Learning benefits evolution if sex gives pleasure

A.R. Griffioen, S.K. Smit and A.E. Eiben

Abstract— **In this paper we investigate the effects of individual learning on an evolving population of situated agents. We work with a novel type of system where agents can decide autonomously (by their controllers) if/when they reproduce and the bias in the agent controllers for the mating action is adaptable. Our experiments show that in such a system reinforcement learning with the straightforward rewards system based on energy makes the agents loose their interest in mating, that is, learning counteracts evolution. This effect can be eliminated by introducing a specific reward for the mating action that always gives positive feedback to the agents, as some kind of pleasure or "orgasm". Using such a combination, individual learning is able to keep non-optimal agents alive, where evolution only leads to extinction. Despite that it preserves a viable population that is able to acquire the necessary survival skills, we also found a disadvantage of learning, namely a hiding effect of ill adapted non-optimal performing agents.**

I. Introduction

In this paper we investigate the effects of individual learning on an evolving population of situated agents. This work fits in the framework of Population-based Adaptive Systems (PAS) with threefold adaptation as described in [6]. In the most general case such PAS's feature evolutionary, individual, and social learning – threefold adaptation. In the present study we only consider the first two.

Combinations of evolution and learning have been investigated before [3], cf. the hundred years of the Baldwin effect [19]. Prominent clusters of related work can be found within memetic algorithms, or hybrid evolutionary algorithms [11], [9], evolutionary robotics [13], [8] and ALife [18], [2], [12], [5], [4]. As we explain below our system has a special combination of features. An important property, implied by these features, is that the population size can change even to extinction. This property is often absent in related work. (Note: Research on predator-prey phenomena is usually not concerned with combinations of evolution and learning.) This also holds for work that claims to model natural systems [15], though it is evident that in nature populations can go extinct. Past research has focussed on the costs and benefits of learning in evolution [7], [10], [12], [13] and on identifying factors that influence this relationship [10], [13]. In this paper we will carry out research towards these topics, but in the context of a changing population size.

The PAS we work with has a specific combination of features that distinguishes it from most other systems in this area:

1) Lack of a crisp optimization criterion. There is no objective function to be optimized (as in typical evolutionary algorithm applications), nor a concrete task to be performed optimally (as in evolutionary robotics). Our agents "only" need to survive in their environment (as in some ALife systems).
2) Agents are situated in their environment. They can sense the environment, see and hear things, and can also change it by their actions. Environmental input is processed by the *controller* of the agent to determine appropriate actions.
3) The evolutionary mechanism is based on natural reproduction.[1] That is, we do not have a fixed size population and an oracle managing the reproduction cycles (parent selection, reproduction, and survivor selection). Instead, offspring creation (parent selection and reproduction) is detached from survivor selection and no central control is exercised on either of them. Considering offspring creation, agents decide autonomously and asynchronously about mating driven by their individual controllers. This is what we call natural reproduction (based on autonomous parent selection). It is complemented by natural selection: asynchronous survivor selection, where an agent dies if it runs out of energy. Note that such a system has two important properties:
   - Populations can grow or shrink, because births and deaths are not related. That is, a new individual can be born without an old one being removed and an existing individual can die without being replaced by a new one. As a consequence, the population size is inherently varying over time.
   - The evolutionary mechanism is partly under control of the agents, because it is the agents themselves who decide if and when to create offspring. This means that the development of agent controllers (through evolution and/or learning) can lead to intensively reproducing agents or just the opposite. Hence, the evolutionary mechanism itself is subject to changes over time.
4) Evolutionary learning and individual learning are acting in the same search space, that of the set of all possible agent controllers. Hence, an agent can be born with controller C, created by an evolutionary operator applied to its parents, and can change C into C', C", etc., during its lifetime by applying the individual learning operator. We postulate that our system is non-Lamarckian: When this agent reproduces only its original controller C is used for creating a child, the individually learned parts of C' etc. do not form inheritable material.

Artificial Intelligence, Vrije Universiteit Amsterdam, The Netherlands, [argriffi,sksmit,gusz]@few.vu.nl

[1] Not to be confused with natural selection.

5) Individual learning is implemented through Reinforcement Learning (RL). In essence, RL changes the controller by regulating agent preferences for actions based on a reward system. Note that in principle RL can strengthen/weaken preferences for all agent actions, including the mating action required for offspring creation. Hereby it is possible that individual learning unlearns reproduction and effectively counteracts evolutionary learning.

