SHORT AND LONG TERM MEMORY IN COEVOLUTION

Phillipa AVERY¹, Zbigniew MICHALEWICZ², Martin SCHMIDT³

¹Department of Computer Science,
University of Adelaide, Adelaide, Australia
E-mail: pippa@cs.adelaide.edu.au

²Department of Computer Science,
University of Adelaide, Adelaide, Australia
E-mail: zbyszek@cs.adelaide.edu.au

³SolveIT Software,
Adelaide, Australia
E-mail: martin.schmidt@solveitsoftware.com

Abstract
Games provide the perfect test bed for measuring the effectiveness of computer generated strategies in a competitive and fun environment. Over the years many different games have been tackled by researchers of computational intelligence with the purpose of creating an intelligent computer player that can challenge human players. In this paper the authors summarize the research performed over the past two years based on the game of Tempo, where the coevolved strategies were represented by logic rule bases with an adaptive memory. The experiments were set up to investigate the effectiveness of various memory structures in a co-evolutionary game system, as well as the effectiveness of various recall processes.

Keywords: Computational Intelligence, Adaptive memory, Tempo

1 Introduction

Coevolution has proved to be an effective way of creating self learning computational systems that develop appropriate strategies for a problem. This
has been shown repeatedly in research where coevolution has been used to create computer players for a variety of games [1][2][3][4][5][6]. The research in this paper follows on from this previous research by including the addition of a memory in the coevolution of a computer player.

As discussed by Ficici in [7], the coevolutionary process can forget previously learned solutions in favor of ones that are more effective against the current opposition. This can be detrimental to the overall usefulness of the system. This issue of forgetting previously good solutions is well documented in coevolutionary research, and when analyzing results from experiments we found this forgetting lead to the creation of strategies that were not highly competitive. By introducing a memory we have attempted to increase the aptitude of our computer players to play against all strategies, not just the current opposition.

The forgetfulness of coevolution is a logical effect of the generational process, as the goal of each individual is to beat the opposition for the current generation only. However this is not always the most desirable outcome. As discussed by Cliff and Miller [8], a later generation individual should be able to beat an early generation opponent. This is not always the case however, as ancestors of the opposition could have strategies that the later generations have forgotten about and have no way of beating. One solution to this is to incorporate some form of memory into the coevolutionary process, one that helps the populations to remember previous solutions [4][7].

The use of memory in coevolution to retain previous winning (best of generation) solutions has been researched using different memory mechanisms. Some involve retaining the best of generations for insertion or replacement into populations in latter generations [9], while others use the historical memory to compete against and influence the fitness [4]. Whichever mechanism used, there remain the same questions regarding the historical population’s representation and selection. These questions can be summarized as [9]: how should individuals be selected for insertion into the history, what size should the history be, and how should an individual from the history be selected for use in the evolution.

Most systems choose to use best of generation individuals, however this is not the only option as discussed by Bader-natal and Pollack [10]. For the purpose of this research however, we are using the history to remind current generations about the successful strategies of past generations, so only the best will be stored. Having decided which individuals to be inserted, the next step is to decide when the individuals should be selected and stored. The selection of individuals directly affects the question relating to the size of the population,
and the selection of an individual from the historical population for use in the evolution. For example, storing the best individual from both populations at every generation can cause the historical population to grow rapidly, and the selection of individuals will be influenced by the larger scale.

We examined a variety of methods for the representing the historical population. One particular choice was whether to periodically remove older (and perhaps less efficient individuals) or constantly grow the population until program termination. We chose the latter for the experiments, due to possible relevance of older individuals. A number of experiments on the effectiveness of this method is discussed in section 4. We chose to use the history by selecting individuals and playing against them, which then affects the fitness of the individual. This differs from other mechanisms such as insertion of older solutions as described by Branke [9], or even using the memory to initialize the population as in [11]. This research focuses mainly on the mechanisms for selecting what individuals to use from the memory however, and not on what is done after selection.

This paper will also discuss the Tempo game and the computationally intelligent mechanisms used to create a computer player for the game. The previous research performed by Johnson et al. [6][12] is discussed in relation to the improvements made for the purposes of this research, particularly the change from a Gaussian membership function for the fuzzy rules, to a triangular one.

The research described in this paper is the culmination of two years research by the authors on the addition of memory to coevolution. The paper summarizes our earlier results reported in [13][14], and provides further experimental results for inclusion of migration crossover.

The paper is organized as follows. Section 2 gives a background of the main topics of this paper, including a description of the Tempo game and the previous research performed by Johnson et al., and a discussion on the way the human memory system works. Section 3 discusses the different methods we have used to investigate and test the addition of memory to the system. Section 4 outlines the results of the experiments performed, and a discussion of observations from the results. The final section concludes with a discussion on the overall findings.

2 Background

For a background to the research described in this paper, we discuss the game used; the Tempo military planning game. The discussion includes the game play involved in Tempo, and the previous computationally intelligent
system developed for the game.

Following this, a background is given into the psychology of human memory. This gives a very brief introduction into how the human memory system works, and in particular focuses on the way humans use short and long term memory.

2.1 The Tempo military planning game

The use of strategic thinking is not limited to game playing, and many of the strategies used by players can be carried into a real-world situation. This can be seen in business and defense organizations, where they are essentially competing against rival organizations for a strategic dominance in their field of expertise. This real-world situation can be directly compared to a zero-sum game of strategy, where there are two or more competitors, and only one can win.

