

Modelling anger caused by perceived unfairness in the game of nines

Joke Kalter

1700006

August 2015

Master Project

Human-Machine Communication
University of Groningen, The Netherlands

Supervisor:

Prof. Dr. Niels Taatgen (Artificial Intelligence, University of Groningen)

Secondary supervisor:

Dr. Christopher A. Stevens (Artificial Intelligence & Cognitive Engineering, University of Groningen)



university of
 groningen

faculty of mathematics and
 natural sciences

artificial intelligence

Abstract

The current study attempts to model anger in an interactive cognitive agent. The goal of the first experiment was to find behavioural patterns caused by anger in a negotiation game called the game of nines. The anger manipulation consisted of giving false feedback to participants, which suggested that their opponent was playing unfair. Results indicated that participants reached fewer agreements, quit more trials, and had more trials ending in a timeout due to the anger. In addition, after the false feedback, participants tended to lie more and insist on an offer more often. The findings of the first experiment were modelled in the cognitive architecture ACT-R (Anderson, 1995). The results of the model indicated that it effectively simulated the findings of the first experiment. A second experiment was done to see how the model would do in a game against a participant. Results of the second experiment showed no differences in trial outcomes between the control and anger condition, but did show that the model lied more often as a result of the feedback and that participants insisted more often in the anger condition. The total scores in the second experiment showed that the model on average obtained a negative score, suggesting that it accepted too many offers that were too low. Overall, the results of the cognitive model suggest that it is possible to model behavioural results due to anger, though the results of the second experiment suggest that the model is not effective enough when playing against a person. Suggestions for future research are discussed in the discussion.

Acknowledgement

I would like to thank Niels Taatgen and Christopher Stevens for their support throughout this project. The weekly meetings really helped both in terms of keeping me motivated and in finding solutions for the problems I ran into during the project. I would especially like to thank you both for your help and patience during the development of the cognitive model, which turned out to take more time and effort than I originally planned.

Furthermore, I would like to thank Irma, Rudi, Alie, Hans, Marjon, and Nick for their support both during my master's project and throughout my years at the university in general. Lastly, a thanks to my fellow HMC students for making it easier and fun to start my master's degree at a new faculty.

Contents

1. Introduction	6
2. Theoretical background	7
2.1 Anger	7
2.2 ACT-R	8
2.3 Game of nines	10
3. Experiment 1	11
3.1 Method	11
3.1.1 <i>Participants</i>	11
3.1.2 <i>Materials</i>	11
3.1.3 <i>Procedure</i>	12
3.1.4 <i>Analysis</i>	12
3.2 Results	13
3.2.1 <i>Questionnaires</i>	13
3.2.2 <i>General variables</i>	14
3.2.3 <i>Trials outcomes</i>	14
3.2.4 <i>Outcomes of coded variables</i>	16
3.3 Discussion first experiment	17
4. Model	19
4.1 Declarative memory	19
4.1.1 <i>Base level activation</i>	19
4.1.2 <i>Spreading activation</i>	19
4.1.3 <i>Partial matching</i>	20
4.1.4 <i>Noise</i>	20
4.2 Current model	20
4.2.1 <i>Technical description of the model</i>	21
4.3 Model results	22
5. Experiment 2	25
5.1 Method	25
5.1.1 <i>Participants</i>	25
5.1.2 <i>Materials</i>	25
5.1.3 <i>Procedure</i>	26
5.1.4 <i>Analysis</i>	26
5.2 Results	26
5.2.1 <i>Questionnaire</i>	26
5.2.2 <i>Points</i>	27
5.2.3 <i>Outcomes</i>	27
5.2.4 <i>Lies</i>	28
5.2.5 <i>Insistence</i>	29
5.3 Discussion results experiment 2	29
6. Discussion	31
7. Conclusion	33
8. References	34
Appendix A: Questionnaires in experiment 1	37
Appendix B: Cognitive model	41

1. Introduction

People negotiate every day, from finding a happy middle between where you and your partner go on holidays, to deciding which route is most efficient in a road trip. While in these negotiations a certain outcome may not have a big impact, in negotiations where money is exchanged, for example when buying or selling a car, there can be a big impact. Generally, people who work in an environment that requires them to negotiate are trained before starting their job to make sure they can get the best deal for their company and also to make sure the customer is happy. Though often these trainings consist of person-to-person or person-to-group settings, there are more and more courses available that make use of computer simulations. The pros of computer simulated negotiation training are that it is cheaper than person-to-person training (e.g. less instructors have to be trained and hired) and it's more efficient (e.g. more people can be trained in the same time as long as they have access to a computer). Though computerized negotiation and negotiation training has been researched for many years (e.g. Gauvin, Lilie & Chatterjee, 1990; Saunders & Lewicki, 1995; Ross, Pollman, Perry, Welty & Jones, 2001; Kersten & Lai, 2007; Williams, Farmer & Manwaring, 2008), the question remains whether computerized agents can reliably simulate human negotiation behaviour.

In every interaction human behaviour influences the outcome, which can lead to better or worse outcomes for both parties involved. People do not always use the most logical or efficient solution to solve a problem. One common type of human influences are emotions. In negotiation research, the influence of several different types of emotions have been researched, for example disappointment (van Kleef, de Dreu & Manstead, 2006; Lelieveld, van Dijk, Beest & van Kleef, 2013), worry, guilt, regret (van Kleef et al., 2006), trust (e.g. Kong, Dirks & Ferrin, 2014), and anger (e.g. Nabi, 2002; Srivastava, Espinoza & Fedorikhin, 2009; Adam & Shirako, 2013). These emotions tend to elicit certain behavioural patterns and negotiation strategies which can help a negotiator get better results when applied well, but can also work against them. Though these behaviours themselves can be modelled or programmed, is it also possible to model the underlying cause of these behaviours, i.e. why a certain strategy is used in a particular negotiation.

The current study focuses on whether anger, caused by perceived unfairness in a negotiation, can be modelled in a computer simulated agent playing a negotiation game called the game of nines. The first experiment will be exploratory. In this experiment, participants are asked to play the game of nines against each other via a chat-screen in which they can freely converse. To induce anger, participants receive false feedback, letting them think the other person is not playing fair. The goal of the first experiment is to find behavioural or strategic patterns that are a result of anger. Therefore, the chats of the first experiment are analysed for behavioural and strategic patterns, such as conceding, demanding or lying more often as an effect of getting angry. The outcome of the analysis of the first experiment is the basis of a cognitive model that can react to an unfair game in a more human way and simulate the effects of anger. The goal of this model is to both assess that it is performing badly due to another agent playing unfair, and to adapt its behaviour accordingly. In the second experiment, the cognitive model is made into an interactive agent against which participants can play the game of nines. The goals of the second experiment are to replicate the behavioural data of participants in the first experiment, and to check whether the cognitive model adapts its behaviour due to anger caused by an unfair game in a human versus computer setting (rather than a computer versus computer setting).

2. Theoretical background

2.1 Anger

Many studies have focussed on how expressing anger in negotiations affect the process. In general, expressing anger in negotiations tends to lead to more concessions being made by the other person (e.g. Sinaceur & Tiedens, 2006; Sinaceur, van Kleef, Neale, Adam & Haag, 2011). Wang, Northcraft and van Kleef (2012), however, found that, although participants tended to concede more when the other person expressed anger, they were also more prone to covert retaliatory actions. In their experiment, participants negotiated against a confederate who expressed anger in the manipulated condition and stayed emotionless in the neutral condition. Participants who were in the anger condition reached more agreements and made fewer demands in the negotiation than participants in the neutral condition. After the negotiation, participants were asked to what extent they would like their opponent to perform four different tasks, two of which were thought to be unpleasant and two pleasant. Participants in the anger condition engaged in covert retaliation, by stating they would like their opponent to perform the unpleasant tasks, more often.

Van Dijk, van Kleef, Steinel and van Beest (2008) also found that expressing anger in a negotiation can pay off. In their first experiment, they found that participants made higher offers to opponents who expressed anger. In addition, even though participants themselves got more angry as a result of their opponent's anger, they did not lower their offers. In van Dijk et al.'s (2008) second experiment, however, they found that fear of rejection was a mediating factor. When participants did not care whether their opponent would reject their offer, they were more deceitful. Furthermore, opponents who expressed anger wound up with lower outcomes. In addition, in their third experiment van Dijk et al. (2008) found that when participants did not perceive there to be high consequences to a rejected offer they made lower offers to opponents who expressed anger.

Several more mediating factors in expressing anger in negotiations have been studied. Deghani, Carnevale and Gratch (2014), for example, found that when a person attaches a moral significance to the issue they concede less when the other person expressed anger than when they did not attach any moral significance to the issue. Lui and Wang (2010) found trust to be a mediating factor in expressing anger in negotiations. In addition to anger leading to more feelings of distrust, they also found that distrust mediated how competitive participants were about the outcome of the negotiation. Côte, Hideg and van Kleef (2013) also found trust to be a mediating factor, though in their study this only seemed the case when the anger expressed was fake, which led to more demands by the participant receiving the faked anger expressions. When the anger was heartfelt, however, they found a decrease in demands made by the participants.

Though in the studies listed above anger was explicitly induced, either by means of a confederate or by instructing participants to use anger in negotiations, there have also been studies looking at why people get angry. One factor leading to anger in negotiations is the fairness of the process. Baron, Byrne and Branscombe (2006) describe that people assess three types of justice to judge whether a situation is fair, namely distributive, procedural and transactional justice. Distributive justice states that rewards should be distributed in a way that is in accordance with people's contribution. Procedural justice concerns the procedure used to divide rewards and its fairness is judged by the following factors: Consistency of the procedure, accuracy, the possibility for corrections, the extent to which decision makers are biased, and ethicality. Finally, transactional justice entails the way in which people are informed about the division of rewards, this is mediated by two factors: the rationality of the reasons given, and the courtesy and sensitivity with which the information is transacted.

Hegtvædt and Killian (1999) found that procedural fairness in negotiations is negatively related to negative feelings, i.e. the fairer people perceive a situation, the fewer negative emotions are experienced. In addition, they found that when it comes to distributive fairness, participants experience less negative feelings when the pay is fair to themselves, but experience more negative feelings when the pay is only fair to the person. Pillutla and Murnighan (1996) also found a relation

between anger and unfairness. In their study, participants had to accept or decline an offer made by an opponent. In one condition, participants knew that the amount to be divided was \$20 (complete knowledge), while in the other conditions they just received the offer. In both conditions, the offer the participant received was low (i.e. either \$1 or \$2). Participants in the complete knowledge condition thus perceived distributive injustice. Pillutla and Murnighan (1996) indeed found that participants in the complete knowledge condition reported feelings of unfairness more often than participants who did not know all the information. In addition, these feelings of unfairness were significantly correlated to reported feelings of anger.

Besides feelings of unfairness eliciting anger, there are also studies showing the mediating effect of anger between perceived unfairness and behaviour (e.g. Chan & Arvey, 2011; Seip, van Dijk and Rotteveel, 2014). In Seip et al. (2014) study, participants had to collaborate by contributing their own points to a common cause. Their points would then be multiplied by 1.5 and equally divided between all parties, despite their contribution. After contributions were made, participants were informed about other's contributions, after which they were given a chance to assign punishment points to one or more players. Punishment would however not only cost the other player, but also themselves. Seip et al. (2014) found that when participants played against a player they thought to be playing unfair (compared to playing against a cooperative player), they got angry and as a result made more use of a punishment system in the negotiations.

2.2 ACT-R

ACT-R (adaptive control of thought-rational; e.g. Anderson, 1995; Anderson, Matessa & Lebiere, 1997; Anderson, Bothell, Byrne, Douglass, Lebiere & Qin, 2004) is a cognitive architecture designed to explain how different parts, termed modules, of the brain work together to complete complex tasks. The ACT-R theory consists of four core modules, namely the perceptual-motor, goal, declarative, and procedural systems. The perceptual-motor system is concerned with all things relating to perceptual input and output, for example focussing on a piece of information on a screen or moving a computer mouse on a screen. The goal system is used to keep track of current goals and intentions. In other words, it keeps track of the task to be accomplished, at what point is that goal reached, and how far along is the process of finishing that task. The declarative system is used to retrieve information from memory. To process the information from these three modules, the procedural system is used. This system uses the information stored in the other modules and coordinates that information so the other modules are updated. For example, if the goal of a task is to find a certain letter (goal system), and an item is detected in the visual field (perceptual-motor system), the procedural system can signal the declarative system to retrieve information about the item in the visual field. The procedural system can then check whether the information retrieved from declarative memory is equal to the item searched for in the goal system, if so, it signals the goal system that the goal is reached. If the goal is not reached, then the procedural system can signal the perceptual-motor system to keep looking for other items.

The information in the perceptual-motor, goal, and declarative systems are kept active in their respective buffers. The perceptual-motor information is stored in a visual or manual buffer, the goal of the task is stored in the goal buffer, and the retrieved declarative information is stored in a retrieval buffer. Each buffer can only keep one piece of declarative information, called a chunk, active at the time. These chunks can consist of several slots, which contain small parts of the overall information in the chunk. For example, if the chunk in the visual buffer refers to an item that has been read from the screen, then that chunk can, for example, contain a single slot listing the size of the item and another slot listing the colour.

The production system does not use a single buffer, but rather consists of a set of productions rules. These productions check the content of relevant buffers on the left hand side and if they match the requirements of the production rule, the right hand of the rule will fire. The buffers checked on a production rule's left hand side can be checked for content (i.e. whether there is a particular chunk stored in there or whether the buffer is empty) and for state (i.e. is the buffer currently actively being used or not). On the right hand side buffers are updated, either by moving

information from one buffer to another, by clearing the buffer's contents, or by creating a new chunk in a buffer. Table 1 gives a small example of declarative memory and a production rule that can be used in a task in which the goal is to find a letter on a screen that is a consonant.

