'Psychoanalysis' of a Minimal Agent

Santosh Manicka and Inman Harvey

Center for Computational Neurosciences and Robotics,
School of Cognitive and Computing Sciences
University of Sussex, Brighton BN1 9QH, United Kingdom
{S.Manicka, inmanh}@sussex.ac.uk

Abstract

The Secretary problem is studied with minimal cognitive agents, being a problem that needs memory and judgment. A sequence of values, drawn from an unknown range, is presented; the agent has only one chance to pick a single value as they are presented, and should try to maximize the value chosen. In extension of previous work (Tuci et al. 2002), Continuous Time Recurrent Neural Networks (CTRNN) are evolved to solve the problem, and then their strategies are analyzed by relating mechanisms to behavior. Strategies similar to the known optimal strategy are observed, and it is noted that significantly different strategies can be generated by very different mechanisms that perform equally well.

Introduction

This study is in the tradition of using Evolutionary Robotics techniques (Cliff et al. 1993; Harvey et al. 2005) to evolve artificial minimal agents with a genetically specified 'nervous system' so as to perform tasks of interest (Beer, 1996; Beer, 2003; Goldenberg et al. 2004). The interest of the Secretary Problem (described in the next section) is that it requires memory and judgment, and the provably optimal strategy requires a strategy of some sophistication.

Tuci et al (2002) evolved CTRNNs with 4 nodes to perform well at this task, and performed a preliminary analysis of the strategies seen. Here we largely replicate their methodology, and go on to look at their activation level patterns from their performance in various scenarios and then interpret them in terms of their behavior with the objective of uncovering any underlying strategy. Whereas Tuci et al had done an overall performance analysis, our strategy was to observe the behavior of the evolved mechanisms in the smallest units of the problem and do it over a strategically chosen range of problems so that we would be able to sensibly describe the observed behavior as a strategy. To put it simply, we analyze the strategies by relating the neural mechanisms to the behavior, in what could be metaphorically called a form of 'psychoanalysis'. The significant results are:

1. A network is found to have evolved a strategy similar to the actual optimal strategy.
2. Two networks with nearly equal fitness values are found to have evolved significantly different strategies.

The Secretary problem can be considered as one in a larger class of problems in probabilistic decision making using a single criterion (maximize rank). While it has been shown, through this work, that evolution 'thinks' like a mathematician in a simple problem of a larger class of decision making problems, it could be interesting to investigate its influence in more complicated problems. One such interesting problem could be the game of poker. CTRNNs could be evolved and their strategies be compared with the game theoretic strategies of poker in search of interesting implications from a cognitive point of view. An even more complicated application could be problems involving multiple criteria like the 'Experts case' problem (Czogala and Roubens, 1989). If successful CTRNNs could be evolved in such problems, a behavioral analysis of their strategies as adopted in this work might help reveal interesting cognitive insights as this approach is fundamentally different from the usual analytical approaches.

Background

The Secretary problem is a problem of choice from among a temporal sequence of random possibilities so that the expected payoff from the choice is maximized or the expected cost of the choice is minimized. A very simple form of the secretary problem version has been described by Ferguson (1989) as follows:

1. There is one secretarial position to be filled
2. The total number of applicants is known
3. The applicants are interviewed sequentially in random order. An order has the same chance of occurrence as any other order.
4. An applicant should either be accepted or rejected at the end of the interview of the applicant and the decision should be made solely on the relative rank of the applicant
5. An applicant once rejected cannot later be accepted
6. The interviewer will not be satisfied unless the chosen applicant is the best in the group (i.e., the payoff is either 1 or 0)
7. If no applicant is accepted before the last applicant, the last applicant should be accepted.

The solution to the problem is quite simple: for a specific integer $r \geq 1$, reject the first r-1 applicants and choose the next applicant who is the best among all the applicants seen until then. Mathematically stated, the probability of choosing the best applicant is $1/n$ if r=1; if r>1 then (Ferguson, 1989)

$$\Phi_n(r) = \sum_{j=r}^{n} P \quad (j^{\text{th}} \text{ applicant is best and you select it})$$

$$= \sum_{j=r}^{n} \left(\frac{1}{n}\right)\left(\frac{r-1}{j-1}\right) = \left(\frac{r-1}{n}\right)\sum_{j=r}^{n} \frac{1}{j-1}$$

For a very large value of n, the value of r is calculated as $1/e$. This translates as "Reject the first ~37% of the interviewees and then pick the first best". Implementation of this analytic solution is quite straight forward; it hardly takes a few lines of code in a computer program. Given the proof arrived at by mathematicians, an interesting question, from an evolutionary point of view, would be to ask "How would evolution shape the cognition of an agent powered by a dynamical system to solve this problem?" This would be interesting in the sense that the agent here is not expected to 'know' mathematics. An experiment performed with such an agent could lead to insights into the mechanisms of cognitive behavior. Such experiments have been conducted in the past by evolving continuous time recurrent neural networks (CTRNNs) (Beer, 1996). Tuci et al (2002) have successfully evolved a CTRNN that could solve the Secretary problem (that maximizes the expected payoff rather than look for the single best item). In this paper we go further to analyze in depth successful CTRNNs.