In the area of defense, this game playing occurs when countries engage in espionage and weapon research and manufacturing with the purpose of maintaining sufficient utilities to win a war against rival countries. The maintenance of only sufficient utilities is important, as this implies that there are enough utilities to win a war, but that the country has not been excessive in its spending. To achieve this fine line, the personnel who create the resource allocation for the defense organizations need to know how to think strategically. These strategies involve the analysis of a great number of complex input fields, with the resulting budget allocation for the various areas of defense for the year. The analysis of the resource allocation is made difficult due to the influences on allocations, such as the political motivations of current (and future) governments, and of course, having other countries with their own resources to compete against [6].

The US Department of defense (DoD) realized the difficulty involved in this task, and in an attempt to give their personnel an education in the strategies required, they created a management system known as the Planning, Programming, and Budgeting System (PPBS)[15]. This system put into place a framework for the decision making of defense budgeting and incorporated a set way of planning for current and future objectives. As part of the set-up of this new system, a major training program was initiated to enable the personnel to use the complex new system. The game of Tempo was created by General Electric’s “think tank” as part of this initiative, and was used by the DoD in the training of their system personnel [6]. The game gave personnel a fun way to practice the strategies they would use in the creation of the resource allocation, and subsequent yearly DoD budget.
Tempo is a zero sum game played between two opposing parties by allocating resources in a cold war style simulation. The goal of the game is to acquire more utilities than the opposition before war breaks out. The decision making process requires allocating a yearly budget resource between offensive and defensive weapons of various strength and cost. The purchase of intelligence is also provided to give insight into opponent’s tactics. Lastly, investment in research and development is available to provide for future weaponry.

The resource allocation involved in the game is in concept relatively simple; determine who needs what and then allocate accordingly. The reality is however very different, as the number of different combinations of allocation plans can be high due to the amount of areas to allocate to. This complexity is then magnified by the changing environment that occurs yearly, such as the increase in the chance of war breaking out, and the addition of new weaponry.

The Tempo game was used in a coevolutionary system by Johnson et al. [6][12] to test the use of evolving a self-learning artificial player for the game. The system followed similar work done by Chellapilla and Fogel [16][2] where a computer player developed its own method of play through coevolution. Instead of using the neural network approach done by Chellapilla and Fogel however, the player was developed by coevolving a set of fuzzy rules.

The system developed in [12] used a Mamdani fuzzy logic system with Gaussian membership functions. The coevolution involved two populations of individuals coevolving against each other. The chromosomes of the coevolutionary process are rule bases with each individual comprising a number of concatenated rules up to a maximum of $m$ rules where $m$ is made up of $w$ weapon rules and $q$ intelligence rules. Each of these rules are built from data relating to: whether the rule is to be used, whether each of the input values for the rule are to be used, the corresponding fuzzy membership function for each input, and the output membership function of the rule. These are then used to decide if a particular intelligence category or weapon should be bought when the yearly budget allocation is performed, using the production operation rule for fuzzy-AND. After each generation the populations are evolved with a chanced crossover and mutation operation.

The fitness of the system involves each individual in each population playing a set number of games against random individuals from the opposition population. The outcome of each game is recorded, and at the end of all game play the fitness of each individual is represented by the calculated average net utilities with a penalty applied to minimize the number of used rules in the rule set, and therefore apply the Occam’s razor principle. The variation operators consisted of a two point crossover and an arithmetic mutation with a small
probability of a major mutation and a large probability of small mutation (for more details see [12]).

The research presented in this paper used the system from [12] as a baseline for the research, and conducted further experiments to refine and improve the former computer player. There were a number of issues that were observed by the research from [12] relating to the ability of the player to compete against humans, and this research is intended to further this work. One particular observation was that the rules being developed by the coevolution process were difficult for a human to beat initially, but easy to overcome with practice. We felt that a large reason for this situation was due to a lack of memory on behalf of the computer player. Without being able to remember previous games, the player was only focusing on the strategy of the moment, instead of developing generic rules of play.

2.2 A discussion on the psychology of human memory

In our experimentation it was decided to investigate the use of two types of memory for selection, one as a long term memory (LTM) and one as a short term memory (STM). The approach was to simplistically mimic the way the human brain stores and uses memory. The concept of applying STM and LTM to heuristics has been widely used over the years, and the use with tabu search has been thoroughly documented by Glover and Laguna [17]. In this research however, we seek to mimic the human mechanism of STM and LTM in a coevolutionary process.

Early theories of human memory consisted of one large memory system. However, in the 1960’s researchers recognized that the memory consisted of two parts: a short term memory that acted as a Temporary storage mechanism, and a long term memory that was a permanent memory storage [18]. This was re-examined in the 70’s and the two memory systems broken down into further levels of processing [19]. The STM was re-termed the working memory, and broken down into articulatory and visuospatial components. The LTM was broken into explicit and implicit memory. Implicit memory involves learning that does not involve active recollection of information, but rather retrieval through indirect performance. Implicit memory includes information associated with: skills and habits, priming, conditioning and non-associative learning [19]. Explicit memory on the other hand is the active recollection of past incidents, and the semantic memory representing general knowledge of the world. Explicit memory includes storage and retrieval of facts and events.

