Modelling Real-World Contract Negotiations

Sander Kremers

October 5, 2014

Master Thesis for Applied Mathematics
Stochastic Operations Research

Assessment Commitee:
Dr. J.C.W, van Ommeren (University of Twente)
Dr. A.V. den Boer (University of Twente)
Prof.dr. R.J. Boucherie (University of Twente)
Prof.dr. M.J. Uetz (University of Twente)
D.P.M. de Poot, MSc (ORTEC)

Abstract

In this research we develop a new mathematical model for negotiations that is able to serve as the basis for a Negotiation Support System. We start by examining the existing models in the literature. We define the key properties of real-world negotiations that are missing in the existing models and propose a new model that does have these properties. We analyse this model by proving several consistency theorems. Because there is no easy closed form expression for the negotiation outcome, we perform numerical studies to determine the effect of the strategy parameters on the outcome of the negotiations.

We then use these observations to develop a procedure with which we estimate the strategy of the other agent and optimize our strategy, given the estimated strategy of the other agent. Finally, we use MATLAB simulations to analyse the performance of this procedure for various strategies of the other agent. From these simulations we conclude that the estimation and simulation procedure yields a significantly better outcome, and should therefore be used whenever possible. We also discuss extensions our proposed negotiation model and give a number of suggestions for future research.

Contents

1 Introduction

Everyone uses negotiation in real-life scenarios. When buying a new car or a new house, people generally negotiate about the price and terms of delivery. Negotiation also occurs often in businesses. For example, consider the negotiation about a software project for which price, delivery date and scope have to be determined. Over multiple sessions, both parties exchange offers and give feedback on each others offers. After a number of sessions, consensus is reached and a contract is signed. The final contract is a legally binding document that outlines the responsibilities of both parties to each other. For large projects, the outcome of these negotiations are very important, because a large amount of money can be involved.

Many business processes can be supported by systems based on mathematical models. At this moment however, no system is available to help negotiators determine a good strategy for negotiations. Given that negotiations are so important, we will develop a mathematical model to support these negotiators. Especially for large projects, if we can improve the negotiation outcome for one of the parties by even a small amount (e.g. reduce the final price of a project by 1%), a negotiation support system could generate a lot of added value for the decision maker. This research is the first step for ORTEC to develop such a negotiation support system.

Negotiation Setting

We will look at the following negotiation setting for this research. Two parties are negotiating about a contract consisting of, possibly, multiple issues. Both parties do not only have conflicting interests, they also have different priorities regarding the issues. We will assume negotiation is conducted over multiple rounds or sessions.

Consider the case where a buyer and a seller negotiate about the delivery and price of a product. The buyer wants to minimize the cost and the delivery time, the seller wants to maximize them. However, the buyer might be in a hurry to get the product, so for him, delivery time is most important. For the seller, the delivery date might not matter much, but he does care about the price.

1.1 Research goal

Outline of a support system

We will create a mathematical model to serve as the basis for a decision support system for contract negotiations. As the name says, a decision support system supports the decision maker, it does not replace him. The negotiator should be able to use the output of the system as guidelines and use his expert opinion to fill in the details. A realistic use for such a Negotiation Support System (NSS) is the following. During the negotiations, after an offer has been made by the other party, the user of the NSS enters this offer into the system. The system then updates and gives a suggestion for a next offer.

Scope

Our model has to support one of the negotiating parties. To do this, our model will only use information that is given by this party or obtained during the negotiations. Also, we do not want to replace the negotiator, instead, we support him in obtaining a good outcome. Our model will be able to update its advice according to the realization of the negotiation. The model will also be suitable for negotiations with multiple dimensions. In this research we will only look at scenarios where there exists a contract that is preferable over disagreement for both parties.

1.2 Research Questions

The objective of this research is to develop a mathematical model that can serve as the basis for a Negotiation Support System. To do that, we must first answer the following research questions:

- How can we model contract negotiations mathematically?
  - What existing models are there?
  - What assumptions do we need to make?
  - How can we model the behavior of the opposing party?
- How can we optimize the negotiation strategy of the user?
- How can we measure the performance of our model?

Approach

Game theory studies the strategic interaction of self-interested agents [19]. In particular, bargaining theory seems to be well suited for this problem. The simplest bargaining model looks at the scenario where two agents must agree on the division of a single pie. This is a good starting point for our research, because it describes a very simple negotiation setting. We will then use concepts from these game theoretical bargaining models as the basis for our new model. We will start with a simple model and expand this by broadening its scope and relaxing its assumptions. We make sure that the model only needs data that is realistically obtainable and that it provides output that is actually useful to the user. We will also test the performance of our model. We do this by performing simulations for different types of behavior by the other party.

1.2.1 Structure of the Report

In the remainder of this report, we will first look at the relevant literature on this topic. We then present a basic model and some initial analysis on this model. We will extend this basic model to a more general one and perform detailed analysis (both numerical and analytical) on the performance. As a final step, we provide a method for estimating the strategy of the other party and optimizing our strategy, given this estimate. We will evaluate the performance of this method by simulations. We end by drawing conclusions from these results and suggesting a number of directions for future research.

2 Literature Study

In this chapter we will discuss the literature relevant to negotiations. We will start by looking at some general concepts from game theory which are useful to analyze negotiations. We will then look at models that analyze negotiations in various settings. The next research area we look is automated negotiations. Automated negotiation has been an active area of research in the area of computer science and artificial intelligence in the last few years. The goal of automated negotiation is to design systems that can negotiate autonomously, an example of an application for such systems is (automated) negotiation in e-commerce. We also briefly discuss some past initiatives in the area of negotiation support systems. Finally, we determine the limitations of the current theory.

2.1 Game Theory

Game theory studies the strategic interaction of multiple players that are all trying to maximize their outcome. In game theory, we assume that the players are perfectly rational. This perfect rationality consists of two parts [2]:

1. Rationality: Agents maximize their own utility (and will always take the action that accomplishes this)
2. Rationality is common knowledge: Agents know that all other agents are rational (and that all agents know this, and they know that all agents know this ad infinitum)

We need these assumptions on the behavior of the players to be able to analyze the outcome of games. If we look at simple games, that is, games consisting of a single play where players act simultaneously, we can use the Nash Equilibrium [17] to determine equilibrium behavior. A set of strategies for each player form a Nash Equilibrium if no player can improve his outcome by changing his strategy. In other words, all strategies are best responses to each other. All of the game theoretical analysis assume perfect rationality and analyse the equilibrium outcome of the game.

In games consisting of multiple periods, or games were players act sequentially, we use a game tree to model the game. In such a game, the notion of a Nash Equilibrium is not strong enough. In general there may be many Nash Equilibria in such games. A sub-game perfect equilibrium is a refinement of the Nash Equilibruim [19]. A set of strategies is a sub-game perfect equilibrium if it is a Nash Equilibrium not only in the whole tree, but also in every sub-game. A sub-game is a "new" game starting from one of the decision nodes in the game tree. By definition every sub-game perfect equilibrium is a Nash Equilibrium, but the converse is not true. It is beyond the scope of this thesis to list all implications of this refined definition of an equilibrium, but the interested reader is referred to [5].

Information setting

The simplest games we can look at are game of perfect information. Perfect information means that all players know everything there is to know about the state of the game [19]. That is, they know the current state of the game, the decisions available to all players and the payoffs that are obtained by all players. Chess is an example of a game of perfect information.

It may be the case that players do not know the exact payoffs of the game, or that we only know that the opponent can be any of a number of types (each having different payoffs). In such a case, players are said to have incomplete information.

Lastly, players may not know what state the game is in, in such a case, a game is said to have imperfect information. An example of a game of imperfect information is poker. The state of the game is determined by the cards in each players hands, but each player only knows the cards in his own hand.

2.2 Negotiation Theory

We will now turn our attention to a more specific area of game theory, namely negotiation theory.

2.2.1 Definitions

Pareto optimality A solution is Pareto optimal if there is no other solution that improves the payoff of one agent, without making the other agent worse off [7]. A Pareto optimal solution is preferable to a non-Pareto optimal solution. However, obtaining a Pareto optimal solution might be hard when both parties have private information about their utilities [23].

Utility In all of the multiple-issue models, a utility function is used to compare solutions. A utility function $u(x_1, \ldots, x_n): \quad \mathbb{R}^n \to \mathbb{R}$ assigns a single value (usually in [0,1]) to every possible contract. Instead of comparing the negotiation outcome by comparing the scores on all issues x_i individually, we use the utility to compare the various outcomes. Each agent has its own utility function and they are assumed to be Von Neuman/Morgenstern expected utility maximizers [25]. We say a utility function is (linear) additive if $u(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i \cdot u(x_i)$. Here w_i is the weight of issue i and $u(x_i)$ is the utility of issue i. If a utility function is linear additive, the utilities of the various issues are independent in the sense that we can determine the outcome of a contract by looking at the outcome of each issue individually.

Reservation Utility The reservation utility of an agent is the minimal acceptable outcome of the negotiation. This does not necessarily have to be zero. In game theoretical models, the reservation utility is assumed to be known. In the non-game theoretical models (e.g. [11]), authors suggest determining the

Best Alternative Outcome to Negotiation (BATNA), and set the reservation utility equal to the utility of this BATNA.

Zone of Agreement When both parties have their own reservation utility, we can define the zone of agreement as the set of all solutions that are acceptable to both players. If players negotiate about the division of a pie of size 1 and both players want at least 40% of the pie, the zone of agreement is to give a piece of pie in the interval of [0.4,0.6] to player one (and the rest to player two).

2.2.2 Nash bargaining solution

Nash [17] used an axiomatic approach to predict the outcome of a bargaining situation where two parties negotiate about the division of a single pie. He viewed the negotiation process as a black box and only looked at the outcome of the process. If both agents have public utility functions and obtain a utility of 0 for disagreement, he showed that the only solution that satisfied his four axioms of rationality is the solution that maximizes the product of the utilities of both agents. The interested reader is referred to [21] for a thorough review of these axioms.

2.2.3 Alternating Offer Protocol

Instead of Nash' axiomatic approach, Rubinstein [22] specified a negotiation protocol and analyzed the behavior of rational agents that use this protocol. In order to ensure that agents reach an agreement, he assumed that agents either incur a fixed cost per negotiation period, or that they both have a discount factor δ_i. The interpretation of this discount factor is that a pie that is worth 1 in the first period to player i, is only worth δ_i in period 2, δ_i^2 in period 3, etc. Either of these two assumptions is needed to ensure that agents have an incentive to reach agreement sooner rather than later.

In his setting, agents have perfect information about each other. Two agents negotiate about a single issue and they both have their own discount factor δ_1 and δ_2. Agents alternate making offers. In particular, in every time period, the "active" agent has the following options:

1. Accept the previous offer
2. Reject the offer and propose a new offer
3. Quit the negotiations

Rubinstein shows that, in this setting, the only Subgame Perfect Equilibrium is that agreement is reached in the first time period, and that agent 1 receives a fraction $\frac{1-\delta_1}{1-\delta_1\delta_2}$ of the total. Notice that the player that starts the negotiations has an advantage if both players have equal discount factors.

2.2.4 Strategy choice

We can identify two classes of strategy in the literature. Time dependent strategies and tit-for-tat strategies. We will describe both of them.

Time Dependent Strategy A time dependent strategy that is often used in literature (e.g. [10, 13, 7]) is:

$$u_i(t) = 1 - (1 - ru_i)\left(\frac{t}{T_i}\right)^{\frac{1}{\beta_i}} \tag{2.1}$$

Here ru_i is the reservation utility of agent i, T_i is his deadline and β_i is a parameter controlling the behavior of agent i. The reason that it is used often, is that it controls the time dependent behavior of the negotiating agents by just a single parameter β_i. Using different values of β_i in (2.1), we can define three general types of strategies.

1. $\beta < 1$:Boulware: The agent maintains his initial offer until he is close to the deadline, then concedes up to his reservation utility.
2. $\beta = 1$: Linear: The agent concedes utility linearly.
3. $\beta > 1$:Conceder: At the start of the agent, he quickly concedes up to his reservation utility.

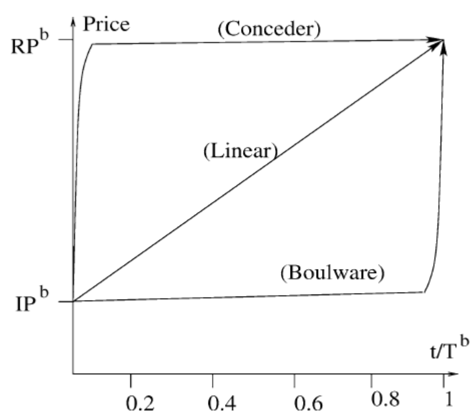

Figure 2.1: Time Dependent strategies[13]

In figure 2.1 each line corresponds to one of these strategies. From equation 2.1 we can see that the utility of an offer from a player using this strategy does not depend on the offer of the other party. So whether the other party makes a small concession or a very large one, we do not change the utility of our next offer.

Behavior dependent strategy Another class of strategies is the behavior dependent strategies. In particular, tit-for-tat is often used (e.g. [9, 6]). In this strategy, we base our behavior on the behavior of the other party. Informally, this strategy tries to match the behavior of the other party. If the other party makes a large concession, we respond by also making a large concession. If the other party does not make any concession, or makes a greedy move, we respond in kind.

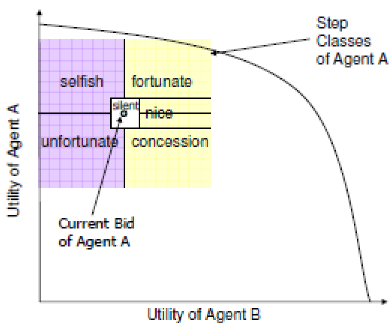

Figure 2.2: Classification of offers [9]

To be able to match the behavior of the other party, Hindriks et al.[9] define a number of categories for the offers of the other party. In figure 2.2 these are displayed. For example, a fortunate move is a move that increases the utility of both parties, whereas a selfish move only increases the utility of a single party. Hindriks et al. show that a tit-for-tat strategy that uses these classifications does well in automated negotiations. Note however, that in order to classify moves in this manner, we must know (or at least have a good idea) about the utility function of the other party.

2.2.5 Incomplete information

Until now, we assumed both agents had complete information about themselves, but also about their opponents. Fatima et al. [23] look at various scenarios where agents use a time-dependent strategy, but have incomplete information about the other party's deadline, discount factor and/or reservation price. They also consider the cases where agents have asymmetric information, e.g. one agent might have complete information, while the other agent does not have any information about the other party. They determine the equilibrium outcome in each of these scenarios. Their analysis is based on the following two theorems:

1. At the deadline, rational agents will offer their reservation utility.
2. Agents use either an extreme conceder, a linear or an extreme boulware strategy.

