A Model of Baldwin Effect in Populations of Self-Learning Agents

Vladimir G. Red'ko
Institute of Optical Neural Technologies, Russian Academy of Science
Vavilova Str., 44/2, Moscow, 119333, Russia.
redko@iont.ru

Oleg P. Mosalov
Moscow Institute of Physics and Technologies
Institutsky per., 9, Dolgoprudny, Moscow region, 141700, Russia.
olegmos_@mail.ru

Danil V. Prokhorov
Ford Research and Advanced Engineering, Ford Motor Company
2101 Village Rd., MD 2036, Dearborn, MI 48124, U.S.A.
dprokhor@ford.com

***Abstract* – We study an evolution model of adaptive self-learning agents. The control system of agents is based on a neural network adaptive critic design. Each agent is a broker that predicts stock price changes and uses its predictions for action selection. The agent tries to get rich by buying and selling stocks. We demonstrate that the Baldwin effect takes place in our model, viz., originally acquired adaptive policy of an agent-broker becomes inherited in the course of the evolution. In addition, we compare agent behavioral tactics with searching behavior of simple animals.**

I. INTRODUCTION

Development of multi-agent systems is a promising research direction of computational intelligence, featuring various phenomena in evolving populations of adaptive agents. One of the most interesting phenomena that can be observed in such populations is the Baldwin effect [1-8]. According to the Baldwin effect, learned features of organisms can be inherited indirectly in subsequent generations of organisms. The Baldwin effect works in two steps. In the first step, evolving organisms obtain an ability to learn a certain advantageous trait through appropriate mutations. The fitness of such organisms is increased, and they are spread throughout the population. However, learning is typically costly because it requires energy and time. Here comes the second step called the genetic assimilation. The advantageous trait can be "reinvented" by the genetic evolution and becomes directly genetically encoded. The second step takes a number of generations. A stable environment and a high correlation between genotype and phenotype facilitate this step. Thus, the advantageous trait originally acquired can become inherited though the Darwinian evolution.

G. Hinton and S. Nowlan [2], D. Ackley and M. Littman [3], G. Mayley [4] and other researchers [5-8] analyzed the Baldwin effect by means of computer simulation. They showed that this effect could play important role in the process of evolution of the model organisms.

In this paper, we design and investigate an evolution model of adaptive self-learning agents; the control system of agents is based on a neural network adaptive critic design (ACD). The ACD includes two neural networks (NNs): model and critic. The model predicts the state of the environment for the next time step, and the critic is used to select actions on the basis of this prediction. These NNs can be optimized by both learning and evolution.

In comparison with other investigations of the Baldwin effect, our study pays the main attention to self-learning autonomous agents. Though our work is similar to that of D. Ackley and M. Littman [3], the control system of our agents is more theoretically justified. First, it is based on the ACD architecture, with the well-investigated temporal difference algorithm [9] as its learning method. Second, our agent control system includes the model NN, thereby allowing the agent to predict future environment states and use its predictions for action selection. Explicit predictions of future states can provide new capabilities in intelligent control systems of autonomous agents.

In our setup, a population of agents evolves. At the birth of each agent, initial synaptic weights of its NNs form its genome. During the agent life, the weights of its NNs are adapted by means of reinforcement learning. Agents that learn well receive large rewards and procreate. Children inherit the genomes of their parents (initial synaptic weights of the NNs). The genomes of evolving agents are subjected to small mutations. If initial weights of some agents drift via mutations toward successful weights obtained in the course of learning, then such agents will learn faster and, hence, obtain more reward. During several generations, the starting weights can reach values which, from the behavioral standpoint, were obtained previously by the successful learning process. Thus, acquired synaptic weights effectively become inherited. This scheme is in contrast with the direct inheritance of learned weights of parents by their children.

This paper consists of six sections. In Section II we describe the agent task. Agent control system is described in Section III, followed by evolution specifics in Section IV. We describe our experiments and results in Section V, followed by conclusion in Section VI.