The actual linking between short and long term memory was originally thought to happen serially; that the information being taken in by our sensors
was processed in the STM and then if it stayed in the STM long enough, it would be transferred into LTM [20] [19]. This was then challenged by Craik and Lockhart [21] who introduced the concept of a *levels-of-processing* framework. This framework had a sequence of analytic stages that show how memory is set. The idea is that memory is not a separate faculty, but instead reflects the outcome of attempts to perceive and comprehend information. Thus, the ability to affect long term memory is directly related to the comprehension or ability to relate the knowledge with the meaning (semantic understanding). In this way, the retrieval of information is not done in a ‘brute force’ search mechanism, but instead the memories are encoded into the whole cognitive system, and there is an increased potential for the same pattern to be repeated on a subsequent occasion [20].

The understanding of how the STM and the LTM work, both independently and together, is still a wide area of investigation. However, the knowledge that the memory is divided into two forms: a short term memory aimed at recalling recent and relevant information, and a long term memory to store necessary information in the long term, are fundamental theories for human memory.

3 Methodology

We have addressed several of the issues mentioned by Johnson et al. [12] regarding system performance, and also investigated memory mechanisms used in addition to the original system. The following section is broken into the improvements we made to the system to increase the processing speed, the method we used to seed the populations with human strategies, what methodologies were used in the memory experiments we performed, and the inclusion of migration into the system.

3.1 Processing improvements

To address the speed of the system, the fuzzy logic rules process was evaluated. While the general concept of the system was well founded, changes to the fuzzy logic system and the representation of the genes could possibly improve the performance of the system.

The original Tempo fuzzy logic system was based on the Mamdani fuzzy logic system with Gaussian membership functions (see [12] for further details). We noted that the use of a Gaussian membership function with floating point precision resulted in the evolutionary process fluctuating over small variations that did not really improve the overall results. This slowed the system with
no gain. To counter this, we changed the Gaussian membership function to a triangular membership function with integer precision. It was thought this would stop the minor fluctuations observed with the floating point precision, as the integer representation involved less mutation variance. This meant the system would be able to spend time looking at a larger area of the search space.

The integer membership function was represented by giving a corresponding integer value for each fuzzy membership function, e.g. 0 for very low, 1 for low, up to 5 for very high. The previous gaussian method differed from this by having floating values center and sigma for each function, which could be mutated by multiplying a randomly allocated minor or major value. The new mutation mechanism for the integer precision involved randomly assigning a new membership function for a major mutation, or moving to an adjacent membership function for a minor mutation. The change was implemented at the start of the experimentation, and the results showed that the processing time was cut down to a third of the previous speed, and the results against the static expert were also marginally improved.

The new representation of the fuzzy logic system meant that the chromosome structure used in [12] needed to be updated. The new structure can be seen in figure 1. As with the old system, there are \( m = w + q \) rules (where \( w \) is the maximum number of weapon rules, and \( q \) is the maximum number of intelligence rules). Each rule is built from the following (Figure 1 expands rule 3): \( U_3 \) is a Boolean defining if the rule is used, \( B_{3i} \) are Booleans defining if input \( i \) is used, \( MF_{3i} \) is the membership function used for the input \( i \), and \( Y_3 \) is the output in range \([0,1]\) for Rule 3. The membership functions were created for each input value and represent a variable number of functions for each input.
3.2 Seeding the populations

One suggestion made by [12] was to seed the population with hand crafted individuals to determine how more human-like individuals would fare in the coevolutionary process. We investigated this area by conducting a simple experiment to determine the reaction of the coevolutionary process. The existing base-line measurement of the system consisted of an expert that used static rules to represent the strategy of; buy weapons based on their utilities per operation cost. The higher the ratio, the more of the weapons would be bought. The term “expert” is used loosely here, as the strategy is a simple concept. However, the strategy has proved capable of winning against novice human players of the Tempo game, and can be considered a good base-line player for measurement of the system performance.

When using the expert for performance measurement, the best individuals from the generations performed relatively poorly on average against the expert. This was to be expected to some degree, in that the populations were focused on beating each other, and they had no incentive to find strategies to beat the expert. We decided to insert the static expert as an individual into each of the population and record how it fared. We already had the measurement abilities in place, as the best individuals from each generation would play the static expert at the end of the generation, and the alien expert would be seen in the results from the static game play. To differentiate between the two functions of the same expert, we named the base-line measurement use of the expert as the “static expert”, and the seeded expert the “alien expert”.

The alien expert mechanism worked by placing the static expert as another individual in each of the populations. The individual would be subject to the same variation operators as all the other individuals. The base-line static expert measurement system remained the same. At the end of each generation, the best individual from each population would play 100 games against the static expert and the amount of times the individual won would create the won ratio against the expert.

The alien expert was used in the experiments with memory primarily to see if the memory helped keep alive strategies to beat the alien expert. This was then further tested to determine if the inclusion of the alien expert did in fact help to kick start the coevolution to create strategies to beat the expert strategy.

3.3 Use of a history

An effect of coevolution has been described as the Red Queen effect [22][8], where two or more populations are constantly changing their traits to compete
with the other players. This can lead to cyclic evolution, where previous strategies are constantly forgotten and rediscovered. After experimenting with the alien expert, we realized that our coevolutionary strategy did just this. The populations were focused only on outdoing each other, and even after expert knowledge was seeded, as soon as the populations had beaten the expert in the other population, they then developed new strategies, which the other population focused on beating, hence ignoring the expert strategy. Occasionally they might rediscover bits of strategies that would beat the expert, but as they were not rewarded for these discoveries, they were promptly forgotten.