They prove the correctness of both theorems. The first one is true in all cases. For the second one, they show that, under the assumption of time-dependent strategies, the extreme conceder, linear and extreme boulware strategy dominate all other strategies. An agent using an extreme boulware strategy does not make any concessions until very close to its deadline, then it (almost) instantly concedes to its reservation utility. On the other hand, an extreme conceder agent (almost) instantly concedes to its reservation utility. This limitation in available strategies makes the analysis by [23] ill-suited for applications in real-world negotiations. In the real world, both parties are expected to use less extreme strategies (even if they are not optimal). They will slowly concede utility until agreement is reached.

2.2.6 Agenda

In all of the above models, we negotiate about all the issues simultaneously. It is also possible that we negotiate on the issues one-by-one. In such a case, it is important to consider the order of the issues on the agenda, as they can greatly affect the (equilibrium) outcome of the negotiations [8, 26].

2.3 Automated Negotiation

2.3.1 ANAC

There is a yearly competition for automated negotiation agents called ANAC (Automated Negotiating Agents Competition) [1]. Here, teams of students or researchers can submit an agent that negotiates competitively against other agents on a predefined domain. In previous years, all of the utility function were linearly additive. This year, non-linear utilities are considered for the first time. All of the successful agents use some sort of opponent model that tries to determine how the utility of the opponent behaves, this works well because of the large amount of offers per negotiation session and the linearity of the utility functions

2.3.2 Mediated negotiations

As we noticed before, in a general setting it may be hard to achieve Pareto optimal outcomes. One of the ways to remedy this is to consider mediated negotiations. In this setting, agents share (some) private information with an unbiased mediator who then suggests possible contracts. Examples of such mediators can be found in [12, 14]. It is beyond the scope of this research to analyse these in detail, because we aim to create a Negotiation Support System (NSS) that supports a single party.

2.3.3 Offer protocols

A different way to approach automated negotiations is to look at the way in which agents generate their offers. Lai et al. [13, 14] assume both agents use

(2.1), with some fixed parameter β_i to determine the utility of an offer. All offers that satisfy Equation (2.1) lie on an iso-utility plane. The authors analyze the following offer selection rule: "Select as the next offer the offer (on the iso-utility plane) that is closest to the previous best offer by the opponent". They show by simulations that the outcome of negotiations (assuming a rather specific form of convex utility functions) under this protocol are close to Pareto optimal.

2.3.4 Opponent Learning

In almost all of the models, the opponents utility is assumed to be private information. There is a wide variety of methods to learn the opponents utility. The offers made by the opponent is used as input for these methods. Among others we can use:

- Bayesian Learning [10]
- Frequency-based Learning [3]
- Fuzzy Constraints [15]
- Genetic Algorithms [16]

These learning models have the following limitations. They require a large amount of offers to be effective, and they generally require utility functions to be linear.

2.4 Negotiation Support Systems

Various research has been done on the design of a Negotiation Support System. A NSS can be used to support a single party, or it can be used as a mediator. Using a NSS as a mediator, we can obtain higher total utilities [12], but both parties would have to agree to use such a mediator, which might be difficult to implement in the real world.

2.4.1 Pocket Negotiator

There is a Pocket Negotiator project by Hindriks et al. [4] from a research group in Delft. Over the past few years, they published a number of articles on various aspects of negotiations. A result of this research program, is that they now have a web-based application called the pocket negotiator that can assist in everyday negotiations. This application includes a method of preference elicitation, visualization of the utility of offers and suggestion of a next offer. However, its scope is different from ours, in that they assume the utility function of both parties to be known (or that both parties use the pocket negotiator to determine them) and are linear additive and that there is only a small amount of issues to be negotiated.

2.5 Limitations of current work

In this section we will discuss the limitations of the works we described above.

2.5.1 Perfect Rationality

Game theoretical models are based on the assumption that agents are perfectly rational and that this rationality is common knowledge (i.e. they know that the other agent is also perfectly rational). This assumption is of great impact on how these models can be used. It causes all of the models to be descriptive models, because perfect rationality implies that agents act according to their optimal strategy. We could compute these equilibrium strategies, but by our assumption, they are already known to the agents, so this analysis would not provide any new information.

2.5.2 Automated Negotiation models (Learning modules)

All of the automated negotiation agents use some sort of opponent model to model the utility of the other agent. In order to identify trade-offs between issues, such a model is necessary. However, the models described in the literature have major draw-backs. They require a large amount of offers by the other party before they can accurately estimate the utility of the opponent. Moreover, most of the learning modules make major assumptions about the form of the utility function of the other party.

2.5.3 Contribution of this research

Because of these limitations, the models and methods from literature are ill-suited for applications in a real-world negotiation. In this research, we take a different approach. We do not assume agents act according to the perfect rationality principle, and we do not need a large amount of historical data for our model.

3 Basic ENO model

In this section we will first look at the properties of real-world negotiations and then describe a basic version of our negotiation model.

3.1 Business Setting

Example

An example of a type of contract we will consider is a cost-plus contract. This type of contract is often used in construction. In this type of contract, we have a buyer and a seller. The buyer contracts the seller to realize a construction project (boat/house/road etc). The buyer reimburses the seller for all costs made and the seller receives an additional fee based on his performance. The structure of this fee is such that it incentives the seller to do well. If the seller performs well according to the performance measures defined in the contract, he will receive a larger fee.

In the contract, negotiating agents decide upon the following:

- Target performance
- Fee
- Legal issues

We will assume the target performance is specified in terms of:

- Cost estimate
- Time estimate

This cost estimate is specified in detailed terms of the estimated value of various cost drivers. For example, the expected amount of labor and cost of this labor, expected amount of material and cost of material, etc.

The fee is based on the performance of the seller. That is, it is based on the realization of the performance measures as specified in the contract. Both parties have conflicting interests about the specification of the target performance. The buyer wants the target performance to be a challenging target. The seller wants the target performance to be set as "low" as possible, such that it is easily achievable and his own profit is maximized. We will only look at the negotiations dimensions which are measurable, so legal issues will not be considered.

Properties of a real-world negotiation

Based on discussions with people that have extensive negotiation experience, we have determined the following key properties for real-world negotiations:

- Their duration is between 5 and 10 rounds

- Agents will make a concession in every round
- There is often no historical information on the behavior of the other party

As we have seen in the literature review, the game theoretical models violate the first two requirements and the automated negotiation models require either a large negotiation history, or much more than 5 to 10 rounds.

3.2 A General Mathematical Model

We first present the general mathematical model for negotiations. We consider a model where two agents negotiate about a contract of r issues $x_1, \dots, x_r$. For example, one could consider a project for which the cost and deadline have to be negotiated. Let $X_1, \dots, X_r$ be the sets of feasible outcomes of each issue. Then $X = X_1 \times \cdots \times X_r$ is the set of all possible contracts. If no agreement is reached during the negotiations, the negotiation ends without a contract being signed, so neither party will receive anything. The negotiations we will look at, consist of multiple rounds $(t_1, \dots, t_N)$, where t_N is the round in which the negotiations end. Agents alternate making offers. That is, in time period t, we will perform action a_t and the other party will perform an action $\overline{a}_t$, where:

$$a_t = \begin{cases} \text{Accept the previous offer } \overline{p}_{t-1} \\ \text{Quit the negotiations} \\ \text{Propose a new contract } p_t \in X \end{cases}$$

Each agent has a utility function $u_i(x)$ to compare different possible outcomes with each other. So $u_i(x) : X \to [0, 1]$. Both agents want to maximize their own utility, but this utility function is private information. That is, agents only know their own utility function, but do not know the utility function of the other party.

3.2.1 Implication of business setting for mathematical model

The business setting we are considering has the following consequences for our model. Everything in the contract except the time estimate can be specified in terms of monetary value. This means that it is possible to define a utility function to compare different contracts. We will assume negotiation is conducted over multiple sessions. Each round corresponds to a time period. A round begins with an offer from our party (the buyer). During a session, the seller will provide feedback on our offer. This feedback is of the form of a counteroffer: $\overline{u_t} \in X$

As discussed in Section 2, we do not want to take the game theoretic approach and assume agents act according to the perfect rationality principle, because this is not a realistic assumption.

3.2.2 Additional Assumption

The model we defined above is incomplete. The missing ingredient is that we need to model the impact of our actions. What happens if we propose a certain contract? What is a good contract to propose? To answer those questions, we have identified two options:

- Heuristically define what constitutes a "good" proposal and look at a single-period problem.
- Explicitly define how the other party responds to our offers and look at all of the periods.

The advantage of the first option is that it reduces the problem to a tractable single-period problem. The challenge is to find a good measure for the quality of a proposal.

The advantage of the second option is that a solution to the model gives us a global solution (over multiple time periods) instead of a local solution. Such a solution will provide a strategy for the whole negotiation, instead of just the current period. The drawback is that we need to explicitly model the behavior of the other party. It is difficult to define this behavior, especially at the start of the negotiation. We think it is unrealistic to define this behavior appropriately. We will therefore look at a heuristic way to measure the quality of a solution in a single period. In the next section, we will propose a model based on such a measure.

3.3 Basic estimated negotiation outcome model

We will now describe the basic version of our Estimated Negotiation Outcome (ENO) model. In the analysis of this model, we will look at a single dimension problem. In Section 8, we will discuss how the model can be extended to multiple dimensions In the remainder of this report, we will use the following notation:

- u_n is the offer made by us in round n
- $\overline{u}_n$ is the offer made by the other party in round n
- If either party accepts the previous offer, we interpret this as making the same offer as the previous one (e.g. $u_n = \overline{u}_{n-1}$ if we accept in round n).

We also assume neither party aborts the negotiations prematurely. Given an offer of the other party $\overline{u}_{n-1}$ and a counteroffer by us u_n, we define the estimated negotiation outcome (ENO) as:

$$ENO = u_n - \frac{1}{2}(u_n - \overline{u}_{n-1})$$

Intuitively, this means that if we are given an offer and a counteroffer , we expect the negotiation to end at a point in the middle of those two offers. We can also use a different estimation technique (i.e. use something more complicated than ending in the middle), but the key assumption is that we estimate the negotiation outcome based on the utilities of the offer and counteroffer. By doing this, we have a well-defined single-period problem that we can solve.

We cannot simply find an offer that maximizes the ENO. Trying to maximize the ENO yields the trivial solution of always sending the same offer (of maximum utility), independent of the actions of the other party. Instead of maximizing the ENO, we define a different objective. At the start of the negotiation we set a target that we want to achieve. Call this T. After every negotiation session t, we want to find the offer u_n such, that the estimated negotiation outcome equals our target:

$$u_n - \frac{1}{2}(u_n - \overline{u}_{n-1}) \quad = \quad T \tag{3.1}$$

This basic model is essentially a tit-for-tat strategy that we have seen in the literature. If the other party makes a large concession, we respond by doing the same. If he makes a very small concession, we do not concede much either. Because we look at all dimensions independently, we do not need a utility function, but we can talk about the value of an offer directly.

3.4 Analysis of basic ENO model

In this section we will analyze the performance of the ENO model if both parties use the strategy given by this method. The first agent is the maximizing agent, he has a Target T. The second agent is a minimizing agent, his target is $\bar{T}$. The utility of the initial offer of the maximizing agent is 1, the utility of the initial offer of the minimizing agent is 0. Starting with the maximizing agent, both parties will make offers according to the ENO model. Recall that we defined the ENO as:

$$ENO = \frac{1}{2}(u_t + \overline{u_t})$$

Where u_t and $\overline{u_t}$ are the offers of respectively the maximizing and the minimizing party.

The first (maximizing) agent will select offers according to Equation 3.2:

$$\frac{1}{2}(u_t + \overline{u_{t-1}}) = T \tag{3.2}$$

Which means their offer will be:

$$u_t = 2T - \overline{u_{t-1}}$$

The second (minimizing) agent will select offers according to Equation 3.3:

$$\frac{1}{2}(u_t + \overline{u_t}) = \overline{T} \tag{3.3}$$

So their offer will be:

$$\overline{u_t} = 2\overline{T} - u_t$$

It is easy to see that we can only reach agreement if $T < \bar{T}$, because agents only accept offers that are better than their targets Moreover, in order for the model to be consistent, we want agents to make a concession in each negotiation round. A sufficient condition for this is that both $u_1 < 1$ and $\overline{u_1} > 0$; once both agents make a concession in the first round, they will then make a concession in every next round as well.

We can now analyze the behavior of both agents analytically and find a closed-form expression for the offers they make in every round.

Theorem 1. *In the basic ENO model, the offers made in round n are given by:* $u_n = 2n(T - \overline{T}) + 2\overline{T}$ and $\overline{u_n} = 2n(\overline{T} - T)$.

Proof. Assuming agreement is reached after round n, we have:

$$
\begin{aligned}
u_0 &= 1 & \overline{u_0} &= 0 \\
u_1 = 2T - 0 &= 2T & \overline{u_1} &= 2\overline{T} - u_1 \\
& & \overline{u_1} &= 2\overline{T} - 2T \\
u_2 &= 2T - \overline{u_1} & \overline{u_2} &= 2\overline{T} - u_2 \\
u_2 &= 4T - 2\overline{T} & \overline{u_2} &= 4\overline{T} - 4T \\
&\vdots & &\vdots \\
u_n &= 2n(T - \overline{T}) + 2\overline{T} & \overline{u_n} &= 2n(\overline{T} - T)
\end{aligned}
\tag{3.4}
$$

□

The first time either $u_n \leq \overline{T}$ or $\overline{u_n} \geq T$, the negotiations end (and no further offers are made). We are interested in the number of rounds it would take to reach agreement in this model, let N_{agree} be the number of this round. In Theorem 2, we see that we can find a closed-form expression for this.

Theorem 2. *In the basic ENO model, negotiations last for N_{agree} rounds, where $N_{agree} = \left\lceil \frac{T}{2(\overline{T}-T)} \right\rceil$*

Proof. We want to find the lowest possible value for $n \in \mathbb{N}$, such that:

$$2n(T - \overline{T}) + 2\overline{T} \leq \overline{T} \quad or \quad 2n(\overline{T} - T) \geq T$$

We can rewrite this as:

$$2n(\overline{T} - T) \geq \overline{T} \quad or \quad 2n(\overline{T} - T) \geq T$$

Or in a single equation:

$$2n(\overline{T} - T) \geq min\{\overline{T}, T\}$$

We know that we can only reach agreement when T is less than $\overline{T}$, so we know that the minimum is attained at T. Furthermore, because $2n(\overline{T} - T)$ is increasing in n, we can find N_{agree} easily:

$$N_{agree} = \left\lceil \frac{T}{2(\overline{T} - T)} \right\rceil \tag{3.5}$$

□

Using Equation 3.5, we can see that the number of rounds before agreement is reached behaves as we would intuitively expect. As we increase our target, we will need more rounds to reach agreement. Also, as the zone of agreement $(\overline{T} - T)$ gets smaller, we will need more rounds to reach agreement. When:

$$\left\lceil \frac{T}{2(\overline{T} - T)} \right\rceil = \left\lceil \frac{\overline{T}}{2(\overline{T} - T)} \right\rceil$$

The offer of the maximizing party in round n will be accepted, otherwise the offer of the minimizing party will be the one that is accepted. Now that we know when agreement is reached, we can use this to determine (the utility of) the point at which agreement is reached, by substituting n in Equation 3.4.

