On Computations and Strategies for Real and Artificial Systems

Amr S. Ghoneim[1], Daryl L. Essam[1] and Hussein A. Abbass[1]

[1] School of Engineering and Information Technology (SEIT)
University of New South Wales at the Australian Defence Force Academy (UNSW@ADFA)
Canberra ACT 2600 Australia
h.abbass@adfa.edu.au

Abstract

A challenge in reproducing life is to reproduce cognition. We propose a methodology by which human actions are analyzed in a real-setting and are then used to evolve artificial neural networks capable of reproducing these actions. It is also demonstrated that analyzing human actions can be used for skill-assessment, where we introduce a model for in-silico computational psychology to assess skills and competency of human plays. The same methodologies can be used by coaches and mentors to diagnose skills for their players and juniors in an attempt to improve their abilities. Results demonstrate interesting patterns in the way expert players develop their skills overtime and that it is possible to reproduce these skills in an artificial context.

Introduction

Establishing a methodology to analyze human actions has a wide spectrum of applications for ALife research. Understanding how human develop their expertise overtime can shed more light in the black-box of human intelligence. In this paper, we look at the dynamics of learning in real human, how skills develop and how the trajectory of skill-development for a human playing a complex game can be assessed. We use these findings to guide the evolution of artificial neural networks to play similarly to the human.

In an early paper in the ALife field, Stewart (1992) argued that life is cognition, that our knowledge and the way we make decisions are particularly crucial determinants for how we evolved. As was put by Varela (1995):

> Yet when it comes to a re-understanding of knowledge and cognition I find that the best expression to the use for our tradition is abstract: Nothing characterizes better the units of knowledge that are deemed most natural.

Many studies focused on understanding the dynamics of learning and evolution (Floreano and Urzelai, 1998). In this paper, we analyze learning based on real human and map it to an artificial model.

In this paper, we consider the game of GO as an example where a human player needs to start from the lowest skill level, working his/her way up to establish themselves as advanced players. We needed to select a gaming environment in general and the game of GO in particular as our test platform for a number of reasons. The beauty of GO lies in the fact that: it has simple rules but large and complex search space. First, we need an unambiguous scoring or ranking scheme. In the game of GO, this is readily available known as the system of kyu and dan ranks. Second, computer games offer a low-risk environment for prototyping artificial learning. Third, online game engines are easy and cheap sources of large amount of data. Fourth, a game such as GO is complex in its strategies, where it relies on human ability to capture spatial patterns and connect information and patterns across the whole board, an important characteristic when we design artificial games or game-theoretic models on networks. The methodology is too generic that it can be applied to both real and artificial spatial game playing.

We structure the rest of the paper in three main sections. First, we present a tiny coverage of the literature related to this paper, taking into consideration that space constraints forced us to remove many references. Second, we present the methodology and analysis using real-human players. Third, we use this analysis to evolve artificial neural networks to reproduce similar behaviours. Finally, conclusions are drawn.

Background Material

Skills and Competency

The term *skills* refer to the learned capacities, whether general or domain-specific, that would be crucial/useful to perform a particular job (Bassellier, et al. 2001). Skills are the component competencies that collectively create the overall *competency*; i.e., the set of skills, knowledge, and qualities or *"behavior patterns"* which are needed to allow an agent to perform tasks/ functions with proficiency (Woodruffe, 1993).

Currently, evaluating the skills of strategic board-game players depends entirely on the degree to which the game's objectives are achieved *(i.e. final outcome)*. Ranking systems – whether online or offline – are virtually the only objective method for automatically assessing the players' experience. However, subjective detailed assessments can frequently be obtained from experts, where different aspects of a player's skills may be evaluated. These types of studies have traditionally been answered through psychological and skill assessment tests (Groth-Marnat, 2009). We extend this approach to a computational environment to overcome the – sometimes – instability inherent in subjective assessment and reduce the resources required to do an assessment.

Learning

Learning can be defined as: given a *task*, a *training-experience*, and a *performance-measure*, a system is assumed to be learning *"if its performance at the task improves with experience"* (Thrun & Pratt, 1998). A similar model can be found in (Osherson et al., 1986), where the learning process classically requires—beside a *learner*—an *item-to-be-learned*, an *environment* wherein the learner is shown the item-to-be-learned, and finally the *hypotheses* arising to the *learner*—given the *environment*—regarding the item-to-be-learned.

This characterizes the relationship between the learning process and *experience*; a concept greatly discussed—whether explicitly or implicitly—in topics related to the *"analysis of human performance"*, or in *"studies of learning and training"* (Farrington-Darby & Wilson, 2006). Experience does not necessary lead to more powerful thinking strategies and/or acquiring directly-perceivable cues—which the inexperienced are usually aware of—but rather to a more efficient employment of the strategies and cues based on the experience-base (Klein & Hoffman, 1993). Hence, experience can *"describe skills, knowledge, or abilities, in tasks, activities, jobs, sport and games"*, and it can *"refer to a process such as decision making or […] to an output such as a decision"* (Farrington-Darby & Wilson, 2006).

Analysing online behaviour and interaction was also investigated in (François et al., 2007), where Self-Organizing Maps where used to classify online interaction between Autistic children and robots to detect the different play styles since *"interaction is decisive in the process of learning through play."* Also, analysing and displaying users' activity and interaction in an online system/community, whether in a competitive way *(e.g. ranking scores)* or an non-competitive way *(e.g. activity statuses)*, was found to draw users attention and motivates users participation (Deiml-Seibt et al., 2009).

