Is Human-like and Well Playing Contradictory for Diplomacy Bots?

Markus Kemmerling, Niels Ackermann, Nicola Beume, Mike Preuss, Sebastian Uellenbeck, Wolfgang Walz

***Abstract*— This paper presents a *non-player character* (NPC, bot) for the strategy game *Diplomacy*. The bot is able to communicate with other players and thus shows a human-like behavior. We investigate how far the playing abilities can be improved without corrupting the human-like behavior. Is there a trade-off at all or do these skills complement one another? Different versions of the bot are tested against other bots and humans which requires means to automatically measure believability. We derive such a measure after a general approach and apply it for monitoring the believability criterion while improving the playing strength of our bot.**

I. Introduction

Creating a *non-player character* (NPC, bot) for the game *Diplomacy* [1] is a great challenge. The game not only requires tactical decisions for a group of entities but also negotiations with other players to form alliances or intimidate enemies. These negotiations distinguish Diplomacy from other simpler war games and thus are the crucial part of the game which deserves closer attention.

While previous research on Diplomacy bots [2] concentrated on the tactical playing abilities, this work additionally presents concepts for negotiation skills. These negotiations are the main interaction among players and an important fun factor of the game. For human players clever playing bots who the player cannot 'talk' to, do not fit well the idea of the game. While they are more fun to play against than weak playing bots without communication, they are still not *believable* [3] as they lack any close cooperation with human players.

The growing market for massively multiplayer online role-playing games (MMORPG) - such as World of Warcraft - indicates the desire for human players to interact with one another in video games. In order to reproduce the MMORPG success, the need to simulate human players in single player games becomes apparent. Unfortunately, these interactions cannot be modeled as a simple score-hunting task like in 3D shooters where NPCs are successful with rising number of defeated enemies. Interestingly, *Diplomacy* may serve as simplified (model) *realtime strategy* (RTS) game as it is not strictly turn-based (all moves happen concurrently).

Most games which rely on interaction contain simple rule-based NPCs whose actions can easily be predicted. But gameplay fun gradually decreases for the human player if interaction with the adversaries gets monotonous and predictable. Varied, yet believable behavior would motivate players to interact with NPCs in the long term and make the game funnier and more interesting.

All authors are with the Chair of Algorithm Engineering, Computational Intelligence Group, Dept. of Computer Science, Technische Universität Dortmund, Germany. E-mail: firstname.lastname@tu-dortmund.de

In order to increase game fun, we seek to improve the interaction abilities and the human-like behavior of Diplomacy bots since we assume that human players tend to prefer playing against bots that behave more similar to themselves.

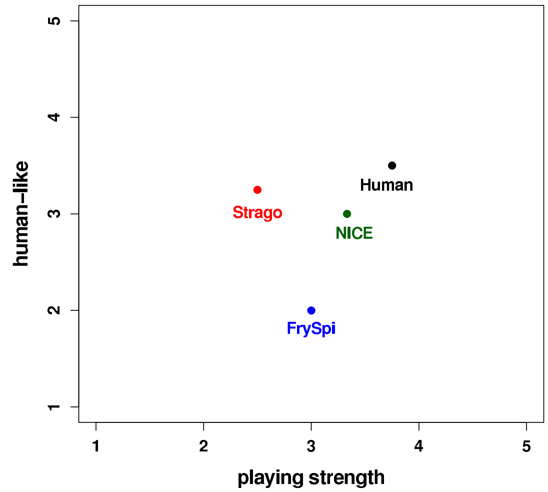

Fig. 1

Initial Turing test estimations of human-like play and strength of three newly implemented bots targetted towards human-like playing styles, higher values are better.

In a first attempt to realize more human-like behaving bots, the *Stragotiator*, the *NICE*-bot and the *FrySpi* have been implemented and compared within a recent student project at our department. Figure I displays the result of a Turing test of 4 games with 1 to 3 human players per game and a fair distribution of the three bots. The plotted points resemble the estimations of the active players concerning playing strength and human-like play, not knowing which power was played by whom. In a large-scale automated test, it was found that generally, the *FrySpi* played better than the *NICE*-bot, the *Stragotiator* being last. Our impression is, that human players are usually playing better as well as more human-like (as found in figure I) than all bots and that it is hard to implement a bot which realizes both attributes in a satisfactory way. A similar experience had been made before during the creation of the *NICE*-bot which is a variant of the *Diplominator* [4], enriched with more communication and negotiation features. Rather than better, the new version plays a bit weaker but much more human-like (the *Diplominator* does not communicate except for default messages like peace to all).

For creating a bot that plays strongly and is assessed a high believability, one could envision two different approaches:

978-1-4244-4815-9/09/$25.00 ©2009 IEEE

Either enriching a well playing bot with more communication features, or improving the playing strength of a bot that is already seen as very human-like. During the Turing test reported above, the *Stragotiator* was the only bot ever being held for a human player (3 times).[1] From this, the main motivation of this work emerged: To improve the *Stragotiator* to satisfactory strength while keeping its human-like playing style (believability).