Remark

If both parties use the ENO strategy, the strategies are equivalent to a linear concession strategy where agents concede a fixed amount of utility in each time period. However, there is an important difference between the ENO strategy and the linear concession strategy we saw in the literature review. The strategy we saw in the literature is independent of the offers from the other party, so if the other party deviates from his strategy, we would still make the same concession. The basic ENO strategy *does* depend on the behavior of the other party, reduces to a linear concession strategy if and only if the other party also uses a basic ENO strategy.

There is a small disadvantage to being the first party to make a concession. If we randomly assign targets ($T < 0.5$ and $\overline{T} > 0.5$) and compute the negotiation outcome in the way we described above, we can see at what point in the interval of possible agreements $[T, \overline{T}]$ the outcome lies. The mean point of agreement lies at around $0.45 \cdot (\overline{T} - T)$, so this basic model slightly favors the minimizing party. This effect is explained by the fact that the maximizing party has a disadvantage, because he has to make the first offer.

Limitations

The basic ENO model is very limited. We exactly mirror the behavior of the other agent. For example, if the other agent makes a concession of 0.1, we respond by doing the same. Also, an offer is accepted by an agent whenever it is better than his target, even though he might be able to obtain a better outcome if he would continue to negotiate. To solve these limitations, in the next section, we present a general ENO model that does not have these problems.

4 General ENO model

In this section we will provide a more general ENO model. In this model, we no longer assume negotiations end exactly in the middle of the last two offers. We also provide a method for updating our target during the negotiations, so that we can achieve a better outcome.

Concession rate

We first rewrite the ENO Equation 3.1 to:

$$u_{n+1} = T + (T - \overline{u_n})$$

Now, we can define a more general ENO strategy. Given our target T and the previous offer by the other party $\overline{u_n}$ the ENO strategy selects the next offer u_{n+1} according to:

$$u_{n+1} = T + \gamma(T - \overline{u_n})$$

With some parameter γ that represents the concession rate. The interpretation is that if γ equals 1, then we expect the negotiations to end exactly in the the middle between the last two offers and this model is reduced to the basic model we analysed in the previous section. If γ is large, that means the negotiator will start far away from his target, and make large concessions in each negotiation round. When γ is low, our initial offer will already be close to our target, but we are not willing to concede much throughout the negotiations. This kind of strategy can often be seen when negotiating about the price of a house.

Target adjustment rate

Instead of setting a target once, at the start of the negotiations, we can also dynamically adjust our target to obtain a better outcome. If the other party also uses an ENO model to determine his offers, he will never make an offer equal to his target. This means, that if he makes an offer that exceeds our target (we are once again the maximizing party), we should not directly accept it, because he is still 'willing' to concede even further. In particular, with some parameter δ, we can set our new target to be:

$$T_{new} \quad = \quad \overline{u_{n-1}} + \delta(u_{n-1} - \overline{u_{n-1}}) \tag{4.1}$$

Or, for the minimizing party:

$$\overline{T_{new}} = u_n - \overline{\delta}(u_{n-1} - \overline{u_{n-1}}) \tag{4.2}$$

We update our target whenever Equation (4.22) gives us a T_{new} higher than our old target T. In doing this, we again need to be aware that the same two things might go wrong is we chose δ too large.

The target adjustment rate δ can be seen as the aggressiveness of the negotiator. If δ is low, this means our target will not get updated much. If δ is large however, we update our target very aggressively. In Section 5 we will see that we can expect a better outcome if we employ an aggressive strategy, however this will also lead to longer negotiations. The basic ENO model did not update its target, so it used $\delta = 0$.

When both parties use this model, neither of them will ever accept any offer by the other party. To remedy this, we will say that the negotiations end as soon as the difference between the last two offers is less than some fixed value ϵ, which should be set by the user.

Possible pitfalls

We need to be careful in adjusting our target. In particular, we need to be aware that two things might go wrong:

1. The model might tell us to not make any concession in the next round, so: $u_{n+1} > u_n$
2. There might be no agreements possible with our new target. That is: $T_{new} > \overline{T}$

Both of these are the result of choosing δ too large, or, in other words, updating our target too ambitiously. The first condition is an internal consistence condition. It does not depend on the behavior of the other party. The second one is an external consistency condition, as it depends on the (to us unknown) target of the other party. In the next section we will analyse these conditions in detail.

Example To give some intuition of when these problem occur, consider the following example. We are negotiating about buying an item and we want to pay at most 30 euro. The other party wants to sell it for at least 25 euro (this is of course not known to us). Now assume that in the last round of negotiation, our offer was to pay 16 euro, and their counteroffer was 26 euro and that our δ is $\frac{1}{4}$. Equation 4.23 will tell us to adjust our target to $26 - \frac{1}{4}(26 - 16) = 23.75$ which is less than the target of the seller. Intuitively, this problem occurred because the other party's offer was much closer to their own target than our last offer was, so he conceded much faster than we expected.

4.1 Formal Description of ENO strategy

The ENO strategy uses the modelling parameters γ, δ and T_0, where T_0 is the initial target of the maximizing party. Similarly, the minimizing party has the parameters $\overline{\gamma}, \overline{\delta}, \overline{T}_0$. Furthermore, without loss of generality, we will assume $u_0 = 1$, $\overline{u}_0 = 0$ and that the maximizing party has to make the first concession. In round n, the ENO strategy for the maximizing party will then be to select an offer according to:

$$u_{n+1} = T_n + (T_n - \overline{u_n}) \tag{4.3}$$

$$\text{Where: } T_n = max\{\overline{u_{n-1}} + \delta(u_{n-1} - \overline{u_{n-1}}), T_{n-1}\} \tag{4.4}$$

The ENO strategy for the minimizing party is:

$$\overline{u}_n = T + \overline{\gamma}(\overline{T}_n - u_n) \tag{4.5}$$

$$\text{Where: } \overline{T_n} = min\{u_n - \overline{\delta}(u_{n-1} - \overline{u_{n-1}}), \overline{T}_{n-1}\} \tag{4.6}$$

Negotiations will end in the first round N where either:

$$u_N - \overline{u}_N < \epsilon \text{ or } u_N - \overline{u}_{N-1} < \epsilon \tag{4.7}$$

Here, ϵ is the stopping criterium, which is defined by the user.

4.2 Analytical Results

In this section we will analyse the general ENO model. In particular, we derive conditions for when the model is consistent with the properties of real-world negotiations. We will assume both agents use an ENO strategy. We will derive results and consistency conditions for the parameters of the maximizing agent. Because the minimizing agent also uses an ENO strategy, the conditions for the parameters of the minimizing agent are similar.

Definition 3. A negotiation model is internally consistent when for all rounds n the inequality $u_{n+1} < u_n$ holds.

This condition follows from the requirement that in real-world negotiations, agents make a concession in every round. In this analysis, we will focus on the maximizing agent, but we can easily define a similar condition for the minimizing agent. In the analysis, we will see that the model is internally consistent when:

$$u_1 < 1 \text{ and } \overline{u}_1 > 0 \tag{4.8}$$

and when the following inequality holds:

$$\delta < \frac{1}{1+\gamma} \tag{4.9}$$

Definition 4. A negotiation model is externally consistent when for all rounds n the inequality $T_n < \overline{T}_n$ holds.

This condition ensures that the Zone of Agreement never becomes empty. In the ENO model, agents try to adjust their target according to Equation 4.22. In the analysis, we will see that, in order for the model to be externally consistent, we require that:

$$T_0 < \overline{T}_0 \tag{4.10}$$

and:

$$\delta < \frac{\overline{\gamma}}{1+\overline{\gamma}} \tag{4.11}$$

The following theorem illustrates the importance of external consistency.

Theorem 5. *If a model is not externally consistent, no agreement will be reached.*

Proof. It is trivial to prove this. An agent will only ever accept an offer that lies in the Zone of Agreement. If this zone is empty, because the model is not externally consistent, no agreement can be reached. □

We are now ready to present the main theorem of this section.

Theorem 6. *Whenever the negotiation parameters are such that Equations 4.8,4.9,4.10 and 4.11 hold, the ENO model is internally and externally consistent, and negotiations end in finite time.*

The remainder of this section is dedicated to proving Theorem 6. We will do this by first proving this theorem for a simple case where neither agent updates their target (i.e.$\delta = \overline{\delta} = 0$) and both agents use the same concession rate (i.e. $\gamma = \overline{\gamma}$). We then show that this same results hold for the general cases where both agents do adjust their targets and do not have the same concession rate.

4.3 ENO model analysis: Same concession rate

We will start by looking at the case where both agents use a static ENO strategy with the same concession rate (i.e. $\gamma = \overline{\gamma}$) . That is, they both set a target at the start of the negotiations and accept any offer that is better than their target. This means that $\delta = \overline{\delta} = 0$. Because of this, in order to be externally consistent, we only require that Equation 4.10 holds. The maximizing agent selects their next offer according to equation 4.12 and the minimizing agent selects offers according to equation 4.13.

$$u_n = T_0 + \gamma(T - \overline{u}_{n-1}) \tag{4.12}$$

$$\overline{u}_n = \overline{T}_0 + \overline{\gamma}(\overline{T} - u_n) \tag{4.13}$$

Either agent will accept the offer of the other agent, whenever this offer is better for them than their target. Whenever an offer is accepted, the negotiations end.

We want to show two things. First, we want to show that the model is internally consistent, that is, both agents make a concession in every round. Next, we want to show that the sequences of offers (u_n) and $(\overline{u}_n)$ converge to a value in $[T, \overline{T}]$ (for ease of notation, we will use T and $\overline{T}$ instead of T_0 and $\overline{T}_0$). We can now explicitly write down the offers of both agents for every negotiation round n and derive a closed-form expression for the offers made in round n:

Lemma 7. *When $\gamma = \overline{\gamma}$ and $\delta = \overline{\delta} = 0$*, the offers made in all rounds n ,where agreement is not yet reached, are given by: $u_n = T\sum_{i=1}^{2n}\gamma^{i-1} - \overline{T}\sum_{i=1}^{2(n-1)}\gamma^i$ and $\overline{u}_n = \overline{T}\sum_{i=1}^{2n}\gamma^{i-1} - T\sum_{i}^{2n}\gamma^i$.

Proof. We can prove this by simply applying Equations 4.12 and 4.13 to the offers made in every round.

$$
\begin{aligned}
u_0 &= 1 & \overline{u}_0 &= 0 \\
\\
u_1 &= (1+\gamma)T & \overline{u}_1 &= (1+\gamma)\overline{T} - \gamma u_1 \\
& & \overline{u}_1 &= (1+\gamma)\overline{T} - (\gamma+\gamma^2)T \\
\\
u_2 &= (1+\gamma)T - \gamma\overline{u}_1 & \overline{u}_2 &= (1+\gamma)\overline{T} - \gamma u_2 \\
u_2 &= (1+\gamma)T - \gamma\left((1+\gamma)\overline{T} - (\gamma+\gamma^2)T\right) & \overline{u}_2 &= (1+\gamma)\overline{T} - \gamma\left((1+\gamma+\gamma^2+\gamma^3)T - (\gamma+\gamma^2)\overline{T}\right) \\
u_2 &= (1+\gamma+\gamma^2+\gamma^3)T - (\gamma+\gamma^2)\overline{T} & \overline{u}_2 &= (1+\gamma+\gamma^2+\gamma^3)\overline{T} - (\gamma+\gamma^2+\gamma^3+\gamma^4)T \\
&\vdots & &\vdots \\
u_n &= T\sum_{i=1}^{2n}\gamma^{i-1} - \overline{T}\sum_{i=1}^{2(n-1)}\gamma^i & \overline{u}_n &= \overline{T}\sum_{i=1}^{2n}\gamma^{i-1} - T\sum_{i}^{2n}\gamma^i
\end{aligned}
$$

Which proves the Lemma. □

Before we can move on to the analysis of the model, we will define the concessions $c_n, \overline{c}_n$ of the maximizing resp. minimizing agent in round n as:

$$
\begin{aligned}
c_n &= u_n - u_{n-1} \\
\overline{c}_n &= \overline{u}_{n-1} - \overline{u}_n
\end{aligned} \tag{4.14}
$$

Theorem 8. *Suppose Equation 4.17 holds then, when both agents use an ENO strategy with parameters $\gamma = \overline{\gamma}$ and $\delta = \overline{\delta} = 0$, the model is internally consistent.*

Proof. We will prove this by showing that in each round, both agents make a concession. By using the definition of the concession in each period, we can easily show the following:

$$
\begin{aligned}
c_n &= u_n - u_{n-1} \\
&= T + \gamma(T - \overline{u}_{n-1}) - T + \gamma(T - \overline{u}_{n-2}) \\
&= \gamma(\overline{u}_{n-2} - \overline{u}_{n-1}) \\
&= \gamma(\overline{c}_{n-1})
\end{aligned} \tag{4.15}
$$

Similarly, we can show that:

$$
\overline{c}_n = \gamma(c_n) \tag{4.16}
$$

By our assumption that $u_1 < 1$, we have that $c_1 = u_0 - u_1 > 0$. Because $\gamma > 0$, we know that $\overline{c}_1 > 0$, but this means that c_2 will also be larger than zero. This will continue for every n. □

An interesting result of this theorem is that the only period in which the ENO strategy could fail to suggest a concession is in the first period. From Lemma 7, we can directly see that this condition for internal consistency is equivalent to:

$$(1+\gamma)T < 1 \text{ and } (1+\gamma)\overline{T} - (\gamma+\gamma^2)T > 0 \tag{4.17}$$

Once we are past the first period, the strategy will tell us to make a concession in *every* period after that. The reason the model can fail in the first period (even when $T < \overline{T}$) is that the targets could be too close to the initial offers u_0 or $\overline{u}_0$. If this is the case, the ENO strategy will suggest a concession of *negative* size.

By Theorem 8 we know that the sequence of offers for the maximizing agent is decreasing and the sequence for the minimizing agent is increasing. This is not quite enough to guarantee an outcome of the negotiations, both sequences could converge to a different value. It could be the case that the sequence of the maximizing player converges to some value larger than $\overline{T}$ and that the minimizing sequence converges to a value less than T. Fortunately, in Theorem 9 we prove this will never happen.

Theorem 9. *Suppose parameters are such that Equations 4.10 and 4.17 hold, then when both agents use an ENO strategy with parameters $\gamma = \overline{\gamma}$ and $\delta = \overline{\delta} = 0$, negotiations will end in finite time, for any value of γ. Moreover, the model is internally and externally consistent.*

Proof. We have already seen that Equations 4.10 and 4.17 are sufficient conditions for internal and external consistency. To prove that negotiations also end in finite time, we will split this proof three cases for different values of γ. For $\gamma = 1$, we have already shown this is true in Section 3.4. First, we will look at $\gamma < 1$. To prove negotiations will end, it suffices to show that for some finite value of n, $u_n < \overline{T}$. By applying Lemma 7, we know that:

$$\begin{aligned} u_n &= T\sum_{i=1}^{2n}\gamma^{i-1} - \overline{T}\sum_{i=1}^{2(n-1)}\gamma^{i} \\ &= (T-\overline{T})\sum_{i=1}^{2n}\left(\gamma^{i-1}\right) + (1+\gamma^{2n-1})\overline{T} \end{aligned}$$

So we require that for some n:

$$(T-\overline{T})\sum_{i=1}^{2n}\left(\gamma^{i-1}\right) + (1+\gamma^{2n-1})\overline{T} \quad < \quad \overline{T}$$

Or, by rearranging the terms:

$$(\overline{T}-T)\sum_{i=1}^{2n}\left(\gamma^{i-1}\right) > (\gamma^{2n-1})\overline{T} \tag{4.18}$$

Because $\overline{T} - T > 0$ and $\gamma < 1$, as n increases, the left-hand side of this equation will converge to some positive value. The right-hand side will converge to 0. So clearly for some finite n, Equation 4.18 will hold and negotiations end.