Methods

As the emphasis of this work is analysis, we used a proven method of evolution of a CTRNN to solve the Secretary problem; we used a very similar approach as used by Tuci et al (2002). In the experimental set up by Tuci et al, the logical sequence of values (worthiness of the interviewee) is presented in the form of a temporal sequence of inputs to the CTRNN; for each one the binary input is switched to value 1 for a length of time proportional to the current value (referred to as 'exposure time'), and then cleared to zero. A sequence (also referred to as a 'trial') contains 20 unique items (integers), from which the network is expected to choose one. The single thresholded output is then tested for a binary accept/reject decision, before moving on to the next value (item) in the sequence. Overall, the network is expected to maximize its payoff by choosing an item with a relative rank in the sequence as large as possible. The presentation of a trial is terminated as soon as the network accepts an item or after all the 20 items are presented (in which case the last item is considered accepted). Between consecutive presentations of items, the network is cleared by setting the input to zero for 2 (simulated) seconds. Before the start of each trial, the network is reset by setting the output values of the nodes to zero. The input is always fed to node 1 and is time-based (Tuci et al. 2002) i.e., the external input will remain at the value '1' for a particular number of iterations (value of an item in the trial/time-step size). In our experiment, the time-step size value was 0.2. So, if an item's value is 27, the external input will be '1' for 135 iterations of network-update.

The network is run through a set of 60 trials, each of length 20 during its evolution. Each trial is defined as follows (Tuci et al. 2002)

$$c = 1,\ldots..,60 \qquad \begin{Bmatrix} l_c = c; \\ h_c = l_c + 29 \end{Bmatrix}$$

Where, l_c is the lowest possible value of an item of trial c and h_c is the highest such possible value.
Each neuron of the 4-neuron CTRNN we evolved uses the following state equation (Tuci et al. 2002):

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^{k} w_{ji} z_j + g I_i$$

with

$$z_j = \frac{1}{1+\exp\left[-\left(y_j + \beta_j\right)\right]}, i = 1,\ldots,4$$

y_i = cell potential
τ_i = decay constant
w_{ji} = strength of connection from neuron j to i
z_j = firing rate of neuron j
I_j = external input to neuron i
g = sensory gain factor
β_j = bias

When the activation of the output neuron exceeds 0.5 at the end of presentation of an item, the item is considered chosen.
The initial strength of the population, the fitness function and the evolutionary parameters that we used were the same as what were used in (Tuci et al. 2002) except for a few changes (as we could not replicate the experiment with the original parameters): the decay constants were mapped to $[10^0, 10^{1.8}]$ instead of $[10^0, 10^{2.8}]$. We used mutation probability 0.2 instead of 0.3 with explicit elitism. The cut-off values for the elitism varied between top 5% and 8%. The number of generations was varied between 5000 and 25000 in the runs.
We evolved 2 fairly-well performing CTRNNs using the above mentioned parameters from 7 evolutionary runs. We consider a network to perform fairly-well when its fitness value is comparable to that of the best network evolved by Tuci et al i.e., a fitness value of 0.85. Henceforth, we will refer to these networks as N1 and N2. Below, we describe their morphologies and their various performance measures.

Results of evolution

Morphology and performance of network N1

We have presented the morphologies of the evolved networks here so that any experiment with these networks can be

replicated with ease without having to resort to re-evolution. Otherwise, we have not explored any direct influence of the network parameters on the results of our analyses.
The mean maximized rank choice (on a scale of 0 to 1) of N1 is: 0.79

Morphology:

Weight matrix (connection from node *j* to node *i*)

	i=1	i=2	i=3	i=4
j=1	-2.853552	-2.087797	0.612068	2.975119
j=2	-0.585138	3.029250	-1.806782	0.330331
j=3	-3.342187	1.828053	1.184740	-4.909308
j=4	0.125955	0.641515	-1.510127	-1.685360

Table 1a. Evolved Weight matrix for N1

Other parameters of the i^{th} neuron

i	1	2	3	4
τ_i	3.563501	46.563980	0.327475	39.911218
βi	0.898813	-0.569968	-0.122038	-0.734500

Table 1b. Other evolved parameters for N1

Gain = 3.352800

The percentage of actively expressed preference (choice of any item other than the last item) and the average acceptance position per trial are plotted below in figures 1a and 2a after performing 100 simulations of 60 trials each.

