Analyzing Human Negotiation using Automated Cognitive Behavior Analysis: The Effect of Personality

Pedro Sequeira (psequeira@ccs.neu.edu) and **Stacy Marsella (marsella@ccs.neu.edu)**
Northeastern University, 360 Huntington Ave.
Boston, 02115 MA

Abstract

In this paper we study the influence of personality traits in negotiation by using a methodology for *automated cognitive behavior analysis* (ACBA). This methodology uses genetic programming (GP) for hypothesis generation and testing of human behavior with the goal of explaining the underlying mental structures guiding people's actions during a task. ACBA iteratively generates programs—the *hypotheses*—capable of explaining the behavior exhibited by an individual during a multi-level, multi-issue, sequential bargaining task against an artificial agent. Our study focuses on the influence of the personality traits of social-value orientation (SVO) and Machiavellianism (Mach). The results show that by using ACBA, we are able to identify differences in the outcomes of programs emerging from GP that are consistent with the influences that different SVO and Mach profiles have in human negotiation behavior.

Keywords: Cognitive Behavior Understanding; Genetic Programming; Artificial Intelligence; Negotiation; Social Value Orientation; Machiavellianism

Introduction

We propose a methodology aimed at explaining the *underlying psychological structures* within the human mind responsible for the production of overt behavior. By psychological structures we refer to mental processes influencing decision-making during the performance of complex tasks. We refer to this methodology as *Automated Cognitive Behavior Analysis* (ACBA) to contrast with standard cognitive task analysis techniques relying on observation and thinking-aloud protocols (van Someren et al., 1994).

The main idea of ACBA is to use genetic programming (GP) as hypothesis generation and testing of human behavior, as illustrated in Fig. 1. In particular, we follow an information processing perspective to characterize human cognitive processes—symbols are created to represent knowledge about the self and the task; instructions are formed from these symbols and combined to generate plans in the form of programs; decision-making mechanisms then select the most appropriate plan to be executed in a given situation (Miller et al., 1960; Newell & Simon, 1961; Simon, 1978). This analogy is depicted in the top part of Fig. 1. The idea is to form possible hypotheses about the underlying cognitive structures of behavior—the program is a *hypothesis* of the response functions operating in an individual if it leads to outcomes consistent with those exhibited by his/her behavior.

Importantly, we follow this perspective to analyze behavior as being produced at a psychological rather than neurophysiological or physical level.[1] In addition, a key goal of ACBA is the identification of the underlying *invariant* behavioral structures across different situations rather than evolving programs solving for a particular instance. This involves aggregating the outcomes of GP in order to properly identify the common structures governing the behavioral phenomenon of interest.

In general terms, ACBA works by iteratively generating *candidate programs* capable of explaining the observed behavior of an individual after having performed some cognitive task. GP stochastically creates programs by combining certain operators with symbols encoding relevant task information. The fitness of each candidate program is evaluated according to how consistent its output is in relation to the outcomes of the exhibited behavior, as depicted in the bottom-right part of Fig. 1. By using selection and mutation operators, GP progressively generates candidates attaining higher and higher fitness. When GP finishes, the most fit programs are chosen as hypotheses of, *i.e.*, possible *solutions* for, the underlying structure of the observed behavior.

Contributions

In this paper we use ACBA to study human negotiation behavior. Negotiation is appealing for the purposes of our study for several reasons: it is a situation prone to conflict, involving complex decision-making in order to distribute limited amounts of resources (Pruitt, 1983); it is a task where people have to be strategic in order to gain the best possible outcome (Thompson, 1990); the behavior of an individual is highly influenced by that of his/her opponents (Galinsky & Mussweiler, 2001; Thompson, 1990); the problem is well defined, specifically the goals and actions of each negotiator and whether success in a negotiation was reached.

We are particularly interested in using ACBA to capture the variability of negotiation behavior according to the subjects' personality. A key question is *whether the solutions emerging from GP confirm the negotiation predispositions attributed to different personality traits* as described in the negotiation literature (*e.g.*, de Dreu & van Lange, 1995; Gunnthorsdottir et al., 2002). Given their influence on negotiation goals, dynamics and outcomes, we focus on the dimensions of *social value orientation* (SVO), assessing the weight attached to the welfare of others (Murphy et al., 2011), and *Machiavellianism* (Mach), which relates to tendencies of being deceptive and manipulative (Christie & Geis, 2013). To achieve that, in this paper we present a set of techniques to aggregate the outcomes of GP regarding the behavior of different individuals in the same negotiation task—the idea is to identify the underlying shared structures governing their behavior according to their personality traits.