II. Agent Task

Inspired by [10], we consider an adaptive agent-broker. It predicts change of a stock price and tries to increase its wealth by buying and selling stocks. The agent has its resource distributed into cash and stocks. The sum of these is the net capital of the agent $C(t)$. The state of the agent is characterized by the variable $u(t)$, which is the fraction of stocks in the net capital of the agent. The environment is determined by the time series $X(t)$, t = 1,2,…, where $X(t)$ is the stock price at the moment t. The goal of the agent is to increase its capital $C(t)$ by changing the value $u(t)$. The capital dynamics is

$$C(t+1) = C(t)\ \{1 + u(t+1)\ \Delta X(t+1)\ /\ X(t)\}\times [1 - J\ |u(t+1) - u(t)|], \qquad (1)$$

where $\Delta X(t+1) = X(t+1) - X(t)$ is the current change of the stock price, J is a parameter that takes into account expenses of the agent when buying/selling stocks. The factor in the braces corresponds to the change of the capital as the result of stock price rise/drop. The factor in the square brackets is the expenses of the agent when buying/selling stocks. Following [11], we use the logarithmic scale for the agent resource, i.e., $R(t) = \log C(t)$. The current agent reward $r(t)$ is defined by the expression: $r(t) = R(t+1) - R(t)$:

$$r(t) = \log\ \{1 + u(t+1)\ \Delta X(t+1)\ /\ X(t)\} + \log\ [1 - J\ |u(t+1) - u(t)|]. \qquad (2)$$

For simplicity and unlike [10], we assume that the variable $u(t)$ takes only two values, $u(t)$ = 0 (all in cash) or $u(t)$ = 1 (all in stock).

III. Agent Control System

The agent control system is a simplified ACD. Our adaptive critic scheme consists of two neural networks: model and critic (see Fig.1). The goal of the adaptive critic is to maximize stochastically utility function $U(t)$ [9]:

$$U(t) = \sum_{j=0}^{\infty} \gamma^j r(t+j),\ t = 1, 2, \ldots \qquad (3)$$

where $r(t)$ is an instantaneous reward obtained by the agent, and γ is the discount factor ($0 < \gamma < 1$). Assuming $|\Delta X(t+1)| << X(\mathrm{t})$ for all t, we suppose that the ACD state $\mathbf{S}(t)$ at moment t is characterized by two values, $\Delta X(t)$ and $u(t)$: $\mathbf{S}(t) = \{X(t), u(t)\}$.

The role of the model is to predict changes of the stock time series. The model output $\Delta X^{Pr}(t+1)$ is based on N previous values of ΔX, $\Delta X(t\text{-}N+1)$, … , $\Delta X(t)$, which are used as the model inputs. The model is implemented as a multilayer perceptron (MLP) with one hidden layer of tanh nodes and linear output, and it is trained by the usual backpropagation method.

The critic is intended to estimate the state value function $V(\mathbf{S})$ (estimate of U in (3)) for the current state $\mathbf{S}(t) = \{\Delta X(t), u(t)\}$, the next state $\mathbf{S}(t+1) = \{\Delta X(t+1), u(t+1)\}$, and its predictions $\mathbf{S^{pr}}_{\mathrm{u}}(t+1) = \{\Delta X^{pr}\ (t+1), u\}$ for two possible actions, u = 0 or u = 1. The critic is also a MLP of the same structure as the model, but it is trained by the temporal difference method [9].