Table 1: Example code of declarative memory chunks and a production rule in ACT-R

<pre>(add-dm (goal1 isa find-letter-goal goal-letter-type consonant) (letter-fact1 isa letter-fact letter A type vowel) ... (letter-fact26 isa letter-fact letter Z type consonant))</pre>	<p>These are examples of how information is stored in the ACT-R's declarative memory. A memory chunk takes the form: (chunk's-name isa chunk-kind slot1-name slot1-content slot2-name slot2-content)</p>
<pre>(p read-letter =goal> isa find-letter-goal goal-letter-type consonant =visual-location> screen-x =x-value screen-y =y-value letter =current-letter ?retrieval> buffer empty state free ==> +retrieval> isa letter-fact letter =current-letter -visual-location>)</pre>	<p>The production rule is called 'read-letter' The current goal is 'find-letter-goal' The letter to be found is the letter 'L', which is stored in the 'goal-letter' slot.</p> <p>In the visual buffer, the location on the screen that currently has the focus is listed. In addition, the letter presented on the screen is stored in the letter slot as the variable 'current-letter'.</p> <p>The production rule also checks whether the retrieval buffer is empty, meaning there is no piece of information already retrieved from the declarative memory.</p> <p>If everything on the left hand side (above) returns true, then execute the right hand side (below)</p> <p>A request is made to retrieve a piece of information, called a letter-fact, with a slot containing the letter that has been read from the screen. When the matching chunk is retrieved, the slot containing the letter-type will also be placed in the retrieval buffer and can be compared to the slot in the goal buffer.</p> <p>This removes the content from the visual buffer so a new request (i.e. new search) to this buffer can be made in another production rule.</p>

The information stored in the memory of a model has the form of chunks, which can be retrieved by the declarative system. The declarative system can search the memory based on information it has been given in a retrieval request. A chunk that best matches the slots listed in the retrieval request is then placed in the retrieval buffer. In addition to retrieving preprogrammed information from declarative memory, new chunks can also be added to the memory. To learn and store new information, the imaginal buffer can be used. To create a new chunk, the desired chunk

type along with relevant information is stored in the imaginal buffer on the right hand side of a production rule (e.g.: +imaginal> isa chunk-type slot1 value), later, when the imaginal buffer is cleared (i.e.: -imaginal>), that information is stored as a chunk in the declarative memory (i.e. chunk-type1 isa chunk-type slot1 value) and can later be retrieved if necessary.

2.3 Game of nines

The game of nines (e.g. Kelley, Beckman & Fischer, 1967; Schoeninger & Wood, 1969; Mascarenhas, Marques, Campos & Paiva, 2013) is a game in which 2 people have to negotiate about how to divide 9 points every round in order to get the highest score. For every round, each player is given a number that is their Minimum Necessary Share (MNS). The MNS value is the minimum number of points players have to get in that round in order not to lose points. A player's MNS value is not known to their opponent, though in most versions of the game, players are allowed to tell the opponent their MNS value, in which case they can either lie or be honest.

After the MNS values are given, a round is started by a player making a first offer (e.g. 'I want 6 points, I offer you three points'). Generally, players take turns starting the rounds. The amount of points added to each player's total score depends on both the amount of points they get out of the negotiation and their MNS value for that round. For example, if player one has an MNS value of 3 and player two an MNS value of 4, then they have to get at least 3 and 4 points, respectively, in order to break even. When a player gets more points than their MNS value then they gain points for their total score. For example, if player one gets 4 points in one round, then the amount of points added to player one's total score is 4 minus their MNS value, which in this example means $4 - 3 = 1$ point. By extension, player two would get 5 points (i.e. $9 - 4 = 5$) in that round, which would result in $5 - 4 = 1$ point gained on their total score. In a similar manner, if a player gets fewer points than their MNS value in that round, points are subtracted from their overall score. For example, if player one would get 2 points, while their MNS value is 3, then they would get $2 - 3 = -1$ points.

In every round, players can make offers back and forth until they reach one of three outcomes; an agreement, a quit, or a timeout. In the case of an agreement, each player gets the amount of points they negotiated minus their respective MNS values. When the players cannot reach an agreement, either player can quit the round. In this case, no points will be lost or gained by either player and thus the total score remains the same. In most versions of the game, there is a certain time limit set for each round. When this time limit is reached, the round is over and no points will be gained or lost.

3. Experiment 1

The goal of the first experiment is to find behavioural and outcome patterns in a negotiation game as a result from participants getting angry. The game used in this experiment is the game of nines, described in paragraph 2.3. Previous research has found several behaviours related to anger in negotiations, for example concessions (e.g. Sinaceur et al., 2011), deceitful behaviour (van Dijk et al., 2008), and demands (Côte et al., 2013). Since some of these behaviours are contradictory, the current experiment takes a rather open form. In this experiment participants are able to freely communicate with each other via a chat program. From the resulting conversation files, all relevant variables can then be extracted. In addition, rather than giving one participant the assignment to act angry, both participants in the experimental condition will receive the same manipulation. This is done to create genuine feelings of anger and, in addition, any behaviour resulting from the manipulation might be construed as unfair or uncalled for by the opponent, which might add to the anger manipulation.

3.1 Method

3.1.1 Participants

40 participants were recruited via Facebook. 11 participants were male, 25 female (4 participants did not give their gender, age and nationality information). The mean age was 22.8, (SD= 3.2, Min.= 19, Max.= 34). Participants received a monetary reward for participating.

The data of three dyads (6 participants) were removed since they reported knowing about the manipulation in the experiment and/or explicitly discussed the feedback with their opponent. This left a total of 34 participants (9 male, 22 female, 3 unknown) with a mean age of 22.6 (SD= 4.5, Min.= 19, Max.= 30). Of these 34 participants, 18 (9 dyads) were in the manipulated condition.

3.1.2 Materials

The experiment was conducted on 3 macbook pro's, each set-up in its own cubicle in the ALICE-lab at the University of Groningen. The chatprogram Pidgin¹ was used to allow for communication between the participants and the experimenter. In logging the communication during the experiment the time-stamp of each message consisted of hours, minutes and seconds.

An instruction sheet was given to the participants, explaining how the game worked. The instructions also encouraged participants to actively negotiate and to feel free to persuade the opponent to accept their offers. This was done to ensure more verbal responses instead of having participants simply type out integer offers. Participants were also told that in any given round there was a possibility for both of the players to get at least one point. This was thought to let participants more easily believe that the other person could be lying when receiving negative feedback.

Two conditions were tested, namely a control condition and an anger condition. The conditions only differed in the type of feedback participants received. In the anger condition participants were led to believe the other player was playing better as a result of playing more deceitful in order to trigger angry feelings towards the other player. In this condition the participants received the following feedback: 'Judging from your score you are not doing very well, try to step it up in the second half!' followed by: 'Your opponent has a lot more points than you, maybe don't believe everything they say...?' Participants in the control condition were told the following: 'Judging from your score you are doing great, keep it up!'.

The series of MNS (Minimum Necessary Share) values in the game was also designed to increase feelings of distrust towards the other participant. The first 12 trials (all trials before feedback was given) were the same for each dyad and included two so called impossible trials. In the impossible trials both participants received an MNS value of 5, making it impossible for both players to receive points or break even. In addition, participants had two trials in which they had an MNS value of 6, making it harder for them to negotiate points. The MNS values participants received after the feedback was randomized by pair and also included two impossible trials and two instances

¹ <https://pidgin.im/>

where participants had an MNS value of 6. All MNS value pairs can be found in table 2. If players obtained a perfect deal in each round (i.e. where the player gets 8 points and the opponent gets 1), they could obtain 52 points in the first 12 rounds, and 56 in the last 13 rounds, while the opponent would obtain a score of -32 and -35, respectively. In contrast, if every trial ended in a 5-4 division, players could obtain between 4 and 16 points in the first half, and between 4 and 17 in the second half of the game.

Table 2: MNS value pairs used in the experiment

Round	1	2	3	4	5	6	7	8	9	10	11	12	
Player 1	3	5	4	1	5	2	3	3	6	5	6	1	
Player 2	4	2	3	6	5	6	3	5	1	5	3	1	
Round	13	14	15	16	17	18	19	20	21	22	23	24	25
Player 1	3	6	4	1	2	5	4	6	5	1	2	5	4
Player 2	3	1	4	1	6	5	4	2	4	2	6	5	5

Players received feedback after the 12th round. The order of the first 12 MNS pairs was the same for every dyad. The order of the value pairs in round 13 to 25 was randomized.

Participants were asked to fill out two questionnaires. The goal of these questionnaires was to check whether the manipulation worked, and thus whether participants got angry. The first questionnaire was presented before the game was played and consisted of questions such as ‘I feel angry’ and ‘I feel wronged’, along with distracter questions such as ‘I feel unsafe’. The second questionnaire was presented after the game of nines was played and contained the same questions as the first questionnaire. In addition, the second questionnaire asked participants to rate their opponent by answering questions such as ‘My opponent was honest’ and ‘My opponent played fair’. Participants could answer the questions by selecting a number on a 5 point Likert scale, ranging from ‘1: never/not at all’ to ‘5: always/very much’. The second questionnaire also contained an open question, asking participants what they thought the study was about. The complete set of questions can be found in appendix A.

3.1.3 Procedure

Participants arrived at the lab at the same time and were placed in separate cubicles. Participants were instructed to first fill out the first questionnaire and then start reading the instructions to the game of nines. After both participants finished reading the instructions the game started. Each participant received their MNS value in a private chat window between the participant and the experimenter. A common chat window, in which both players and the experimenter were present, was used for negotiation. At the beginning of each round, the experimenter gave both players their MNS value and announced which round it was and whose turn it was to start in the common window. After the 12th trial, feedback was given to the players in their respective private chat windows. After the game was finished, both participants were given the second questionnaire. When the experiment was over, participants were debriefed about the purpose of the study together, given their monetary rewards, and were thanked for their participation.

3.1.4 Analysis

Results were analysed using Rstudio². The answers to the questions that were the same in the first and second questionnaire were analysed using a 2 (control vs anger condition) X 2 (before vs after game) mixed design ANOVA. The final scores and the questions in the second questionnaire, relating to the evaluation of the opponent, were analysed using one way between-subjects ANOVA. Since the

² <http://www.rstudio.com/>

distracter questions did not give any significant nor relevant results, their results will not be further discussed.

The negotiations between participants were coded for further analysis. First off, trials were coded by outcome. There are three outcome possibilities, namely an agreement (deal), the case in which either participant quits (quit), or a trial ending because time ran out (time out). The negotiation leading up to an outcome was further coded by content. Every typed line was coded to be either an offer (e.g. 'I offer you 3 points'), a request (e.g. 'I want 5 points'), a final offer (e.g. '5 is my final offer' or 'I can't go any lower than 5'). Anything that did not fit into those three categories was labelled as a comment.

In addition, conversations were coded for lies, demands, concessions and insistence. A comment was labelled a lie when one of the following three things occurred: The participant explicitly states their MNS and it is different than their actual value, the participant says they cannot go any lower when the current offer is 1 or more above their MNS, or when the participant states their MNS is high when it is 4 or lower. Demands entail requests that are higher than what the opponent is offering (e.g. 'I want 5 points'), but also raising their own request within the same trial (e.g. player 1 requested 5 points at first, but later in the same trial asks for 6 points). Concessions include every case in which a participant lowers their own offer in the same trial (e.g. player 1 requested 5 points at first and later in the same trial lowers it to 4). In the case of insistence the participant will not budge from their offer in the same trial. Every time the same offer in one trial is repeated by a player it is coded as insisting.

Furthermore, certain specific utterances were coded, namely accusations, specific rejections, comments on niceness, comments on fairness, and apologies. Accusations involves any case in which the participant accuses their opponent of lying or doubt their opponent (e.g. 'Is that really your MNS value?' or 'I don't believe you have fewer points than me'). Explicit rejections entail instances in which the participant explicitly says 'no' to the opponent's offer. Comments on niceness or fairness involve any case in which the participant says the opponent is not being nice or that a certain offer is or is not fair, respectively. Finally, apologies entail any utterance of the word 'sorry'. Though these explicit cases were found in the data, there were not enough data nor instances to do a reliable analysis. Therefore, these five variables will not be further discussed in the results.

The coded data is analysed using a general linear mixed effects regression (glmer) model. The manipulation condition (control vs. anger) is used as a between subjects factor and time (before vs. after feedback) as a within-subject factor.

3.2 Results

3.2.1 Questionnaires

When asked the questions 'I feel angry' and 'I feel wronged', participants rated an average of 1.4 (sd = 0.57) and 1.4 (sd = 0.73), respectively, on a 1 to 5 Likert scale ranging from '1: not at all' to '5: very much'. When looking at how angry participants felt, no main effects were found for both the condition (control vs. anger condition) and the time the questions were asked (before vs. after playing the game, regardless of condition; condition: $F(1, 32) = 0.55$, $p = 0.47$; time: $F(1, 32) = 0.00$, $p = 1.00$) nor was there an interaction effect between the condition and the time ($F(1, 32) = 0.64$, $p = 0.43$). The same was true for how wronged participants felt (condition: $F(1, 32) = 0.20$, $p = 0.66$; time: $F(1, 32) = 0.98$, $p = 0.33$; interaction: $F(1, 32) = 0.19$, $p = 0.67$). These results indicate that the manipulation did not make the participants feel angry nor made them feel wronged as a result of the manipulation.

When looking at how participants rated their opponent, no significant difference was found between the conditions in rating how nice their opponent was ($M_{\text{total}} = 3.8$, $sd = 1.10$; $F(1, 32) = 2.3$, $p = 0.14$), whether they thought their participants were better at the game ($M_{\text{total}} = 2.7$, $sd = 1.29$, $F(1, 32) = 2.5$, $p = 0.13$), or whether they thought their opponent was angry with them ($M_{\text{total}} = 1.6$, $sd = 0.99$, $F(1, 32) = 0.7$, $p = 0.41$). There were, however, significant differences in their ratings of how honest the opponent was and whether they thought the opponent played fair. Participants in the

anger condition rated their opponent to be less honest ($M_{\text{anger}} = 2.7$, $sd_{\text{anger}} = 1.14$) than participants in the control condition rated their opponents ($M_{\text{control}} = 3.8$, $sd_{\text{control}} = 1.13$; $F(1, 32) = 7.8$, $p < 0.01$). In addition, participants in the anger condition thought their opponent played less fair ($M_{\text{anger}} = 2.8$, $sd_{\text{anger}} = 1.22$) than participants in the control condition thought their opponent did ($M_{\text{control}} = 3.7$, $sd_{\text{control}} = 1.20$; $F(1, 32) = 4.8$, $p < 0.05$).

Even though participants did not indicate they actually got angry, these results do indicate that the manipulation partially worked. More specifically, participants in the anger condition did have a bigger tendency to believe their opponent was lying to them and did not play fair.

3.2.2 General variables

On average participants scored 9.3 points in the first 12 trials ($sd = 3.8$) and 9.5 in the last 13 trials ($sd = 4.0$). After correcting for the difference in number of trials, by dividing the scores of the last 13 trials by 13 and multiplying it by 12, no significant difference was found between the scores of participants in the two conditions ($F(1, 32) = 0.22$, $p = 0.64$). No main effect for time (before vs. after feedback) was found either ($F(1, 32) = 0.32$, $p = 0.58$), nor was there an interaction effect between the condition and time ($F(1, 32) = 0.03$, $p = 0.88$). These results suggest that there were no overall differences in scores due to the fact that feedback was given, regardless of condition. In addition, no differences between the conditions were found in participants' scores.