[1] Our work is concerned with abstractions over the mental processes leading to observable behavior while putting aside the task of explaining the brain regions realizing such cognitive functions.

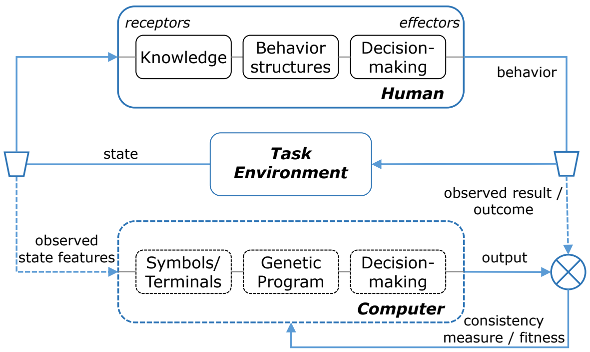

Figure 1: Our vision for *genetic programming as hypothesis generation and testing* of human behavior.

Background

Related Work

A common method for understanding human behavior is by performing *cognitive task analysis* (CTA) (Clark et al., 2008) or using *thinking-aloud* (TA) protocols (van Someren et al., 1994). These methods use diverse interview and observation strategies to capture the knowledge, thought processes and goal structures used by experts when performing some task. The output is a set of precise descriptions of the underlying cognitive processes of the experts, that can be used as records of task performance to *e.g.*, train novices by developing expert systems or build models of team performance. In fact, Newell & Simon (1961) recorded traces of humans reasoning while solving particular tasks to write programs using primitives that best simulated the observed behavior.

The main problem with these protocols is that they require subjects to interpret and rationalize their behavior while performing a task, making it more deliberate and slower (Simon, 1978). Moreover, subjects' interpretation of their behavior might seem more coherent and planful than it actually was (van Someren et al., 1994). Unlike these protocols, ACBA generates programs given relevant information features about a task,[2] as depicted in Fig. 1. As such, our method tries to explain behavior based on observable information, *all in an automated manner*. We argue that, as long as it can be accessed by an external observer, such features might be sufficient to explain an individual's behavior. Although the features need to be selected and encoded into the GP procedure, we claim that data collection using ACBA is more *objective*—does not depend on the subjects' opinions and introspection capabilities and does not interrupt task performance.

Some previous works also used GP as a way to learn models of human behavior. Namely, the approach in (Fernlund et al., 2006) combines context-based reasoning and GP to learn models of strategic behavior through observation of humans performing a driving task in a simulator. The approach is more concerned with learning behavior models of individuals in one task and in a particular instance and therefore cannot generalize within or across individuals. In contrast, our approach tries to capture common, invariant properties leading to behavior in some task.

The procedure used in (Addis et al., 2016) is very similar to our own in that it uses GP to automatically generate and test cognitive theories. In particular, GP is used to evolve the sequence of cognitive functions leading to action selection whose accuracy, variance and duration best match the performance of individuals in a delayed match-to-sample task. Despite the similarity in the approach, the work tries to discover the architectural details underlying the behaviors, which requires a designer to define the architecture components to be optimized via GP. As a consequence, the architectural assumptions of the model will constrain the types of solutions that GP can generate. Conversely, ACBA abstracts from the structure of the underlying mental processes and does not restrict information processing during GP—we allow several possible solutions to be generated and then analyze their characteristics within some group of subjects *a posteriori*. This allows not only interpreting the underlying strategies but also discovering behavior trends that possibly contradict existing theoretical assumptions.

We also note that the cognitive tasks used in these works are more simple compared to how we use ACBA to discover and interpret strategies in a complex sequential negotiation task and then relate those strategies to trait characteristics.