Next, we will look at $\gamma > 1$. We again use the concessions c_n and $\overline{c}_n$ that we defined in 4.14. To show that negotiations must end, it suffices to show that the sequence $(c_1, \overline{c}_1, \ldots c_n, \overline{c}_n)$ is strictly increasing. If both agents make increasingly large concessions, at some point they will "reach" each other. For this, we will again use Equations 4.13 and 4.12:

$$\begin{aligned} c_n &= \gamma(\overline{c}_{n-1}) \\ \overline{c}_n &= \gamma(c_n) \end{aligned}$$

Because $\gamma > 1$, it follows that $c_n > \overline{c}_{n-1}$ and $\overline{c}_n > c_n$, so the sequence $(c_1, \overline{c}_1, \ldots c_n, \overline{c}_n)$ is strictly increasing and we are done. □

4.4 ENO model analysis: Different concession rates

We will now look at the case where both agents have a different values of γ, so now $\gamma \neq \gamma$. By applying the definition of the ENO strategy, we can easily see that Equation 4.8 is now equivalent to:

$$(1+\gamma)T < 1 \text{ and } (1+\overline{\gamma})\overline{T} - \overline{\gamma}(1+\gamma)T > 0 \tag{4.19}$$

We will again show that the ENO strategy will make a concession in every round and that negotiations still end in a finite amount of time. In this more general case, it is trivial to see that Theorem 8 still holds. The only difference is that instead of $\overline{c}_n = \gamma(c_n)$, we now have $\overline{c}_n = \overline{\gamma}(c_n)$. Because $\overline{\gamma} > 0$, the same arguments hold.

Theorem 10. *Suppose parameters are such that Equations 4.19 and 4.10 hold, then when agents use an ENO strategy with parameters γ and $\overline{\gamma}$ and $\delta = \overline{\delta} = 0$, negotiations will end in finite time, for any values of γ and $\overline{\gamma}$. Moreover, the model is internally and externally consistent.*

Proof. The idea is to show that if both agents have a different γ negotiations will end faster than if both agents would use γ^*, where $\gamma^* = min\{\gamma, \overline{\gamma}\}$. To do this, we again look at c_n and $\overline{c}_n$.

$$\begin{aligned} c_n &= \gamma(\overline{c}_{n-1}) \\ &\geq \gamma^*(\overline{c}_{n-1}) \qquad (4.20) \\ \overline{c}_n &= \overline{\gamma}(c_n) \\ &\geq \gamma^*(c_n) \qquad (4.21) \end{aligned}$$

Combining the inequalities in 4.20 and 4.21, we see that when both agents have a different value for γ, negotiations would end *faster* than if both agents would use $\gamma^*$and that in every period, both agents make a concession that is at least as large as when they would both use γ^*. Because of Theorem 9, this directly implies the theorem holds. □

Again, as we have seen in the proof of theorem 9, we require that $u_1 < 1$ and $\overline{u}_1 > 0$. If this requirement does not hold, agents will make concessions of negative size, so instead of converging to an agreement, they diverge and will never reach any agreement.

4.5 General ENO model analysis

Now we are ready to move to the most general case, where agents do update their Targets (i.e. $\delta, \overline{\delta} \geq 0$). Recall that in the previous section, we have defined the way in which an ENO agent updates his target:

$$T_{new} \quad = \quad \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1}) \tag{4.22}$$

Or, for the minimizing agent:

$$\overline{T}_{new} = u_n - \overline{\delta}(u_{n-1} - \overline{u}_{n-1}) \tag{4.23}$$

We first look at the consistency conditions in this general case and then prove the finite time convergence property.

External consistency

Lemma 11. *The general ENO model is externally consistent, whenever the inequality* $\delta < \frac{\overline{\gamma}}{1+\overline{\gamma}}$ *holds.*

Proof. We have already shown the model is consistent whenever targets are not updated, so we only need to look at the rounds in which a target is updated. Let n be any round in which the maximizing agent updates his target. For ease of notation, we will use $\overline{T}$ instead of $\overline{T}_n$. We know that:

$$\begin{aligned} T_n &= \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1}) \\ &= \overline{T} + \overline{\gamma}(\overline{T} - u_n) + \delta\left(u_n - \overline{T} - \overline{\gamma}(\overline{T} - u_n)\right) \\ &= (1 + \overline{\gamma} - \delta - \delta\overline{\gamma})\overline{T} - (\overline{\gamma} - \delta - \delta\overline{\gamma})u_n \end{aligned} \tag{4.24}$$

By our assumption, the following inequality holds:

$$\delta < \frac{\overline{\gamma}}{1 + \overline{\gamma}}$$

We can rewrite this to:

$$0 \quad < \quad (\overline{\gamma} - \delta - \delta\overline{\gamma})$$

We know the other party did not accept our last offer, so $u_n > \overline{T}_n$. This means we can multiply both sides of this equation by $(\overline{T}_n - u_n)$ to obtain:

$$0 > (\overline{\gamma} - \delta - \delta\overline{\gamma})(\overline{T}_n - u_n)$$

We now add $\overline{T}$ to both sides:

$$\overline{T} \quad > \quad (1 + \overline{\gamma} - \delta - \delta\overline{\gamma})\overline{T} - (\overline{\gamma} - \delta - \delta\overline{\gamma})u_n$$

Finally, we note that the right hand side of this inequality is equal to Equation 4.24, which means that $\overline{T} > T_n$, proving the theorem. □

Internal consistency

Lemma 12. *The general ENO model is internally consistent, whenever the inequality $\delta < \frac{1}{1+\gamma}$ holds.*

Proof. We use the definition of the ENO strategy to show this. We only need to show that whenever we update our target, the internal consistency condition holds, because in the previous part, we have shown that it holds whenever we do not update our target.

We can rewrite the offer we make in the next period as:

$$\begin{aligned} u_{n+1} &= T_{new} + \gamma(T_{new} - \overline{u}_n) \\ &= (1+\gamma)\overline{u}_n + (1+\gamma)\delta(u_n - \overline{u}_n) - \gamma\overline{u}_n \\ &= \overline{u}_n + (1+\gamma)\delta(u_n - \overline{u}_n) \end{aligned} \tag{4.25}$$

By assumption, we know that the following inequality holds:

$$\delta < \frac{1}{1+\gamma}$$

Or, equivalently:

$$(1 - \delta - \delta\gamma) > 0$$

Because we know the other agent did not accept our last offer, we have that $u_n > \overline{u}_n$, so this condition reduces so we can multiply by $(u_n - \overline{u}_n)$:

$$(1 - \delta - \delta\gamma)\,(u_n - \overline{u}_n) > 0$$

This is the same as:

$$(1 - \delta - \delta\gamma)\,u_n > (1 - \delta - \delta\gamma)\,\overline{u}_n$$

Which we can rearrange to obtain:

$$u_n \quad > \quad \overline{u}_n + (1+\gamma)\delta(u_n - \overline{u}_n)$$

Here we note that the right-hand side of the equation equals Equation 4.25. Which proves that $u_n > u_{n+1}$ which is the definition of internal consistency. □

The final lemma we need to prove Theorem 6:

Lemma 13. *For any values of the parameters, such that the ENO model is externally consistent, negotiations end in finite time.*

Proof. We prove this by comparing this general model to the case where $\delta = 0$, which we already analysed in Theorem 10. Let $T^\star = sup_n\{T_n\}$ and $\overline{T^\star} = inf_n\{\overline{T}_n\}$. Also, let $u_n^\star$ and $\overline{u}_n^\star$ be the offers made when both parties use a strategy with a fixed Target ($\delta = 0$) equal to $T^\star$ and $\overline{T^\star}$. By definition, we have that for all n, $T_n \leq T^\star$ and $\overline{T_n} \geq \overline{T^\star}$. If we insert these inequalities into Equations 4.12 and 4.13, we can easily see that:

$$T_n + \gamma(T_n - \overline{u}_{n-1}) \leq T^\star + \gamma(T^\star - \overline{u}_{n-1})$$

This means that for all n, $u_n \leq u_n^\star$.Similarly, we can show that $\overline{u}_n \geq \overline{u}_n^\star$. This means that, in the general model where δ and $\overline{\delta}$ do not necessarily equal 0, negotiations end more quickly than when both agents would use strategy with a fixed targets equal to $T^\star$ and $\overline{T^\star}$. By applying Theorem 10, we now know negotiations end in finite time. □

If we combine the last three Lemmas, we have proven Theorem 6.

Conclusion

In this section we have defined consistency conditions. We have shown for which values of the parameters the ENO model is consistent. In the next section, we will perform further numerical analysis on the ENO model.

5 Numerical Analysis

In this section, we will analyze the behavior of the ENO strategy numerically by using a MATLAB simulation.

5.1 Motivation for numerical Analysis

The negotiation outcome is fully determined by the value of all of the parameters $\gamma, \delta, T, \overline{\gamma}, \overline{\delta}, \overline{T}$. Unfortunately, we can not find a nice closed form formula for the negotiation outcome, because in every round, either party might adjust it's target according to 4.22. Let T_n and $\overline{T}_n$ be the current target at round n. Then we have that:

$$T_n = max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} \tag{5.1}$$

The offer at round n is:

$$u_n = T_n + \gamma(T_n - \overline{u}_{n-1})$$

Plugging in the definition of T_n from equation 5.1, we obtain:

$$u_n = max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} + \gamma(max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} - \overline{u}_{n-1}) \tag{5.2}$$

We go one step further, and look at $\overline{u}_n$.

$$\overline{u}_n = \overline{T}_n + \overline{\gamma}(\overline{T}_n - u_n)$$

Using the definition of $\overline{T}_n$ and equation 5.2, we get:

$$\begin{aligned}\overline{u}_n \quad = \quad & min\left\{\overline{T}_{n-1}, u_n - \delta(u_n - \overline{u}_{n-1})\right\} + \overline{\gamma}[min\left\{\overline{T}_{n-1}, u_n - \delta(u_n - \overline{u}_{n-1})\right\} \\ & -max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} \\ & +\gamma(max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} - \overline{u}_{n-1})]\end{aligned}$$

This equation expresses $\overline{u}_n$ in terms of the offers made in round $n-1$:u_{n-1} and $\overline{u}_{n-1}$. We could continue, and express $\overline{u}_n$ in terms of the offers of any previous round, but it is clear that this quickly becomes a mess of nested *min* and *max* operators. So while it is possible to find a closed form equation for the offers in any round, this equation will not be very useful. The best way to evaluate the offers in any round n, is by doing it iteratively, that is, we first compute u_1, then $\overline{u}_1$, etcetera. We implemented the ENO strategy in MATLAB. In the next section we will look at the numerical results of this.

Performance Measures

To measure the performance of a negotiation strategy, we will define two performance measures. Recall that T is the target of the maximizing party and $\overline{T}$ the target of the minimizing party. Let N be the number of the round where agreement is reached and $\overline{u}_N$ the utility of the agreement. The performance measures we are interested in are:

1. Number of the round where agreement is reached: N
2. Point in the Zone of Agreement at which agreement is reached: $R = \frac{\overline{u}_N - \underline{T}}{\overline{T} - \underline{T}}$

By definition, no agent will accept an offer that is worse than their target, so if agreement is reached, it will always lie in the Zone of Agreement $[\underline{T}, \overline{T}]$. This means that R will lie in the interval $[0, 1]$. When R equals one, $\overline{u}_N = \overline{T}$. This is the best possible outcome for the maximizing agent. When R equals zero, $\overline{u}_N = \underline{T}$. This is the best possible outcome for the minimizing agent.

5.2 Performance

We can now look at the performance of our ENO strategy.

Performance against a Random Strategy

We define the following strategy for the minimizing party: Accept any offer for which $u_n \leq \overline{T}$, otherwise offer:

$$\overline{u}_n = \overline{u}_{n-1} + x(\overline{u}_{n-1} - \overline{T})$$

Where x is a random variable distributed uniformly on $[0, 1/2]$. We limit this to half the distance, because we want to guarantee that the dynamic ENO strategy is able to adjust its target without problems, and because we want to prevent the random agent from being able to offer its target in the first round. In Table 1 and Table 2, the results of an ENO strategy with $\gamma = 1, \delta = 1/4$ versus an opponent with a random strategy are displayed. For other values of the parameters, we obtain similar results.

Targets	Mean number of rounds: N	Mean outcome R
$\underline{T} = 0.48, \overline{T} = 0.52$	7.8	0.69
$\underline{T} = 0.48, \overline{T} = 0.57$	5.6	0.69

Table 1: Dynamic ENO ($\gamma = 1, \delta = 1/4$) versus Random, we are the maximizing party. Results are significant at $p < 0.05$

Targets	Mean number of rounds: N	Mean outcome R
$\underline{T} = 0.48, \overline{T} = 0.52$	6.8	0.31
$\underline{T} = 0.48, \overline{T} = 0.57$	5.2	0.30

Table 2: Dynamic ENO ($\gamma = 1, \delta = 1/4$) versus Random. 10000 runs, we are the minimizing party. Results are significant at $p < 0.05$

We see that the ENO agent performs significantly better than the random agent. In Table 1, the ENO agent is the maximizer, so it wants a value of R as close to 1 as possible. In Table 2, the ENO agent is the minimizer and wants a value of R close to 0. Recall that in our model, the maximizing agent is always the first one to make a concession. When we compare the outcome values in both tables, we see that it does not matter whether the ENO agent is the minimizing or maximizing agent. It does influence the number of rounds the negotiations last.

5.2.1 Model Analysis

We will now look at the outcome of negotiations when both agents use an ENO strategy. First, we will fix the targets, and analyze the performance when the agents have different values for γ and δ. Then we will consider the case where both parties have the same parameters γ and δ, but different targets. The goal of these analyses is to provide insight on how the various parameters effect the outcome of the negotiations. Because there are a lot of different parameters, we will only vary them one by one. In all of the plots we present, we have used a step size of 0.01.

Symmetrical case: Variable γ We first look at the symmetrical case. Here both parties will use the same strategy.

In the top graph of Figure 5.1, we can see the effect of the value of γ on the outcome of the negotiations. On average, the negotiations end in the middle of the Zone of Agreement, however we can see that the outcome for different values of γ fluctuates around $R = 0.5$. The main thing to conclude from these fluctuations is that, even if both agents use the same strategy, negotiations do not end always at $R = 0.5$.