Symbiotic Adaptive Neuro-Evolution

Symbiotic Adaptive Neuro-Evolution SANE (Moriarty & Miikkulainen, 1997) is an approach to neuro-evolution where two separate populations are evolved simultaneously instead of evolving a complete network. The two populations are neurons *(explicitly decomposing the search space by acting as local solutions)* and network blueprints *(exploring the best combinations of neurons)*. Blueprints are considered a better alternative to building the networks out of randomly selected neurons. Usually, SANE develop three-tiered feed-forward NNs, evolving neurons for its single hidden layer. Each neuron defines a fixed number of weighted connections that are randomly assigned to both input- and output-layer nodes.

When applied to evolve Go player, each board intersection is represented by an input node for each player, and a single output node. It is illegal to activate both nodes representing an intersection. The first input node – per intersection – is activated iff the corresponding intersection is occupied by a white stone, and vice versa if the intersection is occupied by a black stone, the second input node is activated. An empty intersection is indicated by deactivating both input nodes. A sigmoid activation function is used for the output nodes. The next move is represented by the highest value *(corresponding to the best predicted move)*. If the selected move is illegal, the move corresponding to the next highest activation is selected. However, the network passes if all its output values are below a predefined threshold *(that is, 0.5 in our experimentation)*.

The evolved NNs – the blueprints population – are evaluated by playing a game(s) of Go against the selected opponent, the fitness value is merely the final score(s). As for the neurons population, the fitness value for a neuron is the normalized summation of the fitness values of the blueprints in which it participated. Single point crossover is then applied on mates selected from the elite one third of the blueprints, and *25%* of the neurons, creating two offspring that replace the worst individuals. Mutation is then applied conservatively to the neuron population, and more aggressively to the blueprints *(to maintain high diversity among the network)*.

The Game of Go

The game of Go is the oldest strategic board game in the world, and is also one of the most popular. Though the game is hard, the rules of the game are few and simple, easy to learn, and flexible enough to accommodate any board size as well as the standard *19×19* board. This two-player game, where players alternate placing stones on the intersections of the board, is theoretically in the same category as Chess, as both games are intellectually stimulating, requiring high-level strategic thinking, while also giving the chance for players to apply their tactical skills (Chikun, 1997). The differences between Go and other games (including Chess) in complexity measures is obvious in (Allis, 1994), with the complexities of Go far larger than that of any of the other perfect-information games. Unlike Chess, there are no Go programs that can challenge strong human players (Van der Werf, 2004), nor even moderate human players (Ernandes, 2005). Also, although *9×9* Go boards have a complexity between that of Chess and Othello (Bouzy & Cazenave, 2001), existing Go programs are still immature.

It is worth mentioning that the best known computational model for GO is Monte Carlo Simulation. No neural network or biologically inspired model exists as yet that can outperform Monte Carlo Simulation. As such, this study is a first step to potentially take a different approach towards building neuro-players.

Methodology

The main idea of the proposed methodology is to exploit the possible computational building block(s) of human's actions to assess their skills and competency. The methodology estimates human's skill and competency levels through models trained on historical data of human with known skills and competency levels. The methodology has five main steps:

Subject Identification and Selection: The human subjects to be selected to form the training data need to have gone through multiple competency levels. In other words, this is a longitudinal study that commences with these subjects at a low competency level then moves up to higher ones during the data collection exercise. This is the most expensive step in the whole methodology, time-wise and dollar-wise, in the real world. The game of Go traditionally uses the ranking (rating) system of kyu and dan ranks. In this paper, players with ranks ranging from *30* to *20* kyu are collectively referred to as

Beginners, ranks from *19* to *10* kyu are *Casual* players, *9* to *1* kyu are *Intermediate* amateur players, and finally from *1* to *7* dan are *Advanced* amateur players. Due to some ambiguities in defining the *Professional* dan ranks in the game records, we have decided not to include those ranks in the analysis. We collected the games from *No Name Go Server (NNGS)* online game-archive (Adam, 2009). The cases – game records – were selected from the years' span *1995* up till *2005*. Two datasets are selected separately, a Training Dataset *'trainDS'* which is used to train the proposed classifier, and a Testing Dataset *'testDS'* from which we will select a set of Go players to observe their behaviour.

We selected *381* games for training (*127 for each category; Casual, Intermediate, and Advanced)* based on some strict rules that the games should be complete, with a registered-name, and compatible players. We did not select *'Beginner'* cases because this category contains so much noise. The reason for this noise is that it contains all players who newly joined the server, not necessarily that they are beginners but they have not played enough on this server to establish a rank.

Data Identification and Collection: Every action performed by the human gets recorded. In the context of a game, actions are simply the board moves. In the case of a computer board game, the state of the board at each step of the game gets saved. The training data *(i.e. the data set that will be used to build the model)* needs to be labelled *(i.e. training subjects' skills and competency levels have been assessed by some other means)*, preferably with no missing values, carrying a reasonable number of records for each subject over time and that spans the subject moving from one skill level to another, and of reasonable size. The richness of the data collected per subject, as our experiments demonstrated, means that we do not require a huge dataset to build the skill-assessment model.

Four hundred games were selected for the *testDS* with only *16* games found to be common between the two datasets. The *400* games were played by *246* distinct registered-names *(i.e. players)*. We imposed a threshold of at least *10* games, yielding a final set of *15* players (Table 1) to be used for testing. In the first phase of the experiments, we will run our system using the *trainDS*.

Model Knowledge Initialization: Skill assessment requires a richer understanding of the domain, probably more than what is needed in a traditional data mining task. What is being recorded from the interface is mostly raw data that needs to be grouped, and possibly transformed to a different representation, before it can be used properly for skill assessment. These initial features form the basis for building the actual model.