Personality and Negotiation

Regarding the effects of SVO in negotiation, de Dreu & van Lange (1995) analyzed how different profiles can influence individuals' behaviors. In particular, *prosocials* adopted cooperative strategies initially but became noncooperative as soon as partners repeatedly failed to reciprocate—they made lower demands and higher concessions when compared to *pro-self*[3] oriented subjects, especially after the second round. In contrast, pro-self subjects adopted noncooperative strategies even when partner consistently cooperated. As for the Mach variable, Huber & Neale (1986) showed that *high-Mach* individuals initially aim for higher goals and achieved greater final payoffs when compared to *low-Machs*. In another study, Gunnthorsdottir et al. (2002) showed that low-Machs reciprocate trust in one-shot negotiation while high-Machs strongly defected when it was advantageous for them.

Methodology

As a validation setting, we used data (Xu et al., 2017) collected of people negotiating in a turn-based, multi-level, multi-issue bargaining setting (Thompson, 1990) against an agent using a fixed strategy. The task involved turn-based, sequential offers corresponding to full partitions over 3 records, 2 lamps and 1 painting. Notably, the subjects were unaware of playing against an artificial agent, thus making the collected data suitable for the purposes of our study. Besides the behavioral data, we collected information on the subjects'

[2] Such features may include task-related information, *e.g.*, the state of a game or problem-solving task, as well as subject-related information, *e.g.*, previous actions, task knowledge, goals, etc.

[3] Individualistic and competitive subjects are often merged into one "pro-self" or "egoistic" group (de Dreu & van Lange, 1995).

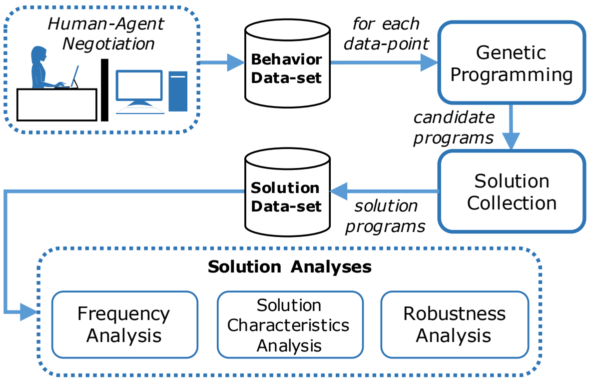

Figure 2: The ACBA methodology for predicting and understanding human behavior.

personality using standard instruments to assess their SVO (Murphy et al., 2011) and Mach (Christie & Geis, 2013).

The challenge of using ACBA in this scenario is to explain the behavior of the human subject for each collected negotiation instance. We assume that: 1) subjects make offers based on an underlying, *i.e.*, non-observable, *target payoff* responsible for the generation of the observed outcome, *i.e.*, the offer made, at each round; 2) such target payoff can be recovered through a mathematical combination of symbols encoding the observable state features. The goal of GP is then to generate programs that, given a negotiation instance, *predict the whole sequence of offers made by the human subject*. By using these assumptions we ensure that the inputs and outputs for the individual and the GP procedure are comparable and coherent, as illustrated in Fig. 1. We reinforce here the idea that only when the output of programs is consistent with the observed behavior, we can conjecture about the underlying psychological structures operating in the subjects' minds.

Behavior Data-set

Fig. 2 depicts the GP-based methodology used in this paper to analyze the behavior of human negotiators. We first create a *behavior data-set* from the data collected in (Xu et al., 2017).

Definition 1 (Data-point)**.** *A data-point corresponds to all information pertaining some negotiation instance, including all information about the offers and counter-offers made by the human and agent negotiators therein.*

Formally, let $d = \langle \mathbf{O}^h, \mathbf{O}^a \rangle$ denote a data-point, where $\mathbf{O}^h = o_1^h, \ldots, o_K^h$ is the sequence of offers made by the *human* negotiator and $\mathbf{O}^a = o_1^a, \ldots, o_K^a$ the sequence of counter-offers made by the *agent*. Each offer $o_k \in \mathcal{O}, k = 1, \ldots, K$ represents a particular partition over the set of items proposed by the respective negotiator at round k, with K representing the *length* of the negotiation in number of rounds. Each offer is in the form $[\#records, \#lamps, \#paintings]$ and identifies the items that the proposing negotiator wishes to get for itself. $\mathcal{O}$ is the set of all possible partitions between the items and hence we have $|\mathcal{O}| = 24$. The agent used in (Xu et al., 2017) used a fixed-strategy based on the one in (de Dreu & van Lange, 1995). Its sequence of offers was: $\mathbf{O}^a = [3,2,1]$, $[2,2,1]$, $[2,2,0]$, $[2,1,1]$, $[2,1,0]$, $[1,2,1]$, repeating the last offer thereafter. To facilitate the analysis in this paper, we attribute the following payoffs to each item:[4] one record is worth 11, one lamp 5 and the painting 2. Therefore, each offer $o \in \mathcal{O}$ has associated a unique *payoff*, denoted by $\mathsf{P}(o) \in \mathbb{R}$, obtained by summing the values of each item therein. As such, in our task we have that $\max_{o \in \mathcal{O}} \mathsf{P}(o) = 45$, and thus a *fair* value is considered to be around 23.