The main research questions we consider here are the following. Given an evolving system of situated agents:

1) What is the effect of adding individual learning through reinforcement learning?
   a) On the viability of the population?
   b) On the performance of the population?
   c) On the evolutionary engine?
2) How does this depend on the rewards used by RL? In particular:
   a) When rewards are energy-based.
   b) When rewards are hard-wired by the user.

The paper is organized as follows. In Section II we briefly describe the underlying system and experimental software platform NEW TIES, developed by the NEW TIES project [2]. In Section III and Section V we discuss the experimental setup including the specifics of the environment, the agents, the learning mechanisms, and the system monitors and measures used to generate data we use to find answers to our research questions. These data are presented and analyzed in Section IV and Section VI. The paper is concluded by Section VII rewieving our findings and giving an outlook to ongoing and further research.

II. NEW TIES

Our investigations are carried out in the NEW TIES system [3] that facilitates setting up different types of worlds, agents, and adaptive mechanisms. A general description can be found in [6]. The system was developed with a specific type of application in mind: socio-biological simulations. NEW TIES agents live in a "simulated physical" world carried by space, time and energy. Space, time and energy in NEW TIES are discrete. Space is implemented as a rectangular grid, time shifts by atomic timesteps, and energy is administered in basic units. Agents can move over the grid and interact with other agents and objects such as plants or tokens. Agents can perform a number of actions, like move, turn, eat, mate, talk, pick up, etc.

Agents have to maintain their energy level: everything, even inactively surviving a timestep, costs energy and running out of energy means that the agent dies. To gain energy, an agent must eat food (plants). The laws of nature governing the environment determine the preconditions and the results of actions, e.g., they specify the amount of energy a plant yields when eaten, the costs of movement, the maximum lifetime for agents, or a minimum age and energy level at which agents can mate. Agents decide on their actions using a controller. In other words, the controller is the decision making unit inside an agent that maps inputs, i.e., perceptions of the agent of the world and its own internal state, to ouptuts, i.e., actions of the agent. In general, one could distinguish between the agents' body properties (colour, shape, sex, weight, etc.) and brain properties (the controller). Here we focus on the adaptation of controllers.

Obviously, agents and plants form a predator-prey system. From the agents' perspective this represents a survival game where the only objective is survival of the individual and the agent population. To survive agents have to adapt to their environment. To this end, there are three adaptive mechanisms: evolution, individual learning, and social learning. One of the main objectives of the NEW TIES project is to investigate the interactions among these adaptive mechanisms.

A. The challenge

It is clear that different circumstances regarding the physical world and the plants require different agent behavior to survive and prosper. In other words, a particular setup represents a particular challenge or learning task that agents must solve through adaptation. For the present investigation we have chosen a known problem to represent the learning task: The poisonous food challenge where agents must learn to distinguish between poisonous and edible plants, [14], [12], [18]. In our scenario there are two types of plans, edible and poisonous, and both types of plants can be eaten by the agents. Eating an edible plant increases energy level of the agent, while eating a poisonous plant reduces it. Agents adapt successfully and solve the challenge if they learn not to eat poisonous plants.

B. Agents

NEW TIES agents can perceive their environment, i.e., obtain input data, use these data to assess the actual situation, and decide about an appropriate action in that situation. By a fundamental design decision NEW TIES agents observe their environment through "seeing" the elementary features (color, shape, etc.) describing entities in the world. For instance, an agent can see an object with color 1, shape 2, and size 3. The fact that this object is a plant will be only "known" to the agent after internal processing of the input data. To reduce the dimensionality of the raw data, it is aggregated into concepts (for details see [6]). These concepts are stored in an agent's ontology and are used to provide a description of a given situation at a higher level than the original raw data.

Processing the incoming information produces a set of concept instances, stored in short-term memory. This short-term memory can be accessed by the controller during decision making to select the next action.

[2]EU FP6, FET Open, project number FP6-502386.

[3]All software used for this paper can be found on our website www.new-ties.org

1) Decision-making: The controller of NEW TIES agents is a decision tree, a so-called decision Q-tree (DQT)[4], where each branch in the tree ends with an action and thus can be regarded as a rule deciding on an action. Decision making amounts to traversing this tree. The way of tree traversal, hence the final decision, depends on 1) the actual situation (effect via test nodes), 2) the individual preferences of the agent (effect via bias nodes). If reinforcement learning is used in addition to evolutionary learning, then the mode of reinforcement learning (exploration or exploitation, see Section II-D) also affects the traversal of the DQT. In general, a DQT has four types of nodes: test nodes, general bias nodes, action bias nodes, and action nodes.