We use spatial analysis of the board to establish what we call reasons for each move. Assume a move is played in a cell, the spatial analysis will see the different shapes that are newly formed by this move. These 48 reasons are then grouped into seven categories: a category of what it seems a bad move *(anti-suji)*, a category for attack, a category for defence, a category for gaining an advantage, a category for deep planning, a category for end of game and an overall category of all reasons put together. These seven categories are named as: *"Not Recommended"*, *"Considered an Attack"*, *"Considered A Defence"*, *"Explicit Gains"*, *"Thoughtful"*, *"End of the Game"* and *"All Reasons"* respectively.

Player ID	Number of Games Played	The Averaged Experience Range Covered by the Corresponding Player's Games
1	*74*	*Upper-Beginner to Lower-Intermediate*
2	*38*	*Lower-Intermediate to Mid-Intermediate*
3	*32*	*Mid-Intermediate*
4	*46*	*Mid-Beginner to Mid-Casual*
5	*20*	*Lower-Intermediate*
6	*16*	*Mid-Casual to Lower-Intermediate*
7	*35*	*Mid-Casual to Upper-Casual*
8	*13*	*Mid-Casual to Upper-Casual*
9	*36*	*Upper-Intermediate*
10	*26*	*Lower-Intermediate to Mid-Intermediate*
11	*50*	*Lower-Advanced*
12	*11*	*Lower-Advanced*
13	*57*	*Upper-Intermediate to Lower-Advanced*
14	*10*	*Mid-Intermediate*
15	*34*	*Lower-Intermediate to Mid-Intermediate*

Table 1: The final test dataset

It is obvious from the plain definition of each category that these categories can overlap. The Frequencies *'F'*, Frequencies per Step *'FS'*, and the Percentages *'P',* are applied as measurements for the aggregated subsets of the generated-reasons per game. Subsequently, and between each distinct pair of experiences, the Wilcoxon-test and a two-sample T-test were applied to statistically signify the ability of the calculated medians/means to differentiate between the corresponding distinct pair of experiences.

Model Building: The model can vary in its characteristics, ranging from simple statistics to complicated neural networks, decision trees, or classifier systems. The choice of the features in the previous step and the right model in this step are critical and can create all the differences between good or bad skill assessment models.

The *Median* and the *Median Absolute Deviation (MAD)* (Davies & Gather, 1993) were chosen as robust univariate measures in case the dataset is contaminated by outliers *(i.e. observations which appear to be inconsistent with the remainder of the dataset)* and thus subject to masking and/or swamping effects. Human players can be of a wide range of experiences, spanning from beginners to professionals. Given the set of experiences $E = \{e_1, e_2, \dots e_n\}$, let D_e denote a subset of the dataset of all games D where the experience of both opponents is e. The median can be estimated as:

$$Median_{e,s,\varphi} = \left(\varphi_{\lceil(|D_e|+1)/2\rceil:|D_e|,R_s} + \varphi_{\lfloor |D_e|/2\rfloor+1:|D_e|,R_s}\right)/2$$

where φ is the measurement function *(i.e. denoting F, FS, and P)*, R_s is the s*th* reasons subset, $|D_e|$ is the number of games in D_e, and $\varphi_{1:|D_e|} \cdots \varphi_{|D_e|:|D_e|}$ are the order statistics of $\varphi_1 \dots \varphi_{|D_e|}$. Accordingly, MAD can be estimated as:

$$MAD_{e,s,\varphi} = Median\begin{pmatrix} \left|\varphi_{1:|D_e|,R_s} - Median_{e,s,\varphi}\right|, \dots \\ \dots, \left|\varphi_{|D_e|:|D_e|,R_s} - Median_{e,s,\varphi}\right| \end{pmatrix}$$

The medians of the different reasons subsets can model how the general strategy is decomposed into characterizing sub-

Measurements	Reasons' Subsets	Casual Games		Intermediate Games		Advanced Games	
		Median	*MAD*	*Median*	*MAD*	*Median*	*MAD*
Frequencies (F)	*Not Recommended*	*28*	*8*	*35*	*9*	*34*	*7*
	Considered An Attack	*262*	*80*	*337*	*112*	*357*	*84*
	Considered A Defence	*400*	*79*	*471*	*84*	*484*	*72*
	Explicit Gains	*123*	*13*	*138*	*13*	*139*	*14*
	Thoughtful	*134*	*40*	*175*	*47*	*189*	*44*
	End of the Game	*0*	*0*	*1*	*1*	*2*	*1*
	All Reasons	*840*	*174*	*1021*	*191*	*1053*	*163*
Percentages (P)	*Not Recommended*	*3.493450*	*0.5431392*	*3.217822*	*0.4167558*	*3.222919*	*0.3764555*
	Considered An Attack	*30.89655*	*3.544815*	*33.26510*	*3.895654*	*33.63148*	*2.681764*
	Considered A Defence	*46.64372*	*1.643718*	*45.64995*	*1.862967*	*45.76271*	*1.503475*
	Explicit Gains	*15.23702*	*3.438733*	*13.83588*	*3.526759*	*13.05903*	*2.960512*
	Thoughtful	*15.57943*	*1.648398*	*16.92677*	*1.553367*	*17.68140*	*1.462031*
	End of the Game	*0*	*0*	*0.1154734*	*0.1154734*	*0.1552795*	*0.09492900*
Frequencies Per Step (FS)	*Not Recommended*	*0.1206226*	*0.02429229*	*0.1275862*	*0.02576802*	*0.1269231*	*0.02250541*
	Considered An Attack	*1.039024*	*0.2743185*	*1.261649*	*0.3359982*	*1.322222*	*0.2477437*
	Considered A Defence	*1.628099*	*0.1956418*	*1.730375*	*0.2213487*	*1.801394*	*0.1727017*
	Explicit Gains	*0.5088968*	*0.06054765*	*0.5152838*	*0.06837375*	*0.5154639*	*0.06208771*
	Thoughtful	*0.5527273*	*0.1229400*	*0.6518518*	*0.1434485*	*0.7003484*	*0.1255453*
	End of the Game	*0*	*0*	*0.00387596*	*0.00387596*	*0.00666666*	*0.00361788*
	All Reasons	*3.418118*	*0.4618815*	*3.810169*	*0.5740072*	*3.941176*	*0.4627970*