When looking at the two players separately, however, a significant difference was found in their scores. Player one scored more points ($M_{p1} = 10.8$, $sd_{p1} = 4.05$) than player two ($M_{p2} = 8.0$, $sd_{p2} = 3.22$; $F(1, 30) = 9.2$, $p < 0.005$). No interaction effects were found between the player and the condition or time. Though participants were randomly selected as being either player one or two, player one always started the first trial, after which players took turns starting trials. In the first half of the experiment, both players started rounds where players both had an MNS value of 5 (impossible trial). In addition, when looking at the average MNS values on trials where participants started versus where the opponent started, the values hardly differed per player (MNS when player starts: $M_{p1} = 4.5$, $M_{p2} = 4.2$; MNS when opponent starts: $M_{p1} = 2.8$, $M_{p2} = 3.2$). In the second half, the MNS values were randomized, meaning players had the same odds of starting a trial containing a certain MNS value. Furthermore, no difference was found in how the different players rated their opponents in terms of honesty, fairness, or whether they thought their opponent was nice or angry. Overall, the found difference between players seems a fluke, perhaps caused by player one starting the first trial and thereby setting the tone.

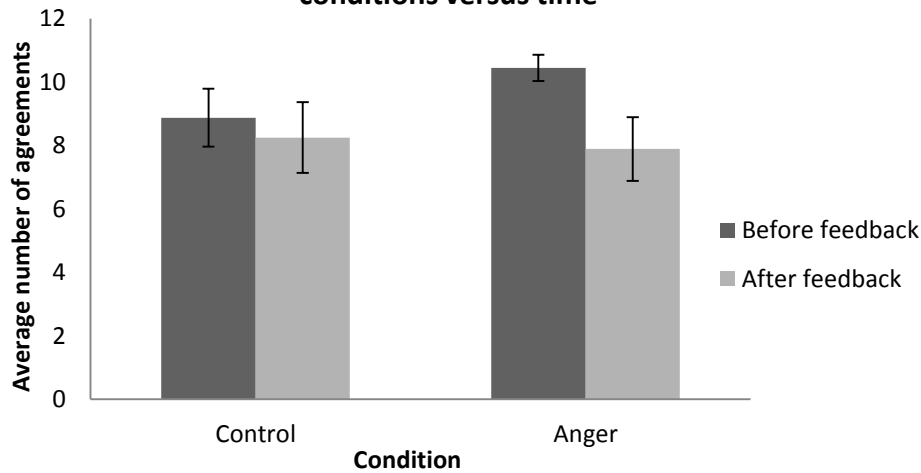
Overall, trials lasted an average of 80 seconds ($sd = 14.15$). In the control condition, trials lasted an average of 82 seconds ($sd = 16.16$) and in the anger condition trials lasted an average of 78 seconds ($sd = 11.86$). No main or interaction effects were found for the durations of trials between the conditions and time ($F(1, 28)_{\text{condition}} = 0.00$, $p = 0.99$; $F(1, 28)_{\text{time}} = 4.07$, $p = 0.053$; $F(1, 28)_{\text{interaction}} = 1.40$, $p = 0.25$).

3.2.3 Trial outcomes

To correct for the difference in number of trials before and after feedback, the 25th trial is removed from the analysis. Since the MNS values given after the feedback were randomized, it is assumed that removing the 25th trial will randomly remove a data point without any specific influence on the data.

Before the feedback, participants reached an average of 8.9 deals ($sd = 2.6$) in the control condition, and 10.4 ($sd = 1.2$) in the anger condition. After the feedback, participants reached 8.3 deals ($sd = 3.2$) in the control condition, and 7.9 ($sd = 3.0$) in the anger condition (figure 1). A general linear mixed effect regression (glmer) model showed no main effects for condition ($Z = 1.24$, $p = 0.21$) or time (before vs after feedback; $Z = -0.89$, $p = 0.37$). The glmer did, however, show a significant interaction between the condition and the time ($Z = -2.23$, $p < 0.05$). More specifically, dyads in the anger condition came to fewer agreements after the feedback than before the feedback, while the number of agreements among dyads in the control condition stayed the same before and after the feedback.

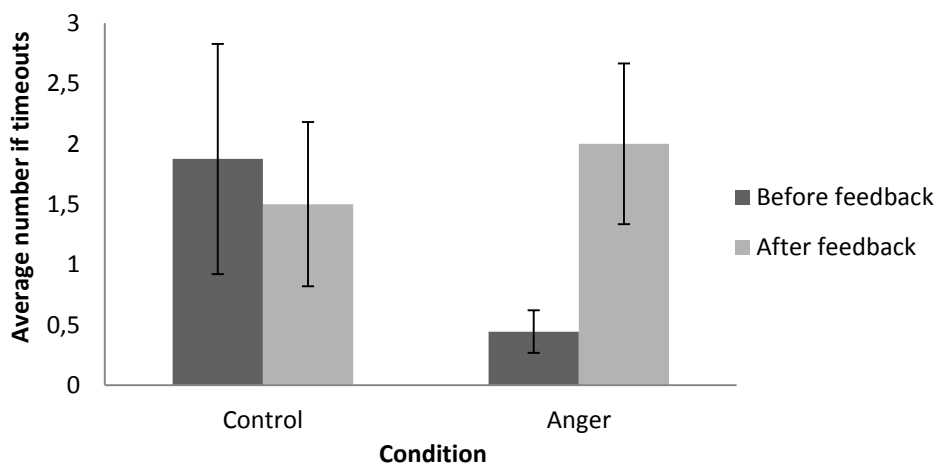
Figure 1: Number of agreements for the different conditions versus time



Overall, 49 trials ended in a timeout. On average, dyads in the control condition had 1.9 timeouts before the feedback ($sd = 2.7$) and 1.5 ($sd = 1.9$) after the feedback. Dyads in the anger condition had an average of 0.4 trials that ended in a timeout before the feedback ($sd = 0.5$) and 2.0 ($sd = 2.0$) after the feedback. Figure 2 shows the average number of timeouts and the standard errors. A glmer showed an interaction effect between condition and time ($Z = 2.8$, $p < 0.005$), but no main effects for condition ($Z = -1.42$, $p = 0.15$) or time ($Z = -0.71$, $p = 0.48$). These results suggest that dyads in the anger condition had more trials ending in a timeout both because of the feedback and the content of the feedback. However, given the low average count of timeouts before the feedback compared to the average count of timeouts the dyads in the control condition had before the feedback, these results should be taken with caution.

In total, 57 trials ended because one of the players quit. On average, participants in the control condition had 1.3 trials ending because someone quit ($sd = 2.1$) before the feedback and 2.3 ($sd = 3.7$) after the feedback. For the participants in the anger condition, these numbers were 1.1 ($sd = 1.5$) and 2.1 ($sd = 3.3$), respectively. Figure 3 shows the average number of trials ending in a participant quitting for both the condition and the time. Though no main effect for condition was found ($Z = 0.20$, $p = 0.84$), there was a significant difference in the number of quits due to the feedback ($Z = 2.0$, $p < 0.05$). Overall, it seemed participants quit more trials after, rather than before, the feedback. No interaction effect was found between condition and time ($Z = -0.13$, $p = 0.90$).

Figure 2: Average number of trials that ended in a timeout for condition versus time



3.2.4 Outcomes of coded variables

A move was counted as a concession when a player lowered their offer in a round compared to their earlier made offer in that same round. Participants in the control condition made an average of 11.3 concessions (sd = 7.1) before the feedback and 12.3 (sd = 9.3) after the feedback. In the anger condition, this was 11.1 (sd = 4.9) and 16.6 (sd = 14.9), respectively. In total, 437 concessions were made in the 425 trials. A glmer showed no significant main or interaction effects ($Z_{\text{Condition}} = 1.4$, $p = 0.17$; $Z_{\text{Time}} = 0.06$, $p = 0.95$; $Z_{\text{Condition*Time}} = -0.85$, $p = 0.40$). When looking at the number of demands participants made in the different conditions and before versus after the feedback, no effects were found either ($Z_{\text{Condition}} = 1.6$, $p = 0.11$; $Z_{\text{Time}} = 0.72$, $p = 0.47$; $Z_{\text{Condition*Time}} = -0.78$, $p = 0.44$). A demand was counted when a player asked for more than what the other player was offering or when a player upped their own offer compared to their earlier offer in that same round. On average, participants in the control condition made 7.4 demands (sd = 3.7) before the feedback and 8.8 (sd = 3.0) after the feedback. Participants in the anger condition made 7.9 demands (sd = 2.7) before the feedback and 12.0 (sd = 5.5) after the feedback. In total, participants made 308 demands in the 425 trials.

Figure 3: Average number of trials ending by one of the players quitting split out for condition vs time

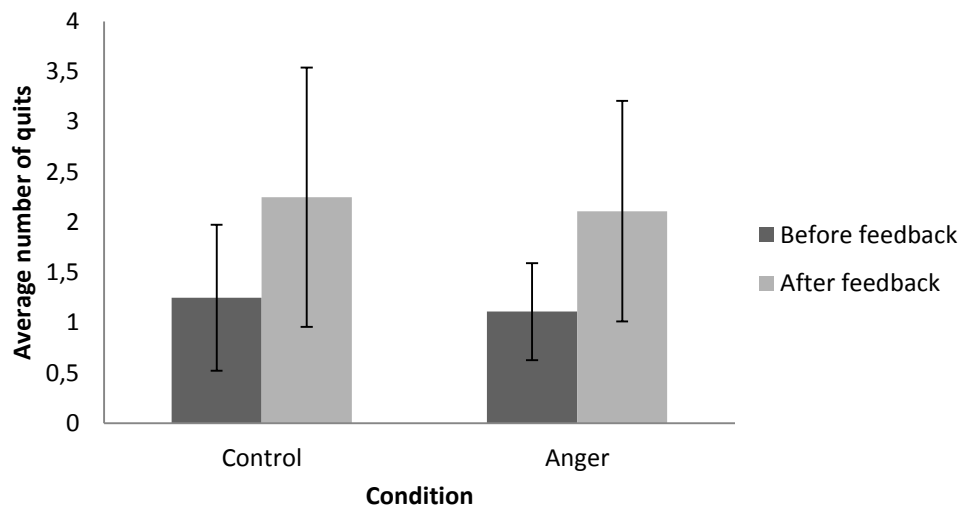
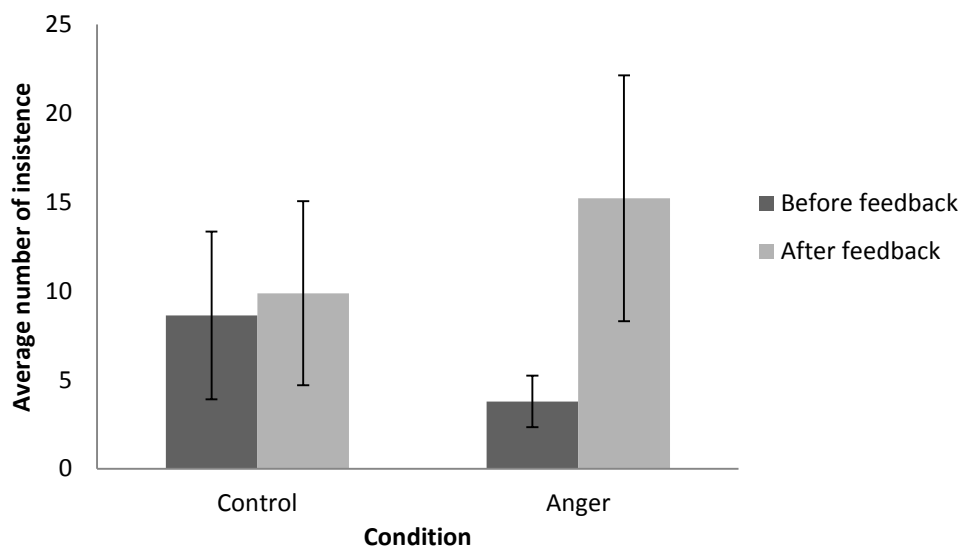


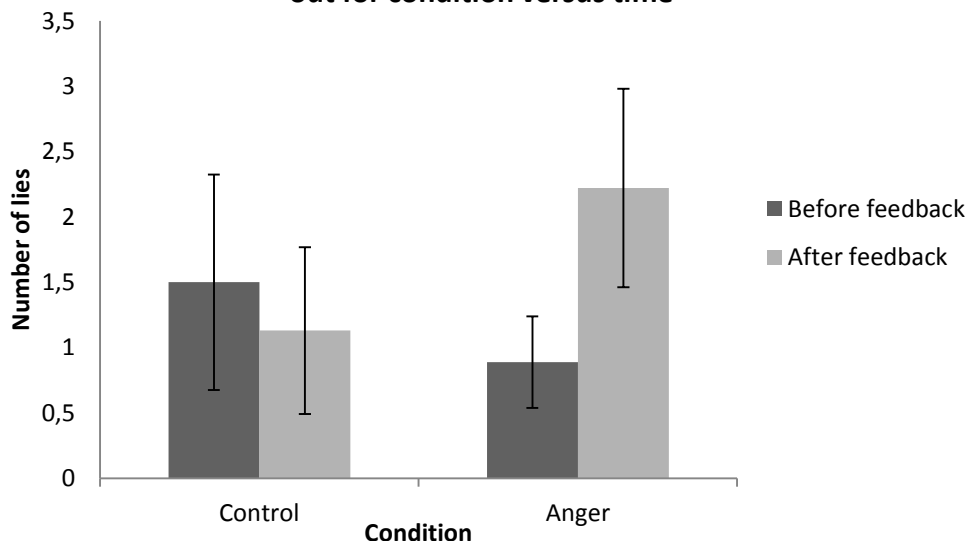
Figure 4: Average number of times a participant insisted, split out for condition versus time



A move was considered insisting when a player repeated the offer they already made in the same trial, which happened 319 times in the 425 trials. On average, participants in the control condition insisted 8.6 times ($sd = 13.4$) before the feedback and 9.9 times ($sd = 14.7$) after the feedback. For participants in the anger condition these numbers were 3.8 ($sd = 4.4$) and 15.2 ($sd = 20.7$), respectively. A glmer showed no main effects for condition ($Z = 0.00$, $p = 0.99$) or time ($Z = 0.21$, $p = 0.84$). The analysis did, however, show an interaction effect between the condition and the time ($Z = 2.88$, $p < 0.005$). As figure 4 suggests, participants in the anger condition insisted an offer more often after they received feedback, while participants in the control condition did not seem to change their tendency to insist.

In total, participants lied 49 times. A lie was counted as an instance where a person explicitly lied about their MNS value, when a person said they had a high MNS value when it actually was 4 or lower, or when a participant claimed they could not go any lower when the current offer is 1 or more points above their MNS value. On average, participants in the control condition told 1.5 ($sd = 2.3$) in the trials before the feedback and 1.1 ($sd = 1.8$) after the feedback. In the anger condition, participants told an average of 0.8 lies ($sd = 1.1$) before they received feedback and 2.2 ($sd = 2.3$) after the feedback. A glmer showed significant main effects for both the condition ($Z = -24.4$, $p < 0.001$) and the time ($Z = -129.0$, $p < 0.001$). In addition, an interaction effect was found between the condition and time ($Z = 351.2$, $p < 0.001$). Figure 5 shows a plot of the average number of lies, and their standard errors, participants told in the different conditions split out to before and after the feedback was received. It seems that overall participants told more lies as the game progressed. In addition, participants in the anger condition lied more often than participants in the control condition. Finally, the interaction effect suggests that the feedback the participants in the anger condition got was a mediating factor leading to them telling more lies. However, given the low count of lies (49 in 425 trials), these results should be taken with caution.