Genetic Programming

The GP component depicted in Fig. 2 searches in a space of mathematical expressions for programs that best fit the human negotiator's behavior—referred to as *symbolic regression*.

Definition 2 (Program)**.** *A program corresponds to a specific combination of symbols and mathematical operators. The input is some negotiation data-point and the output is the value resulting from executing the mathematical expression according to the information stored in that data-point.*

Programs can be represented as syntax trees, where nodes correspond to either mathematical *functions* or numerical *terminals*. For the functions we considered the set $\mathcal{F} = \{$`(x+y)`, `(x-y)`, `(x/y)`, `(x*y)`, `max(x,y)`, `min(x,y)`, `(x^y)`, `(x?y:z:w)`$\}$, where `(x?y:z:w)` represents a conditional operator that selects `y` if `x` $= 0$, `z` if `x` > 0, or `w` otherwise. For the terminals we used $\mathcal{T} = \mathcal{C} \cup \mathcal{V}$, where $\mathcal{C} = \{0, 1, 2, 3, 5, 6, 7, 11, 23, 34, 45\}$ are the *constants*—we included the payoffs and available quantity of each item as well as quarter values of the maximum payoff of 45. As for the *variables* we used $\mathcal{V} = \{$`Round`, `OffPayoff`, `InitOffPayoff`, `PropPayoff`, `InitPropPayoff`, `OffRecords`, `OffLamps`, `OffPaintings`, `PropRecords`, `PropLamps`, `PropPaintings`$\}$ encoding all relevant information available for the human negotiator at each round. Namely, given a data-point d, at each round $k = 2, \ldots, K^d$ we have that: `Round` returns k; `PropPayoff` returns $\mathsf{P}(o_{k-1}^h)$; `InitPropPayoff` returns $\mathsf{P}(o_1^h)$; `OffPayoff` returns $45 - \mathsf{P}(o_{k-1}^a)$; `InitOffPayoff` returns $45 - \mathsf{P}(o_1^a)$; `PropRecords`, `PropLamps` and `PropPaintings` return the quantity of the corresponding item in o_{k-1}^h; `OffRecords`, `OffLamps` and `OffPaintings` return the quantity of the respective item offered by the agent, *i.e.*, in o_{k-1}^a.

Fitness Function. Recall that our objective is to use GP to derive programs predicting the sequence $\mathbf{O}^h = o_2^h, \ldots, o_K^h$.[5] Because each offer has associated a unique payoff, the offer sequence has associated a unique payoff sequence $\mathsf{P}(o_2^h), \ldots, \mathsf{P}(o_K^h)$. As such, we use the output value of a program as an *estimate* of the payoff of offers at each round. Formally, let $\hat{\mathsf{P}}_k^d(\mathrm{p})$ denote the value resulting from executing some program p according to the information stored in data-point d at round k. To calculate a program's fitness, we first derive an estimated offer at each round, denoted by $\hat{o}_k$, whose payoff is the closest (greater than or equal) to the program's output. We then define the fitness of a program regarding a data-point

[4] These payoffs are based on the descriptions of the items' relative values provided to the subjects prior to the task (Xu et al., 2017).

[5] Single-shot negotiations where $K = 1$ were not considered.

d, denoted by $\mathsf{F}^d(\mathtt{p})$, as the negative root-mean-square error (RMSE) between the payoffs as *predicted* by the program and the ones *observed* from the negotiator's actual offers, *i.e.*:

$$\mathsf{F}^d(\mathtt{p}) = -\sqrt{\sum_{k=2}^{K} \left(\hat{\mathsf{P}}_k^d(\mathtt{p}) - \mathsf{P}(o_k^h)\right)^2 / K} \quad (1)$$

Evolutionary Procedure. Given a *behavior data-set*, the goal of the GP component in Fig. 2 is to iteratively generate *candidate programs* explaining the behavior of the human negotiator for all data-points as measured by Eq. 1.