Table 2: The Medians and Median Absolute Deviations (MAD) of the different subsets, among diverse experiences

strategies, and demonstrates the variations in the strategies employed by human Go players of different experiences. To confirm the potential hypotheses suggested by the data, both a two-sample T-test and a two-sided Wilcoxon rank sum test are used. By permuting reasons subsets, estimated measurements, and pairs of different experiences, the T-test and Wilcoxon-test will respectively examine the null hypothesis that the data – *measurements per game* – have equal means/medians against the alternative that the means/medians are not equal.

The two-sample T-test tests a null hypothesis H_0 that the two independent samples come from normal distributions with unknown variances and the same mean, against the alternative that the means are unequal. The test is two-tailed, and performed at a significance level $\alpha = 0.05$, i.e. the probability of mistakenly rejecting H_0 *(Type I error)* is no more than *5%*. Alternatively, the Wilcoxon-test tests a null hypothesis H_0 that the two independent samples come from identical continuous distributions with the same median, against the alternative that the medians are unequal. The Wilcoxon-test is also performed at $\alpha = 0.05$.

In this study, a three-tier ensemble is used to predict the class label of a game of Go as *Casual*, *Intermediate*, or *Advanced*. The first-tier is based on Random Forests (RFs) (Breiman, 2001); ensembles of Classification Decision Trees (CDTs). In order to analyze the reasons, we are looking for a robust white-box model, which can handle data without requiring a lot of data preparation. These requirements suggest the use of CDTs. Each individual classifier *(i.e., RF)* is trained to classify a class and its complement; for example, a RF is trained to classify *Casual* games versus *Not-Casual* (i.e., *Intermediate* and *Advanced*) games, and so on. Thus, each RF outputs two probabilities 'Pr'; i.e. for the previous example, a probability that a given game is originating from the *Casual* class: Pr(C), and a probability that the same given game is originating from the *Not-Casual* class*:* Pr($\neg C$).

In our experiments, a RF is an ensemble of – a maximum of – *1000* classification decision trees. A forest's attributes are determined according to the *Error* and the *Size*; respectively, the minimum error *(i.e., misclassification probability for the out-of-bag observations)* recorded during the process of adding up trees while creating the forest and the ensemble size *(i.e., number of trees)* corresponding to that error value.

The second-tier creates an ensemble of RFs *(i.e., a Forest of Random Forests)* for each class then the joint probability distribution is calculated for two cases: that the instance belongs to the class and that the instance does not belong to the class. The third and final tier combines the results from the second-tier forests using a final gate-function to create for each observation *(i.e., game)* a single probability 'Pr_{final}' per class. The final gate-function combines the probability that an instance is from one class and the probabilities that this instance does not belong to other classes.

Table 2 shows the medians and median absolute deviations among the *127* games per skill level and reasons' subsets. Table 2 shows a statistical difference between casual/intermediate and casual/advanced human Go players, yet it fails to differentiate between intermediate/advanced players. The medians tend to get higher with experience considering both F and FS as measurements; the only exceptions are when the medians of the advanced are lower than or almost equal to the corresponding intermediate in both subsets *Not Recommended* and *Explicit Gains*. Though this apparent correlation between the F/FS medians and the growing experience is expected to some extent, because more experienced players tend to play longer games, the two previously mentioned cases highlights the possibility that more experienced human players are less attracted by direct/instant gains and are more considerate when it comes to not recommended moves. This possibility is supported by medians reported for the measurement P, where the medians of both subsets almost decrease with growing experience.

Using P again, the medians of the subset *Considered A Defence* somewhat decreases with growing experience, suggesting that a more aggressive strategy is applied by well-experienced human players, as opposed to a more defensive strategy by their less-experienced counterparts. The later suggestion is supported by the medians reported for the subset *Considered An Attack* which increase in correlation with growing experience in view of all of the three measurements. The medians reported for both subsets *Thoughtful* and *End of the Game* also increase in correlation with growing experience in view of all measurements.

Thus we can generally claim that, with rising experience, a human player's strategy evolves to a more thoughtful and aggressive strategy, a strategy that cares more about the final steps and eludes the not recommended moves, and last but not least, a strategy that is less lured by direct gains. This claim is statistically supported for human players who progress from casual to both intermediate and advanced experiences.

While using reasons greatly simplified and abstracted the typical knowledge used by humans, the use of aggregated sets of reasons additionally shortened the available reasons and allowed for the highest possible level of strategic abstraction. The three proposed measurements proved to be reasonable in quantifying the strategical aspects of the varying experiences.

Using the features generated per game, an initial preprocessing step is carried on by applying the Minimum Covariance Determinant *(MCD)* algorithm for outliers' detection. A MCD α-value of *0.7* was selected, and all the games tagged as outliers were excluded from the *trainDS*. In this study, outliers are not considered noise or error, rather they are assumed to carry important information that accounts particularly for any unaccounted for parameters when selecting the dataset *(for example, the length – number of moves per game)*. This step is followed by growing RFs that aim to use the previously calculated features to classify the games according to players' ranks. The preprocessing step showed that the measurement *FS* appears less affected by the potentially different or varying mechanism responsible for the outliers. Thus, *FS* is selected as the reliable measurement to monitor the players' competency and skills.