A general problem in this context is how to measure believability automatically. Test games against humans need hours and are only convincing if coming in greater numbers. In order to generate a measure without human interaction, we observed players and bots and identified certain patterns which we reward or punish and subsume in formulas. This took inspiration from the approaches of Yannakakis and Hallam ([5], [6]) to intuitively define or select from data the constituents of a measure for game fun. We adapt our measure to a corpus of played and humanly evaluated games to match human relative assessment as close as possible. Of course, believability is finally a subjective measure if quantified, as e.g. game fun. However, the differences between diplomacy bots appear quite large if compared to the ones found in players' preferences so that we expect a reasonable measure to at least point into the right direction. This is needed to dramatically decrease turnaround cycles in bot development when targeting not only playing strength.

We conduct experiments to investigate the usefulness of the believability measure approach which we expect is quite general and then compare the different bot versions according to playing strength and believability. Eventually, we are looking for an answer to the question if the *Stragotiator* can be improved in strength without losing too much believability.

The methods applied to enliven the bots stem from *Computational Intelligence* as it provides techniques to robustly deal with imperfect information. This way non-deterministic or probabilistic behavior can be modeled that is yet subjected to certain concepts, which shall result in human-like and non-predictable behavior of a believable character. We use evolutionary algorithms for several tasks of parameter tuning. Fuzzy sets are invoked within the modeling of the bot's emotions.

The next section introduces the game Diplomacy, followed by a description of the first version of our bot (sec. III) and the improved version in section IV. Section V lays the foundations for our believability measure which is introduced in Section VI. Section VII reports on comparisons of the different bot versions among themselves and to other bots. Finally, Section VIII recapitulates our insights and depicts future research questions.

II. Diplomacy

Diplomacy is originally a board game developed by A.B. Calhamer [7] and became popular in the 1960s. Situated in the Europe of the first world war, 7 great powers fight for dominance. The board is divided into 56 land and 19 sea regions whereas 34 land regions contain supply centers. Each of the 7 players commands the entities (armies and fleets) of a particular power. The aim of the game is to conquer provinces, namely the ones with supply centers that support one entity each. The game ends when a single power or an alliance of countries attains supremacy with at least 18 supply centers. Figure 2 shows a mid-game scenario.

Diplomacy consists of rounds called seasons (spring, fall), whereas two seasons form a year counting from 1901. A season starts with a negotiation phase followed by a movement phase. Each year the number of entities is adapted to the number of supply centers. New entities can only be build in one of the 22 home supply centers which are a subset of the supply center. In each province one entity can remain at most. A province can be conquered only with a superiority of units against an enemy. The game does not incorporate any random effects.

The most important moves are to attack an enemy, to support an attack, to hold the position or to support a hold. Negotiations are the heart of the game. The players may arrange arbitrary agreements, which they are not forced to keep. Diplomatic discussions are not subjected to any rules and so tactics and psychology are important factors.

In the early years, conventions have been held to gather players and postal Diplomacy arouse. With the internet a huge online community [8] emerged playing a computer-based version. A protocol of messages (DAIDE syntax [8]) has been established to perform the negotiations. The lexis is denoted by the so called press-level from 0 (no messages) to 8000 (free text). Most bots available for the DAIDE environment do not communicate at all or implement press-level 10 at most, which contains peace and alliance messages. On press-level 20, additionally suggestions of moves (XDO), and demilitarized zones are possible. We developed a set of bots 'talking' press-level 20 of which we here concentrate on the most promising one for high believability values, as described above.

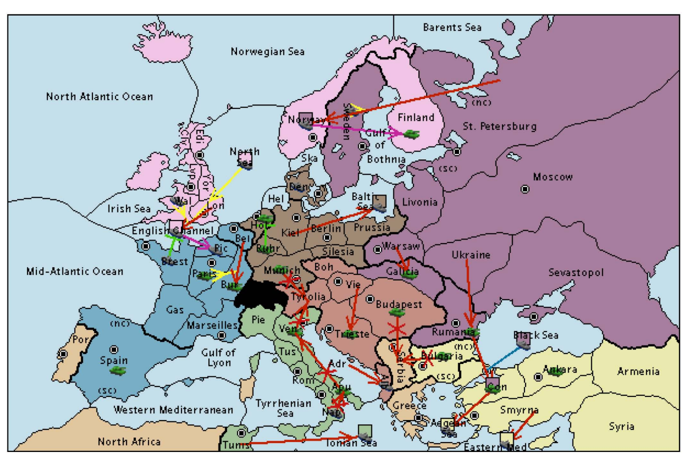

Fig. 2

A screenshot of the Diplomacy game. Arrows give planned moves, crossed arrays indicate failed moves due to conflicts.

[1]However, the test games have been short (5 years) so that humans could have been tricked by sophisticated default communication strategies. Nevertheless, from practical experience we estimate that 4–5 years is a lower bound for attaining a well-founded opinion on the enemy bots.

III. Stragotiator 1

While planing the *Stragotiator*, the main goal was human like behavior. We made the decision to model a particular personality after the classification given in [9]. From the available four character types, we chose to implement the Classicist, a player type that incessantly desires to have at least one ally, seeks for long-term alliances and plays to the success of his alliances. The first Stragotiator version, named *Stragotiator 1* in the following, possesses three major modules, an opponent modeling component, the negotiator, and the strategy core.