Definition 3 (Candidate program)**.** *Corresponds to a program generated* during *the GP evolutionary procedure.*

For each data-point, GP starts by creating 100 independent populations of 500 programs by combining terminals and operators. Then, for each generation (maximum of 5000), GP uses tournament selection and standard crossover and mutation operators to generate a new generation of candidate programs. The output of GP is a set of candidate programs for each data-point analyzed.

Solution Collection

The Solution Collection component in Fig. 2 is the first step towards understanding the behavior of the human negotiators.

Definition 4 (Solution program)**.** *A solution program, or simply solution, represents an* hypothesis *for the underlying behavior structures of the negotiator in a given data-point.*

Solutions are selected from the set of candidate programs for a data-point according to the accuracy with which they predict the sequence of actions of the first negotiator. Specifically, we follow Occam's razor principle by selecting from the existing hypothesis—candidates attaining maximal fitness—the ones with the fewest assumptions, *i.e.*, that use less functions and terminals. Based on the insights from a previous study using synthetic data,[6] we excluded data-points whose solutions' length is higher than 25, and those that originate a significantly-higher number of solutions when compared to all other data-points. In addition, we excluded data-points where $K^d < 3$, *i.e.*, negotiations that were too short and thus not rich enough (Xu et al., 2017).

Augmenting the Solution Set. Because we are analyzing human data, the resulting programs might be complex, *e.g.*, long, deep and deal with several variables, meaning that it will be hard for GP to generate programs attaining a fitness of 0 according to Eq. 1. Therefore, in this paper we generate within Solution Collection a set of *plausibly-equivalent solutions* given a solution set. Formally, let us denote by $\mathtt{c} \in \mathbb{R}$ the value of some constant and by $\mathtt{v}_k^d \in \mathbb{R}$ the value of variable $\mathtt{v} \in \mathcal{V}$ at round k according to the data in d. We then augment $\mathbf{S}^d$ by creating copies of all programs $\mathtt{p} \in \mathbf{S}^d$ and replacing the constants $\mathtt{c}$ therein for variables $\mathtt{v} \in \mathcal{V}$ satisfying $\mathtt{v}_k^d = \mathtt{c}, k = 1, \ldots, K^d$, *i.e.*, by variables whose value is constant during the negotiation.

The output of Solution Collection is then a solution-set for each non-excluded data-point.

[6]We found that random strategies lead to very long solutions while simple strategies to too many when compared to all others.

Table 1: Mean values of features regarding our procedure. Bold values indicate significant differences ($p < 10^{-2}$).

Group	Size	Count	Vars.	Length
Prosocial	196	34.9 ± 41.9	$\mathbf{4.4 \pm 1.6}$	$\mathbf{11.4 \pm 4.7}$
Pro-self	104	43.7 ± 51.2	$\mathbf{5.3 \pm 2.2}$	$\mathbf{14.2 \pm 6.0}$
Low-Mach	189	35.1 ± 42.0	$\mathbf{4.7 \pm 1.7}$	12.4 ± 5.0
High-Mach	111	42.7 ± 50.6	$\mathbf{4.8 \pm 2.1}$	12.6 ± 6.0
Overall	300	38.0 ± 45.5	4.7 ± 1.9	12.5 ± 5.4

Solution Analyses

As mentioned earlier, a key goal in this paper is to understand the behavior of negotiators according to their trait characteristics, *i.e.*, across different negotiation instances, in order to uncover the invariant structures responsible for the behavior within a specific group. To that end, the set of components at the end of the pipeline in Fig. 2 analyze the solution programs across different partitions of the data-set. Namely: Solution Characteristics Analysis gathers statistical information about the set of solutions for one or more data-points, allowing us to analyze general characteristics of the underlying structures and how the behavior complexity varies according to different trait characteristics; Frequency Analysis measures the frequency of solutions and their sub-programs[7] to identify common structures emerging from GP across data-points; Solution Robustness Analysis computes the mean fitness of all solutions and their sub-programs when used to evaluate different data-points according to Eq. 1, thus assessing the *robustness* of each (sub-)program in predicting the respective behaviors.