Using the uncontaminated *trainDS*, *30* RFs are trained to differentiate between each experience level and its complement. The *30* RFs trained – per experience, and using the *FS* – are combined to form the second-tier ensemble.

Model Testing: Once the model is built, it gets tested with subjects that were not included in the model building exercise. Upon successful testing, the model is ready for use.

Using the *testDS*, the games for each player are temporally ordered and then reasons were extracted to estimate the strategic reasoning behind the moves. *FS* is then applied – and combined according to the aggregated reasons' subsets – thus creating the final feature set for each game. The proposed classifier generates three final probabilities for each game: $Pr_{final}(C)$, $Pr_{final}(I)$, and $Pr_{final}(A)$ for Casual, Intermediate and Advanced respectively. Given the number of available experience-levels as $N_{classes}$, and the total number of games per a single player as N_{games}, three Competency *Monitoring-Curves* are plotted for each player; each representing the 'un-weighted' Cumulative Moving Average (CMA) for an experience-level, with a maximum *window size* of *50* games.

For space limitations, we will only show the results for one player with average predictive results to make the discussion more interesting. Figure 1 presents a 2-dimensial line graph with two *y*-axes for the player. The left *y*-axis – *'Player's Experience Curves'* – displays the value of the three *Monitoring-Curves (i.e., generated probabilities)*, while the right *y*-axis – *'Ranks' Categories'* – displays the *Player's Rank* according to the online *NNGS* archives. The *Player's Rank* curve is also a CMA of the actual rank-values. A label on the right *y*-axis represents the center of the respective rank category. The *x*-axis displays the game number, with imposed temporal frames for the corresponding dates (months/years).

The monitoring-curves in all of the resulting figures (including those not-shown) show a clear consistency between the experience level of a player and his/her probabilities' curves. That is, as the player 'assumingly' gains more experience with time, the probabilities' curves reflect this learning activity by either declining or rising. Player#1 advances from an Upper-Beginner to a Lower-Intermediate experience over the course of the *74* games selected. Concurrently, the Casual-monitoring-curve of the mentioned player declines from *0.875* to around *0.5625*, the Intermediate-curve also converges to around *0.5625* rising from *0.4375*, and the Advanced curve is also rising from *0.3750* to a little bit higher than *0.5*.

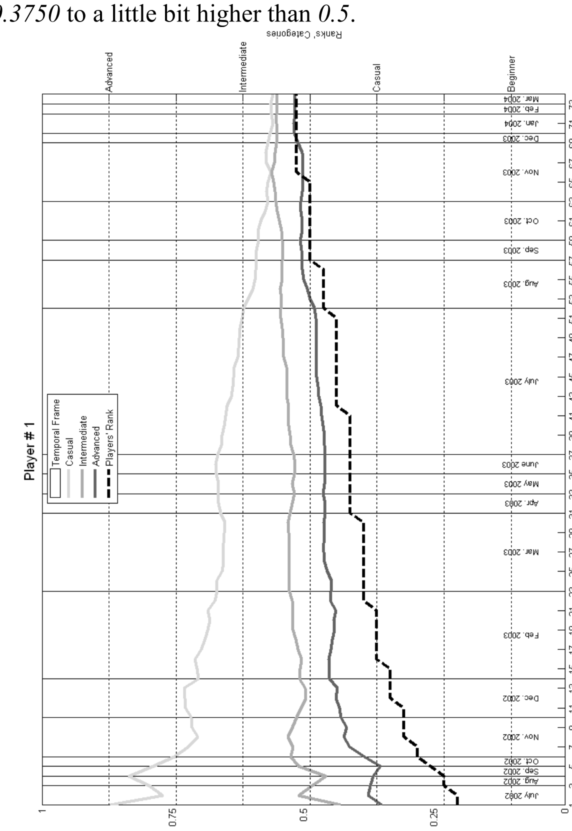

Figure 1: The Competency Probability-Curve for Player 1

Though, on strict classification bases, this player is classified as a Casual during the whole period *(since the Casual probability curve is higher than both the Intermediate and Advanced)*, the clear trend in the curves assures that – with more games – the player is going to be correctly classified as Intermediate. Obviously, classifying a Beginner player as a Casual is reasonable in this context, since no Beginner cases where included in *trainDS*.

In this context, we would like to point out that even though the classifiers are trained using cases only from the mid-range of an experience level – in order to minimize the 'strategical' overlapping between the different levels – this is not the only reason for misclassifying games from around the boundaries between ranks. Alongside the potential personal-influences, we would like to refer to the case that expecting from an expert – for instance – a consistent performance at that level in all subtasks might be a mistake (Klein & Hoffman, 1993). Thus, a player who advanced from being a Casual to the Intermediate level is not expected to show this level of Intermediate-like proficiency in all aspects of the game.