A. Opponent Modelling

The opponent modeling takes responsibility for the friend-foe matrix and the intelligence modeling. The friend-foe matrix is a 7x7 matrix with real-valued entries that indicate the relationship between two powers. The higher the value, the more kindly is the relationship. Its functionality is oriented towards the description in [10].

The intelligence model is a real-valued array that represents for each power how intelligent its acts, according to our criteria. The values are calculated with the *strategy core* of the *Stragotiator* so that it evaluates the opponents moves from this point of view. Thus, the *Stragotiator* deems an opponent the more intelligent, the more similar its moves are to the ones the *Stragotiator* would employ in his situation. Following from that, 'intelligent' may rather be understood as 'comparable' here [11].

B. Negotiator

The Negotiator is responsible for all interactions with the other players. It talks to opponents, responds to requests and decides about peace and ally pacts. The Negotiator is capable of press level 20, with some features missing, as *demilitarized zones* (DMZ). According to our decision to model the Classicist, the main goal was permanent interaction with other players and to complement the actions of allied players well.

A lot of effort has been devoted to sending XDOs (movement suggestions). Depending on the friend-foe matrix, the intelligence of an opponent and the trust in him, the *Stragotiator* decides whether or not to ask for or to accept peace or alliance pacts with this power, respectively. When the *Stragotiator* entered into such a contract, its strategy core will calculate moves for all units in the party, including that of his contracting party. A subset of the best found moves are then suggested to the partnering power.

If an opponent proposes moves (via XDO) to the *Stragotiator*, it first checks if the sender is trustable like described in [11]. If the *Stragotiator* strongly relies on him, the proposal is accepted with a probability of 0.75. If the Stragotiator does not trust the sender, the proposal is rejected with a probability of 0.75. If the decision cannot be taken according to this heuristic, the move is evaluated by the strategy core and, according to the answer, a response is generated.

The second responsibility of the Negotiator is trust modeling. For this task, we choose a linguistic variable with four fuzzy sets, namely mistrust, neutral, trust, and full trust. To which of these sets a power specific trust value belongs determines if the Stragotiator relies on that power. The trust value increases, if

- an ensured move (an accepted suggestion) was indeed executed
- a good move is proposed

and decreases, if

- an ensured move was not executed
- a bad move is proposed
- peace or alliance proposals are made that do not include the sending power

Defuzzification into a single discrete trust level is done according to maximum membership to one of the four classes.

C. Strategy core

The strategy core calculates and evaluates moves. It does not use knowledge about the power actually played. Instead it is parametrized to calculate and evaluate moves for a certain power. However, it is informed about the powers which are considered as enemies. This enables constructing move collections of mixed powers, seeking the advantage of all friendly powers at once, including the actually played power, as is suggested by the implemented Classicist character.

The strategy core prefers moves in the following order:

1) Attacking support
2) Support for a hold
3) Attack
4) Normal move
5) Hold

When calculating moves for a group of powers, the strategy core creates a move for each unit of these powers. For any unit, the move type is chosen according to the given order if any feasible move of that type is detected. Under all possible target provinces for that move type the one with the highest value is chosen. The value of a provinces is determined as given in table I.

TABLE I
Province values according to support center status and nearby units.

Province status	value
Own home supply center held by enemy / endangered	50
Supply center owned near enemy unit	20
Supply center owned by enemy	10
Supply center owned by a power in peace status	-18
Supply center owned by a power in ally status	-20
Normal province with unit of power in peace status	-18
Normal province with unit of power in ally status	-20
Normal province with unit of enemy power	10
All other cases	0

Evaluation by the strategy core is not in all cases done for deriving a move set. For the first round, a number of preferred moves is known in common opening books (see [12]). One of the three or five best moves is chosen uniformly at random. Further it shall be emphasized that when the negotiator has ensured a move to another power, it will be executed by all means.

When evaluating a move, the calculation depends on the value of the move's target province and the environmental conditions. The value of an attacking support move is increased by the number of defenders multiplied with ten and decreased by the constant value -100 if there is no defender. An attack gets the value of the target province multiplied with 1.2 if a neutral power is attacked or with 1.5 if an enemy is attacked.

IV. STRAGOTIATOR 2

The enhanced version 2 of the *Stragotiator* employs an *evolutionary algorithm* (EA) for improving move packages and comes with a more sophisticated province value evaluation. A move package consists of all units of given powers. We chose an EA as easy to apply and versatile optimization method as the nature of the objective function is largely unknown.

The EA is a $(\mu + \lambda)$-EA with μ=20 and λ=40. It uses a uniform recombination probability of 0.4 and runs for up to 500 generations. The EA is working on a given set of units with the task of detecting the move set that leads to the highest function value in addition. Mutation of a move set is done by means of picking each unit with probability $1/\#units$ and change its destination. The new destination decides the new order type: hold if it is the source province, support if a friendly unit holds or attacks the province, or else attack.

To quantify the value v of a move, the importance of the target province has to be taken into account. E.g., the home supply centers should be heavily defended if threatened, or free to build on, if not. Likewise, enemy centers and units are more important than neutral ones.