Figure 5: Average number of lies participants told, split out for condition versus time



Discussion first experiment

The goal of the first experiment was to see what types of negotiation behaviours occur when people get angry due to an unfair game. Participants who were in the manipulated condition did report they thought the other player played unfair, suggesting the manipulation worked. Although participants did not explicitly report getting angry, some significant differences in behavioural patterns were found between participants in the control versus the anger condition. Participants in the anger condition came to fewer agreements after they received their feedback, while participants in the control showed no difference in the number of agreements before and after the feedback. In both

conditions participants tended to quit more trials after they received feedback, but significantly more timeouts after the feedback only occurred in the anger condition.

The current study also did not find an increase in demands due to the manipulation. In addition, unlike previous research (e.g. Sinaceur & Tiedens, 2006; Sinaceur et al., 2011; Wang et al., 2012), participants did not make more concessions in the anger condition. One explanation for this is that participants in the current study rarely explicitly stated being angry. In addition, in the current study both participants received feedback suggesting the opponent was playing unfair. This may have caused participants to think that their opponent's anger was not justified, which in turn caused participants to not change their behaviour due to their opponent's angry behaviour.

The current study did, however, find a difference in insisting behaviour. Participants in the anger condition insisted on a certain offer more often after they received feedback, while participants in the control condition did not change their behaviour after the feedback. This suggests that participants in the anger condition did stand their ground more often and were more unwilling to give in during negotiations after they perceived their opponent to be an unfair player. The increase in insisting behaviour can also explain the increase in trials ending in a timeout. When participants insist on a certain offer, more communication has to take place in order to get to an outcome, which would take more time than simply accepting an offer.

Finally, the current study found that participants in the anger condition tended to lie more often. This is congruent with the study of van Dijk et al. (2008), who found participants to be more deceitful when the anger the opponent expressed was fake. It could be that participants did not believe any expressions of anger, explicit or implicit, their opponent expressed, because they thought they themselves were the ones who were being duped.

The cognitive model, discussed in the next section, will be based on the current findings. It will model the game of nines and try to simulate the behaviours caused by the unfair setting. Specifically, it will try to model a decrease in agreements, caused by more trials ending because of a player quitting or ending in a timeout. In addition, the model will try to simulate the difference in strategy caused by a perceived unfair setting, i.e. causing it to insist and lie more often.

4. Model

The cognitive model was developed in ACT-R (Anderson et al., 2004). The goal of the model was to simulate the first experiment, both by simulating the game of nines as well as the resulting behaviour from the anger manipulation. Specifically, the variables involved will be trial outcomes (i.e. deals, quits, and timeouts) and strategic behaviours (i.e. lies and insistence). In the model, the bargaining steps are stored in chunks in the declarative memory. These chunks contain information about the opponent's offer, based on which the model will choose its move. To incorporate the anger manipulation a new buffer is used, namely a performance buffer, which influences the model's choices via spreading activation.

This chapter will start out by explaining how chunks are retrieved from declarative memory and how spreading activation is involved in retrieval. Next, the chunks used in this model will be discussed as well as the implementation of a performance buffer. Finally, the results from the model are discussed.

4.1 Declarative memory

As was discussed in paragraph 2.2, the declarative memory in ACT-R is set up using chunks which can contain several slots. In addition, each chunk has an activation level, which reflects how active a certain chunk is in the memory. When a retrieval request is made to the declarative module, the memory is searched for a matching chunk. If there are multiple chunks that match the request, the chunk with the highest activation level is chosen. In order to be retrieved, the activation level has to be above the retrieval threshold. The activation value of the chunks in the current model is determined by the following formula:

$$A_i = B_i + S_i + P_i + \epsilon_i$$

In this formula, the activation level of chunk i (A_i) is the sum of the base level activation (B_i), spreading activation (S_i), partial matching (P_i), and a noise factor (ϵ_i). The individual components are discussed below.

4.1.1 Base level activation

The base level activation is based on the number of presentations a certain chunk has had and is calculated using the following formula:

$$B_i = \ln\left(\sum_{j=1}^n t_j^{-d}\right) + \beta_i$$

In this formula, n represents the number of presentations of chunk i , t_j is the time since the j th presentation, d is the decay parameter, and β_i is a constant offset. The decay parameter determines the rate with which the activation level of a chunk drops over time, to simulate a chunk fading from memory. In the current model the decay parameter is set to 0.5. In the current model, the constant offset is set to 1. This was done to insure chunks can be retrieved at all times, since the type of information used in the current model is not expected to disappear with time.

4.1.2 Spreading activation

The spreading activation determines the effect that the content of another buffer has on the retrieval process. The following formula is used:

$$S_i = \sum_k \sum_j W_{kj} S_{ji}$$

The element k represents all the buffers in the model (e.g. goal buffer). Element j represents the source of activation (i.e. a specific chunk). W_{kj} represents the activation from source j in buffer k and S_{ji} is the associative strength from source j to chunk i (i.e. a chunk in declarative memory).

By default, W_k is set to 0 for all buffers except for the goal buffer, which defaults to 1. Any buffer can be used for spreading activation by explicitly setting the activation parameter for that buffer. W_{kj} is calculated by dividing W_k by the number of chunks in source j . For example, if the goal buffer contains the chunk '(goal1 isa goal slot1 nil slot2 20 slot3 50)', W_{kj} would be $W_k/2$. Note that W_k is not divided by three since slot1 is nil and thus empty.

If a chunk i in declarative memory does not contain slots that match the chunks in the source j , then S_{ji} is 0. Otherwise, the following formula is used:

$$S_{ji} = S - \ln(\text{fan}_j)$$

S is the maximum associative strength, which is set to 5 in the current model. fan_j represents the number of chunks in declarative memory that contain source j .

4.1.3 Partial matching

Partial matching allows the retrieval process to retrieve a chunk that is not a complete match to the retrieval request. Matching is calculated using the following formula:

$$P_i = \sum_k PM_{ki}$$

The elements k are the slot values of the retrieval request. M_{ki} represents the similarity of chunk i and value k . Since the partial matching is added to the total activation of a chunk, a mismatch between a chunk i and a value k results in a negative value. To give more weight to mismatching chunk (resulting in a lower activation level) the mismatch penalty parameter P can be set. In the current model, that parameter is set to 2.

4.1.4 Noise

The noise factor in ACT-R has two components; transient and permanent. The transient component is calculated each time a chunk is being retrieved while the permanent component is linked to the chunk as it is added to declarative memory. In the current study only transient noise is used, which is a random number drawn from a logistic distribution.

4.2 Current model

The current model is set up to play against a second version of itself, in order to simulate a two-person negotiation (i.e. the first model is player 1, the second model is player 2). Individual trials are played by putting information about a trial, for example the model's MNS value and the opponent's offer, in the goal buffer of the model. Using the information in the goal buffer, a retrieval request is made to the declarative memory. In this model, the declarative memory is pre-programmed with the possible moves a player can make. This means that the strategy choices made are not a result of the model choosing different production rules, but rather by retrieving a specific declarative memory chunk containing information about the move the player will make. This way of setting up declarative memory chunks in order to make the model choose different moves has been proven effective in previous studies (Stevens, Taatgen & Cnossen, in prep.). The chunks used here are similar to the chunks used in Stevens et al. (in prep.) and were adapted to best fit the current model's goals. More detailed information about the slots these chunks contain can be found below.

After a move is retrieved from the declarative memory, the goal buffer is updated with the new information (e.g. the model's new offer). Via lisp code, the information about the offer made by a player is then transferred to the other player's goal buffer, who then goes through the same process of retrieving a move from declarative memory based on the information in the goal buffer. A

trial is ended when one of the two players retrieves either a chunk containing an 'accept' or 'quit' slot, or when there is a timeout. After a trial is over, the points each player made in that trial are calculated and the trial's outcome is saved in lisp code for analysis purposes. Though the model does not actively keep track of its own total score, it does keep track of how well it is doing. This is done by evaluating every trial as either positive or negative and by using these performance evaluations to influence the next trial. The way these evaluations are set up is discussed in more detail below.

4.2.1 Technical description of the model

As described above, the model chooses its moves by retrieving a move-chunk from the declarative memory. These moves are not only retrieved based on the model's own MNS value and offer, but also on what the opponent is doing. Specifically, move-chunks were set up in the following way:

```
(chunk-type move evaluation agent1mns-bid-difference agent1points agent2action agent2move  
agent1action agent1move agent1mns agent2value new-mns)
```

In which evaluation contains information about how the model is performing, agent1mns-bid-difference is the difference between the model's MNS value and its previous offer, agent1points is how much points the model would get from the opponent's current offer, agent2action is the opponent's current action (i.e. a bid, final offer, or quit), agent2move represents how much the opponent has moved since its previous offer, agent1action is the action the model is going to make in this round (i.e. a bid, final offer, concede, insist, accept, or quit), agent1move contains how much the model is going to change its current offer (i.e. current offer – agent1move), agent1mns contains the model's MNS value, agent2value represents the opponent's MNS, and new-mns represents the MNS value the model presents to the opponent (which can be a lie).

To incorporate the anger manipulation, a performance buffer was created. At the beginning of each trial, an evaluation chunk is retrieved, containing an evaluation slot with either the value 'positive' or 'negative'. This slot is then placed in the performance buffer, which, by using spreading activation, influences the choice the model makes when retrieving a move. The idea for this set up comes from an article by Barret (2005), who suggests that emotions are not discrete categories, but rather a summation of valence evaluation over time. In the current model this is thus applied by not setting the model in a constant angry state, but rather let it accumulate negative or positive evaluations, resulting in a different strategy chosen.

To accumulate these evaluations, at the end of each trial the model evaluates the outcome. A trial is considered positive when the model does not lose points and negative when it does. When a trial ends in a timeout, an evaluation is randomly chosen. This was done since in the first experiment there were negotiations that ended in a timeout simply because participants were still typing, but not necessarily because the negotiation was going particularly good or bad. If a trial ended in one player quitting, the trial was always evaluated as being negative. The evaluations were processed by retrieving a positive or negative evaluation chunk from the declarative memory, and thus giving that chunk a higher activation level, making it more likely that that chunk would be retrieved in the next trial. The feedback used in the first experiment was simulated by letting the model retrieve a positive or negative chunk multiple times, making it more likely for multiple trials in the second half to be played with a negative evaluation in the performance buffer.

To influence the strategy the model chooses, chunks containing a slot in which agent1action was either insist or quit, more often contained an evaluation slot that was set to negative. A lie was simulated by letting the model change its MNS value at the beginning of a trial. This was done by choosing an opening offer, which contained an evaluation slot and a new-mns slot. While in most chunks the agent1mns and new-mns slots had the same value, in some they differed. If the model had a negative evaluation slot in the performance buffer, the new-mns slot was more likely to differ from the real MNS value. While in experiment 1 most lies told pertained to not being able to accept a lower offer while a participants MNS value was 1 point or more below the offer, it was thought that by raising the model's MNS value it would also give a final offer sooner. Finally, the timeouts were

implemented using the (mp-time-ms) function, which keeps track of the systems time. The model code can be found in appendix B.

4.3 Model results

The model was set up to play against itself. In other words, the model was duplicated and the two instances of the model represented the two players. A log file kept track of the trial outcomes (i.e. deals, quits, and timeouts), points, and strategies (i.e. lies and insistence). The figures of the trial outcomes of the first experiment and 1000 simulations of the model can be found in figure 6. The numbers of lies and insistence can be found in figure 7.

As can be seen in figure 6, the model captures the trend that fewer agreements are reached in the anger condition after the feedback compared to before the feedback. In addition, more trials end in a timeout or a player quitting in the anger condition as a result of the feedback. For the control condition, the model also captures the distribution of the outcomes well, except for the number of trials ending in a player quitting. Here the model shows a significant drop in quit trials after the feedback. This could be explained intuitively by saying that people who think they are doing well in a game become more relaxed and feel they do not have to be too competitive to reach a good outcome.

Figure 7 also shows that the distribution of the model's outcomes and those of the first experiment are similar. For both the number of lies and the number of insistence the model shows that they increase in the anger condition as a result of the feedback. While for the number of lies the distribution of the data from the control condition is also nicely captured by the model, the number of insistence shows a large drop in the model's data, but not in the data of the first experiment.

Overall, the model captures the distribution of the results found in the first experiment. The actual counts, however, tend to deviate. However, given the small sample size in the first experiment ($N_{\text{control}} = 16$, $N_{\text{anger}} = 18$) and the rather large standard errors, it is difficult to pinpoint with certainty what the correct counts would be and to correctly model these counts. Attempting to do so with the current data would probably lead to overfitting of the model.