A **test** node evaluates a Boolean query based on concepts known to the agent, e.g., "Is there some plant ahead?" or "Is there an agent nearby?", and depending on the answer (Yes or No) the tree is further traversed through either of the two child nodes. A full path between the root node and a leaf (an action to be performed) represents a conjunction of statements that together provide a situation description in terms of the agents' concepts.

Bias nodes facilitate individual choices of the agents driven by their own preferences. Such preferences are expressed by the so-called biases and for reasons of convenience we distinguish two types of bias nodes: a general bias node and an action bias node. A general **bias node** can be anywhere in the DQT and it may have $n > 2$ child nodes. The choice among the child nodes, i.e., branches under a bias node, is probabilistic during tree traversal. The probabilities belonging to the child nodes are calculated from the biases belonging to the child nodes. Here we distinguish two types (sets) of biases: Genetic biases $\{g_1, \ldots, g_n\}$ and learned biases $\{l_1, \ldots, l_n\}$. Genetic biases are not changing during the lifetime of an agent and are inheritable, that is, are propagated to offspring agents via recombination and mutation. Learned biases are exactly the opposite, they can change during the lifetime of an agent (if individual learning and/or social learning are used) and are not inheritable, that is, are not passed to the offspring of the given agent. The two types of biases determine the choice probabilities together. In other words, our system allows for inherited and learned preferences as well (cf. the Nature and Nurture dichotomy). Formally, the probability p_i belonging to the i-th child node is calculated through

$$\overline{l}_i = max(0, l_i) \; / \; \sum_{j=0}^{n} max(0, l_j) \tag{1}$$

$$\overline{g}_i = max(0, g_i) \; / \; \sum_{j=0}^{n} max(0, g_j) \tag{2}$$

$$p_i = \overline{l}_i + \overline{l}_i \cdot \overline{g}_i \; / \; \sum_{j=0}^{n} (\overline{l}_j + \overline{l}_j \cdot \overline{g}_i) \tag{3}$$

[4]Q hints at the reinforcement learning that implements the individual learning mechanism in NEW TIES

An **action bias node** is similar to a general bias node except that it is always at the last but one level (above the leaves) and its set of children is the complete set of actions. In all types of nodes the genetic biases belong to the node and if an offspring individual inherits the node, it also inherits the corresponding genetic biases.

An **action node** is a leaf node that contains one action. It is carried out provided the environment allows it.

C. Adaptation mechanism 1: Evolution (EL)

The role of the adaptation mechanisms in our system is to change the controllers of the agents such that they exhibit appropriate behavior in a given scenario (challenge). In general we distinguish three adaptation mechanisms, evolutionary learning (EL), individual learning (IL), and social learning (SL), [6]; here we focus on the first two only. Their roles can be roughly distinguished as follows: EL is seeking good tree structures and genetic biases for the controllers, while IL (here: reinforcement learning that will referred to as RL) changes the learned biases. Since the probabilistic choices during decision making are influenced by both types of biases, IL can lead to learned behavior that supresses inherited behavior. It is one of the main questions in this paper, whether this is advantageous or not. This is similar to neural network research, in which evolution is providing the network structure and/or the initial weight matrices of the network and learning is further tuning the network matrices (e.g [16], for an overview see [20].

1) Selection operators: **Survivor selection** NEW TIES uses a truly environmental selection method, i.e., not based on any task related notion of (centrally calculated) fitness. Agents die if they run out of energy or reach the maximum age M.

Parent selection In principle, an agent can decide at any time to mate (subject to some constraints). By choosing the action MATE it selects itself as a would-be parent. To become a real parent, it needs to find and "convince" another agent. To do this, it sends a special message, a mate proposal, whose code and interpretation are hard-wired and the same for all agents. If the other agent accepts this mate proposal the two agents become real parents and produce a child. The environmental constraints on reproduction are that both agents are older than the MateAge threshold value and have to be within mating reach.

In order to give a newborn child a viable start, both parents donate one third of their current energy to the child at birth. Note, that this makes mating a possibly very costly action.

2) Variation operators: The genome consists of the controller, which is a tree.

Recombination DQTs are recombined by random subtree exchange as in standard Genetic Programming [1]. In both trees a random crossover point is chosen. The tree of the child is created by taking the tree of the mother and replacing the subtree residing under the crossover point by the subtree under the crossover point in the tree of the father. The genetic biases are simply copied together with the node that the bias belongs to.

Mutation In NEW TIES mutation complements recombination. Thus, while in some evolutionary algorithms, mutation can be used as a standalone operator to create offspring from a single parent, in NEW TIES it always follows recombination. We have genetic bias mutation and subtree mutation. Genetic bias mutation perturbs the given bias g by a random value drawn from a normal distribution $N(0, 0.5)$, enforcing lower/upper bounds by a simple boundary rule. Subtree mutation first chooses a random mutation point in the tree. The subtree from this point on is replaced by a random tree of equal size. If used each of these mutation operators is applied with a probability of five percent.