Supply centers are valued as given in table I. However, no information is available under this scheme for provinces p without a direct revenue. Some may be attractive because they open up possibilities of attacking valuable adjacent provinces only. Thus, we introduced an influence map that reflects the worth of important provinces in a weaker form onto their neighbors to indicate advantageous directions to turn to. This blurring by propagation of the values of provinces is done iteratively in 4 steps $k = 1, \ldots, 4$ according to equation 1. The factors $g(k)$ correspond to an exponential decrease of the influence on neighboring provinces in relation to their distance k to the current province.

$$v_k(p) = v_{k-1}(p) + \sum_{\forall \text{ neighbors } p_i \text{ of } p} g(k) \cdot v_{k-1}(p_i) \qquad (1)$$

$$\text{with } g_{\text{spring}}(k) = \{0.12, 0.022, 0.01, 0.004\},$$
$$g_{\text{fall}}(k) = \{0.04, 0.022, 0.01, 0.004\}.$$

The EA target function's goal is employed to determine superiority, thus outnumbering the enemy, or recognize a disadvantage, by an unfavorable friend-foe relation in proximity. For successfully moving into a province, possible defenders (immediate neighborhood of the target province) have to be taken into account. These may in turn be neutralized via cutting of their support by attacking them, or in equal terms, supporting the moving unit. If all support is cut off, the province can be conquered with one supported attack (2 units). If this support is not given, we are not able to conquer, thus attacking this province is not worthwhile. For holding a province, equal terms are sufficient, even if it is not clear which unit is moving towards us. Thus, supporting is the best bet to keep the status quo. Equations 2 to 4 denote how the value of a move is derived, where x is the previously computed worth of the province which is the move's target.

$$move(x) = \begin{cases} f(x) & \text{if \#supporting units} \geq 1 \\ f(x)/2 & \text{else} \end{cases} \qquad (2)$$

with

$$f(x) = x/(1 + \text{\#def.} - (\text{\#sup.} + \text{\#attacked def.})) \qquad (3)$$

for attack moves and

$$f(x) = x/(\text{\#att. units} - \text{\#sup.} + \text{\#attacked att.}) \qquad (4)$$

for hold and support moves.

V. MEASURES FOR HUMAN-LIKE BEHAVIOR

We seek to measure the grade of *believability* of a bot since this aspect seems to be an important factor of player satisfaction. In a kind of Turing test human players and observers evaluated bots and humans. The results do not show a consistent picture as different persons have different criteria of believability. Due to this fact and the high necessary effort we decided against evaluations of our bots by persons. Instead we analyzed the features reported as non-believable by the test persons and developed formulas to measure these factors on the basis of the game data. Certain behaviors of bots and players have been identified and classified as either non-human or human indications and so are either punished or rewarded. We subsume these values in one whereas punishments are subtracted and rewards added. This values are calculated in each round and aggregated to an evaluation of the game.

As non-human factors we consider invalid moves, repeated actions, and apparently unpromising moves. As an invalid move, we identified *self-seizing* were more than one entity of the same player tries to stay in or move to one province. The simple rule that at most one unit can remain in each province can be followed easily by humans and so this behavior is punished as non-human like, with a constant -10 multiplied by the number of self-seizings. If a bot could reach an unoccupied supply center but does not capture it, we punish this missing purposefulness heavily (each occurrence by $pen_{max} = -400$).

A move can only be repeated in subsequent rounds if it has not been executed since it was not successful. Repeating moves is punished quadratic in the number of subsequent rounds and entities. Retrying a move can be clever but repeating several for several rounds most probably not, that is why these numbers are squared. This pattern is called 'deadlock' in the following. Also toggling (pattern 'JumpingJack') between two positions does not seem to be quite purposeful and is punished quadratic in the number of rounds. The same penalty is calculated for repeated messages. Moving entities chain-like is also punished quadratic in the number of entities as these moves are very risky: if the first move fails all others entities are stuck as well. One unit can never conquer a hostile province as it cannot establish a superiority, so it is useless to move an isolated entity into an enemy region. This pattern 'ClusterDisconnect' is punished quadratic in the number of entities whereas a unit is considered as isolated in case of a distance of more than three provinces to the next own unit.

When rewarding human-like behavior, we do not consider aspects of playing ability like the number of supply centers. Even among human player there have to be losers, so these kind of failures cannot be attributed to non-human behavior. Instead, we reward successful communication and cooperation.

Each request for support is rewarded by the constant $-pen_{max}$. The success of the communication is measured as the fraction of accepted messages from all messages sent. This fraction is multiplied by $-pen_{max}$. The fraction of realized agreed XDOs is added to 1 and multiplied by $-pen_{max}$.

Messages should mainly be sent to neighboring parties since they mostly cooperate. We divide the map into close (radius of 3 provinces around own units) and far provinces. We calculate the difference between the number of XDO and DMZ messages send regarding close and far provinces. So, the behavior of communication is punished when the 'far' messages prevail or rewarded otherwise.