We wanted to provide a player that would be competitive against all strategies, not just the strategy of the moment. This meant developing players that could beat the current opposition, as well as previous opposition; an ancestral tree where the progeny learn from the ancestors. Following this line of thought, we decided to incorporate a memory structure that would act as a record of historical strategies. We decided to store the best individuals of each population into a separate population that we called the History Bank. We then used the History Bank in the evolution of the individuals of the population, where at the end of the generation, the fitness of each individual would be created by playing against the opposition and the individuals from the History Bank. We wanted this to act as a trigger so that old winning strategies would not be lost in the time line, but would help in developing the future generations.

As discussed before there are a number of different ways we could implement and use this History Bank. We started by implementing a separate History Bank for each population, and each generation the best of the opposition was stored in the other population’s associated History Bank. This was intended to remind Population A of Population B’s winning strategies, and vice versa. After further experimentation and analysis however, we came to the conclusion that to create a truly competitive individual it should be able to beat the opposition, as well as any previous strategies of its own. We decided to experiment with having a single combined (and decidedly larger) History Bank that both populations would add to and play against.

The mechanism used to remind the current players of the history, was to use the History Bank as opposition against the current populations. At the end of each generation, each individual would play a defined number of games against the opposition population, followed by a defined number of games against the individuals in the History Bank. The fitness we use is calculated as the won ratio of the individual (the number of games won divided by the number of games played) adjusted by the amount won by, with a penalty applied to minimize rules and inputs used for the fuzzy rule base. Because the fitness is
calculated against totals of all games played, it is directly influenced by each game it plays. Our strategy uses this by playing each individual against random individuals from the opposition $r_1$ times. By using random selection, each individual plays a different sample from the opposition population and may play more games than other individuals (if chosen to play by the opposition’s random game selection). This is slightly countered in the fitness however, as the win ratio is calculated as an average of all games played. The individual is then played an additional $r_2$ times against individuals from the history, where the selection mechanism changes as explained in section 4 and its subsections. So the total games played to assign the fitness for the individual is $r_1 + r_2$.

We decided to include every generation’s best individual from each population into the History Bank, as we wanted to keep a full history of the past winning strategies used. Initially we implemented the History Bank as only containing unique entries, and later experimented with the inclusion of all individuals regardless of uniqueness.

The History Bank is a continually growing population, with an increase of two individuals every population (when uniqueness is not applied). This means that whatever selection mechanism we apply, it will always be effectively a time based probability distribution. The more generations that pass, the larger the population will become and the less probability an individual will have of being selected for game play. We have included different mechanisms to bias this selection probability, but it can be said that all are still a probability of time. The probability distributions experimented on are: random uniform selection, selection with a linear distribution and selection with a Laplace distribution probability curve.

After experimenting with the History Bank, we investigated the use of a system mimicking the way humans use memory. As discussed, humans have a current short term memory and a larger long term memory that is used to recall past knowledge on the current situation. The short term memory function was achieved by identifying the top ten individuals in the History Bank and playing an additional amount of games against them. The long term memory was then applied by playing another set of games against the entire history. So where previously the fitness was created by playing $r_1 + r_2$ games, now the games against the history is split into $r_{2S}$ for the games played against the short term memory, and $r_{2L}$ for the games against the long term memory. The History Bank individuals selected for the long term memory game play are selected with a linear time based probability, so the more recent individuals have a higher chance of being selected. The short term memory is also selected from with the same linear time based probability, but as the size is a static 10
individuals, as the entire population grows, the probability becomes uniform.

The next selection experiment performed was to further mimic the way the long term memory worked in the human brain. Instead of every single bit of information ever learnt being stored in long term memory, the human brain would sift the information and only store relevant information. We attempted to mimic this by using a ranked long term history. When adding a new individual to the History Bank, the individual would play the top individual from the ranking and if it won it would be ranked above, otherwise it would continue down the ranking until it found its appropriate rank. This was only performed for the top 1000 ranked individuals. If the individual was not in this top 1000, it would not be used in the long term memory. We nicknamed the 1000 top ranked individuals the “gladiators”.

This mechanism provided a very loose ranking, and it was likely that the ranking would be incorrect in some cases. To address this issue, we included a mechanism where at each generation a random individual would be chosen from the gladiators and played against the surrounding 20 neighbours, and adjusted in ranking where required.

The fitness calculation process was then modified as follows. After playing $r_1$ games against the opposition, the individual would be required to play against the short term memory as before ($r_{2S}$). They would then be required to play an additional number of games against the gladiators ($r_{2G}$).

The last selection mechanisms applied was to attempt to mimic the triggering mechanisms of the human long term memory. When a human recalls an event from long term memory, it is normally by relevance to the current situation. This implies that there is some mechanism the brain uses to trigger the appropriate memories. Mimicking this technique is difficult in our system, as the only thing that changes according to the ‘current situation’ is the environment of each game. The way the fitness is calculated using game play, there is no way to evaluate against the current situation of the game. To create a situation where the environmental changes could be used to trigger a relevant rule from the memory, we investigated using different players during the game play itself whose rules were more pertinent to the current situation in the game. Each game would have a number of opposition players used, and would be evaluated on the total outcome. This would effectively create a dynamic enemy that would change as the environment of the game changes. This was an entirely new way of coevolving the Tempo players.

As discussed in [23] it is common in coevolution to see groups or clusters of individuals form that are focused on solving a sub-problem of the end goal. This is particularly useful in our case as the different parts of the game can
be broken up into the different sub areas of strategy that deal with changes in; 
pwar, budget, offensive, defensive and intelligence. By including a clustered
fitness evaluation with the dynamic enemy, we hoped to further encourage this
clustering of sub-solutions with the goal of one day including a mechanism to
form a hierarchy of sub-problems that come together to form a whole solution.
The idea of clustering the populations of a coevolutionary system into the dif-
ferent similar solutions was also investigated in [24], where similar strategies
in the populations were 'packaged' and evolution done on a package level as
well as an individual level. This allowed the fitness to be calculated at both an
individual and package level, and might be something worth investigating at a
later date with our mechanisms.