Figure 6: Trial outcomes of the first experiment and the model

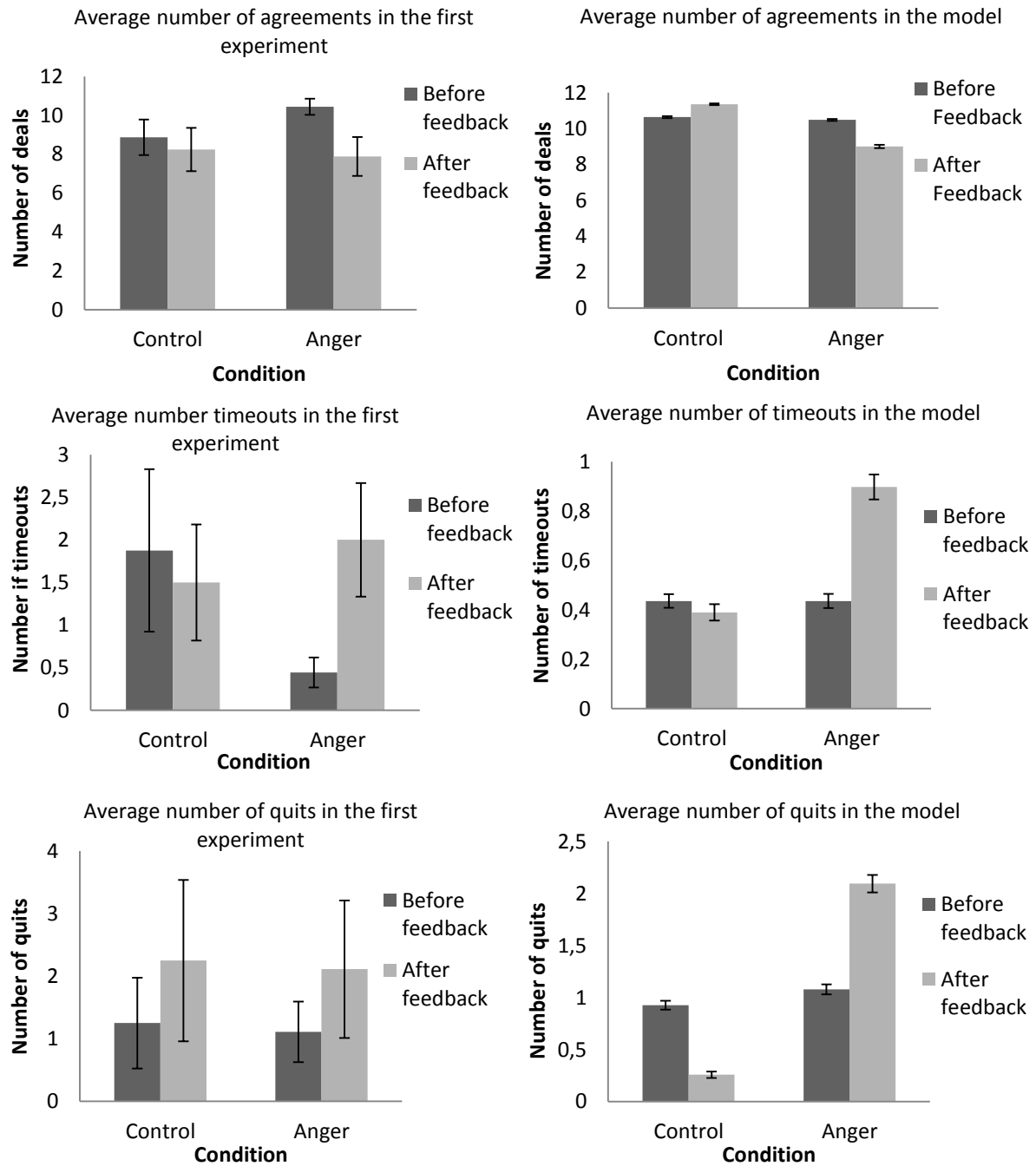
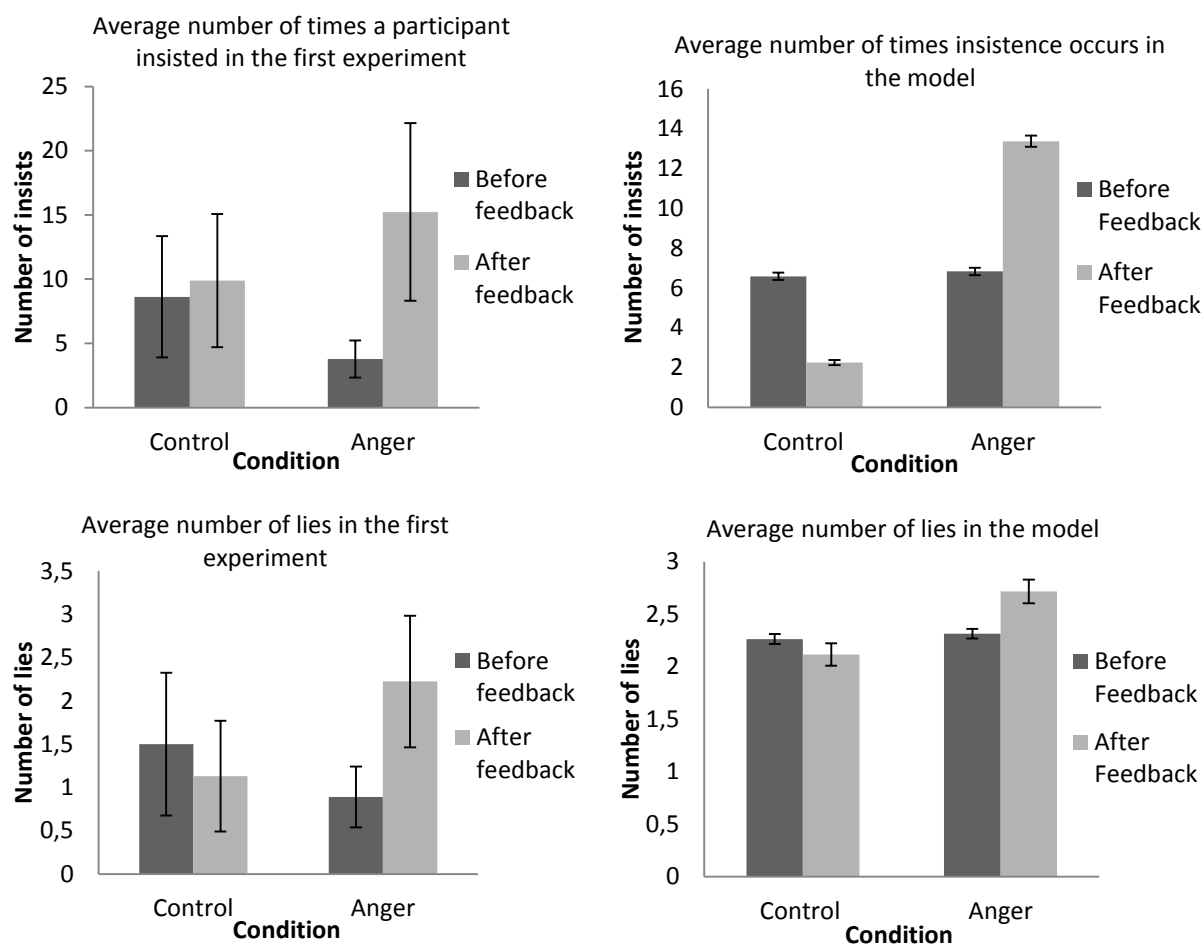


Figure 7: Average counts of insistence and lies in the first experiment and in the model



5. Experiment 2

In the second experiment, participants play against the cognitive model of the game of nines. The goal of this experiment is both to see whether the results of the participants match the results of the first experiment, and to see whether the model is capable of effectively adjusting its behaviour when it keeps track of how it is performing. Similar to the first experiment, in the second experiment the players are given feedback on their performance. This is done to trigger a sense of an unfair game in the manipulated condition.

Overall, it is expected that the dyads in the anger condition reach fewer agreements, have more timeouts and quit trials more often. In addition, it is expected that players lie and insist on an offer more often when they receive negative feedback.

5.1 Method

5.1.1 Participants

40 participants were recruited via Facebook. 11 participants were male. The mean age was 24.8, (SD = 4.9, Min. = 18, Max. = 40). Participants received a monetary reward for participating.

Four participants reported knowing about the goal of the experiment or explicitly reported they did not believe the feedback. These participants were excluded from further analysis. The remaining 36 participants, 8 male, had a mean age of 25.4 (SD = 5.0, Min. = 18, Max. = 40). Of the 36 participants, 13 were in the anger condition.

5.1.2 Materials

The experiment was conducted on a macbook pro, set-up in its own cubicle in the ALICE-lab at the University of Groningen. Participants played against the act-r model via an interface created using the tcl-tk³ package in lisp. Via this interface, participants were given their MNS value for each round and were able to make offers to the model by selecting a number and clicking 'make offer'. This would print out both their request and offer to the chat screen (e.g. 'I want 5 points, I offer you 4 points'). In addition, participants were given the option to select the 'I can't go any lower' box as well as a number, which would print out 'I can't go any lower than [selected number] points'. To end a trial, participants could click the 'accept offer' button to accept the model's offers, or the 'quit round' button to quit the round. Every trial lasted a maximum of 2 minutes, after which the interface would print 'time's up' to the chat-screen. To navigate to the next round after an outcome has been reached, participants could click the 'next round' button. The interface can be found in figure 8.

After the 12th trial, feedback was printed to the screen. Participants in the control condition were told they were playing the hard version of the game and both their and their opponents scores were printed below. In the anger condition, participants were told they were playing the easy version of the game. In addition, their scores were manipulated to make it seem like they were playing worse than the computer. More specifically, the scores of both players were added up and participants were made to believe they only had 1/3 of the total score while the computer had the remaining, and thus more, points.

After the game was played, participants were asked to fill out one questionnaire. This questionnaire contained questions about their gender, age and nationality. In addition, participants were asked to rate on a 5 point Likert scale (ranging from '1: not at all' to '5: very much') how honest they thought the opponent was, whether they thought the opponent was better than they were at this game, whether the opponent played fair, and whether they themselves played fair. Furthermore, participants were asked what they thought the goal of the study was and whether they believed the feedback they received.

³ <https://www.tcl.tk/>

Figure 8: Interface between the model and the participant

The interface is a web-based application with a light gray background. At the top left, there is a text input field labeled 'pp-number:' containing the value '1', and a button labeled 'Start new game'. At the top right, it says 'Participant number: 1' and 'You are player 1'. In the center, a large white box with a thin gray border contains the following text: '*Round 1: player 1 begins*', 'Player 1: I want 5 points, I offer you 4 points.', 'Player 2: I want 6 points, I offer you 3 points.', 'Player 1: I want 4 points, I offer you 5 points.', 'Player 2: Deal.', '*Player 1 gets 4 points, player 2 gets 5 points*', and '*Click \'Next round\' to continue*'. To the left of this box, the text 'Your mns:' is followed by the number '4'. To the right of the box are three buttons: 'Next round', 'Quit round', and 'Accept offer'. At the bottom, there is a row of radio buttons labeled 'Your offer:' followed by numbers 1 through 8. To the right of these is a checkbox labeled 'I can't go any lower' and a button labeled 'Make offer'.

5.1.3 Procedure

After arriving participants were placed in a private cubicle. Instructions to the game were given in written form, explaining the game and showing the interface which they would use to play the game. After participants indicated they understood the instructions, they began playing the game. When they were finished, they were given the questionnaire.

Since multiple participants were present, and did not finish at the same time, participants were given a written debriefing about the purpose of the study. Before leaving, participants filled out the form necessary for their monetary reward and were thanked for their participation.

5.1.4 Analysis

Analysis was done using Rstudio. The post-game questionnaire was analysed using an independent t-test, with condition (control vs. anger) as the between-subjects factor. The game data was analysed using a glmer, using condition (control vs. anger) as the between-subjects factor and time (before vs. after feedback) as a within-subjects factor. The variables logged during the game are the outcome (i.e. deal, quit or timeout), the scores of both players, how many times the model evaluates its performance as either positive or negative, and the number of times either player insists and lies. Insistence is defined as a player making the same offer at least two times in a row in the same trial. A lie is counted when the player uses the 'I can't go any lower button' while the current offer is 1 points or more above their MNS value.

5.2 Results

5.2.1 Questionnaire

Participants were asked four questions after the game of nines were played. On the question 'I think the opponent was honest', participants in the control condition rated an average of 3.31 (sd = 1.11) and participants in the anger condition rated an average of 3.23 (sd = 1.17). This difference was not significant ($t = 0.17$, $p = 0.86$). There was however a difference between the control condition and the anger condition in whether they thought their opponent was better ($t = -2.69$, $p < 0.05$). Participants in the anger condition rated their opponents to have played better than participants in the control condition rated their opponents to have done ($M_{\text{Anger}} = 2.85$, $sd = 1.41$; $M_{\text{Control}} = 1.54$, $sd = 1.05$). When asked whether participants thought their opponent played fair, participants in the anger condition gave a slightly lower rating ($M = 2.85$, $sd = 0.96$) than participants in the control condition

($M = 3.62$, $sd = 1.21$), though this difference was only marginally significant ($t = 1.79$, $p = 0.087$). Finally, the questionnaire did not indicate that participants played less fair as a result of the manipulation ($M_{\text{Anger}} = 2.85$, $sd = 1.28$; $M_{\text{Control}} = 2.69$, $sd = 0.95$; $t = -0.35$, $p = 0.73$).

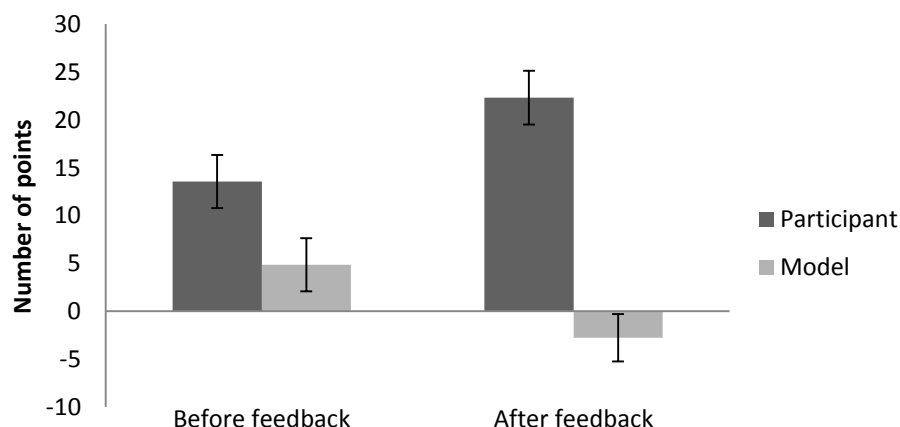
5.2.2 Points

On average, 9.04 ($sd = 13.62$) points were scored by the model and participants in the control condition before the feedback, and 9.73 ($sd = 17.45$) after feedback. For the anger condition these numbers were 9.35 ($sd = 15.90$) and 9.81 ($sd = 19.69$), respectively. A glmer did not show a significant main effect for condition ($Z = 0.17$, $p = 0.86$) nor for feedback ($Z = 0.38$, $p = 0.70$). In addition, no interaction effect was found between the condition and the feedback ($Z = -0.09$, $p = 0.93$).

When looking at the participant and the model separately, participants in the control condition scored an average of 14.31 ($sd = 12.53$) before the feedback and 23.69 ($sd = 11.23$) after the feedback. Participants in the anger condition scored 12.77 ($sd = 16.05$) before the feedback and 20.92 ($sd = 17.27$) after the feedback. These results also showed no main effect for the condition participants were in ($Z = -0.51$, $p = 0.61$), but they did show a main effect for feedback ($Z = 3.32$, $p < 0.001$). These results indicate that participants obtained higher scores in the second half of the game. The lack of interaction effect between condition and feedback ($Z = -0.22$, $p = 0.83$), however, shows that this increase in scores was independent of the condition participants were in.

The modelled opponent in the control condition scored an average of 3.77 ($sd = 13.02$) before feedback and -4.23 ($sd = 9.28$) after feedback. For the anger condition these numbers were 5.93 ($sd = 15.62$) and -1.31 ($sd = 15.53$), respectively. A glmer of these results also showed a main effect for feedback ($Z = -3.57$, $p < 0.001$), though in a different direction than the participants' data. It seems the model obtained fewer points in the second half of the game. The scores were not influenced by the condition the model was in (main condition: $Z = -0.06$, $p = 0.95$; interaction condition*feedback: $Z = 0.42$, $p = 0.69$). An additional glmer indicated that these differences between the model and the participants were significant. More specifically, a main effect was found for the player (model vs participant; $Z = -4.13$, $p < 0.001$), suggesting that overall participants scored more points than the model. In addition, an interaction effect was found between the feedback and the player ($Z = -4.84$, $p < 0.001$), suggesting that while participants scored better after the feedback, the model scored worse. Figure 9 shows the average number of points participants and the model obtained before and after feedback.