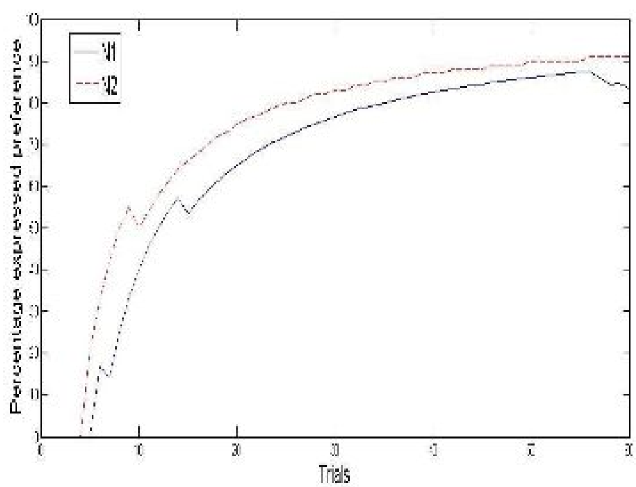

Fig 1a. Percentage of expressed preference per trial

The value 'percentage of expressed preference' at trial *i* denotes the percentage of the number of active choices made by the network until that trial. Acceptance position is the position in the sequence of a trial where an item is accepted (actively or passively) by the network.

Morphology and performance of network N2

The mean maximized rank choice (on a scale of 0 to 1) of N2 is: 0.75

Morphology:

Weight matrix (connection from node *j* to node *i*)

	i=1	i=2	i=3	i=4
j=1	-0.744026	2.395746	-0.296505	2.120096
j=2	0.818743	-4.829388	-3.410042	-3.145099
j=3	-0.344417	1.448848	4.833700	4.198339
j=4	-4.913282	1.830852	-0.096895	1.296698

Table 2a. Evolved Weight matrix for N2

Other parameters of the i^{th} neuron

i	1	2	3	4
τ_i	3.939339	358.841803	17.613176	22.553297
βi	-1.336118	0.079064	-1.928700	-1.408360

Table 2b. Other evolved parameters for N2

Gain = 4.167492

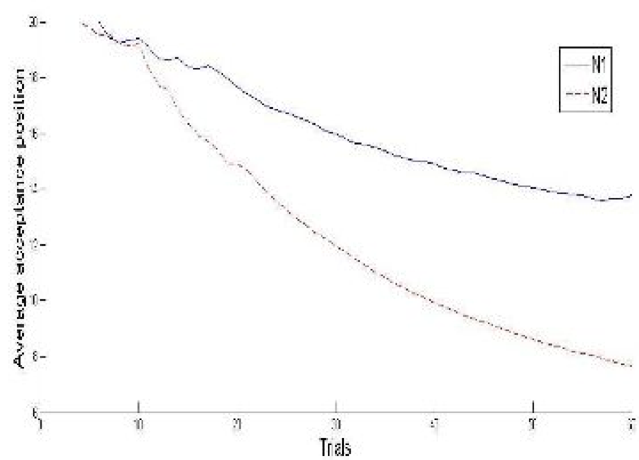

Fig 2a. Average acceptance position per trial

Figures 3a and 3b depict the performance in terms of average relative rank of the item chosen by the networks N1 and N2 in each of the 60 trials averaged over 100 simulations.
The horizontal dotted lines indicate the overall mean performance of the network.

Analysis

It can be seen from figures 3a and 3b that both N1 and N2 perform relatively worse towards the ends of the trial spectrum. We will now look at the activation level patterns of

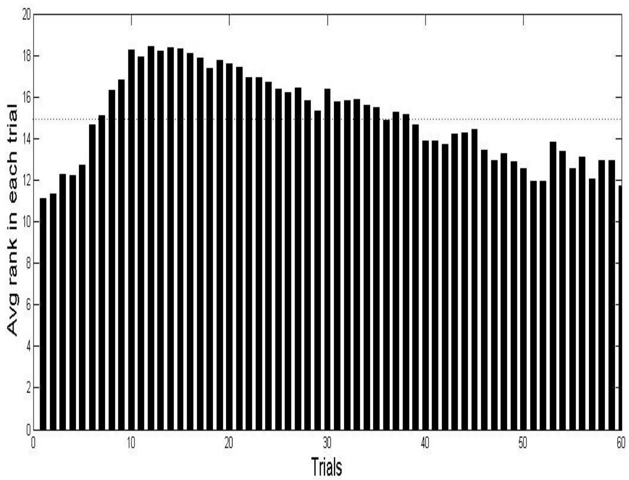

Fig 3a. Average rank choice per trial in N1

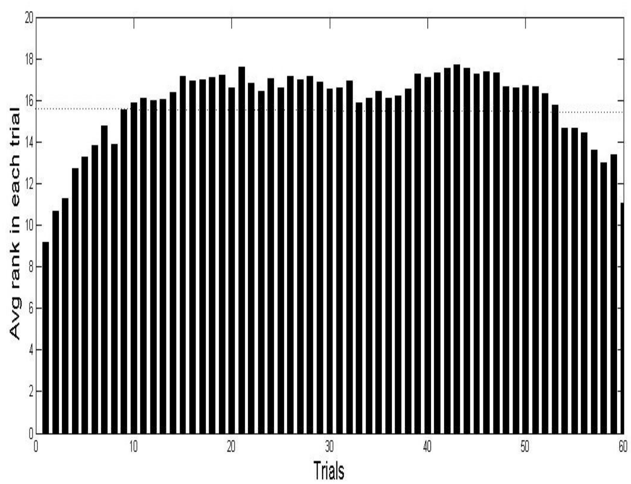

Fig 3b. Average rank choice per trial in N2

each network in specific problem scenarios in an attempt to look at the networks' behavior in detail. A test trial is presented from each of the following categories:

1. A randomly generated trial from the lower trials (c is between 1 and 5) where the network performs the worst
2. A randomly generated trial from the intermediate trials where the network performs the best
3. A randomly generated trial from the higher trials (c is between 55 and 60) where the network performs the worst

Figures 4a through 4d depict the activation level values of the output neuron during the exposure time (see 'Methods' section for definition) for each test trial.