VI. An Adaptive Believability Calculator

In order to determine the believability of a diplomacy bot automatically, we start with the player behaviors identified as indicators for either human-like or non-human like play (§V). Generating a quantitative believability measure for a bot in a recorded game is simply done by computing the values for all of indicating behaviors as found in the game and summing them up, possibly in a weighted manner. However, an intuitive weighting of these behaviors to receive an aggregated measure is highly error-prone as human players have difficulties in deciding which single behavior is more important than any other. Therefore, we adapt the weights according to a number of played games (9) with one human player and different bots for which the human gave a believability estimation on a discrete scale from 1 to 5 ($5 =$ highest). A good weighting would result in a believability ranking for each of the games that closely matches the human one. This can be formulated as an optimization problem by counting the number of wrong relative rankings of each two bots which shall be minimized.

As a simple ad-hoc tool for this problem, we employ a standard $(\mu + \lambda)$-EA that operates on the weights as search space. To reflect the large absolute differences in values for the single behaviors, the weights are cubed before applying them, thereby enabling the EA to modify weights on different scales while keeping a singular mutation step size. The initial population is created from Gaussian distributed random numbers with mean 0 and standard deviation 1. Mutation adds a random number from $[-\sigma, \sigma]$ to every weight. Recombination is discrete: for every weight, the corresponding value is chosen from any of the two parents with probability $\frac{1}{2}$. The optimization process is stopped after a given number of generations (10000). However, note that EA parameters are only manually adapted to the problem so that performance gains may be possible. But we do not suggest that an EA is an ideal tool for solving these kind of problems efficiently. We just report that it detects reasonable solutions easily. In the following, we perform an experimental evaluation of our believability measure creation process.

Research Question. Does learning feature weights from data lead to a reliable believability measure?

Pre-Experimental Planning. The parameters of the EA have been tuned manually, with a (50+50)-ES with recombination probability 0.7 and step size of 4.0 reported as good setting. However, the algorithm performance seems not very sensible to parameter changes. In a first attempt to learn and apply the weight set, it was found that some weights were assigned very high values but their features did not occur at all in the set of games used for the optimization. When applying this measure, the resulting values are highly distorted by this effect. Therefore, we decided to fix the weights for non-represented features (with variance 0 in the data) to 0.

Task. The believability measure shall return better rankings (the absolute values are not considered) than random orderings when optimized and application of the optimization itself shall decrease the error rate significantly. As error we measure the difference to the ranking done by human evaluators. The difference of two orderings is calculated as the number of inversions. As we exclude the humanly played power from counting, 6 nations remain, so that the maximum number of mistakes is $\sum_{i=1}^{5} i = 15$ and the expected number of errors in a random ordering would be 7.5. Furthermore, this shall hold if validated on a set of games not used for learning.

Setup. The test games are set up with one human, two first version *Stragotiators*, two second version *Stragotiators* and two *Alberts 5.1*[2], with randomly distributed powers. According to stability issues with *Albert* in our later experiments (§VII) we also make two games where one of each *Stragotiator* version is swapped with the *NICE* bot. Each game is played to year 1905. As the length of a movement phase is fixed to 4 minutes, this takes nearly one hour per game. Note that it is necessary to add small and slightly randomized waiting times into the communication system of the bots in order to camouflage their bot nature.

[2]`http://ca.geocities.com/stretchy@rogers.com/Albert.htm`

9 games are played by 4 testers, 6 of which are selected for learning and the remaining 3 are the validation set. To guarantee that the two sets are not too similar, we deliberately put the games of one player to the first set and the ones of another player to the validation set only. The other two players have games in both sets. After finishing a game, the bots are rated with a value from 1 to 5 (5 = highly believable) and the bots are ranked according to their value. We then apply the believability calculator with default weights (= 1), start the optimization with the parameters given above and apply the calculator again. For each of the attained two sets of believability values, we rank the bots and compare these rankings to the ones determined by the test players.

Results/Visualization. Figure 3 shows the logarithmically scaled weights of one of several alternative best weight sets, which leads to 29 ordering mistakes (45 would be the random expectation). The module weights are also plotted in Figures 4 and 5 showing the number of ranking mistakes for each game of the learning and the validation set, respectively.

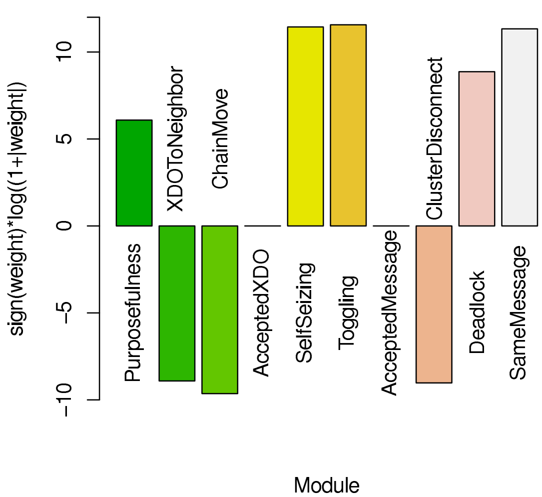

Fig. 3

THE BEST ATTAINED WEIGHT SET, PLOTTED ON LOGARITHMIC SCALE. TWO FEATURES ARE DELIBERATELY FIXED TO 0. NEGATIVE WEIGHTS STAND FOR BELIEVABLE, POSITIVE FOR NON-BELIEVABLE FEATURES.