Analysis of Results

We applied the ACBA methodology over the human negotiation data-set of (Xu et al., 2017). Out of the 405 original data-points, our filtering mechanisms excluded 74 due to the negotiation being too short, 5 for originating very long solutions, and 26 for having a relatively high number of solutions.

Meta-Analysis

We start by examining some high-level characteristics of the negotiations and solutions. Table 1 presents several statistics regarding the solutions and the negotiators' behavior when grouping data-points according to SVO and Mach. "Size" is the number of data-points in each partition. "Count" is the mean number of solutions per data-point. "Vars." is the mean number of unique variables required by solution programs to predict the negotiator's behavior—we use it to assess the amount of memory that would be required by a negotiator if he/she was following the strategy underlying the solution program. "Length" indicates the number of nodes in a program's tree—we use it to assess programs' complexity and richness.

A first observation is that the data-points of high-Machs and pro-selfs originate more solutions. Although the differences are not significant, this is a first indication of more complex underlying strategies used by individuals in these

[7]Given the representation of programs as syntactic trees, a subtree is also a program, *i.e.*, a *sub-program*.

groups. This analysis is also supported by the length of and number of variables in the solution programs. In particular, pro-selfs' solutions are significantly more complex than those of prosocials, requiring more computations and information to predict the negotiators' behavior. This is consistent with the findings in (de Dreu & van Lange, 1995), as pro-self individuals adopt more complex negotiation strategies, *e.g.*, reasoning about issue priorities and the opponent's offers. As for the Mach dimension, results show a similar trend, with high-Machs' solutions involving more variables and slightly more computations (higher length).

Solution Frequency and Robustness Analysis

We now try to make sense of the strategies used by the human negotiators by looking at their solutions' expressions. As indicated in Table 1, the total number of solutions per data-point is too high for us to analyze every single one. As such, we use the outputs of Frequency Analysis to assess (sub-)programs appearing frequently in the expressions of solution programs and the results of Solution Robustness Analysis to determine what are the (sub-)programs that best predict the negotiators' behavior, according to the personality traits. The results are listed in Table 2, with the first line of each group indicating the most *frequent* sub-program where $|\mathsf{s}| > 1$ and the second line showing the most *robust* sub-program and the mean fitness attained when evaluating all data-points of the group.

Overall. Regarding Frequency Analysis, we note that the most frequent terminals appearing in the solutions of data-points (not listed) were `OffLamps`, `OffPaintings` and `OffRecords`, in this order. First, this means that the number of items offered by the agent was a major determinant of the human responses. In fact, two of the most important negotiation strategies, *yielding* and *logrolling*, involve precisely strategizing concessions based on the demands of others and issue priorities (Pruitt, 1983). Moreover, given the relative value of the items in our scenario, the bargaining seems to be more focused on the number of lamps, which incidentally is also the item whose number changes the most throughout the agent's sequence of offers. As for the Solution Robustness Analysis, the results in Table 2 show that we could not find, given all solutions in $\mathbf{S^D}$, sub-programs predicting the behavior of *all* negotiators within any group, thus resulting in negative mean fitness values. Notwithstanding, our interest here is in analyzing the *most robust* solution programs.

Trait Differences. As we can see in Table 2, the solutions for pro-selfs and high-Machs involve calculating the ratio between the payoff offered by the agent in its first offer and the previous one. Given the anchoring effect of first offers in negotiation (Galinsky & Mussweiler, 2001), this ratio can help negotiators assess how the opponent is deviating from its initial offer to determine an appropriate form of retaliation. In fact, we see two slightly different uses for the ratio when looking at the solutions in which this sub-program is used (not listed). For pro-selfs, it is used as a scaling factor for the proposal, *e.g.*, in `max(OffPayoff, (InitPropPayoff*(InitOffPayoff/OffPayoff)))`, thus indicating a strong retaliation when the offered payoff decreases during the negotiation—this result is in line with the characteristic of pro-selfs in maximizing the relative advantage over others' outcomes (Murphy et al., 2011). For high-Machs, it is used as a reduction factor of their proposals, *e.g.*, `(23+(OffPaintings*(max(OffLamps,1)-(InitOffPayoff/OffPayoff))))`, meaning that the higher the offer by the opponent, the less the reduction and thus the higher the proposal will be—which is consistent with the manipulative and exploitative nature of high-Machs (Gunnthorsdottir et al., 2002). We observe a similar trend for Solution Robustness Analysis. Namely, the (sub-)program that best predicts the behavior of pro-selfs corresponds to an almost constant strategy proposing offers whose payoff is in $[23, 24]$, *i.e.*, above-fair proposals that do not take into account the opponent's behavior—this denotes the non-conceding nature of pro-selfs' negotiation behavior (de Dreu & van Lange, 1995). As for high-Machs, we observe a payoff-maximizing strategy depending on the number of initially-proposed records—the highest-valued item—, which is consistent with the tendencies of people with this trait (Gunnthorsdottir et al., 2002).