Behavior of N1

Test trial 1
27, 8, 29, 5, 18, 20, 15, 24, 17, 16, 9, 3, 2, 19, 22, 4, 26, 23, 21, 12
Result: No active choice. Figure 4a shows the behavior of the network towards each item in the trial.

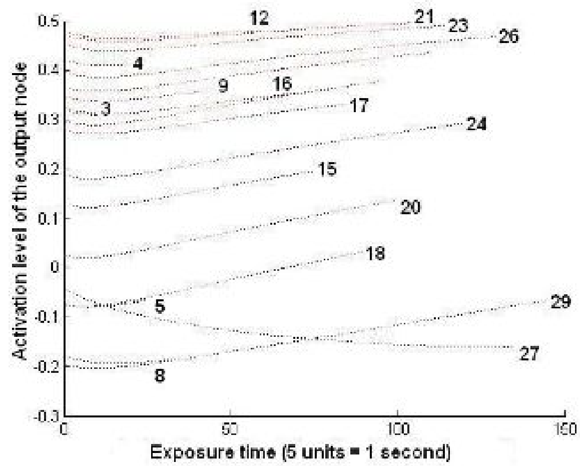

Fig 4a. Activation levels of the output node of N1 for trial 1

Test trial 2
22, 24, 31, 35, 27, 32, 33, 23, 39, 45, 36, 43, 25, 46, 40, 49, 29, 21, 26, 48
Result: Item = 46; Position = 14; Relative rank = 18. Figure 4b shows the behavior of the network towards each item in the trial.

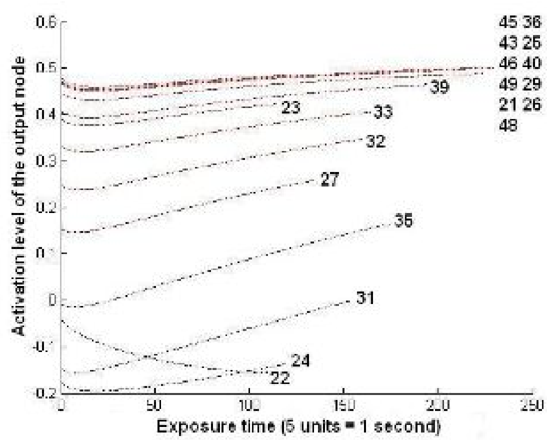

Fig 4b. Activation levels of the output node of N1 for trial 2

Test trial 3
86, 70, 73, 84, 65, 68, 60, 66, 82, 74, 83, 78, 71, 81, 61, 63, 79, 72, 76, 77
Result: Item = 82; Position = 9; Relative rank = 17. Figure 4c shows the behavior of the network towards each item in the trial.

Observations

From figures 4a, 4b and 4c, it can be seen that:

1. The network starts responding with a 'great dip' in all the trials for the first item, regardless of the absolute value of that item. The dip can continue for about 150 iterations (fig 4c) and then start rising. Its maximum destination value of activation could be about 0.05 (see fig 4c, first item = 86; max value could be 89).

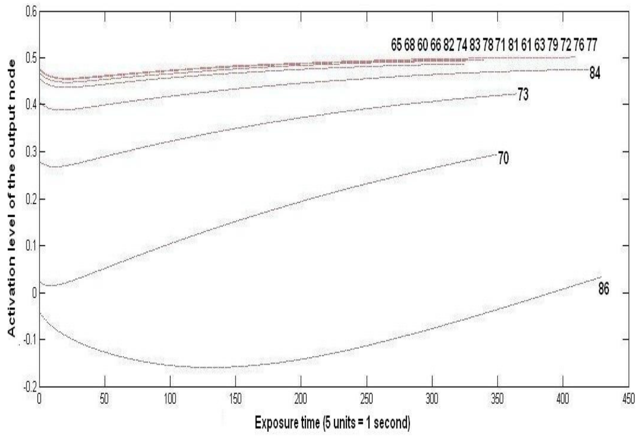

Fig 4c. Activation levels of the output node of N1 for trial 3

2. The number of items that spend their 'lives' below zero-AL ('AL' is short for Activation Level) decreases from test trial 1 to test trial 3.
3. In all trials, the vertical separation between response curves grows smaller with more items seen. They are the closest between activation levels 0.4 and 0.5.
4. In all the trials, for items following the first, there is a slight initial dip and it stays slightly longer for items at higher positions in the trial. Besides, this initial dip is longer for items in trial 3 than in the other two trials.
5. In all the trials, for each item, the network responds with a shoot-up after the initial dip. The slope of the shoot-up is almost the same for all the items in test trial 1 and also test trial 2 (except towards the last few items of the trial). However, this parallelism is less pronounced in test trial 3 where the network tends to flatten-out its response to the items it sees in the higher positions of the trial.