Observations. From playing experience and the attained data, it becomes obvious that human rating of the non-neighboring powers is very unreliable (e.g. there is usually no communication with Turkey when playing England, except peace messages). This is currently not considered and may lead to many ranking errors when comparing human rating to the computed believability value. Concerning the attained best weights, it is surprising that two features have negative values when positive ones were expected: chain moves and disconnected clusters. If at all important, these are held for human playing styles. The most important positively valued (non-believable) features are self-seizing, jumping jack, and repeatedly sending the same message. XDOs to neighbors are valued as believable. However, the alternative best weight

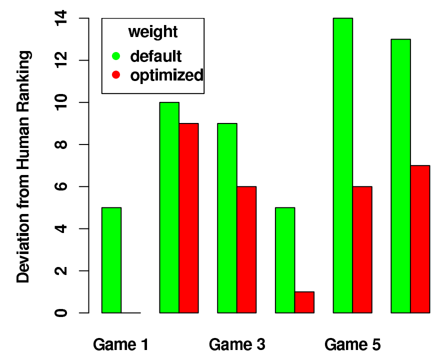

Fig. 4

THE BEST ATTAINED WEIGHT SET (OPTIMIZED) VERSUS THE DEFAULT WEIGHT SET, APPLIED TO THE LEARNING SET OF 6 GAMES.

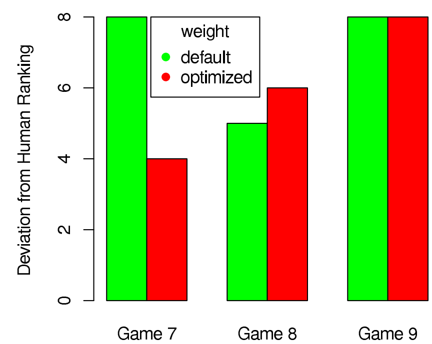

Fig. 5

THE BEST ATTAINED WEIGHT SET (OPTIMIZED) VERSUS THE DEFAULT WEIGHT SET, APPLIED TO THE VALIDATION SET OF 3 GAMES.

instances showed slightly different distributions, so that these should not be overrated.

Discussion. It is easy to see that optimization leads to better weight sets, according to the remaining ranking error numbers. The second goal of attaining a ranking that clearly improves on a random one is reached on the learning set (29 vs. 45 errors) as well as on the validation set (18 vs. 22.5). However, the last game of the validation set stems from a player who did not contribute to the learning set, and it shows the largest number of errors even with the optimized weight set.

This leads to the conclusion that player preferences are very different, and that it is not reasonable to expect the weight set to reflect a preference it has not seen during learning. The same holds true for non-used features (with variance 0 between values of all games in the learning set), they have to be excluded from the believability calculation as their weights cannot be determined. Finally, the number of games has been relatively small in this study. With more games, preferably of one player, the quality of our measure will most likely rise dramatically (although then only rating believability according to a single player's preference). It remains unclear how meaningful the obtained weight values are, so that it is not possible to draw straight conclusions about the single features. This could be investigated by studying many different good weight sets and looking for their similarities. However, this is not done here.

VII. Playing Strength and Believability Evaluation of the Stragotiator

After making available an automated measure for believable which seems to be at least much better than pure guesswork, we return to the original goal of the paper and investigate if the *Stratogiator* in its second version plays stronger while still keeping believability at approximately the same level.

We therefore conduct an experiments under two different settings, the first without press (communication), and the second including press. Without press, our believability measure is doomed to fail, but this may allow for better assessment of the playing strength as it may deteriorate from using press.

Research Question. Does the *Stragotiator 2* play better than the first version while keeping its believability level high?

Pre-Experimental Planning. The two intermediate variants of the *Stragotiator*, Markus and Sebastian, are chosen as representatives of a larger class of possible variants due to their relative playing strengths (and as they enable to evaluate the importance of the single improvements as the insertion of EA-based optimization and blurring).

For the second setting (including press), we originally planned to play against *Albert 5.1* bots. However, it revealed some stability issues that did not occur in the last experiment (probably due to a much higher number of games). Thus, the *NICE* bot is used instead. Note that this bot is a communication-enhanced version of the *Diplominator* which shows the same stability problems as the *Albert 5.1*.

Task. For the no-press setting, the *Stragotiator 2* shall consistently outperform the old version in playing strength. In the press-included setting, this shall also be the case, and ideally, the believability rating (according to the measure generated in the last section) shall not be significantly different for both bots. In addition to visual comparison, we perform Wilcoxon rank-sum tests (the distributions are nearly always non normal) over all played games between the two main *Stragotiator* variants 'Niels' and 'Wolfgang'.