In contrast with these findings, the solutions of prosocials and low-Machs are more integrative and cooperative, *i.e.*, aiming for a fair payoff distribution. In particular, prosocials' solutions frequently compute the difference between an unfair payoff (34) and the opponent's offered payoff. When looking at the solutions in which this sub-program is used (not listed), *e.g.*, `max(23,(34-OffPayoff))`, we see that the resulting value is used by prosocials to establish a minimum acceptable payoff, *i.e.*, their *best alternative to a negotiated agreement* (BATNA) (Galinsky & Mussweiler, 2001). In addition, its most robust solution corresponds to a program targeting fairness—the mean payoff of prosocials' first offers is ≈ 24, meaning that this program outputs a value in $[18, 22]$, thus indicating that, on average, the maximum target payoff is right below fairness. These programs therefore correspond to highly-cooperative strategies, as predicted by this personality trait (de Dreu & van Lange, 1995; Murphy et al., 2011).

As for the low-Mach group, the frequency results show that the number of offered paintings and lamps plays an important role, especially in determining how much is added to a

Table 2: Results of Frequency Analysis and Solution Robustness Analysis overall and for each trait characteristic.

Strategy	Most frequent / robust (sub-)program	
Prosocial	`(34-OffPayoff)`	
	`(18+(PropPayoff/6))`	-4.4 ± 2.8
Pro-self	`(InitOffPayoff/OffPayoff)`	
	`(24-PropPaintings)`	-5.4 ± 3.4
Low-Mach	`min(OffPaintings,OffLamps)`	
	`max(23,(PropPayoff-OffPayoff))`	-4.6 ± 3.0
High-Mach	`(InitOffPayoff/OffPayoff)`	
	`max(23,(PropRecords*9))`	-4.8 ± 3.2
Overall	`(InitOffPayoff/OffPayoff)`	
	`max(23,(PropRecords*9))`	-4.7 ± 3.0

fair payoff as *e.g.*, in `(min(OffPaintings,OffLamps)+23)`. Furthermore, by looking at the expression of the most robust (sub-)program, we see that only when the difference between what the negotiator proposed and his/her opponent offered (`PropPayoff-OffPayoff`) is very high, the right side of `max` is selected, otherwise, fairness will be the target payoff. Given the decreasing nature of the agent's proposals, this sub-program has a higher effect in the beginning of the negotiation. These two facts are highly-related with low-Machs' tendency to adhere to moral standards such as reciprocity (Christie & Geis, 2013; Gunnthorsdottir et al., 2002).

Conclusions and Future Work

We showed how the ACBA methodology can be used to study human behavior in a task involving negotiating over a set of items. To validate the methodology and support our analysis, we resorted to two personality traits having a major role in bargaining behavior: SVO and Mach. Overall, the results of applying ACBA to a previously-collected negotiation dataset showed that: 1) we were able to uncover distinct behavioral structures in the expressions of solution programs for the data-points of different trait groups; 2) such structures have into account task information and lead to outcomes that are consistent with the behavioral trends exhibited by individuals having the corresponding traits, as described in the literature. These analyses more generally reveal that ACBA can be used to expose hypotheses about the underlying motivations of complex sequential behavior in dynamic contexts.