Sawtooth This fluctuation results in a sort of sawtooth function. We can explain this by looking at the bottom graph of Figure 5.1. We can see that for values of γ between 0.5 and 1, the outcome is increasing. In this same interval, the minimizing agent is the first to adjust its target. Because both agents have the same parameters, after the first adjustment of the target by either party, from there on, both agents will adjust their target in *every* round by a fixed percentage of the distance between the last two offers. However, the first time the target is adjusted, the size of this adjustment could be less than this fixed fraction. In particular, the minimizing party will adjust its target in the first round, whenever the first offer by the maximizing party is less than 0.8. This corresponds to a γ of less than 1. As γ gets closer to 1, the minimizing party is able to adjust its target less, so he "loses out" more in his first adjustment.

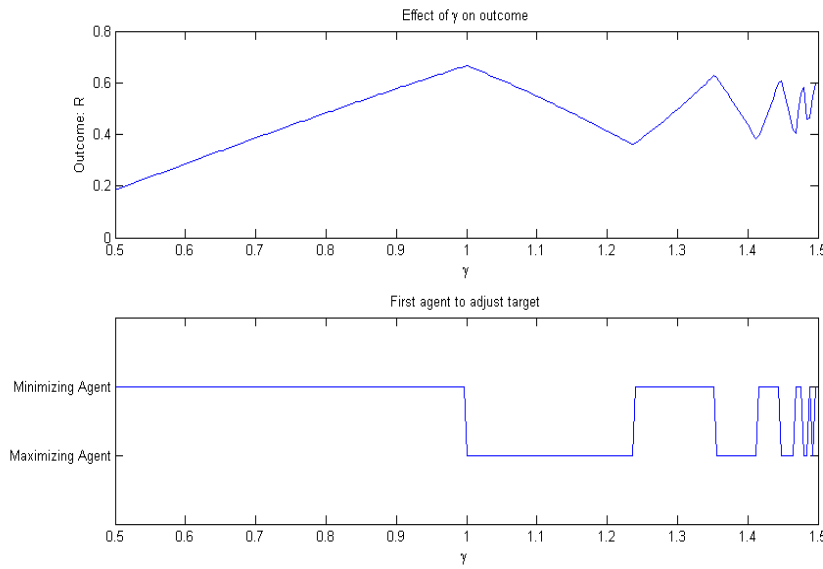

Figure 5.1: Effect of $\gamma = \overline{\gamma}$ on the negotiation outcome.$T = 0.4$,$\overline{T} = 0.6$,$\delta = 1/4$. The top figure shows the effect of γ on the negotiation outcome, the bottom figure shows which agent is the first to adjust his target. (e.g. for $\gamma < 1$, this is the minimizing agent)

In Figure 5.2 we can see the effect of γ on the duration of the negotiations. We note that as γ gets closer to 1.5, the duration of the negotiations increases quickly. This is because we are getting close to the theoretical bound for internal consistency (see previous section) $\delta < 1/\gamma$. We can not increase γ further than 1.5 , because we showed in the previous section that the inequality $(1+\gamma)T < 1$ must hold.

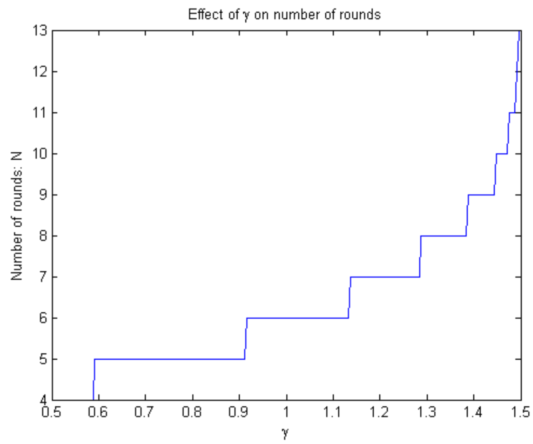

Figure 5.2: Effect of $\gamma = \overline{\gamma}$ on the duration of the negotiations.$T = 0.4$,$\overline{T} = 0.6$,$\delta = 1/4$

Symmetrical case: Variable δ In Figure 5.3 we see the effect of δ on the negotiation outcome. We see a similar pattern to the case of a variable γ. The outcome fluctuates around the value of 0.5. In contrast to the previous case, we now see that the negotiation outcome is increasing whenever the maximizing party makes the first offer. The reason for this is that an increase in δ corresponds to a *larger* first concession rather than a *smaller* concession (which was the case for an increase in γ). We see that as δ gets larger, the fluctuations of the outcome become smaller.

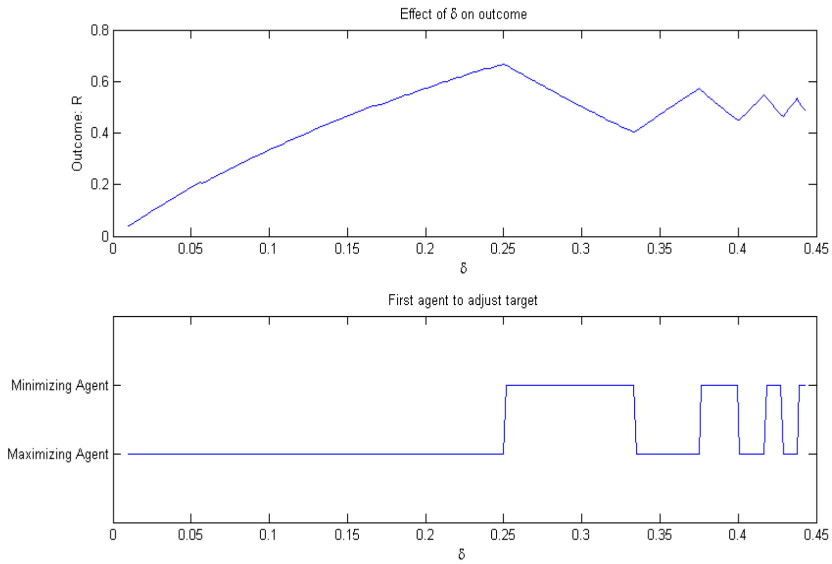

Figure 5.3: Effect of δ=$\overline{\delta}$ on the negotiation outcome.$T = 0.48$,$\overline{T} = 0.52$,$\gamma = 1$. The top figure shows the effect of δ on the negotiation outcome, the bottom figure shows which agent is the first to adjust his target.

In Figure 5.4 we see that the effect of δ on the number of rounds is similar to that of γ. As δ gets closer to its theoretical bound (see previous section), the number of rounds increases quickly.

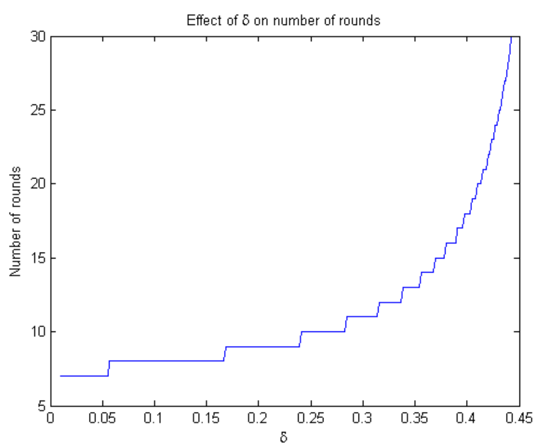

Figure 5.4: Effect of δ=$\overline{\delta}$ on the duration of the negotiation. $T = 0.4$,$\overline{T} = 0.6$,$\gamma = 1$

Variable Zone of Agreement The next set of parameters we can look at is the targets for both agents. First we will look at what happens when we increase the distance between the targets, then we will look at what happens if we move the zone of agreement closer to one of the initial offers. For both cases, the duration of the negotiation behaves as expected. If the zone of agreement is larger, negotiations will end sooner. On the x-axis in Figure 5.5 we have our Target. The Target of the other party is always set to $1 - T$, so the further we move to the right on this axis, the smaller the zone of agreement is. We can see that when both parties use the same γ and δ, there are very large fluctuations of the negotiation outcome. Further inspection tells us that as δ gets larger, these fluctuations become less extreme. Also note that when $\delta = 0$, the peaks lie at $R = 1$ and $R = 0$. So by allowing the agents to adjust their targets, the outcome of the negotiations will be less dependent on the size and location of their initial targets.

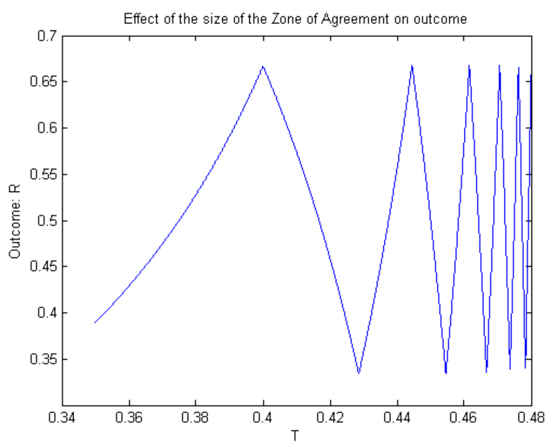

Figure 5.5: Effect of the size of the zone of agreement on the outcome. $\gamma = 1$,$\delta = 1/4$,$\overline{T} = 1 - T$

We also see these large fluctuations if we look at Figure 5.6. Here, the target of the other party equals $T + 0.04$. This means the size of the zone of agreement is constant, but we vary its location. Here, we see that the negotiation outcome depends greatly on the location of the Zone of Agreement.

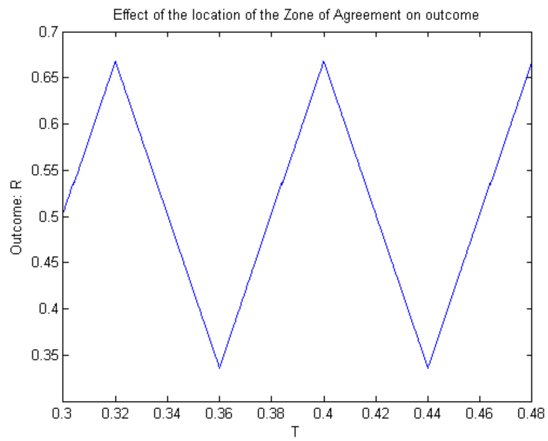

Figure 5.6: Effect of the location of the ZoA on the outcome. $\gamma = 1, \delta = {}^1\!/\!{}_4, \overline{T} = T + 0.04$

Asymmetrical case: γ We now look at the case where agents have a different value of γ. We will keep all other parameters fixed. In Figure 5.7 we see the same peaks that we saw in the symmetrical case (Figure 5.1), but now there is also an upward trend. If γ is larger than $\overline{\gamma}$, we do better than the other agent, if it is less, we will do worse.

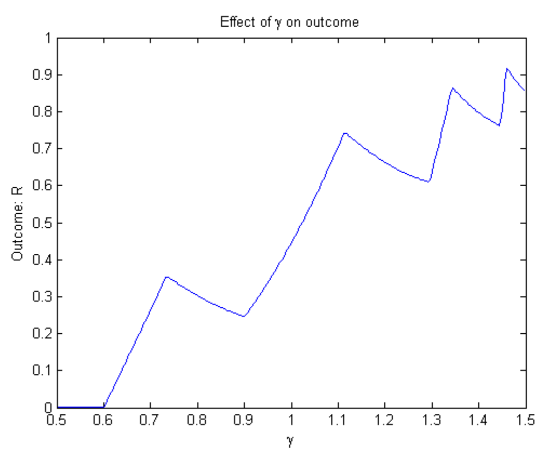

Figure 5.7: Dynamic ENO versus Dynamic ENO, $T = 0.4, \overline{T} = 0.52, \delta = \overline{\delta} = {}^1\!/\!{}_4, \overline{\gamma} = 1$

When we look at the duration of the negotiations in Figure 5.8, we see roughly the same pattern as for the symmetrical case.

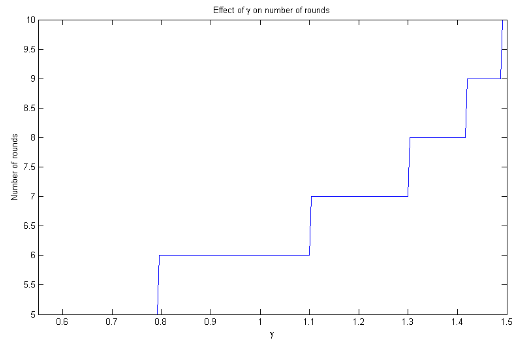

Figure 5.8: Effect of the value of γ on the duration of the negotiation. $T = 0.4$,$\overline{T} = 0.52$,$\delta = \overline{\delta} = 1/4$,$\overline{\gamma} = 1$

Asymmetrical case: δ Finally, we consider the case where both agents have the same γ, but a different δ. In Figure 5.9, we see that this graph is much smoother than the one of γ in Figure 5.7. We also see that if δ is large enough, we are able to obtain the best possible outcome, whereas it is impossible to pick a γ to get the best possible outcome.

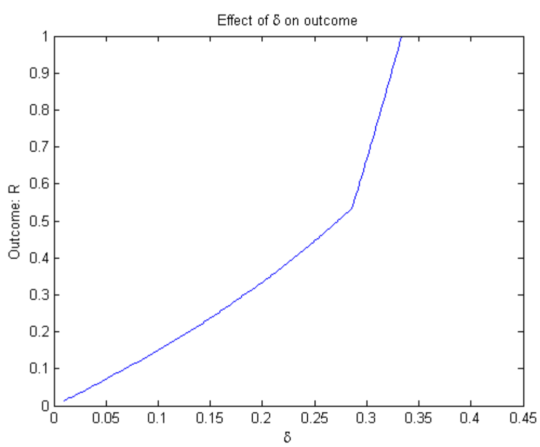

Figure 5.9: Effect of δ on the negotiation outcome. $T = 0.4$,$\overline{T} = 0.52$,$\overline{\delta} = 1/4$,$\gamma = \overline{\gamma} = 1$

In Figure 5.9, we saw that this best possible outcome is obtained for δ larger than 0.325. If we look at Figure 5.10, we see that for this value of δ, negotiations will still end in only 7 rounds.

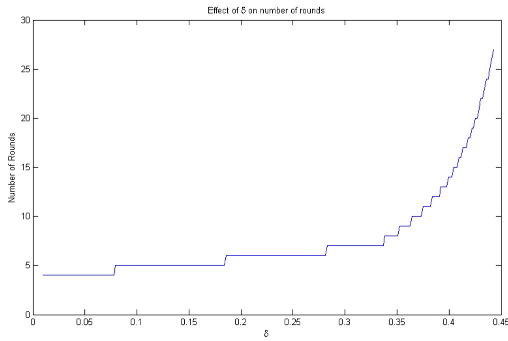

Figure 5.10: Effect of δ on the duration of the negotiation. $T = 0.4$,$\overline{T} = 0.52$,$\overline{\delta} = {}^{1}\!/_{4}$,$\gamma = \overline{\gamma} = 1$

Conclusion

We have seen that increasing γ or δ will yield a better negotiation outcome at the cost of negotiations lasting longer. A decision maker that uses this ENO model will have to make this trade-off.