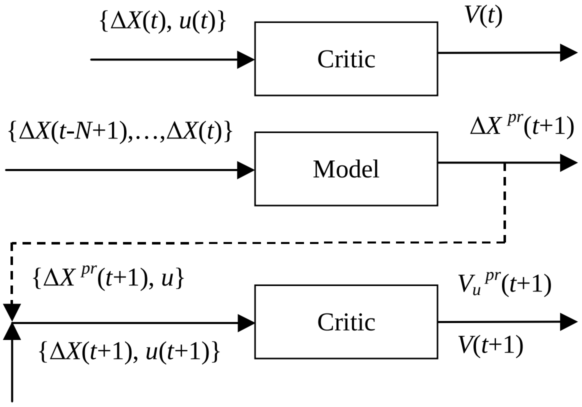

Fig. 1. Our ACD. The model predicts changes of the time series. The critic (the same neural network is shown in two consecutive moments) forms the state value function for the current state $\mathbf{S}(t) = \{\Delta X(t), u(t)\}$, the next state $\mathbf{S}(t+1) = \{\Delta X(t+1), u(t+1)\}$, and its predictions $\mathbf{S^{pr}}_{\mathrm{u}}(t+1) = \{\Delta X^{pr}\ (t+1), u\}$ for two possible actions, u = 0 or u = 1.

At any moment t, the following operations are performed:

1) The model predicts the next change of the time series $\Delta X(t+1)$.

2) The critic estimates the state value function for the current state $V(t) = V(\mathbf{S}(t))$ and the predicted states for both possible actions $V^{\mathrm{pr}}{}_{u}(t+1) = V(\mathbf{S^{pr}}_{u}(t+1))$, where $\mathbf{S^{pr}}_{\mathrm{u}}(t+1) = \{\Delta X^{pr}\ (t+1), u\}$, and u = 0 or u = 1.

3) The ε-greedy rule is applied [9]. With the probability 1-ε the action corresponding to the maximum value $V^{pr}{}_{u}(t+1)$ is selected, and an arbitrary action is selected with the probability ε ($0 < \varepsilon << 1$).

4) The selected action is carried out. The transition to the next time moment t+1 occurs. The current reward $r(t)$ is calculated in accordance with (2) and received by the ACD. The value $\Delta X(t+1)$ is observed and compared with its prediction ΔX^{pr} $(t+1)$. The model NN weights are adjusted to minimize the prediction error using the error backpropagation and the gradient descent with $\alpha_M > 0$ as the model learning rate.

5) The critic computes $V(t+1)$. The temporal-difference error is calculated:

$$\delta(t) = r(t) + \gamma\, V(t+1) - V(t)\ . \tag{4}$$

6) The weights of the critic neural network are adjusted to minimize the temporal-difference error (4) using its backpropagation and the gradient descent with $\alpha_C > 0$ as the critic learning rate.

IV. Evolution of Adaptive Agents

Our evolving population consists of n agents. Each agent has a resource $R(t)$ that changes in accordance with values of agent rewards: $R(t+1) = R(t) + r(t)$, where $r(t)$ is calculated in (2). At the beginning of any generation, all agents have the same initial resource $R(0)$.

The initial synaptic weights of both NNs (model and critic) form the agent genome **G**. The genome **G** does not change during agent life, and it is fixed when the agent is born. However, synaptic weights of the NNs **W** are changed during agent life via learning described in Section III.

Evolution passes through a number of generations, $n_g=1,2,\ldots, N_g$. The duration of each generation n_g is T time steps. At the end of each generation, the agent having the maximum resource R_{max} (n_g) is determined. This best agent gives birth to n children that constitute a new (n_g+1)-th generation. The children genomes **G** differ from their parent genome by small mutations. Mutations affect all elements of **G**, as a normally distributed random value with zero mean and standard deviation P_{mut} (mutation intensity) is added to each synaptic weight.

At the beginning of every new (n_g+1)-th generation, we set for each agent $\mathbf{G}(n_g+1) = \mathbf{G}_{\mathbf{best}}(n_g)$ + **mutations,** $\mathbf{W}(n_g+1) = \mathbf{G}(n_g+1)$, where $\mathbf{G}_{\mathbf{best}}(n_g)$ is selected from the best agent of the previous n_g-th generation. Thus, the genome **G** changes only via evolution, whereas the synaptic weights **W** are adjusted only via learning.

On average, we expect that the best agents will begin to accumulate R earlier with every new generation as the result of the Baldwin effect.