The design of the triggering mechanism was based on the clustering of
relevant information. This meant that if a trigger for a particular situation
came up, the cluster relating to that situation could be called upon and an
individual selected with relevant rules. The problem with this mechanism was
the complexity involved with the amount of clusters needed to represent each
change in the environmental situation. To minimize this complexity, it was
decided to focus on the percentage of war change (pwar) scenario for this
experiment, and the memory was clustered on this single changing factor. The
pwar is used to determine when war will break out, as at the end of each year
a random number is generated, and if the number is smaller than the
pwar, the
game is over and the scores are calculated. As a result, this is one of the larger
influences on strategy used, as it is common to play differently depending on
whether the chance of war is low or high.

To cluster the individuals in the history five vectors are used, one for each
of the membership functions for pwar (very low, low, medium, high and very
high). Each time an individual is added to the memory, if it has one or more
rules using the pwar input, it is classified under the relevant vector(s). The
coevolutionary game play was then modified to make use of these vectors by
changing the opposition as the pwar increased in the current game. This was
done by checking what membership functions would be triggered by the pwar
input, and choosing a relevant vector cluster to pull an opposition from. There
were a number of issues involved in this, such as what would happen for mul-
tiple triggers (i.e. the pwar is very low to low), what if no rules are available
for triggering, and what rule to select from the vector if triggered.

To solve the multiple vectors triggered scenario a very simple mechanism
was implemented that involved selecting an individual from the first vector
triggered, then giving a chance (we used 50%) that the next triggered vector
would be used instead. To improve on this, the next step was to make the
chance dependent on the membership ratio of the pwar input. If no individuals could be found from the clusters for the current pwar, then the individual would be selected based on a linear time bias from the entire long term memory (LTM). Initially the individual selected from the cluster was chosen randomly, and then a time based linear probability distribution was applied. The time based cluster distribution provided a slight improvement to the results, and was used for further investigation.

We were expecting average results from the clustered system for our experiments, as the use of the pwar input to cluster the memory could potentially hamper the memory functionality. Focusing only on pwar means a lot of other possibly better solutions will be ignored if they do not have the pwar input. This could then create a scenario where the populations are evolving against weaker individuals. After running the original experiments however, we realized that the clusters were performing poorly against the static expert. We then implemented a very simple ‘expert’ that made decisions based on pwar like the clusters did. This expert’s strategy was to buy up on offensive weaponry while the chance of war was still low, then start to build up defensive weapons when the chance of getting attacked was getting higher.

3.4 Migration

From the results of the short and long term memory and gladiator experiments, a distinct pattern of one population dominating over the other emerged. This domination was obvious (especially in the case of the short and long term memory) and happened almost every time the system was run. The interesting thing was that the dominating population very quickly achieved and maintained a very high win ratio against the expert, while the opposition player stayed at an average win ratio. We theorized that the games against the history were effectively rewarding one population over the other by adding extra play against the best individual from the ‘better’ population. This meant that the first population to find a strategy that was good against its own solutions as well as the opposition’s would then be able to continuously dominate against the opposition. One question we had however was why the opposition was not able to catch up. Our ideal scenario involved both populations being competitive with the expert strategy, so we decided to investigate the direct influence of integrating individuals from the winning population into the opposition population, and vice versa. To do this we incorporated a method of population migration.

For this experiment we included a migration mechanism that involved a probability of performing crossover with an opposition individual. This was
performed the same as the normal crossover (a two point crossover), except that one parent was chosen randomly from the current population, and the other parent was chosen randomly from the opposition population. The probability of crossover being performed was 30%, and if crossover was chosen there was then a 10% chance that migration crossover was applied. The experiment was then run on the gladiator system.

4 Experiments and Results

This section gives a record of the experiments performed to investigate the different memory mechanisms we used. All the graphs of results in this section depict how well the best individual from each generation performed against the static expert player, which was the baseline measurement. The graphs show the won ratio against the expert on the y axis, and the generations on the x axis. The results depicted show the average of ten runs, all with the same environmental and evolutionary configuration. The system was run each time for 50,000 generations with a population size of 100 for both populations.

Each experiment had the following evolutionary parameters. The ratio of individuals to be replaced in each generation was 90%. The variation operators applied were mutation and crossover. Mutation was applied with a 70% probability of occurrence, where if applied each gene had 50% chance of mutation, with a 10% chance of a large mutation, or otherwise a small mutation occurred. The crossover applied was two point crossover with a 30% probability of occurrence. For further details please refer to [12].

Following the discussion of the experiments and their results is a table summarizing all the experiments and the different mechanisms used in each one.

4.1 Baseline Experiment

As a baseline measurement, we ran the system without any of the memory or seeding mechanisms with the fitness calculated solely from game play against the opposition ($r_1 = 20$). The results can be seen in Figure 2. The results show that the players were not doing overly well against the expert, they were achieving an average of 36% win ratio. The results also showed that there were no trends to beating the expert, just occasional jumps in performance followed by decreases (this is not visible in the results, as the results show the average of 10 runs). This was to be expected, as the coevolutionary process was not training the individuals against the expert, only against the
other population.