The results of this study provide positive indications as for the potential of ACBA in understanding human cognitive behavior. In particular, an interesting application of this work in the context of negotiation is hypothesizing personality trends of opponents and predicting their behavior in the absence of personal information. We are currently developing mechanisms to further process the solutions emerging from GP. We are especially interested in studying the variety of solutions within some group of individuals. We are creating techniques to cluster programs based on their behavioral semantic similarity. Using a similar technique, we are also determining the similarity of data-points, which we will use to group subjects according to how similar the underlying structures of their behavior are—our goal is to discover novel ways of classifying individuals' behavior tendencies.

Acknowledgments

We would like to thank Yuyu Xu for the collection and initial processing of the negotiation data used in this study.

This work was supported in part by the Air Force Office of Scientific Research, under FA9550-14-1-0364. The content does not necessarily reflect the position or the policy of any Government, and no official endorsement should be inferred.

References

Addis, M., Sozou, P. D., Lane, P. C., & Gobet, F. (2016). Computational Scientific Discovery and Cognitive Science Theories. In *Computing and philosophy* (pp. 83–97). Springer. doi: 10.1007/978-3-319-23291-1_6

Christie, R., & Geis, F. L. (2013). *Studies in Machiavellianism*. New York, NY, USA: Academic Press, Inc.

Clark, R. E., Feldon, D. F., van Merriënboer, J. J. G., Yates, K. A., & Early, S. (2008). Cognitive Task Analysis. In *Handbook of research on educational communications and technology* (3rd ed., pp. 577–593). Mahwah, NJ, USA: Lawrence Erlbaum Associates.

de Dreu, C. K. W., & van Lange, P. A. M. (1995). The Impact of Social Value Orientations on Negotiator Cognition and Behavior. *Personality and Social Psychology Bulletin*, *21*(11), 1178–1188. doi: 10.1177/01461672952111006

Fernlund, H. K. G., Gonzalez, A. J., Georgiopoulos, M., & DeMara, R. F. (2006). Learning tactical human behavior through observation of human performance. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, *36*(1), 128–140. doi: 10.1109/tsmcb.2005.855568

Galinsky, A. D., & Mussweiler, T. (2001). First offers as anchors: The role of perspective-taking and negotiator focus. *Journal of Personality and Social Psychology*, *81*(4), 657–669. doi: 10.1037/0022-3514.81.4.657

Gunnthorsdottir, A., McCabe, K., & Smith, V. (2002). Using the Machiavellianism instrument to predict trustworthiness in a bargaining game. *Journal of Economic Psychology*, *23*(1), 49–66. doi: 10.1016/S0167-4870(01)00067-8

Huber, V. L., & Neale, M. A. (1986). Effects of cognitive heuristics and goals on negotiator performance and subsequent goal setting. *Org. Behavior and Human Dec. Proc.*, *38*(3), 342–365. doi: 10.1016/0749-5978(86)90005-1

Miller, G. A., Galanter, E., & Pribram, K. H. (1960). *Plans and the structure of behavior.* New York, NY, US: Holt, Rinehart and Winston, Inc. doi: 10.1037/10039-000

Murphy, R. O., Ackermann, K. A., & Handgraaf, M. (2011). Measuring Social Value Orientation. *Judgment and Decision Making*, *6*(8), 771–781. doi: 10.2139/ssrn.1804189

Newell, A., & Simon, H. A. (1961, dec). Computer simulation of human thinking. *Science*, *134*(3495), 2011–2017.

Pruitt, D. G. (1983). Strategic Choice in Negotiation. *American Behavioral Scientist*, *27*(2), 167–194.

Simon, H. A. (1978). Information-Processing Theory of Human Problem Solving. In *Handbook of learning and cognitive processes: Human information processing* (Vol. 5, pp. 271–295). Oxford, England: Lawrence Erlbaum.

Thompson, L. (1990). An examination of naive and experienced negotiators. *Journal of Personality and Social Psychology*, *59*(1), 82–90. doi: 10.1037//0022-3514.59.1.82

van Someren, M., Barnard, Y., & Sandberg, J. (1994). *The think aloud method: a practical approach to modelling cognitive processes*. London, UK: Academic Press.

Xu, Y., Sequeira, P., & Marsella, S. (2017). Towards Modeling Agent Negotiators by Analyzing Human Negotiation Behavior. In *Proc. 7th Int. Conf. on Affective Computing and Intelligent Interaction.* San Antonio, TX, USA.