V. Results of Simulations

In our computer simulations we use two examples of model time series:

1) sinusoid:

$$X(t) = 0.5(1 + \sin(2\pi t/20)) + 1, \tag{5}$$

2) stochastic time series from [10, Example 2]:

$$\beta(t) = \alpha\beta(t\text{-}1) + \gamma(t),$$

$$p(t) = p(t\text{-}1) + \beta(t\text{-}1) + k\,\lambda(t),$$

$$X(t) = \exp(p(t)/1200), \tag{6}$$

where $\lambda(t)$ and $\gamma(t)$ are two random normal processes with zero mean and unit variance (N(0,1)), and where $\alpha = 0.9$, $k = 0.3$.

Some parameters are set to the same values for all simulations. Specifically, we set population size $n = 10$, discount factor $\gamma = 0.9$, number of inputs of the model NN $N = 10$, number of hidden neurons of the model and the critic $N_{hM} = N_{hC} = 10$, learning rate of the model and the critic $\alpha_M = \alpha_C = 0.01$, expenses of the agent when buying/selling stocks $J = 0$, and the initial resource of newborn agent $R(0)=0$. Other parameters (generation duration T, parameter ε of the ε-greedy rule, mutation intensity P_{mut}) were set to different values, depending on the simulation, as specified below.

For our experiments with the sinusoid (5), we compared the maximum resource in the population R_{max} (n_g) at the end of each generation in two cases:

1) evolution of self-learning agents, as detailed in Section IV (the blue curve in Fig. 2), and
2) pure evolution, i.e., without agent learning (the red curve in Fig. 2).

The results are averaged over 1000 simulations with T = 200, ε = 0.05, P_{mut} = 0.01. Fig. 2 does indicate a significant advantage to combining evolution with learning.

The rest of the paper discusses our results for the evolution of self-learning agents. Detailed dynamics of the best agent resource $R_{max}(t)$ for the sinusoidal $X(t)$ in a particular simulation is illustrated by Fig. 3 for the first five generations (note that $R_{max}(n_g)=R(200k)$ of the best agent, where k=1,2,3,4,5). The parameters of the simulation are the same as for Fig. 2. Fig. 3 demonstrates the sequential improvement of agent policies. The well-adapted agent behaves in the following way. It buys stocks at the moments of predicting stock price rises and sells stocks when predicting stock price falls. This corresponds to the optimal policy. The agent capital periodically increases (when the stock price rises), and it remains approximately constant (when the stock price falls). Fig. 3 shows that, in the 1st generation, the best agent optimizes its policy by learning and finds a rough solution only by the end of this generation. Subsequently, the best agents find a satisfactory (close to optimal) policy faster and faster. By the 5th generation, a newborn agent "knows" a satisfactory policy as encoded in its genome **G**, and the learning does not improve the policy significantly. Thus, Fig. 3 demonstrates that, for the simple periodic dependence $X(t)$, the initially learned policy becomes inherited.

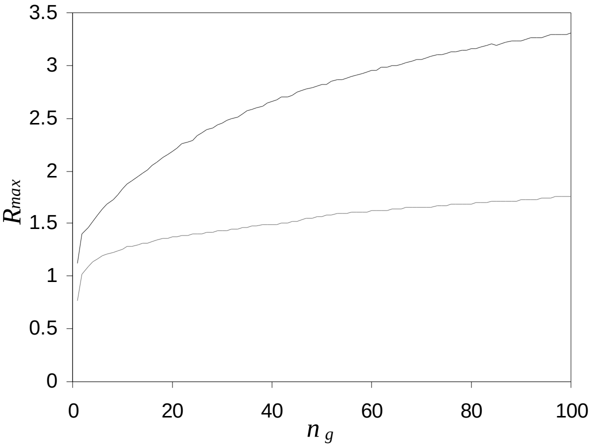

Fig. 2. Two plots of the maximum resource in the population $R_{max}(n_g)$ attained by the end of each generation vs. the generation number n_g . The blue curve corresponds to the case of the evolution of self-learning agents (evolution is combined with learning, as detailed in Section IV), whereas the red curve corresponds to the case of pure evolution (without learning). Each point of the plots represents an average over 1000 simulations, each starting with a different random seed; $T = 200$, $\varepsilon = 0.05$, $P_{mut} = 0.01$