![Graphs showing success ratio against the expert for the Original System](image)

**Figure 2.** Success ratio against the expert for the Original System

To try and encourage the system to include the expert in the process, we included the ‘alien expert’ discussed previously. The expert was inserted as one of the individuals for each population A and B. The results from the insertion looked much the same as the original results. After the addition of the alien expert, there was a very brief increase in the won ratio followed by a sharp drop back to the previous average won ratio. Once again, this was mostly expected, as the nature of coevolutionary systems is to focus on beating the opposition in its current form, and promptly forgets any previous opponents once they have been beaten. The system was only concentrating on beating the opposition, and so the next experiment aimed to include the expert and other past strategies in the whole evolutionary process.

### 4.2 Memory Experiment

For the next experiment, we decided to implement a mechanism to allow the individuals to remember past winning opponents, by including another population of historical individuals. The logic behind this, was that to begin with the expert is the best individual, so it will be placed in the history. When the history is then used in the evolutionary process, there is a chance that there will be repeated promptings to continue to beat the expert.

The initial memory experiments consisted of randomly selecting individuals from the History Bank, and playing a set number of games against them. The fitness was calculated by playing a number of games against the opposition ($r_1 = 20$) and re-adjusting after each time. To include this strategy of evolving against the History Bank, we then provided a mechanism for playing an additional number of games ($r_2 = 20$) against the History Bank, and ad-
justed the fitness of the current individuals in the same manner as that for the opposition games.

The graph of the won ratio against the expert can be seen in Figure 3. The use of a History Bank gave us the better results we were looking for. We were now achieving an average of around 50% won ratio, and were achieving an over 90% top win ratio against the expert for single runs as well as longer trends of winning.

![Figure 3. Success ratio against the expert using random selection from History Bank for population A](image)

These results were promising, but we noted that the populations were still not tending overall towards beating the static expert and were instead regularly staying around the average to below average won ratio. The populations were still ‘forgetting’ previous solutions and focusing more on beating the current opposition. We reasoned that the cause of this was the random nature of the selection from the History Bank. By selecting at random, we were increasing the potential for larger amounts of the earlier, and presumably simpler, individuals from the history to be chosen. What we actually wanted, was for the populations to mainly grow against the harder (more recent) individuals, and then be occasionally prompted not to forget older strategies and so address the Red Queen effect. To create this scenario, we decided to include a probability distribution for game play selection from the History Bank.

### 4.3 Linear Selection

The initial probability distribution was a simple linear time distribution, where the latest historical individuals had a higher probability of getting chosen then the older individuals (see Figure 4).

This distribution would force a higher playing ratio against the more recent individuals, but would still encourage play against the older strategies. The
results of this experiment can be seen in Figure 5, and showed a slight increase in the win ratio average against the expert, but was still not to a satisfactory standard.

The results from a typical single run of the linear probability distribution can be seen in Figure 6. This shows the fluctuation that is occurred in a single run, and how improvement could still be made to implement the growth of wins against the expert.

4.4 Laplace Selection

The next question following on from the previous results, was whether the linear probability distribution was the best distribution, and if perhaps a curved distribution might provide better results. To test this theory, we replaced the linear distribution with a Laplace (otherwise known as Double Exponential) probability distribution function. We were expecting that by introducing a
curved distribution function with a sharp regression, we would force the populations to play more against the most recent players in the history. The results of this experiment can be seen in Figure 7.

![Figure 6](image1.png) ![Figure 7](image2.png)

**Figure 6.** Success ratio against the expert for a single run using Linear Distribution curved distribution function with a sharp regression, we would force the populations to play more against the most recent players in the history. The results of this experiment can be seen in Figure 7.

**Figure 7.** Success ratio against the expert using Laplace Probability Distribution

The individual results showed a slight improvement with more trends towards winning against the expert. However, there was still a lot of fluctuation and the results were still lacking. What we really wanted was to see an overall increase over time in the statistics against the expert.

### 4.5 Short and Long Term Memory Selection

The inclusion of a history to evolve against had produced some nice peaks, however even with the probability selection distribution it still seemed to average around a 60% win ratio with a lot of variation. The time function probability selection we were utilizing allowed a biased selection mechanism, but it was still allowing a higher play rate against the older solutions than we wanted.
Even though the time-based probability selection was biased towards the top of the scale, as the History Bank grew, the chances of playing against the top end decreased. To address this we decided to introduce a specified top end window to the probability selection - the short term memory window. This window would be evolved against a set number of times prior to evolving against the whole history - the long term memory.

The experiments included a window of the top ten individuals and played against individuals from the short term memory \( r_{2S} = 10 \) followed by games against the long term memory - the entire History Bank \( r_{2L} = 10 \). The games played against the opposition remained the same \( r_1 = 20 \). The results were very promising, with a slightly higher ratio for both populations against the expert as depicted in Figure 8. More importantly, we were now seeing trends towards beating the expert. This can be seen in Figure 9, which shows the results for a typical single run of the system. The single runs for the system showed that in most cases, one of the populations was outperforming the other population, and so the overall results were therefore improved as well.

![Figure 8. Success ratio against the expert using Short and Long Term Memory](image)

4.6 Unique vs. Not Unique Memory Structure

One of the decisions made early on in the process was to make the History Bank unique and then increase their probability of getting chosen if the same individual was repeatedly added to the population. We decided to test this decision and performed a run with no uniqueness checking, and instead all the best individuals for all the generations were automatically added to the history.