D. Adaptation mechanism 2: Reinforcement Learning (RL)

The goal of reinforcement learning is to maximize reward by optimizing a policy π. It is updated with SARSA reinforcement learning [17].

Executing a reinforcement learning step is intertwined with decision making. That is, the DQT is not traversed twice (once for making a decision, once for applying a learning step), but the agents learn from the in vivo decisions. Since RL has two modi, exploration and exploitation, this means that DQT traversal for decision making, is also performed under either of these flags. Which of these modi is actually used is determined probabilistically for each time the tree is traversed by the exploration-exploitation ratio ϵ. In experiments reported in this paper ϵ increases in equal step sizes from 0.5 to 0.95 in the first 2000 time steps of an agents lifetime. This means that agents rely more on their own preferences as they get older.

If exploration is chosen, then the biases are neglected and at each bias node and action bias node a uniform random choice is made among the possible sub-branches.

If exploitation is chosen, then genetic and learned biases are taken into account as shown in Formula 1-3. In practice we apply a slight variation to handle cases where learned biases are negative (which is possible by the specific working of RL). If there are only negative learned biases, then the highest (negative) learned value is chosen. If there are positive learned biases, than we ignore all negative values and apply the above formula.

Each node of the decision tree has a learned bias. They are initially set to zero and are changed when its leaf node is chosen as decision. All leaf nodes have been assigned an eligibility trace. Having performed an action in a state, the selected action node is rewarded through the SARSA (λ) rule:

$$\begin{aligned} &Q_{t+1}(s,a) = Q_t(s,a) + \alpha\delta_t e_t(s,a), \text{for all } s,\ a, \\ &\text{where } \delta_t = r_{t+1} + \gamma Q_t(s_{t+1}, a_{t+1}) - Q_t(s_t, a_t) \end{aligned} \tag{4}$$

Alpha is the learning rate and γ is the discount rate. The value r is the immediate reward for an action. For all actions, except for the mate action, the immediate reward is based on the energy gain caused by the action. As long as the agent is a child, when it is younger than its mate-age, the agent is not punished for choosing the action for mating, as if it is not present yet.

All traces of other action nodes are updated with the following rule: For all s, a

$$e_t(s,a) = \begin{cases} \gamma\lambda e_{t-1}(s,a) + 1 & \text{if } s = s_t \text{ and } a = a_t; \\ \gamma\lambda e_{t-1}(s,a) & \text{otherwise} \end{cases} \tag{5}$$

The reward r spreads upwards from the selected action node to the root using formula 6.

$$\begin{aligned} &cdr_{t+1} = cdr_t\gamma + r \\ &v_{t+1} = (c_t v_t)/(c_t + 1) + 1/(c_t + 1)cdr_{t+1} \\ &c_{t+1} = c_t + 1 \end{aligned} \tag{6}$$

Where cdr is the cumulated discounted rewards of the given inner node and v is the learned value of the given inner node. Finally, c counts the number of updates and rewards received by this inner node in the past.

III. Experimental Setup I: poisonous plants drain twice the energy as edible plant yield

The world we use for this challenge is a grid of 200x200. It is initialised with 500 agents, 8000 edible plants and 10000 poisonous plants. There is a maximum to the number of agents, because of the computational cost of an agent and the limited available resources. We use a soft limit, meaning that agents are unable to reproduce when the limit is exceeded, but if the number of agents is below it, a number of agents is allowed that together with the old population can temporarily exceed the limit. Agents and both types of plants are randomly distributed over the grid. We call our atomic time step a day and 365 days a year. The minimum mating age for agents 1000 days, only after this age they are able to reproduce. The maximum age for agents is 7300 days, when they reach this age they die regardless of their energy level. Hence, the maximum lifetime is about 7 to 8 times the mating age. Initially agents are assigned a random age between zero and one year. We use a ratio such that a poisonous plant drains twice the energy that an edible plant yields.

The initial controller of all agents is equal. In this controller some behaviours are pre-wired [5] like looking for food. However, the behaviour for eating the correct food type is not present. This can be acquired by changing the tree structure and/or tuning the biases of bias nodes and action bias nodes, although the probability that the latter succeeds is small in the tree-structure of the initial controller.

We use the adaptation mechanisms as described in Section II, except that tree-mutation is not used in the experiments of this paper.

A. Measures

In order to answer the research questions (Section I) we need measures for indicating the viability of the population, the performance of the population and providing insights into the evolutionary engine.