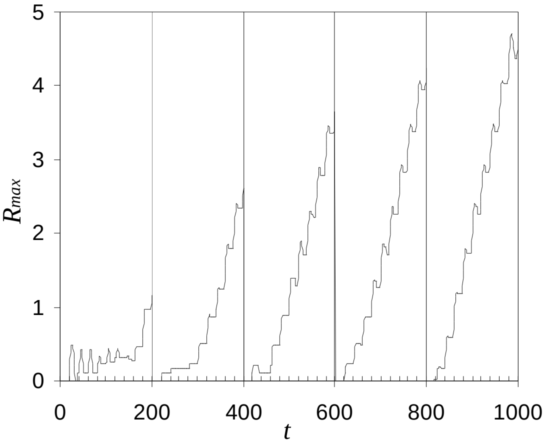

Fig. 3. The resource plot of the best agent in the population for a particular simulation; the case of sinusoidal time series (5), first five generations. The moments when generations end are shown by vertical lines; $T = 200$, $\varepsilon = 0.05$, $P_{mut} = 0.01$. Periodical increases (when the stock price rises) and approximately constant values (when the stock price falls) of R_{max} correspond to optimal policy of the agent, e.g., such policy is observed in 5th generation. See the text for details.

Figs. 4 and 5 illustrate our simulation results for the stochastic time series (6). Fig. 4 shows dynamics of the best agent resource $R_{max}(t)$ for the first twelve generations. Fig. 5 shows dynamics of the best agent action selection during the 2nd generation ($3000 < t < 3500$) (Fig. 5a), in the 12th generation ($28300 < t < 28400$) (Fig. 5b) and in the 48th generation ($118500 < t < 119000$) (Fig. 5c) for the same simulation. The time series $X(t)$ is also shown in all figures. The parameters of this simulation are $T = 2500$, $\varepsilon = 0.03$, $P_{mut} = 0.03$. Fig. 4 demonstrates that the agent resource at the end of each generation features an upward trend. During the early generations (generations 3 to 7), any significant increase of the agent resource begins only in the second half of the agent life. This means that the agent learning process takes a while before it finds an adequate policy. During the later generations (generations 9 to 12), the increase of the resource begins from the start of each generation, demonstrating that the advantageous policy becomes inherited.

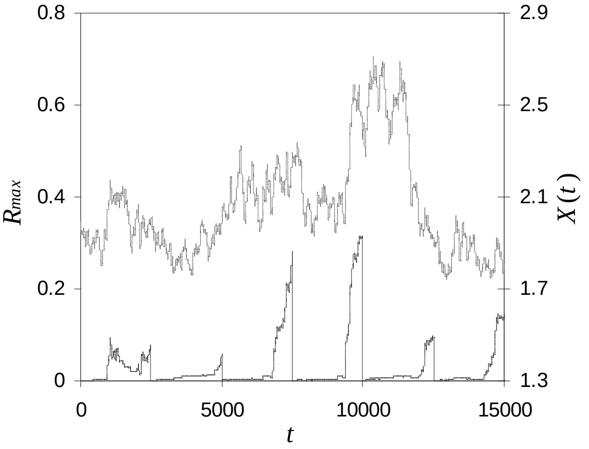

Fig. 4a. The resource plot of the best agent in the population (blue); the stochastic time series; 1-6th generations; $T = 2500$, $\varepsilon = 0.03$, $P_{mut} = 0.03$. The time series $X(t)$ is red.