Although there was not much of a difference, this was actually the first time the populations had achieved an outright 100% win ratio against the ‘expert’. We were not expecting this, as we thought the unique population would have
fared better due to the forced diversity of the populations. It occurred to us that this higher achievement might actually be caused due to the short term memory being filled with the best strategy, and therefore forcing the populations to find a solution to that single strategy.

4.7 Ranked "Gladiator" Selection

The gladiator long term memory system was used in this final experiment, and followed along the same lines as the previous history experiments. The fitness was calculated by playing first against the opposition ($r_1 = 20$), then against the short term memory ($r_{2S} = 10$) and finally against the gladiator long term memory ($r_{2G} = 10$). The results can be seen in Figure 10.
found in section 4.6, but after fixing the ranking of some precision errors we found that the results were drastically improved. The single results for the gladiator system showed that in most cases both of the populations were averaging an above 60% win ratio. However, we were still witnessing the stabilized advantage of one population over the other as mentioned in the previous experiment.

4.8 Migration

To determine if the migrating the individuals from the dominating population to the weaker population could be used to increase the overall fitness of the weaker population, we incorporated a form of migration. We ran the migration by having a percentage chance of crossover being with a parent from the other population. The experiment used the same gladiator system with the same parameters as previous experiments, but with a 10% chance of migration crossover instead of normal crossover. The results from these experiments can be seen in Figure 11.

![Figure 11. Success ratio against the expert with migration incorporated into the gladiator system](image)

These results show a very big change from the gladiator experiments without migration. The results are clearer when the single runs are viewed, and Figure 12 shows some of the single run results for this experiment. Even with the minimal chance of crossover being applied, the results showed that the populations quickly started mirroring each other. The results also seemed to disprove our theory of the memory having the biggest influence in the creation of a dominating population, as by including a small probability of crossover with the opposition, the dominance disappeared and some runs included both populations with an average win ratio well below 50%. These results were very surprising, and further research into why this has occurred is warranted.
4.9 Using the expert

For the previous experiments, we were incorporating a seeded expert into the system to see how it fared against the evolutionary process, and to measure the memory’s ability to keep strategies against the expert alive. The next experiment was designed to test the actual usefulness of the alien expert in keeping good strategies alive. We started by performing an experiment with no Alien Expert inserted. This used the long and short term memory mechanisms with a linear time based selection probability from the long term memory. The fitness was calculated by playing $r_1 = 20$ games against the opposition, followed by $r_s = 10$ games against the short term memory and $r_l = 10$ games against the long term memory. The graph of the won ratio against the expert can be seen in Figure 13.

When compared with the results from section 4.5 (the experiment with the same parameters as this one, but with the seeded alien expert), the results showed that the Alien Expert stabilized the evolution with less fluctuations in the results. However, the fluctuations of the experiment without any expert
produced some better results in some of the peaks of the evolutionary process. Also of interest, was the speed at which the coevolution was able to beat the expert on a regularly. We were expecting the experiment without an Alien Expert to show a slow increase against the expert, and a lot of variation with worse performance overall than the Alien Expert experiment. We were subsequently expecting the seeded Alien Expert experiment to show an immediate increase in the ratio against the Static Expert, and then a stabilization or a slow decrease as the Alien Expert gets diluted in the memory. What we found instead was that the memory by itself quickly improved to beat the expert, even without any mechanism to know of the expert’s existence.

This result led us to deliberate whether the insertion of the Alien Expert into the populations may actually hinder the evolutionary process. As noted by Rosin and Belew [25], having an initial opposition population that is hard to beat can stunt the evolution, as the individuals will not fare well and there will be less variation to guide the search. By inserting the Alien Expert into the population we are essentially adding an individual that will initially beat almost all the randomly generated individuals in the populations that it plays against (from the opposition population or from the history). Without the expert, the populations are free to search and improve without having their fitness decreased by being evaluated against the expert. This means that they are able to explore a more diverse range of solutions and create possibly better solutions against the expert that are not specifically tailored to beat the expert.

4.10 Clustering the memory

For this experiment we applied the clustering mechanism to the long term memory. We performed an experiment with the same environmental and evolutionary parameters as before, with $r_1 = 20$ games against the opposition,
followed by $r_s = 10$ games against the short term memory, and $r_{lc} = 10$ games against the clustered long term memory.

When we ran the clustered system using the same strategy as the previous experiments, the results were lacking when compared to the normal short and long term memory mechanism. We thought that this may be caused by the fact that the current expert had no reliance on $p_{war}$, and our clustered mechanism was tailored to adapt to $p_{war}$ variation. To test this we created a new expert to test against as described in section 3.2. We then ran the clustered mechanism again, this time replacing the static benchmarking expert and the Alien Expert with the $P_{war}$ Expert. The clustered mechanism performed better against this expert, and the results from this can be seen in Figure 14.

![Figure 14. Success ratio against the expert with clustered long term history](image)

The clustered mechanism performed very differently to the normal long term memory selection mechanism, with very large variations and a slower climb towards regularly beating the Static Expert. This slow trend to improve was accentuated by the large variations, and was something we had not seen in prior experiments. The slow trend was something that was partly expected, as the clustering mechanism may overlook better opponents from memory to evolve against if they do not include a rule for $p_{war}$. Taking this into account, it was interesting to see the increase towards better solutions towards the end of the generations. This is rather promising, and our next step in this direction will be to incorporate the other dynamic inputs into clusters, such as the budget and the weapon types and categories. The variation itself was not expected to the degree seen, and reasons for this fluctuation are still to be investigated, but it is possible that the clustered memory contains largely very weak or very strong (due to only applying them by their $p_{war}$ rules) individuals against the expert, and goes through phases of applying these.