Interpretations and reasoning

The great dip is indicative of 'Never choose the first item, whatever it may be' as even if it is 89, it wouldn't be chosen (test trial 3). The extent of the dip will determine how fast the responses of the items can race to the 0.5-AL finishing line. This is the reason behind observation (2). As the network sees more items, it strives to settle for an item with as big a relative rank as possible by being more "cautious". This behavior can be seen in observation (4). The longer initial dip and therefore a more delayed start of the shoot-up, combined with a slightly smaller rate of shoot-up makes sure that only those items with relatively more persistence (longer response due to larger items) can move closer to the finishing line. Observation (4) is in turn the reason behind observation (3) since (4) results in a smaller difference between the initial and the final activation values for an item seen higher in the trial and therefore the following item's starting activation level is closer to that of the previous item. The reason behind observation (5) is that when it sees more values in a higher trial and still strives for rank maximization, it can't continue the trend of shoot-up (with a constant rate) as it does in a lower trial because if it does, items with relatively smaller values (like 65 or 70 in trial 3) can easily cross the 0.5-AL. Therefore, it seems that it has evolved to "stretch" its single strategy of rank maximization to the higher trials by lengthening its initial dip and slowing down its shoot-up. This also could be the reason why the last range of values (60, 89) is a bad performer (fig 3a) as the percentage of expressed preference slightly drops towards the end (fig 1a). That's because the response could flatten out so much that the network eventually refuses to actively accept any item as shown in the response levels in fig 4d below for the following trial:

63, 60, 76, 65, 71, 68, 77, 85, 75, 84, 62, 74, 86, 82, 64, 78, 88, 72, 73, 61

Result: No active choice

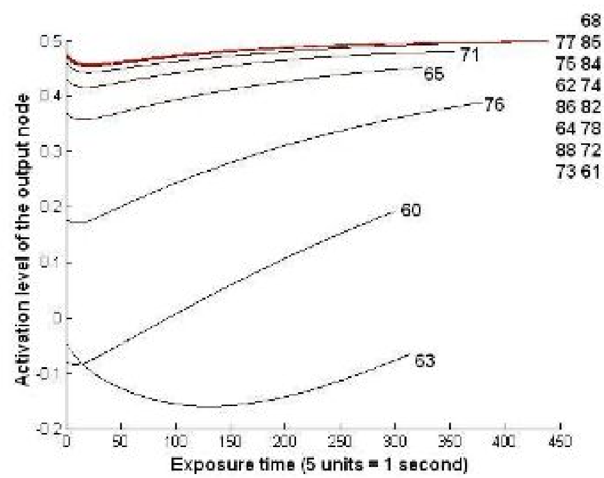

Fig 4d. Activation levels of the output node of N1 for a random trial

Strategy

Has the network evolved a strategy? We have not found a complete answer to this question, but we will present here a few directions along which the answer could be pursued.

The network appears to be following two stages of aspiration-setting: first, when it sees the first item and second, when it sees the rest as the response to the first item is significantly different from the rest. The term "aspiration" can be described as a value such that an item with a value greater than the aspiration could be considered as a candidate for selection. Here we describe a possible approach to uncover the aspiration-setting strategy of the network. See figure 5a. A sample response curve is plotted for a random item at a random position in a trial. Two types of typical responses are depicted past a random point B – a response crossing the 0.5-AL and a response flattening out just below the 0.5-AL line. When the response reaches point C (on the same level as the starting point A), the current item is roughly considered worthy of acceptance. The corresponding value 'p' on the x-axis when multiplied by 0.2 (step size) can be considered as the lower bound of the current aspiration (LBA) set by the network. The actual aspiration can be calculated when the response crosses the 0.5-AL as at point D. Then the actual aspiration is *l**0.2. Yet there is a possibility that for an item, the response totally flattens out before it could cross the finishing line (fig 4d). In that case, an approximate value of the more accurate LBA can be calculated. At point E, in fig 5a, the response starts to flatten out.