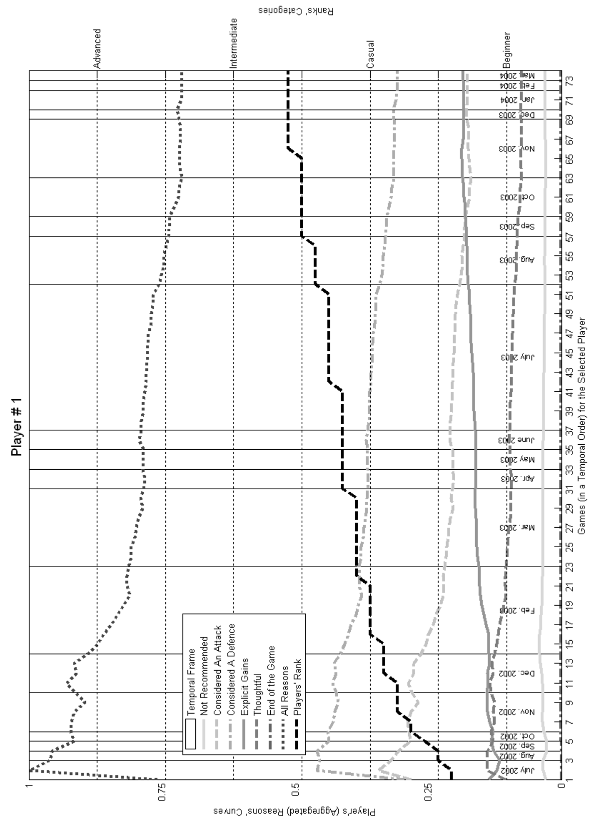

Figure 2: The Skills Probability-Curve for Player 1

Here we reach the final stage of our results, in which we diagnose the skills learning-activity of human Go-players by temporally observing each of the strategies' characteristics. A straight benefit of the figure is the distinctive opportunity to realize how the strategic reasoning of human Go-players is decomposed among the available strategies' characteristics, and how those characteristics evolve temporally with experience. Figure 2 shows the normalized un-weighted CMA of the *FS* directly measured from the games. As a player progresses from being a Beginner to lower-Intermediate – through being a Casual – the categories 'All Reasons', 'Considered an Attack', 'Considered a Defense', and 'Thoughtful' seem – in general – to decline slightly with experience. The 'Explicit Gains' appears to be the only curve rising during the Beginner to lower-Intermediate progression. These findings are obvious in Figure 2. On the contrary, progressing from the Intermediate to Advanced shows precisely the opposite behavior. As players progress through the Intermediate rank and to being Advanced, the categories 'All Reasons', 'Considered an Attack', 'Considered a Defense', and 'Thoughtful' unevenly increase with experience. Unsurprisingly, 'Explicit Gains' is the only curve declining during the Intermediate to Advanced progression. Both the 'Not Recommended' and 'End of the Game' curves show subtle variations during the entire experience range.

Our earlier findings seem to agree with only half of the later findings, that is, the changes occurring as a player progresses through the Intermediate rank and to being an Advanced. This apparent disagreement – where the categories 'All Reasons', 'Considered an Attack', 'Considered a Defense', and 'Thoughtful' seem to decline as a player advances from being a Beginner to a lower-Intermediate – can be attributed to two associated reasons. At first, in Table 2 no Beginner games matched the selection criteria, and therefore, the progression from Beginner to Casual was not investigated. That leads us to the second reason; the window size of *50* games employed in the CMA considers this 'history' of being a Beginner when the player has already advanced onto being a Casual, thus affecting the curves for an additional period.

Each skill monitoring figure characterizes how a player is evolving with the experience he/she is gaining. Plain benefits of such results is the ability to construct customized – *to a specific-personality / level-of-expertise* – learning processes; designed tasks that convey to a learner the missing bits of knowledge/skill/understanding, by which gaining an experience might be assured and/or accelerated. Another potential benefit is the ability to clone a person/level-of-expertise, possibly for creating an automated instructor.

Strategically Aware Fitness Measurement

In this section, we advance the findings of the previous sections by asking whether or not we can use it, not only to analyze human subjects, but also to guide an artificial evolutionary process. This is what we will call a strategically-aware fitness function.

To accomplish this goal, SANE was used to evolve *9×9* neuro-Go players using both the traditional exclusively score-dependent fitness function, and the proposed strategically-aware function. The networks are evolved against the GNU Go engine as an opponent. To compensate for the additional computational cost of estimating the strategies in the Go games, *50* blueprints are evolved instead of the *200* suggested by (Richards, et al. 1998). Due to the nature of the problem in which a network is evaluated by playing a game, and in spite of using elitism, the fitness values across generations fluctuate due to stochastic effect.

After generating the Strategically-Aware component *(TP)*, the traditional score-based fitness function can be modified by simply adding the generated probability to the game score. The effect of the added term can be tuned by the coefficient *α*. The proposed fitness measure *f* for a *Network* is calculated as:

$$f_{Network} = \sum_{i=1}^{N_{Games}} \frac{\alpha\, Score_i + (1-\alpha) TP_i}{N_{Games}}$$

where N_{Games} is the number of games played by *Network* in the evaluation phase, $Score_i$ is a value – in the range from *0* to *1* –

representing the *Network*'s score in game *i*, while TP_i is the *Trained Probability* generated by the RF for the game *i*.

To investigate the effect of the added TP, the coefficient α was varied, using four values; *1.0*, *0.8*, *0.2*, and *0.0*. The first α value represent the traditional score-based fitness function, while α set to *0* represents the case where the networks' evaluation is based entirely on the Trained Probability.

In General, the parameters in the experiments are based on those found effective in (Richards, et al. 1998), except for the number of blueprints which was reduced from *200* to *50*. A single run consists of *500* generations, and *10* different runs were evolved. The *500* generations are twice the number of generations required by SANE to evolve a network capable of defeating Wally on *9×9* boards in (Richards, et al. 1998). Since Wally is a trivial engine when compared to GNU Go, GNU Go's level was set to *1* throughout the experiments instead of the default of *10*. However, GNU Go – even when playing at level 1 – is much more developed than Wally. Therefore, we do not expect to evolve a NN that is capable of defeating GNU Go, but a network that has developed enough strategies to be explored.

The games were scored using Chinese rules. The networks were always evolved to play White, thus never making the first move. The komi value – necessary to avoid a tie – was set to *0.5*, and no handicap stones were given to the networks. An upper bound of *200* moves per game was placed, to ensure that unreasonably long move-sequences that are probably suggested by the untrained networks are not pursued. This experimental setup cost up to a maximum of 7 days for a single run per an α value using a Sun Constellation Cluster.