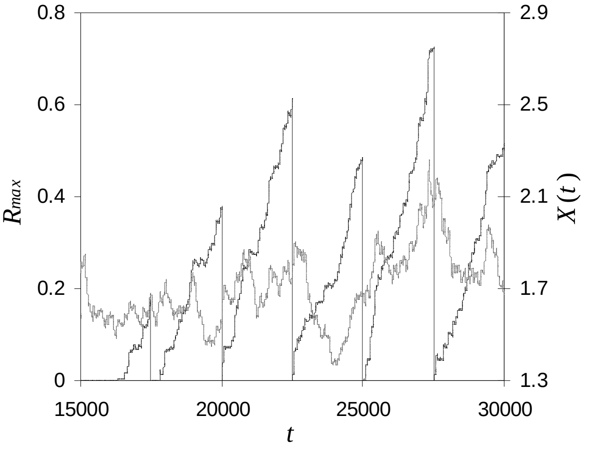

Fig. 4b. The resource plot of the best agent in the population (blue); the stochastic time series; 7-12th generations. The time series $X(t)$ is red.

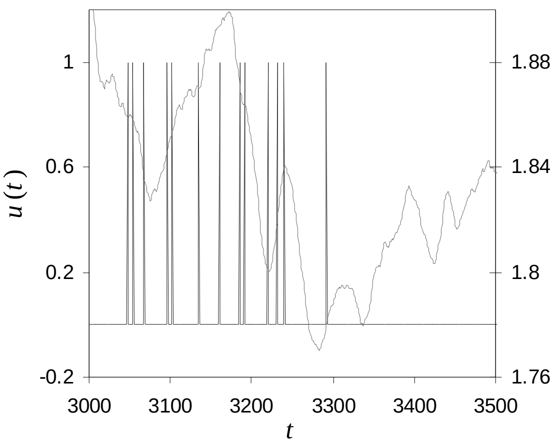

Fig. 5a. The time dependence of actions selected by the best agent in the population (blue); actions are characterized by values $u(t)$: $u = 0$ (all in cash) and $u = 1$ (all in stock); this is the 2nd generation fragment. The time series $X(t)$ is red.

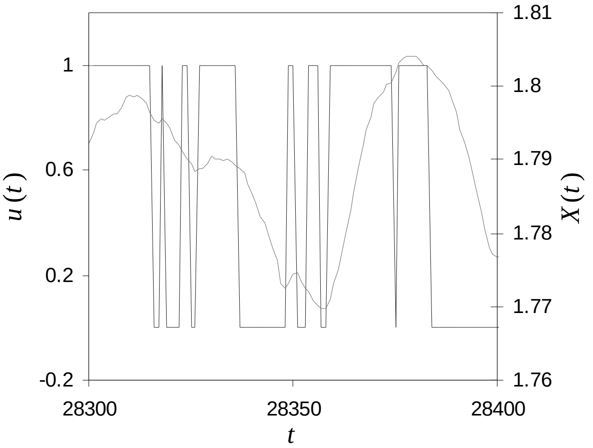

Fig. 5b. The time dependence of actions selected by the best agent in the population (blue); actions are characterized by values $u(t)$: $u = 0$ (all in cash) and $u = 1$ (all in stock); this is the 12th generation fragment. The time series $X(t)$ is red.

