdo político) de todas las posibles coaliciones ganadoras y el orden de preferencia de los partidos políticos basado en las distancias del acuerdo alcanzado a las posiciones ideales de los partidos. Los algoritmos elaborados usan un procedimiento externo para la determinación de la solución basado en métodos de programación cuadrática secuencial (SQP). Además, se desarrolla una técnica "penalti" para determinar si existe intersección entre los espacios factibles de negociación (áreas de negociación) de los partidos que negocian para formar un posible gobierno. El análisis realizado en de Ridder et al. (2007) está enfocado en tres aspectos de formación de coaliciones y se formulan varias hipótesis: sobre procedimiento, flexibilidad y poder. En este estudio, se comprueban las hipótesis mediante la aplicación de los algoritmos introducidos a datos reales de partidos políticos en Holanda, y mediante la aplicación a ejemplos teóricos; además se extraen implicaciones para los partidos políticos.

En el capítulos 4 se describe un juego de dos etapas competitivo con $m > 2$ mercados y $n > 2$ empresas. En la primera etapa del juego, las empresas toman decisiones sobre localización; en la segunda etapa, las empresas deciden las cantidades a suministrar en cada mercado. En cada etapa, las decisiones se toman simultáneamente. El espacio de localización es discreto, en particular se trata de una red en la que se considera que los nodos de la red son las posibles localizaciones de las empresas (siendo también la localización de los mercados).

Se considera libertad de entrada, es decir, el número de empresas que entran al mercado no es conocido de antemano; también se considera que los costes son asimétricos, específicos para cada empresa. En este estudio se derivan soluciones analíticas para las cantidades a suministrar y propiedades para determinar el tamaño de los mercados. El cálculo de las cantidades óptimas a suministrar (el equilibrio) es posible una vez que se determina el tamaño del mercado (definido por el número de empresas que entran). Para encontrar el equilibrio en localización, se desarrollan dos métodos diferentes: por un lado se desarrolla un algoritmo de enumeración completa; y, por otro lado, un procedimiento de búsqueda local. El algoritmo de enumeración completa asegura que se encuentran todos los posibles equilibrios del juego. El procedimiento de búsqueda local se centra en encontrar un equilibrio dado un punto inicial. Con el fin de comprobar como funcionan los algoritmos se han elaborado dos casos. El primer caso se ha tomado de otro estudio preliminar en literatura con $n = 3$ empresas y $m = 6$ mercados. El segundo estudio numérico consiste de 4 casos diferentes. Este segundo estudio es usado para mostrar la viabilidad de los algoritmos cuando se consideran más sofisticadas aplicaciones. Además, mediante el uso de los algoritmos como instrumento computacional sistemático a casos presentados en la literatura existente se detecta un error en Sarkar et al. (1997). En este artículo una de las soluciones dadas no es un equilibrio del modelo.