Results and Discussion

In order to investigate the effect of the proposed fitness function, two different types analysis – to the neuro-evolution process – are to be shown and discussed. We start by showing and discussing the convergence among the varying α values. Then a Tournament between selected players and GNU Go – set to different levels – is held.

Figure 3 shows the convergence of the *50* blueprints and *4000* neurons evolved using SANE for *500* generations. For each α value, the average fitness – of *10* different runs – for the 1) best network, and the 2) entire population are plotted.

The convergence of the fitness values or all of the different combinations enters a relative plateau, staring from around generation number *50* for both α values of *0.2* and *0.8*, and followed by generation number *150* for both α values of *0.0* and *1.0*. The same is true for the convergence of the entire population, except for $\alpha = 0.0$ where the population seems to continue evolving. Notably, the 'relative' difference between the best network and the population in terms of fitness values decreases with an increasing α, except for $\alpha = 0.8$ which shows the lowest difference. A possible explanation is that while depending more on the TP component rather than the score, the evolving networks increasingly fluctuate between the generations. However, setting to $\alpha = 0.8$ shows a less varying fitness than $\alpha = 1.0$, even in other detailed figures that are not shown here due to page constraints.

The first step to investigate the playing capabilities evolved and whether it takes advantage of the engine's weaknesses is by holding a tournament between selected representative players and GNU Go. As we mentioned before, SANE used a weaker player than GNU Go at level 1 as an opponent. The behavior of the evolved players when playing against GNU Go set to different levels will shed a light on the type of the strategies evolved. The tournament involves GNU Go at *10* different levels, starting from *1* (weakest) to *10* (default).

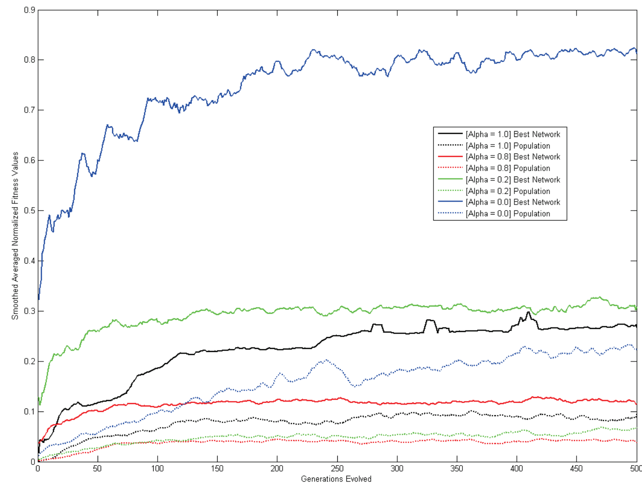

Figure 3: The Convergence of the Fitness Values

A simple and straightforward criteria is used to select a representative player for each of the varying α values; the network achieving the overall best 'game score' across the *10* runs and the *500* generations. Figure 4 shows the best games' scores across the *10* runs, the best score for each α is encircled. The maximum possible score using the Chinese rules – and a komi value of 0.5 – on a *9×9* board is *81.5*.

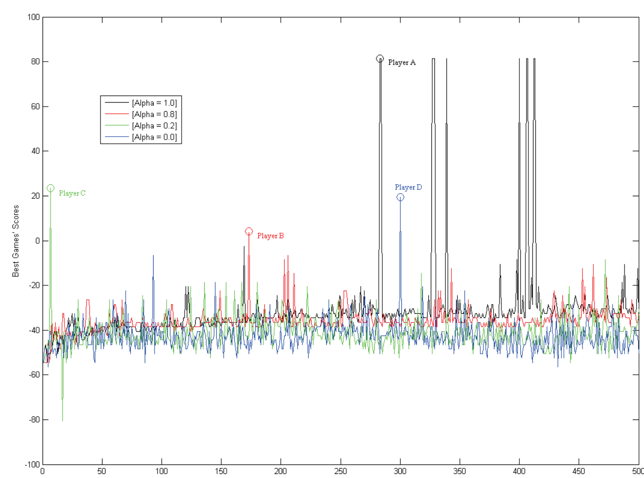

Figure 4: Best Games' Scores across the 10 runs

The tournament consists of the four selected players versus GNU Go at *10* different levels. The players will be named *PlayerA*, *PlayerB*, *PlayerC*, and *PlayerD*; representing respectively the α values of *1.0*, *0.8*, *0.2*, and *0.0*. For each pair – that is, a selected player versus a GNU Go at a single level – *30* different matches were played. The komi value is set to 0.5, the games are scored using Chinese rules, and the GNU Go always starts the games.

Table 3 shows the percentage of Wins of the 4 selected players against GNU Go. Even though none of the players were able to defeat GNU Go at a level higher than the one they were evolved against, as α decreases, the percentages of wins against GUN Go at level 1 increases until is reaches 20% of the games for *PlayerD*. This finding strongly suggests that the networks evolved using the proposed fitness function evolve different varying strategies to defeat the opponent. Even if those players were selected from premature generations; *PlayerC* was evolved in the seventh generation.

		Selected Players			
		A	*B*	*C*	*D*
Details	*Alpha Value*	1.0	0.8	0.2	0.0
	Corresponding Generation	283	173	7	300
The Level of GNU Go	*1*	6.7%	10%	13.3%	20%
	2	0%	0%	0%	0%
	3	0%	0%	0%	0%
	. .	. .	. .	. .	. .
	10	0%	0%	0%	0%

Table 3: Selected Players' Details and Percentages of Wins

The main objective in a game of Go is to secure a territory. The capability of creating and defending a group of connected stones that remains alive – i.e., do not get captured – until the end of the game is fundamental to a go player. Therefore, the final scores of the games, even in cases of loosing, are meaningful to our analysis. Players that can secure bigger territories than other players, and which will be reflected in the final score, are relatively more trained.