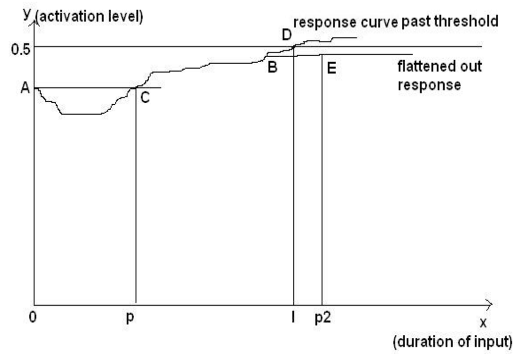

Fig 5a. Aspiration-setting in network N1

The LBA is then calculated as (*p2**0.2). If the item is not big enough (not a long enough response) to help reckon the aspiration, it can be incremented by 1 until it either reaches point D or E. This way, the aspiration at a particular position in a particular trial can be calculated. Such calculations when performed extensively over a wide range of trials and positions might help get a deeper insight into the underlying patterns of aspiration-setting. Further analysis of the great dip's impact also helped reveal a more interesting strategy. We ran about 1000 simulations of the network each containing the regular 60 trials. Fig 6a below depicts the percentage of choice made at a particular position of the trial in these simulations.

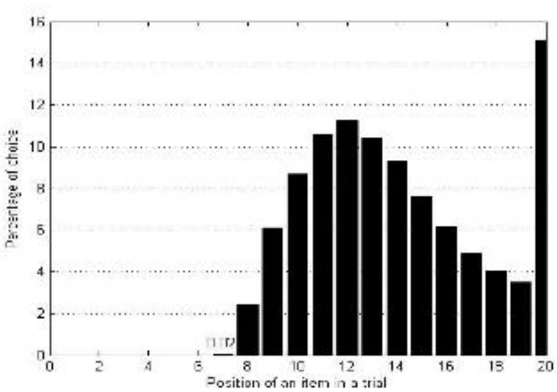

Fig 6a. Percentage of choice per position of N1

It can be seen that no choice is made at any of the first 6 positions. At the 7th position, the percentage of choice is extremely low (0.02%). Only starting from the 8th position, the network actively makes a good proportion of choices. The actual optimal strategy to the Secretary problem is to ignore (make no choice) the first 37% of the trial (Ferguson, 1989) and then to choose the best item that's better than any item seen so far. The length of a trial in our experiment is 20 and 37% of 20 is 7.4. Therefore, our network seems to have evolved a similar strategy as the actual optimal strategy at least as far as ignoring the first few items is concerned.

Comparison with optimal strategy

In this section, we compare the behavior of N1 with the analytic-optimal strategy described in the 'Background' section. As a note, the mean maximized rank choice of the optimal strategy is 0.80 as compared to 0.79 of N1. Fig 6b below depicts the trial-wise average acceptance position of N1 against that of the optimal strategy. It can be seen that the average acceptance of the optimal strategy is almost always about 14. It is because the first best item better than the best in the first 37% of the sequence (i.e., in the first 7) can appear anywhere between positions 8 and 20 with equal probability. Therefore, over a sufficiently large number of simulations (in this case, 1000) the average position will be the average of 8 and 20 which is 14. Consequently, it can be seen that the strategy of N1 is not wholly similar to the optimal strategy after ignoring the first 37% of the sequence. Still there is an overlap between the 2 plots between trials 23 and 27 and also at about 56. Could it mean that N1 behaves in the same way as the optimal strategy in these ranges of values? Further analysis provides the answer 'No'. Figure 6c depicts a trial-wise average rank choice by N1 (fig 3a repeated) against the choice by the optimal strategy. It can be seen that between trials 23 and 27, N1 fairs better than the optimal strategy. At the trials around 56, the optimal strategy performs better than N1. So their performances are different even though their average acceptance positions in these trials are the same.

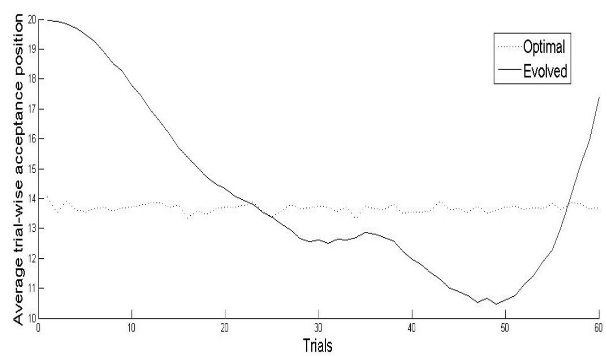

Fig 6b. Average acceptance positions – a comparison

It can also be seen from this figure (6c) and from figure 6b that even if N1's performance is the same as the optimal strategy in trials like 10 and 50, their average acceptance positions are different. The reason for the above 2 observations is that even though the average acceptance position is the same, the standard deviation between the acceptance positions of N1 and the optimal strategy is quite considerable as shown in figure 6d.