6 Optimization and Estimation

In this section we will describe a method to estimate the parameters of the other party and optimize our parameters, given this estimate.

6.1 Optimization

If we know the value of the parameters of the other party $\overline{\gamma}, \overline{\delta}, \overline{T}_0$, we want to be able to find the optimal value of our parameters γ, δ, such that we get the best possible outcome in at most N more rounds. It is important that we include the condition that we reach agreement in at most N_{max} rounds, because otherwise the optimal parameter setting would always (trivially) be to set the parameters to their theoretical bounds (See section 4.2 for these bounds). We can write this as a mathematical programming problem, where $\overline{\gamma}, \overline{\delta}, \overline{T}_0, T_0$ are known parameters:

$$
\begin{aligned}
& \max u_{N_{max}} \\
s.t. \quad & \\
& u_n = T_n + \gamma(T_n - \overline{u}_{n-1}) && \forall n \geq 1 \\
& \overline{u}_n = \overline{T}_n + \overline{\gamma}(\overline{T}_n - u_n) && \forall n \geq 1 \\
& T_n = \max\left\{T_{n-1}, \overline{u}_{n-1} + \delta(u_{n-1} - \overline{u}_{n-1})\right\} && \forall n \geq 1 \\
& \overline{T}_n = \min\left\{\overline{T}_{n-1}, u_n - \overline{\delta}(u_n - \overline{u}_{n-1})\right\} && \forall n \geq 1 \\
& u_N - \overline{u}_N < \epsilon \\
& u_n, \overline{u}_n, T_n, \overline{T}_n \in (0,1) && \forall n \geq 1 \\
& \gamma, \delta \in \mathbb{R}^+ \\
& u_0 = 1 \\
& u_1 = 0
\end{aligned}
$$

We can use standard techniques from linear programming to replace the *max* and *min* operators in the restrictions by using additional binary variables and big-M constraints. We are still left with a quadratic term the constraints involving γ and δ. which means that we have a mixed integer quadratically constrained program (MIQCP). We can see this when we look at the constraint in equation 6.1:

$$u_n = T_n + \gamma(T_n - \overline{u}_{n-1}) \quad \forall n \geq 1 \tag{6.1}$$

We multiply the variables γ with the variables T_n and $\overline{u}_{n-1}$, so this is a quadratic constraint. Because this is an equality constraint, this constraint does not represent a convex region. This makes the problem more difficult and makes it impossible to it solve using mixed integer programming solver. However, because all variables are bounded, we will be able to use the non-linear solver BARON [24] to solve this problem.

Effect of $\overline{T}_0$ When we look at the influence of the inital values of the optimization, the first thing we note is that $\overline{T}_0$ only affects the outcome if it is actually used to determine $\overline{u}_1$. It is only used whenever the inequality holds:

$$\overline{T}_0 < u_1 - \overline{\delta}(u_1 - \overline{u}_0) \tag{6.2}$$

If this inequality does not hold, $\overline{T}_0$ has no effect on both the outcome and the number of rounds of the negotiation. Now let us assume inequality 6.2 holds. Then we can look at the effect of a change in $\overline{T}_0$.

Effect of N_{max} We see that for low values of N_{max}, there is no parameter setting that obtains a good outcome. Because we cannot negotiate for more than N_{max} rounds, the only way to reach agreement is set our parameters to 0. This means that we will instantly make an offer that equals our target and then make the same offer in all the remaining rounds. When N is large, we can obtain the 'optimal' outcome of $\overline{T}_0$. Increasing N_{max} beyond this point will not yield a better outcome, because the other party will never accept any offer worse than it's target.

6.2 Estimation of Parameters

The ENO strategy we have looked at so far only uses the last two offers (and our target) in determining the next action. In this section we will estimate the parameters of the other party, based on the whole history $(u_0, \overline{u}_0, \ldots, u_{n-1}, \overline{u}_{n-1})$ of the negotiations. We will then combine this estimate with the QCP optimization we described in the previous section to improve our parameters γ and δ.

6.2.1 Opponent model

For estimating the parameters of the other party, we will assume he uses an ENO strategy with unknown parameters $\overline{\gamma}, \overline{\delta}, \overline{T}_0$. If he follows the ENO strategy exactly, we can easily deduce his parameters from his offers. In practice, people will most likely not follow the ENO strategy exactly. This can be due to various reasons, for example:

- Agents might round their offer up or down
- Agents might use heuristic guidelines (that only somewhat resembles an ENO strategy) instead of this strategy

To estimate the parameters of the other party, we will assume he does not change his parameters during the course of the negotiations. Let n be the current round. To be able to use the QCP optimization procedure, we need to determine $\overline{\gamma}, \overline{\delta}$ and $\overline{T}_n$. We observe the following behavior of the model in all simulations:

Conjecture 14. *Let N be the first round in which $T_N > T_{N-1}$, then for every $n \geq N$, we have $T_{n+1} > T_n$.*

While we have been unable to find an analytical proof for this conjecture, it holds for every negotiation we simulated and analysed throughout this research.

According to this conjecture, we can split the negotiation history in two parts. Before round N, the other party *never* updates his target, after round N, he updates his target in *every* round. Because we have no information other than the negotiation history, we will also need to determine the value of N from the negotiation history.

Before updating the target In the rounds where the other party does not update his target, the ENO model says he should offer:

$$\overline{u}_n = \overline{T}_n + \overline{\gamma}(\overline{T}_n - u_n)$$

We can separate the parameters we want to estimate from the input:

$$\underbrace{\overline{u}_n}_{y} = \underbrace{(1+\overline{\gamma})\overline{T}}_{\alpha_1} - \underbrace{\overline{\gamma}u_n}_{\alpha_2 x} \tag{6.3}$$

We can use equation 6.3 to apply Linear Least Squares regression [20]. This means we can get a the estimate for $\overline{\gamma}^{est}$ according to:

$$\overline{\gamma}^{est} = \frac{Cov(x,y)}{Var(x)}$$

Furthermore, we can get a $(1-\alpha)$ confidence interval for $\overline{\gamma}^{est}$ as:

$$[\overline{\gamma}^{est} - s_{\overline{\gamma}^{est}}t^*_{n-2}, \overline{\gamma}^{est} + s_{\overline{\gamma}^{est}}t^*_{n-2}]$$

Where $s_{\overline{\gamma}^{est}}$is the standard error of the estimated value of $\overline{\gamma}$, and t^*_{n-2} is the $(1-\alpha/2)$'th quantile of the Student's t-distribution with $n-2$ degrees of freedom. Here we note that we need at least 3 data points before we are able to provide this confidence interval.

After updating the target In the rounds where he does update his target, he will make offers according to:

$$\overline{u_n} = u_n - \overline{\delta}(u_n - \overline{u}_{n-1}) - \overline{\gamma}\overline{\delta}(u_n - \overline{u}_{n-1})$$

This is equivalent to:

$$\overline{u_n} = \underbrace{(1 - \overline{\delta} - \overline{\delta\gamma})u_n}_{\eta u_n} + \underbrace{(\overline{\delta} + \overline{\delta\gamma})\overline{u}_{n-1}}_{\theta\overline{u}_{n-1}}$$

Unfortunately, we can see in this last equation that $\eta = 1 - \theta$. This means that we cannot solve for both $\overline{\delta}$ and $\overline{\gamma}$ from the estimates of η and θ. So we have:

$$\overline{u}_n = \theta u_n + (1-\theta)\overline{u}_{n-1} \tag{6.4}$$

Recall from Section 4.2 the definition of the concession (c_n and $\overline{c}_n$) in round n:

$$\begin{aligned} c_n &= u_n - u_{n-1} \\ \overline{c}_n &= \overline{u}_{n-1} - \overline{u}_n \end{aligned}$$

Then we can rewrite equation 6.4 as:

$$\underbrace{\overline{c}_n}_{y_n} = \theta \underbrace{(u_n - \overline{u}_{n-1})}_{x_n} \tag{6.5}$$

We can use equation 6.5 to apply Linear Least Squares regression with a zero value for the intercept term. [20]:

$$\overline{\theta}^{est} = \frac{\sum_{i=1}^{n}(u_n - \overline{u}_{n-1})\overline{c}_n}{\sum_{i=1}^{n}(u_n - \overline{u}_{n-1})^2}$$

And the $(1-\alpha)$ confidence interval is:

$$[\overline{\theta}^{est} - s_{\overline{\gamma}^{est}} t^*_{n-1}, \overline{\theta}^{est} + s_{\overline{\gamma}^{est}} t^*_{n-1}]$$

Where $s_{\overline{\gamma}^{est}}$is the standard error of the estimated value of $\overline{\theta}$, and t^*_{n-1} is the $(1-\alpha/2)$'th quantile of the Student's t-distribution with $n-1$ degrees of freedom. Here we note that we need at least 2 data points before we are able to provide this confidence interval.

6.2.2 Estimation Procedure

If we would know the first round in which the other party updates his target (N), we could split the negotiation history into two parts. For all the offers in rounds before N, we find a Linear Least Squares (LLS) fit of equation 6.3. For all offers in rounds after N, we find a Linear Least Squares (LLS) fit for equation 6.5. We then find $\overline{\gamma}$ from α_2 and $\overline{\delta}$ from θ. The last parameter we need to find to be able to run the optimization is $\overline{T}_n$ and we can find its value by plugging in the estimated value of $\overline{\delta}$ in $\overline{T}_n = u_n - \overline{\delta}(u_n - \overline{u}_{n-1})$.

Of course, we do not know the round in which the other party first updates his target, but there are only n (recall that n is the current round) possible values for it. So instead of running this estimation procedure once, we run it n times, for n different values of N and look at which model is the best fit for the negotiation history. The best fit is the set of parameters having the lowest squared difference between the predicted offers (by using an ENO strategy with these parameters) and the actual offers. In pseudo code, this looks like algorithm 1.

Algorithm 1 Parameter Estimation

```
Input: Negotiation history
for i=1:LastRound do
        (a_1,a_2)=LLS(all rounds before i)
        (θ)=LLS(all rounds after i)
        Compute \overline{γ_i} and \overline{δ_i} from (a_2, θ)
        R_i=Squared error with (\overline{γ_i},\overline{δ_i})
endfor
N=argmin_i{R_i}
Output: N, γ_N, δ_N
```

7 Simulations

In this section we present the results of numerical simulations in MATLAB. The goal of these simulations is to evaluate the performance of the estimation and optimization method.

7.1 Optimization and estimation

In algorithm 2 we describe the procedure to for optimizing our parameters. In every round, we obtain a new estimate for the parameters of the other party. We then use those parameters as input for the optimization to find better parameters for ourselves.

Algorithm 2 Optimization and Estimation

```
input: History, N_max, n
if 2<n<N_max
        (γ̄, δ̄, T̄)=EstimateParameters(History)
        (γ, δ)=OptimizeParameters(γ̄, δ̄, T̄, N_max−n)
end
u_n=MakeOffer(γ, δ, T, u_{n−1})
output: u_n, γ, δ
```

We again note that before we can get an estimate for all relevant parameters, we need information from at least three previous rounds. The algorithms for the functions EstimateParameters and OptimizeParameters are described in the previous section.

To test the effect of estimating and optimizing the parameters, we use simulations. In these simulations, we will use an ENO strategy with algorithm 2 to update our parameters in every round. The other party uses an ENO strategy with fixed parameters. We will also provide results for the case where the other party uses an ENO strategy with fixed parameters, but with added random noise.

7.2 Implementation

We first go over some of the details of the implementation.

Numerical Optimization There is no non-linear solver available in MATLAB, so instead we do the optimization by a numerical grid search. This means that we simply try every possible value of γ and δ with a stepsize of 0.01 and see which gives the best outcome. This optimization takes 2-5 seconds in the MATLAB implementation, and is the biggest bottleneck in terms of computational power needed for the simulations.

Parameter Selection procedure For every simulation run, we need to select parameters for both parties. We do this by randomly drawing them from the following uniform distributions:

- $\gamma, \overline{\gamma} \sim U(0.5\,,\,2)$
- $\delta, \overline{\delta} \sim U(0\,,\,0.4)$
- $T \sim U(0.25\,,\,0.49]$
- $\overline{T} \sim U(T+0.01\,, T+0.3)$
- $N_{max} \sim U_{discrete}(5\,,\,10)$

When we have selected all parameters, we first run a simulation of the negotiations where neither party updates his target. If these negotiations are too short (less than 3 rounds) or too long (more than 15 rounds) we delete them, because real-world negotiations are expected to last between 5 to 10 rounds, and we are interested in the performance of our strategy in real-world settings.

Estimation and Optimization procedure

We now present the details of the implementation of the estimation and optimization procedure.

Insufficient information on $\overline{\delta}$ If the other party has not updated his target when we estimate his parameters, we cannot get any information on the value of this δ. Because we do need this parameter for the optimization procedure, we need to make an assumption on it. In the simulations, we will determine which of the following two assumptions is best:

- $\overline{\delta} = 0$
- $\overline{\delta} = \delta$

Insufficient information on $\overline{\gamma}$ We have seen that, if the other updates his target in the first round, we cannot obtain an estimate for $\overline{\gamma}$. When we do not have enough information, we will assume $\overline{\gamma} = 1$.

N_{max} **is too low** In the optimization, there might be *no possible* value for our parameters that will ensure agreement before N_{max}. If this is the case, we will run the optimization again for $N_{max}+1$, and if needed again for $N_{max}+2$, up to $N_{max}+5$. This will make sure that the optimization will always give an output such that agreement is reached.

Not updating target in every round The estimation procedure we described in Section 6 is based on the fact that, once the other party updates his target, he will keep updating it in every next round. However, this only happens when *both* parties use fixed parameters. Because we can now change our parameters during the course of the negotiation, this theorem no longer holds. If there is not much variance in the parameters of the other party, we likely have a very good estimate of their parameters once they update their target for the first time. So in every round, instead of only making a new estimate of their parameters, we also compare this new estimate to our previous best estimate. In pseudo code, our estimation algorithm is displayed in Algorithm 3. If we compare this to the estimation algorithm we presented in Section 6, we see that the only diffences are the second and third line.

Algorithm 3 Implementation of Estimation Procedure

Input: Negotiation history , previous Estimate
$(\overline{\gamma_0}, \overline{\delta_0}) = (\overline{\gamma}_i^{old}, \overline{\delta}_i^{old})$
R_0=Squared Error with $(\overline{\gamma_0}, \overline{\delta_0})$
for i=1:LastRound do
    (a_1, a_2)=LLS(all rounds before i)
    (θ)=LLS(all rounds after i)
    Compute $\overline{\gamma_i}$ and $\overline{\delta_i}$ from (a_2 , θ)
    R_i=Squared error with $(\overline{\gamma_i}, \overline{\delta_i})$
endfor
N=$argmin_i\{R_i\}$
Output: N, γ_N , δ_N

7.3 Results

For all simulations, we used a stopping criterion of 0.001. This means that whenever $u_n - \overline{u}_n < 0.001$, the negotiations end in agreement. All of the simulations contain 5000 runs. 5000 runs take roughly 24 hours to complete. All of the results are presented along with one standard deviation.