Setup. For both settings, we play up to 20 years (40 rounds) as this game phase is most interesting for humans (in the environment of the previous experiment, this would mean a game duration of more than 3 hours). Around 2800 games are done for each setting, and for each game, two bot types are selected at random and play 3 against 4 with randomly chosen powers. The number of supply centers is recorded after the fall movement and adjustment phases for every year. The first setting (no press) is only played by different Stragotiator variants, namely 'Wolfgang' (*Stratogiator 2*), 'Sebastian' ('Wolfgang' without blurring), 'Markus' ('Wolfgang' without EA optimization), and 'Niels' (*Stratogiator 1*). The second setting (press level 20) is played in a similar mode (3 against 4, randomly selected), but with 3 bot types only: 'Niels' (*Stragotiator 1*), 'Wolfgang' (*Stragotiator 2*), and 'NICE'. From the server logs, the believability value for each bot in this setting is computed.

Results/Visualization/Observations. For the no-press setting, we obtain three different cases regarding to dominance of one bot type, namely 'Wolfgang' (*Stragotiator 2*) leading, for Italy, England, France (very slightly) and Turkey. 'Sebastian' is playing strongest with Russia, and either 'Markus' or 'Niels' win with Austria and Germany. Figure 6 depicts the three typical cases. Over all games, Wilcoxon rank-sum tests indicate a significant improvement of the *Stragotiator 2* over the first version for the years 1910, 1915 and 1920 while they are very similar for year 1905 ($p = 0.37$).

For the second setting, the same three powers are shown in figure 7. Here, 'Wolfgang' always beats 'Niels' except for Austria and Germany, and in case of Russia it also beats the 'NICE' bot. Figure 8 displays the calculated believability measures for the three bots over all games.

Concerning the playing strength over all games of all powers, Wilcoxon rank-sum tests indicate that 'Wolfgang' is significantly better than 'Niels', at every of the 4 measure points 1905, 1910, 1915, and 1920. Tests between the observed believability values for 'Niels' and 'Wolfgang' show no significant difference, whether 'Wolfgang' is significantly better than the 'NICE' bot in this respect. All reported p-values are below 10^{-10}.

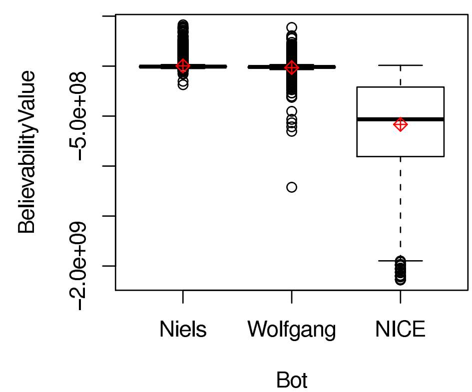

Fig. 8

Absolute computed believability values for the three bots in the press-included setting, higher means more human-like.

Discussion. We can safely state that the targetted improvement of the *Stragotiator* from version 1 to version 2 (including EA optimization and blurring) has been achieved. However, for the 'middle' powers Austria and Germany, the improvement was in fact a decrease. It seems that for these countries, a successful strategy has to be considerably different. Concerning the believability measures, we obtain that the old and the new *Stragotiator* are very similar, and both are better than the *NICE* bot, which nicely fits into the results of the very first comparisons made by human players as reported in the introduction. Thus, the overall goal of keeping believability and improving playing strength of the *Stragotiator* is attained. However, the bot still plays weaker than some other publicly available bots like *Albert 5.1* or the *Diplominator*.

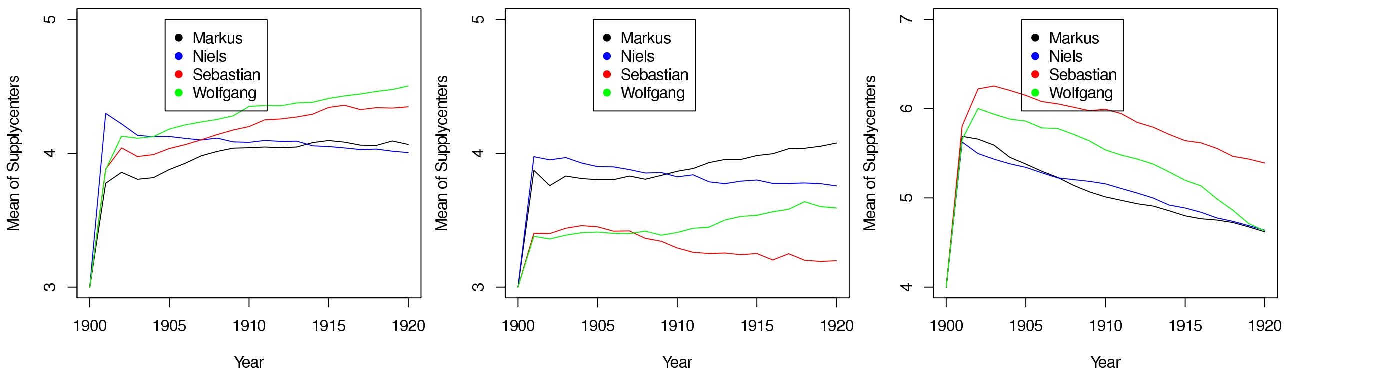

Fig. 6

COMPARISON OF DIFFERENT *Stragotiator* VARIANTS WITHOUT PRESS, 'WOLFGANG' STANDS FOR *Stratogiator 2*, 'SEBASTIAN' LACKS BLURRING, 'MARKUS' HAS BLURRING BUT NO EA OPTIMIZATION, 'NIELS' IS THE *Stratogiator 1*. FROM LEFT TO RIGHT: ITALY, AUSTRIA, RUSSIA.