En el capítulo 5 se describe un problema de localización competitiva en el que se utiliza el concepto de Stackelberg, líder-seguidor. En este modelo existe dos cadenas de suministro, la del líder, que desea abrir un nuevo centro y debe decidir sobre su localización, y la del seguidor, que debe tomar su decisión después del líder. La cuota de mercado captada por las cadenas viene dada por un modelo gravitacional que depende de las distancias a los centros. Es conocido que el problema del seguidor es un complicado problema de optimización global. El problema del líder es incluso más complicado ya que el líder tiene que decidir dónde localizar el nuevo centro anticipándose a la solución óptima que tomará el seguidor. Hasta el momento, en la literatura existente únicamente se han desarrollado técnicas heurísticas para solucionar el problema del líder. En este capítulo se desarrollan algoritmos de ramificación y acotación (B&B). El algoritmo (B&B) para el líder debe resolver dos problemas como el del seguidor: el primero para calcular (y de esta manera anticiparse a) la localización óptima del seguidor (problema de optimización global) y el segundo para obtener una cota superior para el problema del líder (otro problema de optimización global). Con respecto al algoritmo B&B para el problema del seguidor se han derivado cuatro cotas superiores diferentes. El algoritmo evalúa y elige la mejor de las cuatro cotas superiores. La regla de ramificación usa rectángulos y se generan nuevos rectángulos dividiendo el rectángulo original mediante la bisección en el eje de más anchura. Además, se han implementado y comparado dos variaciones: la primera consiste en comenzar con un rectángulo inicial que contiene el área factible; la segunda consiste en comenzar con una partición inicial en rectángulos. En el capítulo se indica que la idea de generar una partición inicial mejora la acotación, pero genera más particiones que las estrictamente necesarias. Adicionalmente se han comparado dos tipos de criterios de selección: seleccionar el último rectángulo generado, "depth first search", y seleccionar el rectángulo con la mejor cota, "best bound search". A continuación, se mide la eficiencia de ambos criterios. La efectividad y la eficiencia de los algoritmos son investigadas con la ayuda de estudios computacionales. Como principal conclusión se ha comprobado que, teniendo en cuenta la ejecución completa del algoritmo, cotas más sofisticadas no son necesariamente más efectivas que cotas más simples basadas en distancias. Es mejor centrarse en medir la calidad de la cota durante la ejecución del algoritmo y elegir la más ajustada. Con respecto al criterio de selección, seleccionar como candidato a la división al rectángulo con la mejor cota (el más prometedor, "best bound search") tiende a resultar en un mínimo esfuerzo en el número de evaluaciones de funciones. Sin embargo, la regla de selección "depth first search" podría suponer menos requisitos de memoria. Debido a que los requisitos de memoria suelen

ser un problema en grandes dimensiones, no es necesariamente un punto importante para el problema de localización en espacios de dos dimensiones.

Las principales conclusiones de este estudio se obtuvieron respondiendo a las cuestiones de investigación. Dentro del ámbito de investigación de este estudio, los modelos de TJ han sido herramientas valiosas para modelar problemas de toma de decisiones involucrando múltiples jugadores, asimétricos y heterogéneos. La contribución dentro de los modelos de TJ es más significativa en los capítulo 2 y 4 puesto que añaden características adicionales a casos observados en la literatura existente. Un factor importante de este estudio es lo que se ha llamado "la codificación en notación de programación matemática"; y que contribuye de manera importante al desarrollo de algoritmos para generar soluciones en problemas de toma de decisiones. Con respecto a los casos estudiados, diferentes retos inspiraron el desarrollo de los algoritmos: aplicación a grandes casos, asimetría, heterogeneidad y multiples jugadores. Para cada uno de los problemas se introducen e implementan algoritmos; éstos están basados en métodos de enumeración, métodos iterativos, soluciones analíticas, métodos SQP, técnica de funciones "penalti" y métodos de optimización global.

Brevemente, las principales contribuciones de este estudio se pueden resumir como sigue. En esta Tesis se han estudiado modelos de TJ y reformulado dichos modelos en problemas de optimización de IO. Este procedimiento de reformular problemas de TJ en problemas de optimización de IO es un paso muy importante para el desarrollo de los algoritmos. Por otra parte, en este estudio de investigación, se han usado tanto datos reales como datos generados de forma aleatoria. En todos los casos aplicados, los algoritmos son diseñados siguiendo el modelo conceptual introducido en el capítulo 1. La idea es que los resultados de los algoritmos implementados tienen como fin proporcionar elementos a los tomadores de decisiones y analistas para comprender el problema de "la toma de decisión". Mientras que dos de los casos estudiados contribuyen a ese fin mediante el uso de datos reales (capítulos 2 y 3), los modelos y métodos en los capítulos 4 and 5 son ilustrados con datos generados de manera aleatoria.