To compare the outcomes between different negotiations, we again use the following performance measure:

$$R = \frac{\overline{u}_N - T}{\overline{T} - T}$$

Recall that this means that R will lie in the interval $[0, 1]$. When R equals one, $\overline{u}_N = \overline{T}$. This is the best possible outcome for the maximizing agent. When R equals zero, $\overline{u}_N = T$. This is the best possible outcome for the minimizing agent.

Deterministic opponent

In table 7.3 we can see the main results of the simulation against a deterministic opponent. We aborted a run whenever the negotiations took longer than 50 rounds (to save time). Any negotiation that lasts over 50 rounds is called a failure. There are two main reasons negotiations could fail:

1. Our target is updated too aggressively ($T_n > \overline{T}_n$).
2. We do not get information on $\overline{\delta}$, causing us to underestimate the aggressiveness of the opponent.

The first failure will only happen when we do not have enough information on $\overline{\gamma}$. In the cases where we do not get an estimate for $\overline{\gamma}$, we cannot guarantee to satisfy the external consistency condition we derived in section 4.2. The second failure is due to not having a good estimate on $\overline{\delta}$. This can happen when he first updates his target after N_{max}, causing us to not take the exact value of $\overline{\delta}$ into account when optimizing our parameters. In this case, if we were to let the negotiations run for more than 50 rounds, eventually agreement will be reached, but it might take a very long time. Inspection of the simulation data tells us that the majority of the failures are of type 2. A failure is counted as a negotiation with $R = 0$.

	Outcome: R	# Rounds	# failures
No Optimization	0.500±0.005	6.5±0.2	0
Optimization with initial $\overline{\delta} = 0$	0.561±0.006	7.9±0.5	367
Optimization with initial $\overline{\delta} = \delta$	0.580±0.005	7.9±0.5	226

Table 3: Effect of the optimization. Failures are counted as $R = 0$. Results are 95% confidence intervals of the mean values over 5000 runs.

For the negotiations in first row in Table 7.3 we disabled the estimation/optimization procedure. We do not change our parameters during the course of these negotiations. We simply randomly chose them once and then keep them fixed. We use the 'No Optimization' scenario as a base case to evaluate the performance of the estimation/optimization procedure.

We see than on average, when no optimization is used, the result is at $R = 0.5$. This means that there is no advantage in being the first (or second) to make a concession. We see that, if we assume $\overline{\delta} = \delta$ whenever there is insufficient historical information to provide a good estimate leads to a better outcome and less failures. We will continue to use the initial assumption of $\overline{\delta} = \delta$ for the remainder of the simulations. Another observation is that when we use the optimization procedure, negotiations will last longer than without it.

A more detailed look at the data tells us that the optimization procedure improves the negotiation outcome in 70% of the simulated negotiations.

Analysis of simulation outcome

In the next analysis, we will look at the effect of the various parameters on the improvement of the outcome. This is the improvement that is gained by using the estimation and optimization procedure described in Algorithm 2, over having fixed parameters during the whole negotiations.

7.3.1 Effect of N_{max}

In figure 7.1 we can see the effect of N_{max} on the improvement of the outcome.

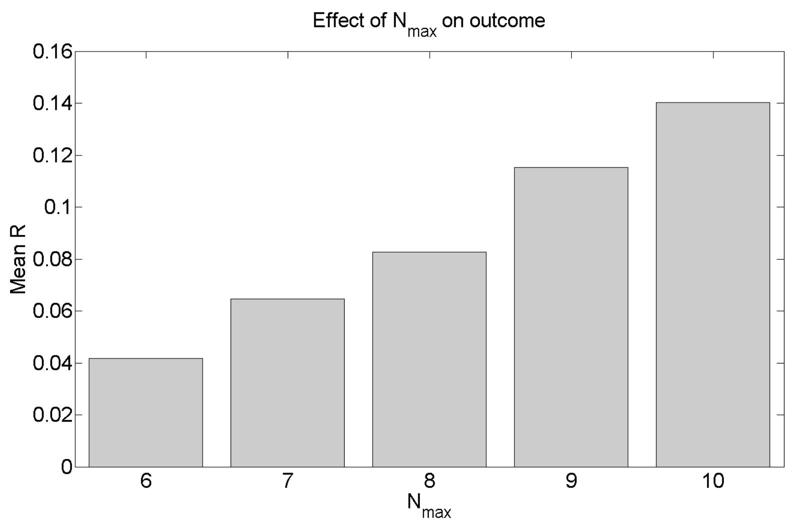

Figure 7.1: Effect of the maximum number of allowed rounds on the outcome.

We see that if we allow the negotiations to last for more rounds, we can expect a better outcome. Figure 7.1 suggests a linear relationship between the outcome and N_{max} in these simulations.

7.3.2 Effect of the size of the Zone of Agreement

Recall that the zone of agreement (ZoA) is defined as $[T, \overline{T}]$. An agreement will always lie within this interval. In figure 7.2.

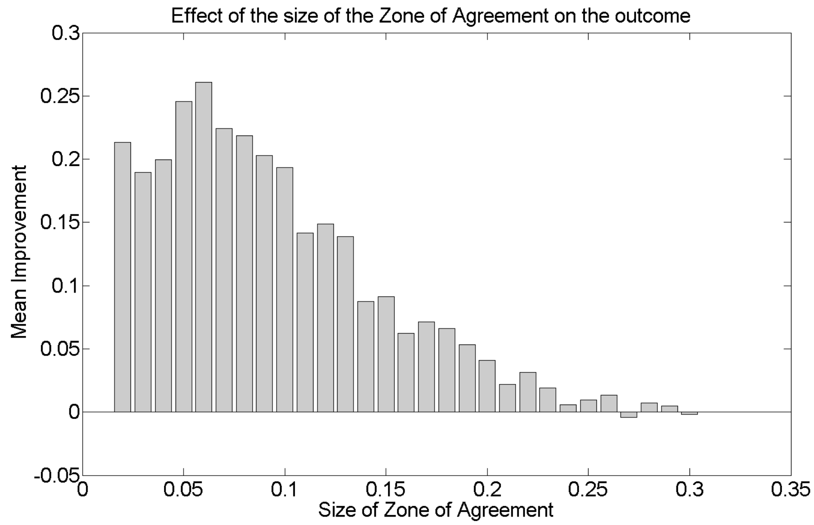

Figure 7.2: Effect of the zone of agreement on the outcome. Step-size bars: 0.01

We clearly see that as the Zone of Agreement gets larger, we get less improvement from our optimization. If both parties are already close to agreement, we cannot to gain much from optimizing. This is what happens when the ZoA is large: both parties will make large concessions early in the negotiations and will already be close to agreement when we start optimizing. Because we start running the optimization at the third round, there will be limited room for improvement.

7.3.3 Effect of δ and $\overline{\delta}$

In the next two figures, we will look at the effect of δ and $\overline{\delta}$ on the average improvement.

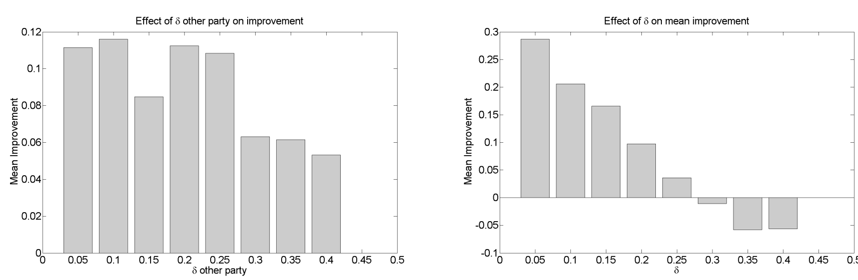

Figure 7.3: Effect of $\overline{\delta}$ and δ on the improvement.

We see that we can obtain the best improvement if either δ, or $\overline{\delta}$ is small.

If either is large, there is much less to be gained by running the optimization. When δ is large, we will often not be able to improve much, because we require that the negotiations end before N_{max}. Also, for large $\overline{\delta}$, the other party will have often updated his target before we are able to get a good estimate of his parameters and optimize our parameters. Because of this, a large part of the Zone of Agreement will not be reachable anymore, so we cannot expect a large improvement.

7.3.4 Effect of γ and $\overline{\gamma}$

There is no effect of either γ or $\overline{\gamma}$ on the mean improvement. In figure 7.4, the value of γ is plotted against the mean improvement.

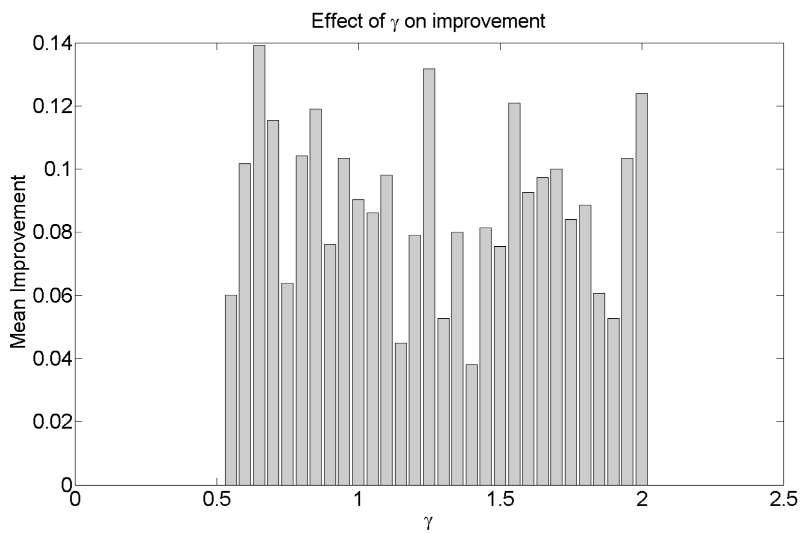

Figure 7.4: Effect of γ on the improvement.

When we look at Figure 7.4, we see that there is no connection between γ and the mean improvement. For $\overline{\gamma}$ we get the same result.

Semi-random opponent

We model a semi-random opponent as follows. Similar to the deterministic case, we draw random values for his parameters $\overline{\delta}, \overline{\gamma}, \overline{T}$ at the start of the negotiations. Then, in every round, instead of using these fixed parameters, we draw a new $\overline{\gamma_i}$ from a uniform interval: $(\overline{\gamma} - \epsilon_\gamma, \overline{\gamma} + \epsilon_\gamma)$ and $\overline{\delta}_i$ from $(\overline{\delta} - \epsilon_\delta, \overline{\delta} + \epsilon_\delta)$. So every round the opponent gets a new set of parameters. We can control the variance by the value of ϵ. In Table 7.3.4, we see the results of an ENO strategy without the optimization/estimation procedure. In Table 7.3.4, the ENO agent does optimise his parameters. We see that a fixed ENO strategy does slightly better against a semi-random strategy than against a fixed ENO strategy.

	R	# Rounds	# Fails
Low Variance:$\epsilon_\delta = \epsilon_\gamma = 0.05$	0.525±0.003	6.9±0.1	0
High Variance:$\epsilon_\delta = 0.1$,$\epsilon_\gamma = 0.2$	0.526±0.002	6.2±0.1	0

Table 4: Effect of the optimization. R is the outcome *without* optimization procedures. Results are 95% confidence intervals of the mean values over 5000 runs.

	R	# Rounds	# Fails
Low Variance:$\epsilon_\delta = \epsilon_\gamma = 0.05$	0.631±0.003	7.5±0.3	10
High Variance:$\epsilon_\delta = 0.1$,$\epsilon_\gamma = 0.2$	0.601±0.003	6.9±0.3	2

Table 5: Effect of the optimization. R is the outcome *with* optimization procedures. Results are 95% confidence intervals of the mean values over 5000 runs.

In both cases, without optimization, $R = 0.50$. This means that a fixed ENO strategy, has the same performance against a semi-random ENO strategy as it has against a fixed ENO strategy. We see the best performance for the optimization procedure in the low variance case. By comparing the results in Table 7.3.4 and 7.3.4 we see a significant improvement is gained by using the optimization method. Another interesting thing to note is that there are less failures against semi-random opponents than there were against deterministic opponents. This is because, even if we get a bad estimate on their parameters, the other party can still randomly make a concession, speeding up the process of reaching an agreement. We see this speedup in the data when we look at the average number of rounds. As we increase the variance, we reach agreement more quickly.

7.3.5 Conclusion

We have seen that using the estimation and optimization procedure produces better results than not using it, against both semi-random and deterministic behavior of the other party. We therefore recommend using this in the negotiation support system.

8 Extention: Multiple Dimensions

In this section we will discuss the application of the ENO model to multiple dimensions. We negotiate about a contract with m issues: $x_1, \ldots, x_m$. We want to maximize all issues, and on all of these dimensions, our initial offer is 1. The other party's initial offer is 0 on all dimensions, and he wants to minimize all issues.

Independent utilities

The easiest case is when the targets for the various dimensions are independent. This means that we have a target of the form: $x_1 = T_1, \ldots, x_m = T_m$. Where T_i is the target for dimension i. In figure 8.1 we see an example of two dimensions.

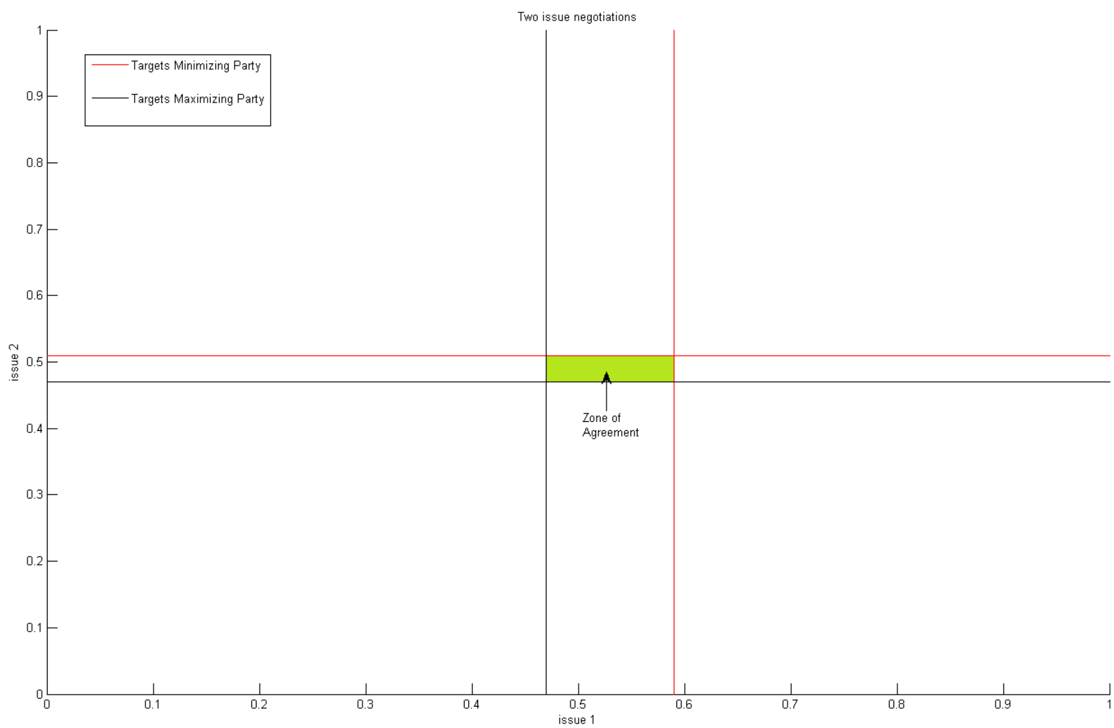

Figure 8.1: Both parties have independent targets on all dimensions. The zone of agreement is the colored rectangle.