Figure 5 shows the average score of the selected players against GNU Go. Since GNU Go always plays as the black, and given the komi value of *0.5*, the minimum possible score for the selected players is *-80.5*. All players report their best results when playing against level *1*. For higher levels, Players *C* and *D* report the minimum possible score. However, *PlayerB* reports better average scores in most of the higher levels than *PlayerA*.

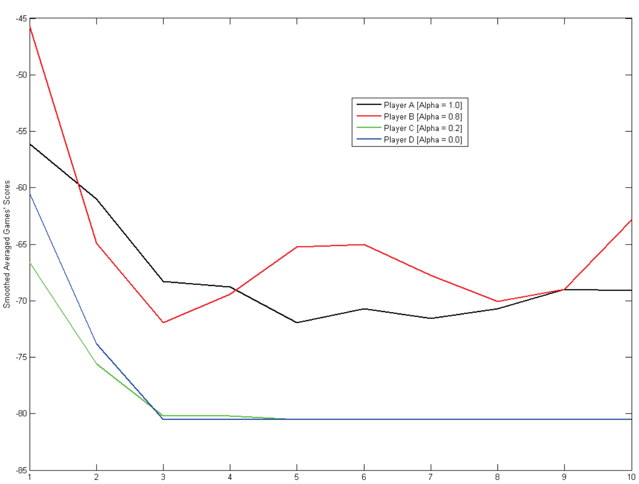

Figure 5: The Average Score against GNU Go

Conclusions and Future Work

We provided a methodology for an automatic and objective assessment and monitoring of human-players' skills and competencies in the game of Go. The generality of the approach entails that the models can be used to assess artificial players as well, which we successfully demonstrated using an Artificial Neural Network. The findings are seen as advancement towards better understanding of human strategies to assess the skill levels of humans. For example, if player's skills are constant for a while, and if the objective is to improve the performance of that player, the artificial life environment or a game environment may switch to some training scenarios to improve the specific skills which have been stagnating. If the aim is to entertain the person, the game may alternate between an easier version and a harder version.

References

Adam, G. (2009) No Name Go Server Archives. http://computer-go.org/pipermail/computer-go/2006-April/005343.html (12-Oct-09)

Allis, L. (1994). *Searching for Solutions in Games and Artificial Intelligence*. Ph.D. thesis, Maastricht University, The Netherlands.

Bassellier, G., Reich, B. and Benbasat, I. (2001). Information Technology Competence of Business Managers: A Definition and Research Model. *Management Information Systems, J.*, 17:159–182.

Bouzy, B. and Cazenave, T. (2001). Computer Go: An AI oriented survey. *Artificial Intelligence*, 132:39–103.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45:5–32.

Chikun, C. (1997). *Go: A Complete Introduction to the Game*. Kiseido.

Davies, L. and Gather, U. (1993). The Identification of Multiple Outliers. *American Statistical Association, J.*, 88:782–792.

Deiml-Seibt, T., Pschetz, L. and Müller, B. (2009). A conversational model to display user activity. In *23rd BCS HCI 09 Conf. on People and Computers*, Swinton, UK.

Ernandes, M. (2005). Artificial Intelligence & Games: Should Computational Psychology be Revalued? *Topoi*, 24:229–242.

Farrington-Darby, T. and Wilson, J. (2006). The nature of expertise: A review. *Applied Ergonomics*, 37:17–32.

Floreano, D. and J. Urzelai (1998). Evolution and learning in autonomous mobile robots. In D. Mange and M. Tomassini (Eds.), *Bio-Inspired Computing Machines: Towards Novel Computational Architectures (1st ed.)*, Chapter 12, pages 317–364.

François, D., Polani, D., and Dautenhahn K. (2007). On-line behavior classification and adaptation to human-robot interaction styles. In *ACM/IEEE Inter. Conf. on Human-robot interaction*, NY, USA.

Groth-Marnat, G. (2009). *Handbook of psychological assessment*. Wiley.

Klein, G. and Hoffman, R. (1993). Seeing the Invisible: Perceptual-Cognitive Aspects of Expertise. In Rabinowitz, M. (ed), *Cognitive Science Foundations of Instruction*. Lawrence.

Moriarty, D., and Miikkulainen, R. (1997). Forming Neural Networks through Efficient and Adaptive Coevolution. *Evolutionary Computation* 5, 4:373–399 Erlbaum Associates, New Jersey.

Richards, N., Moriarty, D., and Miikkulainen, R. (1998). Evolving Neural Networks to Play Go. *Applied Intelligence*, 8:85–96

Osherson, D., Stob, M. and Weinstein, S. (1986). *Systems that learn*. MIT Press Cambridge, Mass., USA.

Stewart, J. (1992). Life=Cognition The epistemological and ontological significance of Artificial Life. In F. Varela and P. Bourgine: Toward a Practice of Autonomous Sys., *Euro. Conf. on A. Life*, MIT Press.

Thrun, S. and Pratt, L. (1998). *Learning to learn*. Kluwer Academic Pub.

Van der Werf, E. (2004). *AI Techniques for the Game of Go*. Ph.D. thesis, Universiteit Maastricht, Maastricht, The Netherlands.

Varela, F. J. (1995). Some Biological Ingredients for a Nouvelle Cognitive Science. In L. Steels and R. Brooks (Eds), *The Artificial Life Route to Artificial Intelligence*, 11–22.

Woodruffe, C. (1993). What is meant by a competency? *Leadership & Organization Development, J.*, 14:29–36.