Este estudio debería ser visto como un primer paso para hacer que la TJ sea más aplicada a problemas de teoría de decisión de mayor dimensión, más complejos. Los casos aplicados contienen características específicas que sirven de guía para la selección de modelos específicos de TJ, modelos de IO y algoritmos. Dentro de los problemas de toma de decisiones, el estudio se ha centrado en el comportamiento racional del tomador de decisiones. Una posible línea futura de investigación consiste en considerar racionalidad acotada en los problemas de toma de decisiones. Con respecto a la TJ, no se han estudiado juegos de información incompleta. Sería de interés enfocar la futura investigación en casos de juegos de información incompleta con un gran número de jugadores y estudiar cómo contribuir tanto en el desarrollo de modelos de TJ y en la codificación de estos modelos de TJ en modelos de IO, así como en el desarrollo de nuevos métodos/algoritmos de IO. Con respecto a los modelos de IO y algoritmos, el desarrollo de los modelos y los algoritmos en este estudio se ha basado en las características de los casos aplicados. Una futura investigación puede incluir el estudio del rendimiento de los algoritmos desarrollados en comparación con otros algoritmos de IO. En resumen, todavía existen nuevos retos en la elaboración de modelos y algoritmos en Investigación Operativa para poner la Teoría de Juegos en práctica.

List of Publications

Publications and Papers submitted to international journals

Sáiz, M.E., Fernández, J., Hendrix, E.M.T. and Pelegrín, B. (2007), On a Branch-and-Bound approach for a Huff-like Stackelberg location problem, Submitted

de Ridder A., Rusinowska A., Sáiz M.E. and Hendrix E.M.T. (2006), Coalition formation: the role of procedure and policy flexibility. Submitted

Sáiz, M.E., Hendrix, E.M.T.(2006), Methods for computing Nash equilibria of a location-quantity game. *Computers & Operations Research*, in Press: doi: 10.1016/ j.cor.2007.02.022

Sáiz, M.E., Hendrix, E.M.T. and Olieman, N. (2006), On the computation of stability in multiple coalition formation games. *Computational Economics* 28(3): 251-275

Finus, M., Sáíz, M.E. and Hendrix, E.M.T. (2004), An empirical test of new developments in coalition theory for the design of international environmental agreements. Submitted to *Environmental & Resource Economics*

Working Papers

Sáiz, M.E., Hendrix, E.M.T., de Ridder, A. and Rusinowska,A. (2007), On the computation of Negotiation Positions and Preferences in a Spatial Coalition Model. Mansholt Working Paper MWP, Wageningen

Sáiz, M.E., Fernández, J., Hendrix, E.M.T. and Pelegrín, B. (2007), On a Branch-and-Bound approach for a Huff-like Stackelberg location problem. Mansholt Working Paper MWP-37, Wageningen

Sáiz, M.E., Hendrix, E.M.T. (2006), Methods for computing Nash equilibria of a location-quantity game. Mansholt Working paper MWP-30, Wageningen

Sáiz, M.E., Hendrix, E.M.T. and Olieman N.O. (2004), Determination of stable coalitions in a CO2 emission game. Mansholt Working paper MWP-12, Wageningen

Finus, M., Sáiz, M.E., Hendrix, E.M.T. (2004), An empirical test of new developments in coalition theory for the design of international environmental agreements. Mansholt Working paper MWP-14, Wageningen

International conferences

Sáiz, M.E., Fernández, J., Hendrix, E.M.T. and Pelegrín, B. (2007), On a Branch-and-Bound approach for a Huff-like Stackelberg location problem. EURO XXII 2007. July 8-11, Prague (Czech Republic)

Sáiz, M.E., de Ridder A., Hendrix E.M.T., Rusinowska A. (2006), On the Computation of Stability and Negotiation Positions in a Spatial Coalition Model. Operations Research 2006 (OR2006). September 6-8, Karlsruhe (Germany)

Sáiz, M.E., Hendrix E.M.T. (2005), Computing Stability in Coalition Formation with Heterogeneous Players. First Spain Italy Netherlands Meeting on Game Theory (SING1). June 24-26, Maastricht (The Netherlands)

Sáiz, M.E., Hendrix E.M.T. (2004), Horizontal Cooperation Incentives in Oligopoly Supply Chains. International Conference Operations Research 2004 (GOR/NGB). September 1-3, Tilburg (The Netherlands)

Sáiz, M.E., Hendrix, E.M.T.(2003), Determination of stable coalitions in a CO2 emission game. International Symposium on Mathematical Programming (ISMP 2003). August 18-22, Copenhagen (Denmark)