When we have independent issues, we can directly apply the ENO model, because we can evaluate an offer on each dimension separately. Updating a target on one of the dimensions does not impact the targets on the other dimensions, so we do not run into any problems.

The user of the model will be able to set his negotiation parameters γ_i and δ_i for each dimension. By doing this, he can assign importance to each of the dimensions. For example, if he decides issue i^* is most important, he can set δ_i^* to a high value. He will then negotiate more aggressively on dimension i.

The advantage of having independent dimensions, is that it is very easy for negotiators to determine targets independently and that they often do so already in practice. The disadvantage is that they are unable to express relationships between the different dimensions. However, we can often overcome this by aggregating several dimensions. We can use aggregation whenever there is an objective relation between two negotiation dimensions. For example, suppose we negotiate about the sale of a product, we have to agree upon a (one-time) purchase cost and a (monthly) maintenance cost. The user of the model wants to minimize the total cost over the duration of the contract. Instead of treating these two issues as separate dimensions, we can compute the total cost for any offer made by the other party and use the single negotiation dimension "total cost" for the ENO model. Figure8.2 displays an example of negotiations among two independent dimensions.

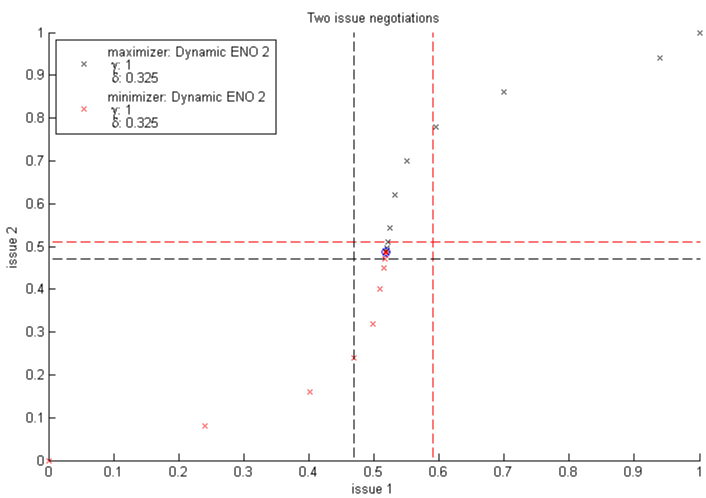

Figure 8.2: Example of a negotiation outcome. Offers are depicted as crosses.

Linearly dependent utilities

When we are dealing with multiple issues that are not independent, we need to be able to evaluate offers. The standard way of doing this is by using a utility function $f(x_1, \ldots, x_m) \rightarrow [0, 1]$. The other party will have his own (different) utility function $\overline{f}(x_1, \ldots, x_m)$.

Each party's target can be specified as a fixed value of their utility function. So we have $f(x_1, \ldots, x_m) = T$ and $\overline{f}(x_1, \ldots, x_m) = \overline{T}$ as the targets. We will first restrict ourselves to utility function of the form:

$$f(x_1, \ldots, x_m) = \sum_{i=1}^{m} \lambda_i x_i$$
$$\text{where: } \sum_{i=1}^{m} \lambda_i = 1$$

So the total utility of an offer is a linear combination of the dimensions. This means that the user can fully specify his utility function by assigning weights to each dimension. In figure 8.3 we see an example of this in two dimensions.

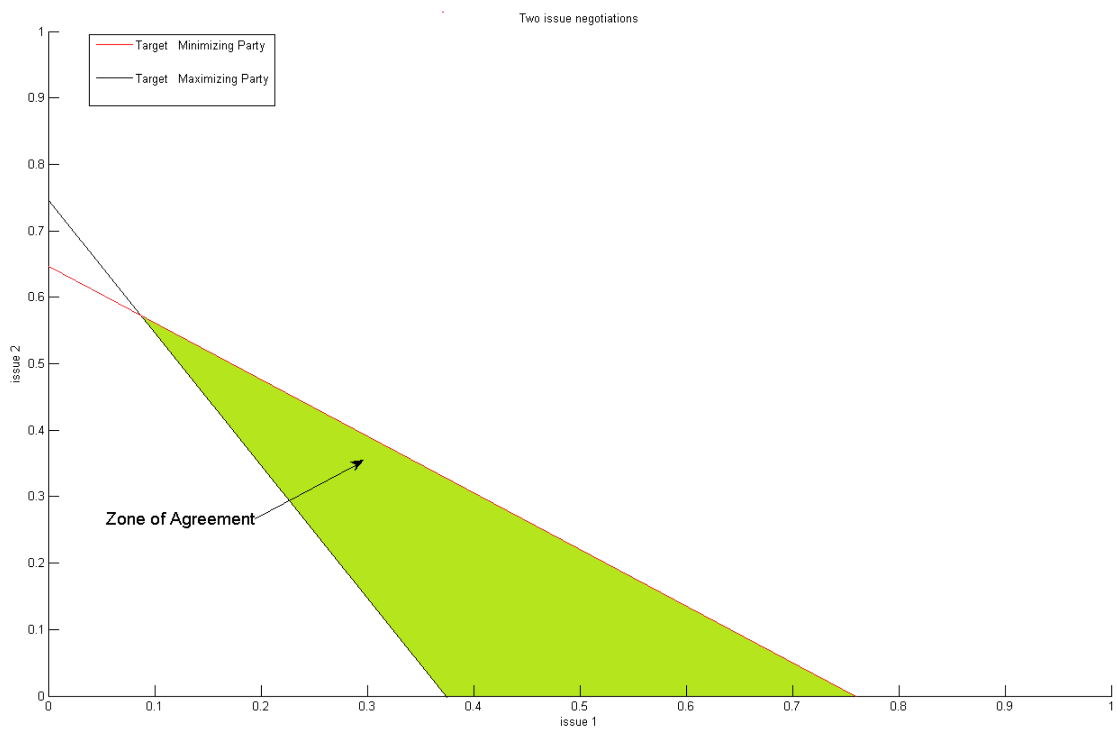

Figure 8.3: Placeholder 2

To apply the ENO model to a setting with linearly dependent utilities, we need to do two things:

1. Select the utility of our next offer
2. Out of all offers with this utility, select the best one

We would like to use the ENO strategy to select the utility of our next offer. Recall that this would mean we select our next offer according to:

$$f(u_{n+1}) = T + \gamma(T - \overline{f(\overline{u_n})})$$

Here, $u_{n+1}$and $\overline{u}_n$ are the offers. However, because both parties use different utility function, we cannot do this. T is defined in terms of our utility function, but $\overline{f(\overline{u_n})}$ lives in the utility space of the other party. If we were to subtract the two, we would be comparing apples and oranges. If the utility functions of both parties are similar (i.e. the weights for all dimensions are similar), we could substitute $\overline{f}(\overline{u}_n)$, by $f(\overline{u}_n)$.

The next step is selecting an offer, Lai et al. [13] have shown that selecting an offer closest to the previous offer by the other party leads to a negotiation outcome that is close to Pareto optimal.

$$\begin{aligned} & min \quad ||u_{n+1} - \overline{u}_n|| & (8.1) \\ s.t. \quad & f(u_{n+1}) = T + \gamma(T - f(\overline{u_n})) & (8.2) \\ & u_{n+1} \in X \end{aligned}$$

It is beyond the scope of this research to examine the case of linear dependent utilities in more detail. It would, however, be an interesting direction for future research. In particular, we would like to know whether selecting an offer according to 8.2 yields a good solution.

Non-linear utilities

The last option is to use non-linear utilities. To extend the ENO model, we would need to first determine a function for our utility. While there are methods to determine a linear utility function (e.g. SMART [18]) and methods that compare multiple options without explicitly constructing a utility function (e.g. AHP [18]), there are no methods to construct general non-linear utility function for the whole negotiation space. It is beyond the scope of this research to develop such a method.

Conclusion

We conclude that it is far from trivial to extend the ENO model to multiple dimensions with dependent targets. The most natural extension is to use linearly dependent utilities, however more research should be done before we can apply the ENO model to this setting.

9 Conclusion

In this research, we developped a new mathematical model for contract negotiation. We did this by answering the following research questions:

- How can we model contract negotiations mathematically?

We looked at various negotiation models from the literature. We have seen that these models have several limitations that make them unusable for a Negotion Support System. In particalar, these models use the unrealistic assumption of perfect rationality of the agents, or require a large amount of historical data to function properly. The ENO model we develop in this research solves these problems.

- How can we measure the performance of our model?

We have done this in three steps. We first defined consistency conditions and have shown, analytically, for which values of the parameters the model meets these requirements. We then defined a measure with which we can compare the outcome of different negotiations. By numerical analysis, we determined the effect of the model parameters on the outcome and duration of the negotiations. Finally, we performed simulations to analyse the performance of the optimization procedure.

- How can we optimize the negotiation strategy of the user?

In Section 6 we describe a method for optimizing the strategy parameters during the negotiation. It first estimates the negotiation parameters of the other agent based on the previous offers he made during the negotiations. By performing simulations, we have seen that applying this procedure, significantly improves the negotiation outcome, even when the other agent does not follow an ENO strategy exactly.

Future Research

There are many possible directions for future research. We will briefly go over the most interesting ones in this section.

- Can we determine a good strategy at the start of the negotiations?

The current optimization procedure provides a suggestion for our δ and γ, given the offers made by the other agent so far. We would also like to create a method to suggest a strategy at the start of the negotiations. This means that if we have historical data at the start of the negotiations, we would like to also be able to suggest an initial offer and initial target.

- How does the ENO strategy perform against real people?

It would be interesting to verify the ENO model in negotiations between real people. How does the strategy we described in this research do against actual people?

- How can the model be extended to (linearly) dependent dimensions?

As we have seen in Section 8, there is still work to be done whenever the negotiation dimensions are dependent. In particular, linearly dependent dimensions seem realistic, but we need some modifications and additions to the ENO model before it can be applied to linearly dependent dimensions. In particular, a method to estimate the weights of the other party should be developed.

- How should we adjust the ENO model when the other agent does not use a fixed strategy?

Throughout this research we assumed the other party uses a strategy with fixed parameters. If the other party does not have fixed parameters (because he might also try to optimize his parameters after every round), how should we adjust our strategy? The optimization method we described will probably need to be adjusted in this case.

Conclusion

The ENO model is a more realistic model of negotations than the extisting models, because it does not have the strong limitiation of the existing models. There are still many possibilities for future research, but the ENO model we presented here can be used as the basis for a Negotiation Support System. We created a prototype of such a NSS in AIMMS. This prototype can now be used by actual negotiators.

References

[1] Automated negotiating agents competion, September 2014. http://www.itolab.nitech.ac.jp/ANAC2014/.

[2] R. Aumann. Backward induction and common knowledge of rationality. *Games and Economic Behavior*, 8:6–19, 1995.

[3] T. Baarslag, M. Hendirkx, T. Hendriks, and C. Jonker. *Measuring the Performance of Online Opponentmodels in Automated Bilateral Negotations.* Springer, 2012.

[4] T. Baarslag, M. Hendirkx, T. Hendriks, and C. Jonker. Pocket negotiator, September 2014. http://ii.tudelft.nl/negotiation/index.php/PocketNegotiator.

[5] C. Camerer. Progress in behavioral game theory. *The journal of economic perspectives*, 11:167–188, 1997.

[6] S. Chen and G. Weiss. An efficient and adaptive approach to negotiation in complex environments. *Frontiers in Artificial Intelligence and Applications*, 242:228–233, 2012.

[7] S. Fatima, M. Wooldridge, and N. Jennings. An agenda-based framework for multi-issue negotiation. *Artificial intelligence*, 152:1–45, 2004.

[8] C. Fershtman. The importance of the agenda in bargaining. *Games and economic behavior*, 2:224–238, 1990.

[9] M. Hendirkx, T. Hendriks, and C. Jonker. The benefits of opponent model in negotiation. In *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on*, volume 2, pages 439–444, 2009.

[10] K. Hindriks and D. Tykhonov. Opponent model in automated multi-issue negotiation. In *Seventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, pages 331–338, 2008.

[11] C. Holsapple, H. Lai, and A. Whinston. Implications of negotiation theory for research and development of negotiation support systems. *Group Decision and Negotiation*, 6(3):255–274, 1996.

[12] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12:111–125, 2003.

[13] G. Lai, C. Li, and K. Sycara. A decentralized model for multi-attribute negotiations with incomplete information and general utility functions. In Takayuki Ito, Hiromitsu Hattori, Minjie Zhang, and Tokuro Matsuo, editors, *Rational, Robust, and Secure Negotiations in Multi-Agent Systems*, volume 89 of *Studies in Computational Intelligence*, pages 39–57. Springer Berlin Heidelberg, 2008.

[14] G. Lai, C. Li, and K. Sycara. *A general model for pareto optimal multi-attribute negotiations*, volume 89 of *Studies in Computational Intelligence*. 2008.

[15] X. Luo, N. Jennings, N. Shadbolt, H. Leung, and J. Ho-man Lee. A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. *Artifical Intelligence*, 148:53–102, 2003.

[16] S. Matwin, T. Szapiro, and K. Haigh. Genetic algorithms approach to a negotiation support system. *Systems, man and Cybernetics*, 21:102–114, 1991.

[17] J. Nash. Noncooperative games. *Annals of Mathematics*, 54:286–295, 1951.

[18] G. Wright P. Goodwin. *Decision Analysis for Management Judgment*. Wiley, 2009.

[19] H. Peters. *Game Theory: A Multi-leveled Approach*. Springer, 2008.

[20] S. Ross. *Probability and Statistics for Engineers and Scientists*. Elsevier Academic Press, 2009.

[21] A. Roth and M. Malouf. Game-theoretic models and the role of information bargaining. *Psychological Review*, 86(6):574–594, 1979.

[22] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.

[23] Nicholas Jennings Shaheen Fatima, Michael Wooldridge. Bargaining with incomplete information. *Annals of Mathematics and Artificial Intelligence*, 44:207–232, 2005.

[24] The Optimization Firm. Baron, September 2014. http://www.aimms.com/aimms/solvers/baron/.

[25] J. Von Neumann and O. Morgenstern. *The Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[26] E. Winter. Negotiations in multi-issue committees. *Journal of Public Economics*, 65:323–342, 1997.