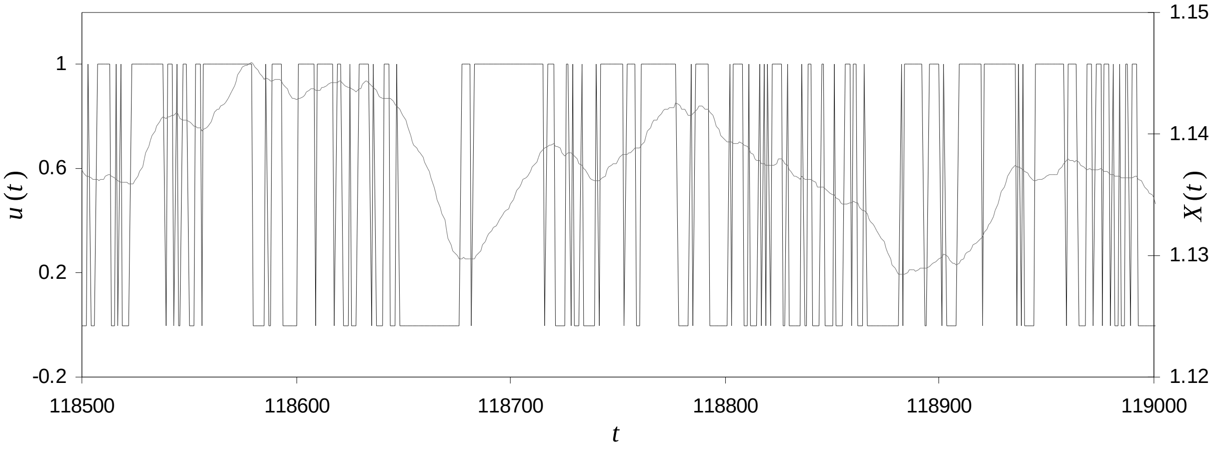

Fig. 5c. The time dependence of actions selected by the best agent in the population (blue); actions are characterized by values $u(t)$: $u = 0$ (all in cash) and $u = 1$ (all in stock); this is the 48th generation fragment. The time series $X(t)$ is red.

Figs. 4 and 5 demonstrate that, at the beginning of evolution, the agents do not find an effective policy. For example, during the 2nd generation the best agent prefers to keep all capital in cash, i.e., $u = 0$ almost all the time in this case (Fig. 5a). However, by the 12th generation the best agent finds a reasonable policy (Fig. 5b). It buys/sells stocks when correctly predicting imminent stock price increases/decreases. It should be noted that the agent policy is not always optimal. For example, at t = 28315, 28332, 28380, the trend changes from the price rise to the price fall. However, the agent does not sell stocks immediately at these moments. It appears to be waiting for the down trend to become more salient, namely, to reach a sufficiently large negative values of $\Delta X(t)$.

In the 48th generation (Fig. 5c) the agent demonstrates an interesting rational behavior. Anticipating moderate stock price increase, the agent usually transforms its capital into stocks. In contrast, anticipating moderate stock price decrease, the agent demonstrates searching behavior. It tries randomly to transform its capital into stocks for a short time (e.g., at t = 118550, 118850, 118980). Such random searching tactic can be useful, if the agent is not confident about future changes of environment. The tactic is to some extent asymmetrical, i.e., the agent prefers to keep its capital in stocks (see also Fig. 5b), as if hoping to get future positive rewards (profits) during possible stock price increases. Thus, the agent switches its behavior between two tactics (buy stocks or sell stocks). Such process of switching appears to have both inertial and random search components.

It should be noted that the behavior with switching between two tactics is analogous to searching tactics of simple animals. For example, some species of caddis fly larvae use similar tactics for case building [12, 13]. The larvae inhabit creek bottoms and build their cases from hard particles of different size. They can use small or large sand particles [12]. Large particles are distributed randomly, but typically in groups of several particles. Using large particles, the larva can build cases more quickly and effectively than with small particles, so its preference is evident. The larva uses two tactics: 1) testing particles in its vicinity and building the case from selected particles, 2) searching for a new place with a collection of appropriate particles. Investigations of larva behavior reveal inertia in switching from the first tactic to the second tactic [12, 13]. If the larva finds a large particle, it continues testing particles until it finds several small particles, and only after repeated failures to find new large particles does the larva switch to the second tactic. During searching for a new place, the larva wanders and sometimes randomly tests particles along its way. It can switch from the second tactic to the first tactic, if it finds a large particle. When switching from the second tactic to the first tactic, it also exhibits inertia. The switching between tactics resembles a random search with inertial effects.

The larva behavior appears to have similarities with our agent-broker behavior. We can view the agent keeping the capital in stocks (u = 1) as an equivalent of the first larva tactic. Indeed, both the agents and the animal can obtain a profit pursuing this tactic: the agent may obtain a positive reward, and the larva may build its case more effectively. The second tactic (keeping u = 0 or brisk switching between different u for the agent, and searching for a new place for the larva) is waiting/searching for conditions for profitable actions. Switching between the tactics appears to include essentially random components for both the agent and the larva, as well as switching inertia. It is reasonable to assume that both the random switching and the inertia are due to insufficient knowledge of both the agent and the larva about their respective environments.