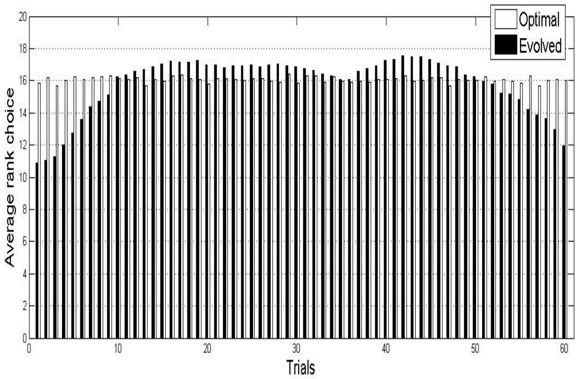

Fig 6c. Average trial wise rank choice – a comparison

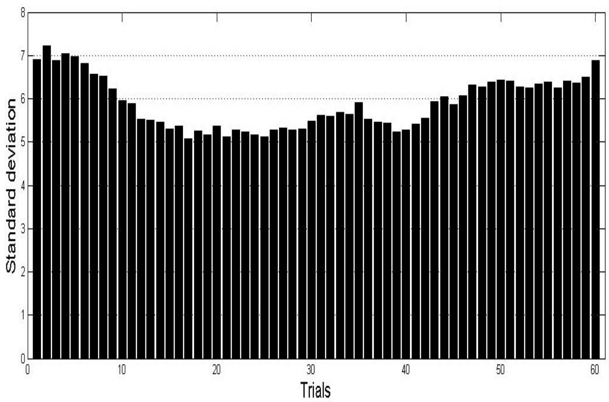

Fig 6d. Standard trial-wise deviation between average acceptance position of N1 and that of optimal strategy.

It can be seen that the minimum standard deviation is about 5. Also, in trial 42 where N1 seems to perform relatively the best (fig 6c, 3a), the standard deviation is approximately 5.5. At trial 17, where the standard deviation is the least i.e., 5 the performance is also among the highest. These observations suggest that the second part of N1's strategy is not as definitive and general as the optimal strategy (standard deviation is neither zero nor constant in the trials; see fig 6d) and yet not fully trial-dependent (there is no pattern displayed in the evolved strategy in fig 6b).

Behavior of N2

In this section, we describe the behavior of the network N2 when it is presented with the same 3 trials as N1 was presented with and compare their responses. The focus here is to point some significant differences between the networks' cognition even if there is no big difference between their overall performances.

Test Trial 1
27, 8, 29, 5, 18, 20, 15, 24, 17, 16, 9, 3, 2, 19, 22, 4, 26, 23, 21, 12
Result: No active choice. See figure 7a.

Test Trial 2
22, 24, 31, 35, 27, 32, 33, 23, 39, 45, 36, 43, 25, 46, 40, 49, 29, 21, 26, 48
Result: Item = 39; Position = 9; Relative rank = 13. See figure 7b.

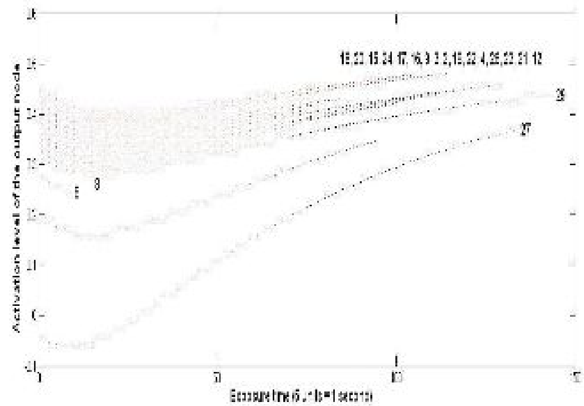

Fig 7a. Activation levels of the output node of N2 for trial 1

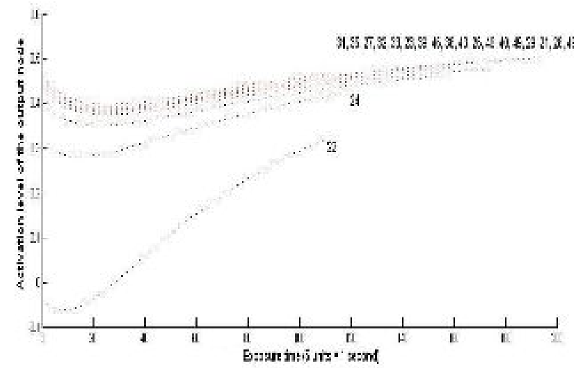

Fig 7b. Activation levels of the output node of N2 for trial 2

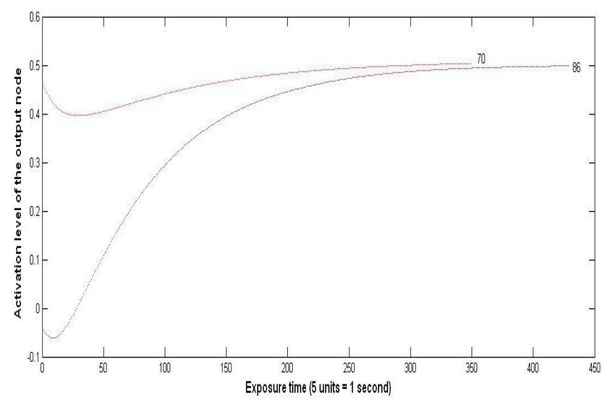

Fig 7c. Activation levels of the output node of N2 for trial 3

Test Trial 3
86, 70, 73, 84, 65, 68, 60, 66, 82, 74, 83, 78, 71, 81, 61, 63, 79, 72, 76, 77
Result: Item = 70; Position = 2; Relative rank = 7. See figure 7c.