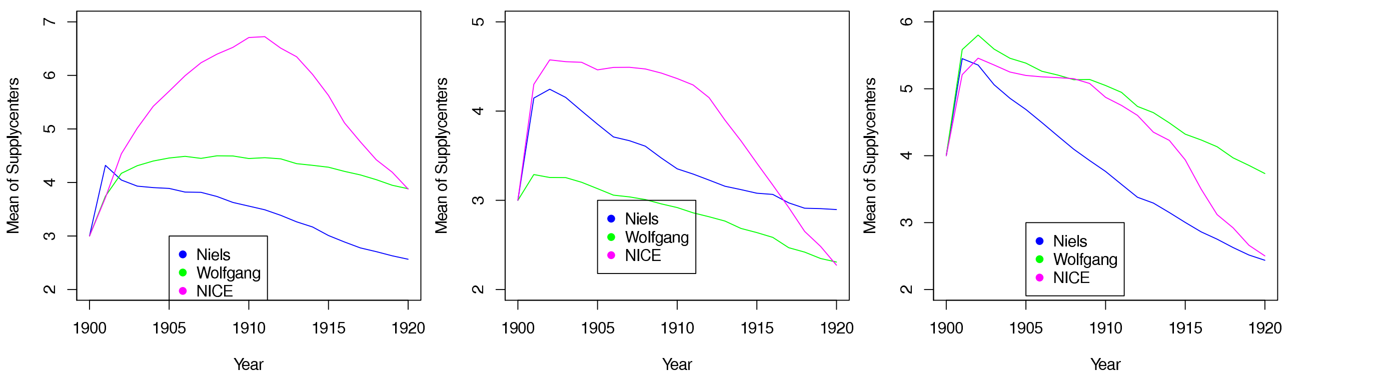

Fig. 7

COMPARISON OF THE OLD AND THE NEW *Stragotiator* WITH THE *NICE* BOT INCLUDING PRESS. FROM LEFT TO RIGHT: ITALY, AUSTRIA, RUSSIA.

VIII. CONCLUSIONS

We discussed an approach to increase playing strength while not decreasing believability of an already human-like playing *Diplomacy* bot. This has been realized by introducing an EA for optimizing move sets and improving the move evaluation and the map weighting. To determine the believability of our bots automatically, we introduced a measure based on features intuitively classified as non-human like. The weights of this measure were optimized by an EA according to believability rankings establish by human players in some test games. The main results are that the derived believability measure works well and that the improved bot plays better than its original version while keeping its high believability level. Thus the title question of the paper can be answered as 'yes, but not always'. It is possible to improve one criterion without losing on the other, but this requires some effort. We also have to state that large-scale automated testing against third-party bots is often difficult as they are not always stable.

At least two issues remain for future work: The believability measure approach shall be tested with larger sets of games, and could also be used for an entirely different game. Concerning Diplomacy, one has to think of different bot-variants for different powers, as e.g. the central powers Germany and Austria have quite different requirements than the others.

Acknowledgements

We thank the student project group 529 for developing the basics of this work.

REFERENCES

[1] R. Sharp, *The Game of Diplomacy*. Arthur Barker, London, 1979.
[2] S. J. Johansson and F. Håård, "Tactical coordination in no-press diplomacy," in *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2005, pp. 423–430.
[3] D. Livingstone, "Turing's test and believable ai in games," *Comput. Entertain.*, vol. 4, no. 1, p. 6, 2006.
[4] A. Webb, J. Chin, T. Wilkins, and J. P. und Vincent Dedoyard, "Automated negotiation in the game of diplomacy," http://www.doc.ic.ac.uk/project/2007/362/g0736203/TheDiplominator/TheDiplominator/finalreport.pdf, 2008.
[5] G. N. Yannakakis and J. Hallam, "A Generic Approach for Generating Interesting Interactive Pac-Man Opponents," in *Proc. of the 2005 IEEE Symp. on Computational Intelligence and Games*, 2005, pp. 94–101.
[6] ——, "Feature Selection for Capturing the Experience of Fun," in *AIIDE'07 Workshop on Optimizing Player Satisfaction*, 2007, pp. 37–42.
[7] A. B. Calhamer, *Calhamer on Diplomacy: The Boardgame Diplomacy and Diplomatic History*. Authorhouse, 1999.
[8] A. Rose and H. Williams, "Daide protocol," http://www.daide.org.uk/daideint.xml, 2003.
[9] P. D. Windsor, "What's your point," http://devel.diplom.org/DipPouch/Zine/S1999M/Windsor/point.html, 1999.
[10] D. Loeb, "The observation module," http://devel.diplom.org/DipPouch/Zine/S1995R/Loeb/Observe.html, 1995.
[11] ——, "Challenges in multi-player gaming by computer," http://devel.diplom.org/DipPouch/Zine/S1995M/Loeb/Project.html, 1995.
[12] M. Hand, "The diplomatic pouch: The library of diplomacy openings," http://www.diplom.org/Online/Openings/textversion/, 1995.