Figure 9: Average number of points the players obtained before and after the feedback in the second experiment



5.2.3 Outcomes

In total, the 650 trials ended in 527 deals, 122 quits, and 1 timeout. In the control condition, an average of 10.08 ($sd = 1.85$) deals were made before the feedback and 11.23 ($sd = 2.05$) after the feedback. In the anger condition these numbers were 9.77 ($sd = 2.01$) and 9.46 ($sd = 2.22$),

respectively. A glmer showed no significant main effect for condition ($Z = -0.25$, $p = 0.80$) nor for feedback ($Z = 0.90$, $p = 0.37$). In addition, no interaction effect was found between the condition and feedback ($Z = -0.80$, $p = 0.42$).

In the control condition, players quit an average of 1.92 (sd = 1.85) trials before the feedback and 1.77 (sd = 2.05) after the feedback, while players in the anger condition quit an average of 2.15 (sd = 2.08) trials before the feedback and 3.54 (sd = 2.22) after the feedback. For these outcomes, a glmer also showed no significant main effect for condition ($Z = 0.55$, $p = 0.58$) nor for the feedback ($Z = -0.29$, $p = 0.77$). In addition, no interaction effect was found between the condition and the feedback ($Z = 1.54$, $p = 0.12$). These results indicate that the manipulation had no influence on the outcome types of the trials.

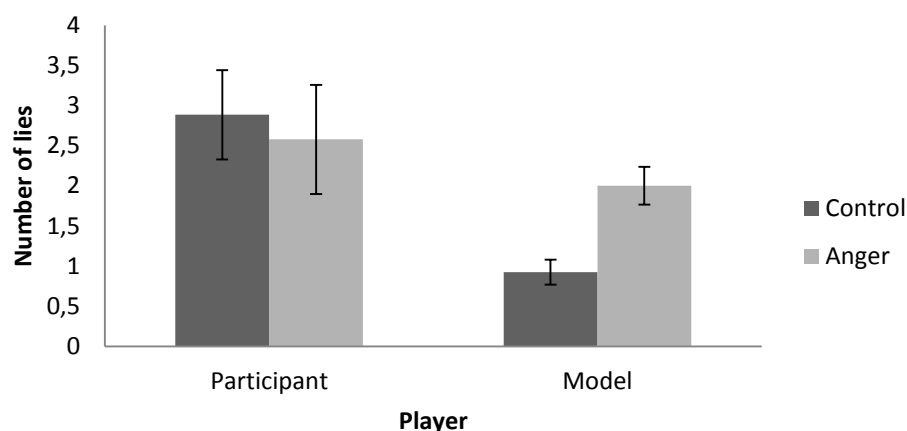
5.2.4 Lies

In total, 218 lies were told in the experiment, 142 of which were told by the participant. In the control condition, an average of 1.54 (sd = 1.88) lies before the feedback and 2.27 (sd = 2.62) after the feedback. In the anger condition, an average of 2.85 (sd = 1.77) lies were told before the feedback and 3.69 (sd = 3.09) after the feedback. A glmer showed no main effect for condition ($Z = 0.04$, $p = 0.97$) and only a marginally significant effect for feedback ($Z = 1.93$, $p = 0.05$). In addition, no interaction effect was found between the condition and feedback ($Z = 0.79$, $p = 0.43$). These results indicate that slightly more lies were told in the second half of the experiment and that this was unrelated to the condition in which the players were.

Participants in the control condition told an average of 2.23 (sd = 2.39) lies before the feedback and 3.54 (sd = 3.18) after the feedback, while participants in the anger condition told an average of 1.54 (sd = 2.33) lies before the feedback and 3.62 (sd = 4.15) after the feedback. A glmer of these results showed no main effect for condition ($Z = -0.93$, $p = 0.35$) and only a marginally significant effect for feedback ($Z = 1.95$, $p = 0.05$). In addition, no interaction effect was found ($Z = 1.10$, $p = 0.27$). These results suggest that participants told more lies after the feedback, thus in the second half of the game, but that this increase was unrelated to the condition participants were in.

The model told an average of 0.85 (sd = 0.80) lies before, and 1.00 (sd = 0.82) after the feedback when it was in the control condition. In the anger condition, the model told 1.69 (sd = 1.03) lies before the feedback and 2.31 (sd = 1.32) after the feedback. A glmer showed only a marginally significant main effect for the condition ($Z = 1.88$, $p = 0.06$), suggesting the model told slightly more lies when in the anger condition, even when it had not received feedback about its performance yet. No main effect was found for the feedback ($Z = 0.41$, $p = 0.68$) nor was there an interaction effect between the condition and the feedback ($Z = 0.29$, $p = 0.77$).

Figure 10: Number of lies participants and the model told in the different conditions



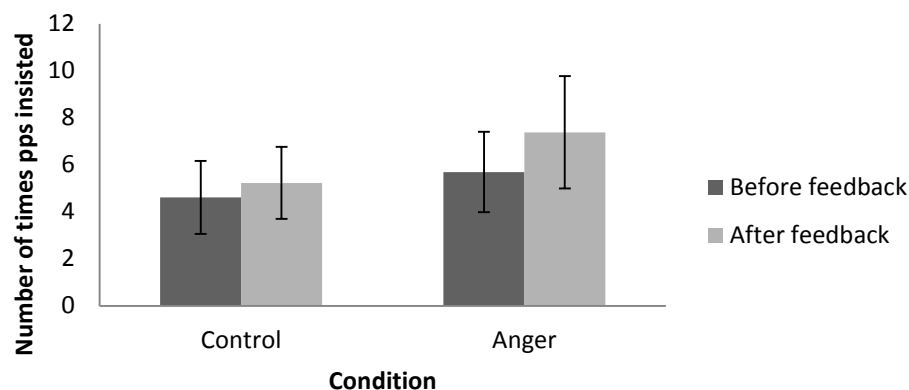
An additional glmer showed a significant difference between the amount of lies the different players (i.e. model or participant) told ($Z = -2.78$, $p < 0.01$). The results suggest that participants told

more lies than the model did. In addition, an interaction effect was found between the condition and the player ($Z = 2.30$, $p < 0.05$). While the number of lies participants told were about the same in the control and anger condition, the model told more lies in the anger condition than it did in the control condition. Figure 10 shows the average number of lies participants and the model told in the different conditions.

5.2.5 Insistence

Due to technical mistakes, only the number of times the participants insisted was logged during the experiment. In total, participants insisted 298 times in the 650 trials. Participants in the control condition, on average, insisted 4.62 (sd = 5.61) times before the feedback and 5.23 (sd = 5.53) after the feedback. In the anger condition, these numbers were 5.69 (sd = 6.17) and 7.38 (sd = 8.64), respectively. A glmer showed a main effect for condition ($Z = 100.9$, $p < 0.001$) as well as feedback ($Z = 53.1$, $p < 0.001$). In addition, an interaction effect was found between condition and feedback ($Z = 57.4$, $p < 0.001$). Figure 11 shows the average number of times participants insisted on an offer in the different conditions, before and after they received feedback. These results indicate that participants tended to insist more often in the anger condition. In addition, participants overall insisted more often after they received feedback. Finally, the interaction effect suggests that participants in the anger condition were more influenced by the feedback than participants in the control condition, resulting in an increase in the number of times they insisted on an offer.

Figure 11: Number of times the participant insisted on an offer before and after the feedback in the different conditions



5.3 Discussion results experiment 2

Overall, the results from the questionnaire indicate that the manipulation in the second experiment had some effect. Participants in the anger condition thought their opponent was playing better and participants in the anger condition had a slight tendency to think that their opponent was playing less fair than participants in the control condition did. The manipulation, however, did not result in differences in trial outcomes (i.e. deals, quits and timeouts). When looking at the number of lies both players told, there were only marginally significant outcomes due to players receiving feedback or due to condition, but no interaction effect between the condition and the feedback. It was found, however, that participants played more deceitful than the model did. In addition, when taking the player into account, there was an indication that the model played more deceitful as a result of the condition than participants. When looking at the number of times a participants insisted on an offer, the expected interaction effect between condition and feedback was found. Participants tended to insist more often on a certain offer after they received feedback in general as well as an increase in insistence when in the anger condition. In addition, the effect of the feedback was more pronounced for participants in the anger condition.

Comparing the results of the second experiment to the first experiment and the model's results, there are no similarities in the distribution of the trial outcomes. More specifically, in the

second experiment, there were no significant differences in the trial outcomes while in the first experiment and in the model's results the anger condition led to fewer agreements after the feedback and more timeouts and quits. There were, however, similarities in the strategies. In the both experiments, as well as in the model's results, there was an increase in the number of lies told and in the number of times a player insisted.

One explanation for the differences in the results of the second and the first experiment is that in the second experiment participants knew they were playing against a computer. Knowing that the opponent was a computer and not another person could have changed participants' strategy choices. In addition, aside from general differences in strategy choices, participants may not have gotten as angry with the computer as they would have with another person. Finally, judging from the model's scores, it seems the model was not very robust when playing against a person compared to playing against a second version of itself. Therefore, participants might not have had to use a very aggressive playing style in order to get the amount of points they wanted since the model did not seem to drive a hard bargain.

6. Discussion

The goal of the current study was threefold. The first goal was to see what types of behaviours occur when people get angry when their opponent in a negotiation game played unfair. This was tested in the first experiment. The results showed that dyads that received negative feedback came to fewer agreements, quit more trials and had more timeouts. In addition it was found that participants in the anger condition tended to insist more on a certain offer after they received feedback and that the opponent was playing more deceitful. The second goal was to model the results of the first experiment in a cognitive model. The model created in this study successfully simulated the distribution of the game outcomes and the behavioural strategies as a result of the negative feedback it processed. The third goal of the experiment was to transform the cognitive model into an interactive agent against which participants could play. The results of this second experiment, did not show the expected distribution in game outcomes. The results did, however, indicate a difference in the strategies used as a result of the condition participants were in. More specifically, the computerized agent tended to lie more often in the anger condition, while the participants insisted more often on an offer after the negative feedback.

The first experiment did not show some of the hypothesized behavioural changes as a result of anger, such as concessions (e.g. Sinaceur & Tiedens, 2006; Sinaceur et al., 2011) or demands (Côte et al., 2013). The increase in number of lies players told as a result of anger is, however, consistent with the study by van Dijk et al. (2008), who found participants to be more deceitful when confronted with an angry opponent. One reason not more of the known behavioural responses were found is because the game of nines does not give much room for negotiation. The range of MNS values and points to be divided limits the amount of demands and concessions players can make before finding an agreeable outcome. In future studies, a multi-issue negotiation setting could be used in which participants have to negotiate on one case for a longer period of time. By having participants negotiate over multiple factors in one negotiation round, there would be more room for demands or concessions. In addition, each factor would be essential for the outcome. While in the game of nines participants might lose track of their total score and treat every trial as a new negotiation, in multi-issue negotiations every issue, or factor, would be part of the same negotiation and thus would be directly related to their total outcome.

The differences in trial outcomes found in the first experiment due to the manipulation might, however, also be construed as a change in behaviour. The increase in timeouts indicates that participants negotiated longer on single trials, which suggests they become more willing to actively fight for their points. The decrease in deals and increase in quit trials after the negative feedback also suggests that participants were less cooperative. A quit trial suggests that participants were not just unwilling to accept a low offer, but also that they did not want their opponents to get any points. This would be congruent with the study by Seip et al. (2014) who found that participants who were confronted with an angry opponent were more prone to retaliatory actions. Future studies should check the reason behind a quit trial, for example by letting participants write down the reasoning behind their actions during the negotiation.

The cognitive model made based on the findings of the first experiment proved to be effective in simulating the behavioural changes and the resulting trial outcomes. The implementation of the performance buffer, based on the idea by Barret (2006) that emotion occurs as a result of multiple valence evaluations over time, seems an effective way to model emotion in a cognitive agent. The results of the second experiment however, while suggesting changes in the strategies used, did not completely prove the model to be useful in an interactive game with participants. The total score outcome of the model averages to a negative number, suggesting the model accepted too many offers that were below its MNS value. Though the current study was more interested in the behavioural factors in negotiation, the low scores of the model suggests that it was probably too lenient when playing against a person. However, the fact that the model ended up with fewer points than the participant could also be congruent with the findings by van Dijk et al. (2008) who found

that participants who express anger wound up with lower outcomes. In addition to the model not being robust enough to negotiate with a person, it is arguable that people in general react differently when playing against another person compared to playing against a computer. It could be that participants remained more rational and calm when playing against the computerized model since they do not perceive the computer to have bad intentions but rather perceive it to play according to a certain set of rules. Future studies should take behavioural and emotional differences caused by a person playing against a computer versus playing against another person into account.

Overall, the current study suggests that behavioural changes as a result of anger can indeed be modelled. The model in the current study, however, does not seem robust enough to be transformed into an interactive agent that can be used for negotiation training. In addition, since the current study was mostly interested in the behavioural differences due to anger, some other factors of negotiation might be left out, for example, choosing the best strategy for a good outcome instead of letting emotions do the choosing. Future studies should focus on how the mechanisms of modelled emotion fit into known computerized training agents. This way, the emotional factor becomes part of a bigger picture, instead of being the entire picture. In addition, future studies should check whether the mechanism used in the current model also works for other types of emotions.

7. Conclusion

The goal of the current study was to find behavioural differences in a negotiation setting due to anger and effectively model these behaviours in an interactive cognitive agent. In the current study, anger was caused by a perceived unfair opponent. The findings of the first experiment show that anger led to fewer agreements, more quits, and more timeouts. In addition, participants tended to lie more and insist on an offer more often as a result of anger. These findings were successfully modelled in a cognitive agent. When transforming this model into an interactive agent, however, the model did not perform as expected. While there were still behavioural differences in both the model and the participants, there were no differences in trial outcomes. In addition, the model did not perform well in terms of total score. Overall it seems that, although anger was successfully modelled, more research has to be done to make an effective interactive cognitive agent that can process anger that can be used in negotiation training.