VI. Conclusion

We demonstrated evolutionary assimilation of acquired features (the Baldwin effect) in a population of self-learning agents, in which agent control systems are based on a neural network adaptive critic design. The agent task was that of a broker, previously considered in [10, Example 2]. We did not intend to improve the results of [10] because our goal was to study the Baldwin effect on a relatively simple but sufficiently illustrative problem.

Our simulations also demonstrated that the agents learn different behavioral tactics in analogy to adaptive behavior of simple animals. Of course, more detailed studies are needed to understand thoroughly the relationship and analogies between these behaviors.

Our work describes a possible approach to investigation of evolution of autonomous adaptive agents. Different learning algorithms for agent control systems may be employed, without changing the fundamental outcome.

Our future research can include:

- a more detailed analysis of interaction between learning and evolution;
- an investigation on the role of prediction in shaping adaptive behavior of autonomous agents;
- a more thorough comparison of agent behavior with adaptive behavior of simple animals.

Acknowledgments

The authors thank Dr. Valentin A. Nepomnyashchikh for valuable consultations on the behavior of caddis fly larvae. This work is supported by the Russian Foundation for Basic Research, Grant No 04-01-00179 and the Program of the Presidium of the Russian Academy of Science “Mathematical Modeling and Intelligent Systems”, Project No 16.2.45.

References

[1] J.M. Baldwin, “A new factor in evolution,” *American Naturalis*t, vol. 30, pp. 441-451, 1896.

[2] G. E. Hinton and S. J. Nowlan, “How learning can guide evolution,” *Complex Systems*, vol. 1, pp. 495–502, 1987.

[3] D. Ackley, M. Littman, “Interactions between learning and evolution” In Langton C. G., Taylor C., Farmer J. D., and Rasmussen S. (Eds.) *Artificial Life II*. Reading, MA: Addison-Wesley. 1992, pp.487-509.

[4] G. Mayley, “Landscapes, learning costs and genetic assimilation,” *Evolutionary Computation*, vol. 4, No 3, pp. 213–234, 1996.

[5] R.K. Belew and M. Mitchell (Eds.), *Adaptive Individuals in Evolving Populations: Models and Algorithms*, Massachusetts: Addison-Wesley. 1996.

[6] P.Turney, D. Whitley, R. Anderson (Eds.), *Evolution, Learning, and Instinct: 100 Years of the Baldwin Effect* // Special Issue of Evolutionary Computation on the Baldwin Effect, vol.4, No 3, 1996.

[7] R. Suzuki and T. Arita, “The Baldwin effect revisited: three steps characterized by the quantitative evolution of phenotypic plasticity,” In *Proceedings of the Seventh European Conference on Artificial Life (ECAL2003)*, 2003, pp. 395-404.

[8] R. Suzuki and T. Arita “Interactions between learning and evolution: outstanding strategy generated by the Baldwin effect,” *Biosystems*, vol. 77, No 1-3, pp. 57-71, 2004.

[9] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction.* – Cambridge: MIT Press, 1998.

[10] D. Prokhorov, G. Puskorius, and L. Feldkamp, “Dynamical neural networks for control,” In J. Kolen and S. Kremer (Eds.) *A Field Guide to Dynamical Recurrent Networks*, IEEE Press, 2001.

[11] J. Moody, L. Wu, Y. Liao, M. Saffel, “Performance function and reinforcement learning for trading systems and portfolios,” *Journal of Forecasting*, vol. 17, pp. 441-470, 1998.

[12] V.A. Nepomnyashchikh, “Selection behaviour in caddis fly larvae,” In *From Animals to Animats 5:* Proceedings of the Fifth International Conference of the Society for Adaptive Behavior / Eds. R. Pfeifer et al. Cambridge, USA: MIT Press, 1998. pp.155-160.

[13] V.A. Nepomnyashchikh “How animals solve bad-formalized search tasks,” In *Synergetics and psychology. Texts*. Issue 3, Moscow, 2004, pp. 197-209 (In Russian)