National conferences

Sáiz, M.E., Hendrix E.M.T. (2004), Facility-Pricing Location Game. Twenty-ninth Conference on the Mathematics of Operations Research (LNMB). January 13-15, Lunteren (The Netherlands)

Blanquero, R., Carrizosa, E., Sáiz, M.E. (2003), Optimización Global en la Estimación de Máxima Verosimilitud con Datos Censurados. 27 Congreso Nacional de Estadística e Investigación Operativa (SEIO) (ISBN 84-8409-955-5). April 2003, Lleida (Spain)

Curriculum Vitae

M. Elena Sáiz was born in Madrid on May 31, 1973 (officially). In 1999 she graduated in Mathematics at the Universidad Complutense de Madrid. From 1999 she was employed at Getronics, and later on at Endesa, as a System Manager, being responsible of databases (Oracle), applications (SAP, Siebel) and operating systems (UNIX, Linux, Windows). After getting experience in ICT, she realized that her real vocation is in research. In 2000 she started working in Madrid as a teacher in a Secondary School (CEF) and in 2001 she moved to Sevilla to start her PhD studies. She combined her PhD studies at the Universidad de Sevilla (Spain) (Sept 2001-Dec 2002) with work at Getronics.

In 2003 she was appointed at Wageningen Universiteit as a guest researcher with a grant from WIMEK for 6 months within the project "STAbility of COalitions"in cooperation with Environmental Economics and Natural Resources Group and Hagen Universität. After that, she was granted with a Marie Curie Host Fellowship within the "Marie Curie Training Site Mansholt Institute, Consumer-oriented Chains" which made possible for her to receive course training at Wageningen Universiteit. From August 2004 she was a junior researcher within the Operations Research and Logistics Group doing her PhD. In her research she got experience with applying several optimization algorithms: enumeration, penalty approach, branch and bound, multistart and local search.

From November 2007 she will be appointed as a Post-doc at the NSM Decision Lab of the Radboud Universiteit Nijmegen to conduct research in the field of social choice theory and cooperative game theory, including coalition formation theory. The research objective is to set up an extensive database to test the empirical performance of social choice theoretic and game theoretic models of collective decision making and coalition formation.

Annex to statement
M. Elena Sáiz
Mansholt Graduate School of Social Sciences (MG3S)
Completed Training and Supervision Plan

Name of the course	Department/Institute	Year	ECTS*
Courses:			
Writing and presenting a scientific paper	Mansholt Graduate School of Social Sciences (MG3S)	2003	1.4
Time Planning and Project Management	Wageningen Graduate Schools	2005	1.4
Mansholt Introduction course	MG3S	2003	1.4
Advanced Operations Research	PhD Course University of Sevilla	2002	14
Advanced Statistics	PhD Course University of Sevilla	2002	14
Combinatorial Optimization I	LNMB	2004	6.3
Combinatorial Optimization II	LNMB	2004	6.3
Operations Research and Logistics	Operations Research and Logistics Group	2004	6
Presentations at conferences:			3
Mansholt Multidisciplinary Seminar	MG3S	2007	
ISMP Copenhagen, Denmark		2003	
LNMB Conference, Lunteren, the Netherlands		2004	
GOR/NGB, Tilburg, the Netherlands		2004	
First Spain-Italy-Netherlands Meeting on Game Theory, Maastricht, the Netherlands		2004	
GOR/ÖGOR/SVOR, Karlsruhe, Germany		2006	
EURO XXII, Prague, Czech Republic		2007	
Total (minimum 30 ECTS)			**55**

* One ECTS on average is equivalent to 28 hours of course work
LNMB stands for Landelijk Netwerk Mathematische Besliskunde
ISMP stands for International Symposium on Mathematical Programming
GOR stands for Gesellschaft für Operations Research
ÖGOR stands for Österreichische Gesellschaft für Operations Research
SVOR stands for Schweizerische Vereinigung für Operations Research
NGB stands for Nederlands Genootschap voor Besliskunde