3. References

- Adam, H., & Shirako, A. (2013). Not all anger is created equal: The impact of the expresser's culture on the social effects of anger in negotiations. *Journal of Applied Psychology, 98*(5), 785-798. doi:10.1037/a0032387
- Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist, 51*(4), 355-365. doi:10.1037/0003-066X.51.4.355
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*(4), 1036-1060. doi:10.1037/0033-295X.111.4.1036
- Anderson, J. R., Matessa, M., & Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction, 12*(4), 439-462. doi:10.1207/s15327051hci1204_5
- Baron, R. A., Byrne, D., & Branscombe, N. R. (2006). Social psychology (11th ed.) Upper Saddle River, N.J., USA: Pearson Education, inc.
- Barrett, L. F. (2006). Valence is a basic building block of emotional life. *Journal of Research in Personality, 40*(1), 35-55. doi:10.1016/j.jrp.2005.08.006
- Chan, M. E., & Arvey, R. (2011). The role of forgivingness and anger in unfair events. *Personality and Individual Differences, 50*(5), 700-705. doi:10.1016/j.paid.2010.12.021
- Côté, S., Hideg, I., & van Kleef, G. A. (2013). The consequences of faking anger in negotiations. *Journal of Experimental Social Psychology, 49*(3), 453-463. doi:10.1016/j.jesp.2012.12.015
- Dehghani, M., Carnevale, P. J., & Gratch, J. (2014). Interpersonal effects of expressed anger and sorrow in morally charged negotiation. *Judgment and Decision Making, 9*(2), 104-113.
- Espinoza, F., Fedorikhin, A., & Srivastava, J. (2006). Anger in ultimatum bargaining: Emotional outcomes lead to irrational decisions. *Advances in Consumer Research, 33*(1), 264-265.
- Gauvin, S., Lilien, G. L., & Chatterjee, K. (1990). The impact of information and computer based training on negotiators' performance. *Theory and Decision, 28*(3), 331-354. doi:10.1007/BF00162704
- Hegtvedt, K. A., & Killian, C. (1999). Fairness and emotions: Reactions to the process and outcomes of negotiations. *Social Forces, 78*(1), 269-303. doi:10.2307/3005797
- Kelley, H. H., Beckman, L. L., & Fischer, C. S. (1967). Negotiating the division of a reward under incomplete information. *Journal of Experimental Social Psychology, 3*(4), 361-398. doi:10.1016/0022-1031(67)90004-2

- Kersten, G. E., & Lai, H. (2007). Negotiation support and E-negotiation systems: An overview. *Group Decision and Negotiation*, 16(6), 553-586. doi:10.1007/s10726-007-9095-5
- Kong, D. T., Dirks, K. T., & Ferrin, D. L. (2014). Interpersonal trust within negotiations: Meta-analytic evidence, critical contingencies, and directions for future research. *Academy of Management Journal*, 57(5), 1235-1255. doi:10.5465/amj.2012.0461
- Lelieveld, G., Van Dijk, E., Van Beest, I., & Van Kleef, G. A. (2013). Does communicating disappointment in negotiations help or hurt? solving an apparent inconsistency in the social-functional approach to emotions. *Journal of Personality and Social Psychology*, 105(4), 605-620. doi:10.1037/a0033345
- Liu, M., & Wang, C. (2010). Explaining the influence of anger and compassion on negotiators' interaction goals: An assessment of trust and distrust as two distinct mediators. *Communication Research*, 37(4), 443-472. doi:10.1177/0093650210362681
- Nabi, R. L. (2002). Anger, fear, uncertainty, and attitudes: A test of the cognitive-functional model. *Communication Monographs*, 69(3), 204-216. doi:10.1080/03637750216541
- Pillutla, M. M., & Murnighan, J. K. (1996). Unfairness, anger, and spite: Emotional rejections of ultimatum offers. *Organizational Behavior and Human Decision Processes*, 68(3), 208-224. doi:10.1006/obhd.1996.0100
- Ross, W. H. J., Pollman, W., Perry, D., Welty, J., & Jones, K. (2001). Interactive video negotiator training: A preliminary evaluation of the McGill negotiation simulator. *Simulation & Gaming*, 32(4), 451-468. doi:10.1177/104687810103200402
- Saunders, D. M., & Lewicki, R. J. (1995). Teaching negotiation with computer simulations: Pedagogical and practical considerations. *Negotiation Journal*, 11(2), 157-167. doi:10.1111/j.1571-9979.1995.tb00057.x
- Schoeninger, D. W., & Wood, W. D. (1969). Comparison of married and ad hoc mixed-sex dyads negotiating the division of a reward. *Journal of Experimental Social Psychology*, 5(4), 483-499. doi:10.1016/0022-1031(69)90039-0
- Seip, E. C., Van Dijk, W. W., & Rotteveel, M. (2014). Anger motivates costly punishment of unfair behavior. *Motivation and Emotion*, 38(4), 578-588. doi:10.1007/s11031-014-9395-4
- Sinaceur, M., & Tiedens, L. Z. (2006). Get mad and get more than even: When and why anger expression is effective in negotiations. *Journal of Experimental Social Psychology*, 42(3), 314-322. doi:10.1016/j.jesp.2005.05.002
- Sinaceur, M., Van Kleef, G. A., Neale, M. A., Adam, H., & Haag, C. (2011). Hot or cold: Is communicating anger or threats more effective in negotiation? *Journal of Applied Psychology*, 96(5), 1018-1032. doi:10.1037/a0023896

- Srivastava, J., Espinoza, F., & Fedorikhin, A. (2009). Coupling and decoupling of unfairness and anger in ultimatum bargaining. *Journal of Behavioral Decision Making*, 22(5), 475-489.
doi:10.1002/bdm.631
- Stevens, C. A., Taatgen, N. A., Crossen, F. (2015). A cognitive model of theory of mind in negotiation. Manuscript in preparation.
- Ulijn, J., Rutkowski, A. F., Kumar, R., & Zhu, Y. (2005). Patterns of feelings in face-to-face negotiation: A sino-dutch pilot study. *Cross Cultural Management*, 12(3), 103-118.
doi:10.1108/13527600510798097
- van Dijk, E., van Kleef, G. A., Steinel, W., & van Beest, I. (2008). A social functional approach to emotions in bargaining: When communicating anger pays and when it backfires. *Journal of Personality and Social Psychology*, 94(4), 600-614. doi:10.1037/0022-3514.94.4.600
- Van Kleef, G. A., De Dreu, Carsten K. W., & Manstead, A. S. R. (2006). Supplication and appeasement in conflict and negotiation: The interpersonal effects of disappointment, worry, guilt, and regret. *Journal of Personality and Social Psychology*, 91(1), 124-142. doi:10.1037/0022-3514.91.1.124
- Wang, L., Northcraft, G. B., & Van Kleef, G. A. (2012). Beyond negotiated outcomes: The hidden costs of anger expression in dyadic negotiation. *Organizational Behavior and Human Decision Processes*, 119(1), 54-63. doi:10.1016/j.obhdp.2012.05.002
- Williams, G. R., Farmer, L. C., & Manwaring, M. (2008). New technology meets an old teaching challenge: Using digital video recordings, annotation software, and deliberate practice techniques to improve student negotiation skills. *Negotiation Journal*, 24(1), 71-87.
doi:10.1111/j.1571-9979.2007.00167.x

Appendix A: Questionnaires in experiment 1

Please fill out:

Gender: male 0 female 0

Age: _____

Nationality: _____

Answer to what extent the following statements are applicable to you on a scale from 1 (Not at all) to 5 (very much)

1. I am competitive

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

2. I am lazy

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

3. I am honest

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

4. I am athletic

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

5. I am easily distracted

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

6. I get angry easily

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

7. I am social

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

8. I am smart

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

The following questions are about your current emotions. Please answer to what extent the statement is currently applicable to you on a scale from 1 (Not at all) to 5 (Very much).

9. I feel happy

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

10. I feel anxious

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

11. I feel angry

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

12. I feel unsafe

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

13. I feel wronged

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

14. I feel relaxed

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

The following questions are about how you view other people. Please answer how often a statement is applicable to you on a scale from 1 (Never) to 5 (Always).

15. I try to think the best of others

1	2	3	4	5
Never	Rarely	Sometimes	Often	Always
0	0	0	0	0

16. I think other people are lazy

1	2	3	4	5
Never	Rarely	Sometimes	Often	Always
0	0	0	0	0

17. I think other people are trustworthy

1	2	3	4	5
Never	Rarely	Sometimes	Often	Always
0	0	0	0	0

18. I think other people are smart

1	2	3	4	5
Never	Rarely	Sometimes	Often	Always
0	0	0	0	0

19. I think other people are selfish

1	2	3	4	5
Never	Rarely	Sometimes	Often	Always
0	0	0	0	0

[THE FOLLOWING QUESTIONS WERE ONLY PRESENTED AFTER THE GAME WAS PLAYED]

The following questions are about your opponent. Please answer to what extent do the statements below apply to your opponent on a scale from 1 (Not at all) to 5 (Very much). Your opponent will not get to see your answers.

I think my opponent ...

20. ...was nice

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

21. ...was honest

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

22. ...was better at this game than me

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

23. ...was angry with me

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

24. ...enjoyed the game

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

25. ...played fair

1	2	3	4	5
Not at all	a little	somewhat	quite a bit	very much
0	0	0	0	0

26. What do you think this experiment was about?

This is the end of the experiment.

Appendix B: Cognitive model

(define-model gonines1

(sgp :esc t :v t :bll .5 :ans .25 :mas 5 :mp 2 :ms 2.0 :w-hook adjust-buffer-w :rt -10 :performance-activation 1 :declarative-first-span 500 :blc 1 :lf 4 :le 0.1)

(chunk-type move evaluation agent1mns-bid-difference agent1points agent2action agent2move agent1action agent1move agent1mns agent2value new-mns)

(chunk-type evalfeedback condition processCount evaluation)

(chunk-type goal my-mns my-action my-last-offer my-offer my-give my-move opp-mns opp-action opp-last-offer opp-offer opp-give opp-move outcome timeout endtime endtrial status)

(chunk-type values values)

(chunk-type evaluation evaluation)

(add-dm

(positive isa values values pos)

(negative isa values values neg)

(conc-1 isa move evaluation negative agent1mns-bid-difference 3 agent1points -2 agent2action bid agent2move 1 agent1action concede agent1move 1)

(conc-2 isa move evaluation positive agent1mns-bid-difference 3 agent1points -1 agent2action bid agent2move 2 agent1action concede agent1move 2)

(conc-4 isa move evaluation positive agent1mns-bid-difference 2 agent1points -1 agent2action bid agent2move 2 agent1action concede agent1move 1)

(conc-5b isa move evaluation positive agent1mns-bid-difference 2 agent1points -5 agent2action bid agent2move 1 agent1action concede agent1move 1)

(conc-6 isa move evaluation negative agent1mns-bid-difference 3 agent1points -3 agent2action bid agent2move 0 agent1action concede agent1move 1)

(conc-7 isa move evaluation negative agent1mns-bid-difference 5 agent1points 2 agent2move 1 agent2action bid agent1action concede agent1move 1)

(insist-4 isa move evaluation negative agent1mns-bid-difference 1 agent1points 0 agent2action bid agent2move 0 agent1action insist agent1move 0)

(insist-1 isa move evaluation positive agent1mns-bid-difference 1 agent1points -1 agent2action bid agent2move 1 agent1action insist agent1move 0)

(insist-2 isa move evaluation positive agent1mns-bid-difference 0 agent1points -2 agent2action bid agent2move 0 agent1action insist agent1move 0)

(insist-9 isa move evaluation positive agent1mns-bid-difference 1 agent1points 0 agent2action bid agent2move 1 agent1action insist agent1move 0)

(insist-3 isa move evaluation negative agent1mns-bid-difference 3 agent1points -3 agent2action bid agent2move 0 agent1action insist agent1move 0)

(insist-5 isa move evaluation negative agent1mns-bid-difference 1 agent1points -1 agent2move 1 agent2action bid agent1action insist agent1move 0)

(insist-6 isa move evaluation negative agent1mns-bid-difference 2 agent1points 1 agent2move 1 agent2action bid agent1action insist agent1move 0)

(insist-7 isa move evaluation negative agent1mns-bid-difference 2 agent1points -2 agent2move 1 agent2action bid agent1action insist agent1move 0)

(insist-8 isa move evaluation positive agent1mns-bid-difference 0 agent1points 0 agent2move 1 agent2action bid agent1action insist agent1move 0)

(accept-1 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 3 agent2move 1
(accept-2 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 2 agent2move 1
(accept-4 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 3 agent2move 1
(accept-3 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 1 agent2move 1
(accept-5 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 0 agent1points 1 agent2move 1
(accept-6 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 0 agent1points 0 agent2move 1
(accept-7 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 0 agent1points 0 agent2move 1
(accept-8 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 1 agent2move 1
(accept-9 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 1 agent2move 1
(accept-10 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 0 agent2move 0
(accept-11 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 0 agent2move 0
(accept-12 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 0 agent1points -1 agent2move 1
(accept-13 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 0 agent2move 0
(accept-14 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 5 agent1points 0 agent2move 0
(accept-16 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 5 agent1points 2 agent2move 1
(accept-17 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 5 agent1points 2 agent2move 1
(accept-18 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 1 agent2move 0
(accept-19 isa move evaluation negative agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 1 agent2move 0
(accept-20 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 1 agent2move 2
(accept-21 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 2 agent1points 1 agent2move 2
(accept-22 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 3 agent1points 1 agent2move 2
(accept-23 isa move evaluation positive agent2action bid agent1action accept)	agent1mns-bid-difference 1 agent1points 0 agent2move 2
(quit-3 isa move evaluation positive agent2action bid agent1action quit)	agent1mns-bid-difference -1 agent1points 0 agent2move 0
(quit-5 isa move evaluation negative agent2action bid agent1action quit)	agent1mns-bid-difference -2 agent1points -1 agent2move 0

; Final offer stuff

(final-0 isa move evaluation positive agent1mns-bid-difference 1 agent1points 0 agent2action bid agent2move 0 agent1action final agent1move 0)
(final-2a isa move evaluation negative agent1mns-bid-difference 1 agent1points 0 agent2action bid agent2move 0 agent1action final agent1move 0)

(final-4 isa move evaluation positive agent1mns-bid-difference 2 agent1points 1 agent2action final agent2move 0 agent1action accept)
(final-5 isa move evaluation positive agent1mns-bid-difference 1 agent1points 0 agent2action final agent2move 0 agent1action accept)
(final-10 isa move evaluation positive agent1mns-bid-difference 0 agent1points 1 agent2action final agent2move 0 agent1action accept)
(final-11 isa move evaluation positive agent1mns-bid-difference 1 agent1points 1 agent2action final agent2move 0 agent1action accept)
(final-7 isa move evaluation negative agent1mns-bid-difference 2 agent1points -2 agent2action final agent2move 0 agent1action quit)
(final-8 isa move evaluation negative agent1mns-bid-difference 1 agent1points 0 agent2action final agent2move 0 agent1action accept)
(final-9 isa move evaluation negative agent1mns-bid-difference 2 agent1points 1 agent2action final agent2move 0 agent1action accept)
(final-12 isa move evaluation negative agent1mns-bid-difference 1 agent1points 0 agent2action final agent2move 0 agent1action accept)
(final-13 isa move evaluation negative agent1mns-bid-difference 2 agent1points 0 agent2action final agent2move 0 agent1action accept)