Observations and interpretations

The network seems to have learnt the smallest and the largest values of the entire range. It can readily accept 89 wherever it sees it (see response for the item 86 in fig 7c) and never accepts 1 (always a dip). Consequently, in the last few trials, it ends up making a very early choice because they are the biggest numbers it has ever seen and in the first few trials, it ends up making no choice as they are some of the smallest. So, in these ranges the average rank is 10 as that is approximately the average rank at any position of the trial out of a maximum rank of 20. Some of the most significant differences between N1 and N2 are discussed below.

Strategy

From the observations above, it appears that N2 has learned to differentiate between the worth of the items it has seen during its evolution rather than differentiate them within a trial. Its aspiration seems to be set by the first item rather than by the first few items as in N1. It can be vaguely stated as "I need an item larger than the previous items but if it is large enough (say greater than 60), I might accept it". Unlike N1, initial activation level of an item is much lower than that of the previous item. It looks like the pre-caution that is taken by N1 in the first few items is taken after each item in the case of N2. Also, though the dip looks deeper in each item than N1, the LBA seems to be almost the same as what is set by N1 (see corresponding sub-figures of figures 4 and 7). Still, the shoot-up is more conspicuously non-linear than N1 and could differ quite drastically with each item. That is one of the reasons why it could become hasty (see trial 3).

Discussion

We note that artificial evolution has resulted in a CTRNN with a strategy strikingly similar to the optimal strategy developed using rigorous mathematical analysis. It makes us wonder how such a strategy could have evolved. Could the transitions in the evolution of the strategy have followed the same analytical steps that a mathematician uses in his method? What is more, there seems to be at least one more strategy to solve the problem as reflected from the behavior of N2 whose performance is quite comparable to that of N1. Though we have not been able to verbalize its strategy as we could do for N1, from a cognitive viewpoint, we have been able to describe how the network has learnt to assess an item in a sequence based on its position. It has particularly interesting implications on the cognition of judgment when we interpret the dips and shoot-ups in the behavior patterns as being weary and being optimistic respectively. It further implies that a definitive strategy like the analytic solution is not necessarily the only way to solve the Secretary problem; 'patterns of cognition' could work too. Of course the lack of the ability to generalize to the variants of the problem (Tuci et al. 2002) could draw criticisms against making such an inference. Still, it can not be conclusively said that the network has learnt (or rather 'memorized') the boundaries of each trial, thereby performing better than the optimal strategy in some trials; the smallest and largest item in a trial is not always the same. Therefore, the network should have learnt, to some extent, to what to expect based on what a 'judgment' from what it has seen. This 'judgment' is what we refer to as 'patterns of cognition'. What makes it interesting is that it does not seem to be a definitive strategy as the optimal strategy and yet yields a comparable performance.

Conclusion

Different CTRNNs were evolved to solve the Secretary problem and their behavior was analyzed. One of them evolved a strategy similar to the analytically optimal strategy. It was also observed that two different networks with almost equal average performances can evolve totally different behaviors. Above all, an investigative study of the neuronal activation levels has proved to be extremely useful in unveiling the CTRNN's behavior. This approach could particularly be useful when a level of analysis higher than the usual dynamical systems theoretical approach is necessitated. This kind of behavioral analytical approach could be a lot simpler to adopt in case of more complicated probabilistic decision making problems.

References

Beer, R. D (1996). Toward the evolution of dynamical neural networks for minimally cognitive behavior. *From Animals to Animates IV*: Proc of the 4th Intl. Conf. of Simulation of Adaptive behavior, pages 421-429.

Beer, R.D (2003). The Dynamics of Active Categorical Perception in an Evolved Model Agent. *International Society of adaptive Behavior,*11(4):209-243.

Cliff, D., Harvey, I., and Husbands, P (1993). Explorations in Evolutionary Robotics. *Adaptive Behavior*, 2(1): 73-110.

Czogala, E. and Roubens, M (1989). An approach to multi-criteria decision making problems using probabilistic set theory. *European Journal of Operational Research*, 43(3):263-266.

Ferguson, T.S (1989). Who solved the secretary problem? *Statistical Science*, 4(3):282-296.

Goldenberg, E., Garcowski, J., and Beer, R.D (2004). May we have your attention: Analysis of a selective attention task. *From Animals to Animats 8*: Proceedings of the Eighth International Conference on Simulation of Adaptive Behavior, July 13-17.

Harvey, I., Di Paolo, E., Wood, R., Quinn, M, and Tuci, E.A (2005). Evolutionary Robotics: A new scientific tool for studying cognition *Artificial Life*, 11(1-2), pp. 79-98.

Tuci, E., Harvey, I., and Todd, P.M (2002). Using a net to catch a mate: Evolving CTRNNs for the Secretary problem*. Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior*.