25
4.11 Experiment Parameter Comparisons

A summary of the different experiments and the difference in parameters can be found in Table 1. The information depicted is as follows. The type of History Bank mechanism used is single for the initial case, then split into short and long, and lastly gladiator. The Probability Selection Distribution depicts the type of distribution used to select an individual for game play from the History Bank. The Game number shows the number of games each individual from each population must play to calculate the fitness: \( r_1 \) is the games played against the opposition, \( r_2 \) is games against the History Bank, \( r_{2S} \) is games against the short term memory, \( r_{2L} \) is games against the long term memory and \( r_{2G} \) is games against the gladiator long term memory. The History Bank lengths show what size each of the history bank mechanisms are, where \( L \) is the single History Bank length, \( L_s \) is the short term memory length, \( L_l \) is the long term memory length and \( L_g \) is the gladiator memory length. The unique field is a Boolean field representing whether the History Bank applied uniqueness of individuals or not. Finally, the Alien Expert field depicts if the populations were seeded with the Alien Expert for the experiment.

Table 1. Summary of the experiments

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline(4.1)</td>
<td>∅</td>
<td>N.A.</td>
<td>( r_1 = 20 )</td>
<td>N.A.</td>
<td>N.A.</td>
<td>no</td>
</tr>
<tr>
<td>Memory(4.2)</td>
<td>Single</td>
<td>Uniform</td>
<td>( r_1 = 20, r_2 = 20 )</td>
<td>( L = \infty )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Linear(4.3)</td>
<td>Single</td>
<td>Linear</td>
<td>( r_1 = 20, r_2 = 20 )</td>
<td>( L = \infty )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Laplace(4.4)</td>
<td>Single</td>
<td>Laplace</td>
<td>( r_1 = 20, r_2 = 20 )</td>
<td>( L = \infty )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Short &amp; Long Term(4.5)</td>
<td>Short, Long</td>
<td>Linear</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2L} = 10 )</td>
<td>( L_s = 10, L_l = \infty )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Unique(4.6)</td>
<td>Short, Long</td>
<td>Linear</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2L} = 10 )</td>
<td>( L_s = 10, L_l = \infty )</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>Gladiator(4.7)</td>
<td>Short, Gladiator</td>
<td>Ranked</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2G} = 10 )</td>
<td>( L_s = 10, L_g = 1000 )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Migration(4.8)</td>
<td>Short, Gladiator</td>
<td>Ranked</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2G} = 10 )</td>
<td>( L_s = 10, L_g = 1000 )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Expert Test(4.9)</td>
<td>Short, Long</td>
<td>Linear</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2L} = 10 )</td>
<td>( L_s = 10, L_l = \infty )</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Clustered(4.10)</td>
<td>Short, Clustered</td>
<td>Linear</td>
<td>( r_1 = 20, r_{2S} = 10, r_{2L} = 10 )</td>
<td>( L_s = 10, L_l = \infty )</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>
5 Conclusions and further work

The experimentation of memory in a coevolutionary system has shown that the short and long term memory approach was beneficial, particularly when applying a structured long term memory. We demonstrated that the sorting and selection mechanism for the history population affects the usefulness of the memory, and that by including a specialized memory selection mechanism represented by the short term memory, the system could make better use of its memory.

The short term memory mechanism works like the human memory mechanism for learning; it allows a person to overcome information overload and recall the most useful and recent memory rather than their large long term knowledge base. The long term memory is still there for prompting historical memories and reminding the person of past successes and mistakes, and allows the person to grow overall. We have attempted to replicate this in the Tempo system, allowing the system to have reminders of past strategies, while still giving the best information for the current situation.

The inclusion of more than one selection mechanism from a history based on long and short term memory has many different areas that could still be investigated. There are areas involving different ways to rank the individuals in the gladiator system, which is currently a very simple stochastic mechanism. There are many ways that the evolutionary parameters could be experimented with to determine what the optimum values for the games played against each memory should be. The clustered mechanism used to try and find a way to mimic the way the human long term memory recalls information also needs to be investigated further, with more information in the clusters leading to possibly better results. Another possible mechanism could follow the case-based memory technique used by Louis and Miles [11]. Lastly, we shall investigate possibilities to incorporate some generic semantics into the system to represent ‘social’ learning for the whole population.

One area we have begun to investigate is to have an adaptive number of evaluation games dependent on the environment and the current stage of evolution. At the moment there are user defined parameters representing each set of games to be played to get the fitness, including games against the opposition and the history. It might be that a better alternative would be to let the system itself decide what number of games to play. To this end, investigation has begun into the use of a fitness threshold, where the number of games played adjusts itself as the size of the fitness variation increases or diminishes.

The use of memory with the Tempo system has given rise to some really challenging computer players, however the individuals are still a static rule
base that human players can adapt to beat over time. The long term goal of our research is to create a system where the individuals are evolving and adapting to beat a particular human player during real-time game play. The use of memory was the first step in this research, and our next step is to find a way to adapt to human game play. We are currently investigating ways of extracting rules representing a human’s strategy from the output of their game play. These rules will then be added to a population of human rules that the Tempo system can evolve against, then each game a new individual can be chosen to play against, which has the advantage of being evolved against the human’s logic. This will allow us to develop a system that grows and improves along with the human they are competing against, a tailored system that provides the best challenge and training for an individual human.

6 Acknowledgments

The authors would like to thank the SAPAC group for their continued support in this research.

References