; Initial offers

(initial-1 isa move evaluation positive agent1mns 1 agent2action bid agent2value 6 agent1action bid agent1move 4 new-mns 2)
(initial-19 isa move evaluation positive agent1mns 1 agent2action bid agent2value 6 agent1action bid agent1move 3 new-mns 1)
(initial-2 isa move evaluation positive agent1mns 4 agent2action bid agent2value 6 agent1action bid agent1move 5 new-mns 4)
(initial-3 isa move evaluation positive agent1mns 3 agent2action bid agent2value 5 agent1action bid agent1move 5 new-mns 3)
(initial-4 isa move evaluation positive agent1mns 5 agent2action bid agent2value 8 agent1action bid agent1move 6 new-mns 5)
(initial-10 isa move evaluation positive agent1mns 4 agent2action bid agent2value 6 agent1action bid agent1move 6 new-mns 4)
(initial-12 isa move evaluation positive agent1mns 5 agent2action bid agent2value 8 agent1action bid agent1move 6 new-mns 5)
(initial-13 isa move evaluation positive agent1mns 2 agent2action bid agent2value 7 agent1action bid agent1move 4 new-mns 2)

(initial-18 isa move evaluation negative agent1mns 5 agent2action bid agent2value 8 agent1action bid agent1move 7 new-mns 5)
(initial-7 isa move evaluation negative agent1mns 1 agent2action bid agent2value 6 agent1action bid agent1move 6 new-mns 3)
(initial-8 isa move evaluation negative agent1mns 6 agent2action bid agent2value 7 agent1action bid agent1move 8 new-mns 6)
(initial-9 isa move evaluation negative agent1mns 3 agent2action bid agent2value 7 agent1action bid agent1move 6 new-mns 3)

```

(initial-14 isa move evaluation negative agent1mns 1 agent2action bid agent2value 6 agent1action
bid agent1move 5 new-mns 2)
(initial-15 isa move evaluation negative agent1mns 4 agent2action bid agent2value 6 agent1action
bid agent1move 6 new-mns 4)
(initial-23 isa move evaluation negative agent1mns 4 agent2action bid agent2value 6 agent1action
bid agent1move 6 new-mns 5)
(initial-16 isa move evaluation negative agent1mns 1 agent2action bid agent2value 5 agent1action
bid agent1move 4 new-mns 1)
(initial-17 isa move evaluation negative agent1mns 2 agent2action bid agent2value 8 agent1action
bid agent1move 6 new-mns 2)

```

```

(initial-21 isa move evaluation positive agent1mns 2 agent2action final agent2value 8 agent1action
quit agent1move 6 new-mns 2)
(initial-20 isa move evaluation negative agent1mns 2 agent2action final agent2value 8 agent1action
quit agent1move 6 new-mns 2)
(initial-24 isa move evaluation positive agent1mns 2 agent2action final agent2value 8 agent1action
quit agent1move 6 new-mns 2)
(initial-25 isa move evaluation negative agent1mns 2 agent2action final agent2value 8 agent1action
quit agent1move 6 new-mns 2)
)

```

```

(p start-trial                                     ;this player starts

```

```

    =goal>
        isa goal
        my-last-offer nil
        my-offer nil
        opp-offer nil
        - opp-mns nil
        outcome nil
        timeout 0
        - my-mns nil

```

```

    ?retrieval>
        state free
        buffer empty

```

```

    ?performance>
        state free
        buffer empty

```

```

    ==>

```

```

    +retrieval>
        isa values
        ;;recently-retrieved nil

```

```

)

```

```

(p start-trial2                                     ;opponent starts

```

```

    =goal>
        isa goal
        my-last-offer nil
        my-offer nil
        opp-offer =oppoffer
        opp-last-offer nil
        - opp-mns nil

```

```

        outcome      nil
        timeout      0
-      my-mns        nil

?retrieval>
  state      free
  buffer     empty
?performance>
  state      free
  buffer     empty
==>
+retrieval>
  isa      values
  ;;recently-retrieved nil
)

(p get-performance2                                ;get an evaluation for the first trial
when the opponent starts
=goal>
  isa      goal
  my-last-offer  nil
  my-offer  nil
  opp-offer =oppoffer
  ;opp-mns  nil
  outcome   nil
  timeout   0
  my-mns    =mymns

=retrieval>
  isa      values
  ;values  =eval

?performance>
  state      free
  buffer     empty
==>
+performance>
  isa      evaluation
  evaluation =retrieval
-retrieval>
)
(p get-performance1                                ;get an evaluation for the first trial when this player starts
=goal>
  isa      goal
  my-last-offer  nil
  my-offer  nil
  opp-offer  nil
  ;opp-mns  nil
  outcome    nil
  timeout    0
-      my-mns    nil

```

```

=retrieval>
    isa          values
;    values      =eval

?performance>
    state        free
    buffer       empty
==>

+performance>
    isa          evaluation
    evaluation    =retrieval
-retrieval>
)
(p retrieve-offer1                                ;Retrieve first bid when this player
starts
    =goal>
        isa          goal
        my-last-offer nil
        opp-last-offer nil
        my-offer      nil
        opp-offer      nil                    ;i.e. opponent has not made
any offer yet
        ;opp-mns      nil
        outcome        nil
        timeout        0
        my-mns         =mymns

    =performance>
        isa          evaluation
        - evaluation    nil
    ?retrieval>
        state        free
        buffer       empty
==>

+retrieval>                                ;retrieve first move
    isa          move
    agent1action bid
    agent1mns     =mymns
    :recently-retrieved nil
=performance>
)
(p retrieve-offer2                                ;Retrieve first bid when opponent starts
    =goal>
        isa          goal
        my-last-offer nil                    ;no previous offers
        opp-last-offer nil
        my-offer      nil
        opp-offer      =oppoffer            ;i.e. opponent has made an offer
        ;opp-mns      nil
        outcome        nil
        timeout        0

```

```

my-mns =mymns
opp-action =oppaction
=performance>
isa evaluation
- evaluation nil
?retrieval>
state free
buffer empty
==>
+retrieval>
isa move
agent1action bid
agent1mns =mymns
agent2action =oppaction
; agent2value =oppoffer
:recently-retrieved nil
=performance>
)

(p make-first-move ;when either player begins
=goal>
isa goal
my-last-offer nil
my-offer nil
my-give nil
;opp-offer nil
;opp-mns nil
opp-last-offer nil
outcome nil
timeout 0
my-mns =mymns
;opp-action nil
my-move nil

=retrieval>
isa move
agent1action =myaction
;agent1mns =mymns
agent1move =myopeningbid
new-mns =newmns

=performance>
isa evaluation
evaluation =eval

!bind! =mymove (- =myopeningbid =newmns)
!bind! =mygive (- 9 =myopeningbid)

==>
=goal> ;update the goal
with player 1's move and strategy
my-offer =myopeningbid

```

```

my-give          =mygive
status           wait                               ;wait for opponent
my-action        =myaction
my-move          =mymove
my-mns           =newmns

-retrieval>                                           ;remove current
move
=performance>
!eval! (if (< =mymns =newmns) (setf *lie1* (+ *lie1* 1)))
)
(p respond-opp-move1                                ;respond to
opponent's openingsbid
=goal>
  isa
  my-last-offer nil goal ;still first offers, so
no previous offers
  opp-last-offer nil
  my-offer =myoffer ;both
players have made their first offers
- my-give nil
  opp-offer =oppoffer
  opp-give =oppgive
  outcome nil
  timeout 0
  my-mns =mymns
  opp-mns =oppmns
  opp-action =oppaction
; my-move =mymove
  status active
  ;opp-move nil

=performance>
  isa evaluation
  evaluation =eval

?retrieval>
  buffer empty
  state free
!bind! =mymove (- =myoffer =mymns)
!bind! =oppmove (- =oppoffer =oppmns)
!bind! =agent1points (- =oppgive =mymns)
==>
=performance>
+retrieval>
  isa move
  agent1mns-bid-difference =mymove
  agent2move =oppmove
  agent1points =agent1points
  agent2action =oppaction
:recently-retrieved nil

```



```

=goal>
    ;my-move
    opp-move
)
(p respond-opp-move2
opponent's openingsbid
=goal>
    isa
    my-last-offer
    opp-last-offer
    my-offer
    - my-give
    opp-offer
    outcome
    timeout
    my-mns
    opp-mns
    opp-action
    my-action
    status
=retrieval>
    isa
    agent2action
    agent1action
    agent1move

=performance>
    isa
    - evaluation
    !bind! =mynewoffer (- =myoffer =mymove)
offer - agent1's-move/mymove
    !bind! =mygive (- 9 =mynewoffer)
==>
=goal>
    my-last-offer
    my-offer
    my-give
    opp-last-offer
    opp-action
    my-action
    my-move
    opp-offer
    status
=performance>
-retrieval>
)
(p respond-other-moves
=goal>
    isa
    my-last-offer
    opp-last-offer
    my-offer

```

```

=mymove
=oppmove

goal
nil
nil
=myoffer
nil
=oppoffer
nil
0
=mymns
=oppmns
=oppoldaction
=myoldaction
active

move
=oppaction
=myaction
=mymove

evaluation
nil
;new offer is old

=myoffer
=mynewoffer
=mygive
=oppoffer
=oppaction
=myaction
=mymove
nil
wait

goal
=mylastoffer
=opplastoffer
=myoffer

```

```

-      my-give      nil
      opp-offer    =oppoffer      ;
      opp-give     =oppgive
      outcome      nil
      timeout      0
      my-mns       =mymns
;      my-move     =mymove
      opp-action   =oppaction
      status       active
      ;endtime     =endtime
?retrieval>
      buffer       empty
      state        free
=performance>
      isa          evaluation
!bind! =oppmove (- =opplastoffer =oppoffer)
!bind! =mymove (- =myoffer =mymns)
!bind! =agent1points (- =oppgive =mymns)
==>
=goal>
      opp-move     =oppmove

+retrieval>
      isa          move
      agent2move   =oppmove
      agent1mns-bid-difference =mymove
      agent2action =oppaction
      agent1points =agent1points
      :recently-retrieved nil
=performance>
)

(p respond-other-moves2
=goal>
      isa          goal
      my-last-offer =mylastoffer
      opp-last-offer =opplastoffer
      my-offer      =myoffer
-      my-give      nil
      opp-offer    =oppoffer      ;
      outcome      nil
      timeout      0
      my-mns       =mymns
      status       active

=retrieval>
      isa          move
      agent1action =myaction
      agent1move   =mymove
=performance>
      isa          evaluation
!bind! =mynewoffer (- =myoffer =mymove)

```

```

!bind! =mygive (- 9 =mynewoffer)

==>
=goal>
    my-last-offer
    my-offer
    my-give
    opp-last-offer
    opp-offer
    my-action
    status
=performance>
-retrieval>
)

(p deal
    =goal>
        isa
        outcome
        timeout
        ;my-action
        my-offer
        opp-offer
        endtrial
    =performance>
        isa
    =retrieval>
        isa
        agent1action
        agent1move

    goal
    nil
    0
    nil
    =myoffer
    =oppoffer
    0
    evaluation
    move
    accept
    nil

==>
=goal>
    my-action
    my-last-offer
    opp-last-offer
    opp-offer
    endtrial
    -performance>
    -retrieval>
;!eval! (setf *outcome* 'deal)
)

(p timeout3 ;when timeout on other player's turn
    =goal>
        isa
        ;opp-action
        outcome
        -
        status
        timeout
        endtrial
    goal
    quit
    nil ;outcome is quit or deal. (or
    active
    1
    0

```

```

=performance>
  isa
==>
  =goal>
  -performance>
)
(p quit
  =goal>
    isa
    outcome
    timeout
    ;my-action
    my-offer
    opp-offer
    endtrial
  =performance>
    isa
  =retrieval>
    isa
    agent1action
==>
  =goal>
    my-action
    my-last-offer
    opp-last-offer
    opp-offer
    endtrial
  -performance>
  -retrieval>
;!eval! (setf *outcome* 'quit)
)

(p other-quit-deal
  =goal>
    isa
    ;opp-action
    outcome
  -
  status
  timeout
  endtrial
  timeout?)

  =performance>
    isa
==>
  =goal>
  -performance>
)

(p eval-positive

```

evaluation

goal
nil
0
quit
=myoffer
=oppoffer
0

evaluation
move
quit

quit
=myoffer
=oppoffer
nil
1

goal
quit
nil ;outcome is quit or deal. (or
active
0
1

evaluation

```

=goal>
    isa
    outcome
    endtrial
    ?retrieval>
        buffer
        state
    ?performance>
        buffer
!eval! (>= *mytrialpoints* 0)
==>
    +retrieval>
        isa
        values
; !eval! (setf *positiveeval1* (+ *positiveeval1* 1))
)

(p eval-negative
    =goal>
        isa
        outcome
    ?retrieval>
        buffer
        state
    ?performance>
        buffer
!eval! (<= *mytrialpoints* 0)
==>
    +retrieval>
        isa
        values
; !eval! (setf *negativeeval1* (+ *negativeeval1* 1))
)

(p eval-positive2
    =goal>
        isa
        outcome
        timeout
        endtrial
    ?retrieval>
        buffer
        state
    ?performance>
        buffer
==>
    +retrieval>
        isa
        values
;
; !eval! (setf *positiveeval1* (+ *positiveeval1* 1))
)

```

1

goal
agreement

empty
free

empty

values
pos

goal
agreement

empty
free

empty

values
neg

0

goal
timeout

empty
free

empty

values
pos

```

(p eval-negative3
  =goal>
    isa
    outcome
    endtrial
  ?retrieval>
    buffer
    state
  ?performance>
    buffer
  ==>
    +retrieval>
      isa
      values
;
;!eval! (setf *negativeeval1* (+ *negativeeval1* 1))
)
(p save-eval
  =goal>
    isa
    - outcome
  =retrieval>
    isa
    values
  ==>
    !eval! (if (not (eq =eval 'neg)) (setf *positiveeval1* (+ *positiveeval1* 1)) (setf
    *negativeeval1* (+ *negativeeval1* 1)))
    -retrieval>
    -goal>

)

(p eval-negative-feedback
  =goal>
    isa
    condition
    processCount
  ?retrieval>
    buffer
    state
  ==>
    +retrieval>
      isa
      values
    =goal>
      evaluation
  !output! 'negative
)

(p eval-positive-feedback

```

```

=goal>
    isa                evalfeedback
    condition          (CONTROL)
    processCount =Pcount
?retrieval>
    buffer      empty
    state       free

==>
+retrieval>
    isa                values
    values             pos
=goal>
    evaluation         pos
!output! 'positive
)
(p loop-feedback
    =goal>
        isa                evalfeedback
        processCount =Pcount
    =retrieval>
        isa                values
    -    values             nil
!eval! (<= Pcount *process*)
==>
    !bind! =newPcount (+ Pcount 1)
    =goal>
        processCount =newPcount
    -retrieval>
)
(p save-feedback
    =goal>
        isa                evalfeedback
        processCount =Pcount
    =retrieval>
        isa                values
    -    values             nil
!eval! (>= Pcount *process* )
==>
    -retrieval>
    -goal>
)
)

```