[5]Pre-wired is different from hard-wired, because in the pre-wired case the behaviour can be modified by the adaptation mechanisms

To measure the intensity of the population we use the population size. A successful run is a run in which the population size did not reach zero before the end (set to 30000 days here).

To measure the performance of the population, we use a measure monitoring how well agents eat, the total and average energy of the population and the total and average age. We named the measure how well agents eat g, which is

$$\frac{\text{correct (successful) eat actions}}{\text{total (successful) eat actions}} \quad (7)$$

To measure the performance of the evolutionary engine we will monitor the average number of mate-agreements of the agents.

IV. Results: Setup I

A. EL only and combination EL and RL with energy based rewards

To answer our research questions (Section I) we started research with runs with evolution only. The results in Figure 1(b) shows clearly that evolution only survives for approximately 1000 time steps and thus is not viable ((indicated by the striped line (EL)). The next step is to find out the effect of adding reinforcement learning to evolution on viability. The results, depicted in Figure 1(b) the dotted line (EL-RL (e)), show that energy based reinforcement learning is able to sustain for a longer period than evolution alone, but that it is not viable in the long run, because after 15000 time steps the population is almost decimated.

An important result is that the combination with energy based reinforcement learning is thus unable to make a population viable. A reason for this is that reinforcement learning is unlearning reproduction, since it costs energy and therefore receives negative rewards for reproduction. The reward for other actions, except the eat action, are also negative, but most often the reward for reproduction is the most negative, because it is costing 1/3 of the agent's energy. The EL-RL (e) curve in Figure 1(d) proves that reward based reinforcement learning is unlearning reproduction, because the total number of mate-agreements is going to zero. Furthermore, Figure 1(c) indicates that agents do not reproduce enough to sustain the population. Agents reproduce once every 3000 timesteps, while the average age in the population is only 1000.

There are two reasons why despite the negative reward agents still reproduce. Firstly, because they have to try to reproduce at least once to unlearn it. Secondly, during exploration agents can still choose the mate-agreement action, even when they unlearned it. The periodic behaviour of the curve is a side-effect of how the age of the initial population of agents is initialized and the miminal reproduction-age.

To find out whether populations can be viable within the current setup, we will have to carry out other experiments with different mate-rewards in order to encourage agents to reproduce.

B. Combination of evolution and reinforcement learning with a hardwired reward

The results in the previous section suggested that reproduction is unlearned or becomes so low such that the agent population is unable to sustain. The explanation for this is that agents receive a negative reward for reproduction in case of energy based reinforcement learning. To test this explanation we will introduce a special reward for reproduction. Its only role is to make mating actions attractive, regardless that it cost energy. Since the values are equal to or larger than zero, they can be regarded as kind of pleasure or orgasm.

Viability The main result is that with a hardwired positive reward the population is viable. Note that a reward of zero is working as a reward, because the other actions cost money and except for eating do not give any reward.

Performance of the population In general, the results show that the higher the reward for reproduction the better the performance. For instance, in the g measure this is expressed in the steepness of curve, where the curve for a reproduction reward of 1000000 is steeper than for a value of 10000. The average age (see Figure 1(c)) seems to contradict the general trend that a higher value is better, but these results are because the population size approaches its limits of zero and (artificial) maximum of 2000.

An interesting finding with respect to g is that in all different simulations, including that of evolution (EL) alone, in the first 1000 time steps it has approximately a similar value. This means that the combination of reinforcement learning and evolution is unable to learn the task during this period. This implies that the combination keeps agents alive that would be dead in the case of evolution alone. To find out how agents were able to survive, we analyzed the results by tracking the average number of actions agents were doing over some period for every type of action, similar to counting the number of mate-agreements for the evolutionary engine. From this data analysis it appears that agents are choosing often the NULL action. Agents thus learn to save their energy. This is suggesting a hiding effect: the combination keeps agents alive with a non-optimal strategy.

Evolutionary engine The intensity of the evolutionary engine is measured by the number of mate-agreements. The general trend is that the higher the mate-reward the higher the number of mate-agreements.

V. Experimental setup II: poisonous plants drain as much energy as edible plants yield

The results of the g measure of the previous experiments suggest that the individual learning preserves non-optimal individuals. This is known as the Hiding effect [10]. In order to test this we construct an environment in which evolution alone is able to make the agent population viable. In this environment everything is similar to the environment of setup I, except for the poisonousness. In this setup, we use a ratio so that a poisonous plant drains an equal amount of energy that an edible plant yields. We also use the same measures as used in experimental setup I.

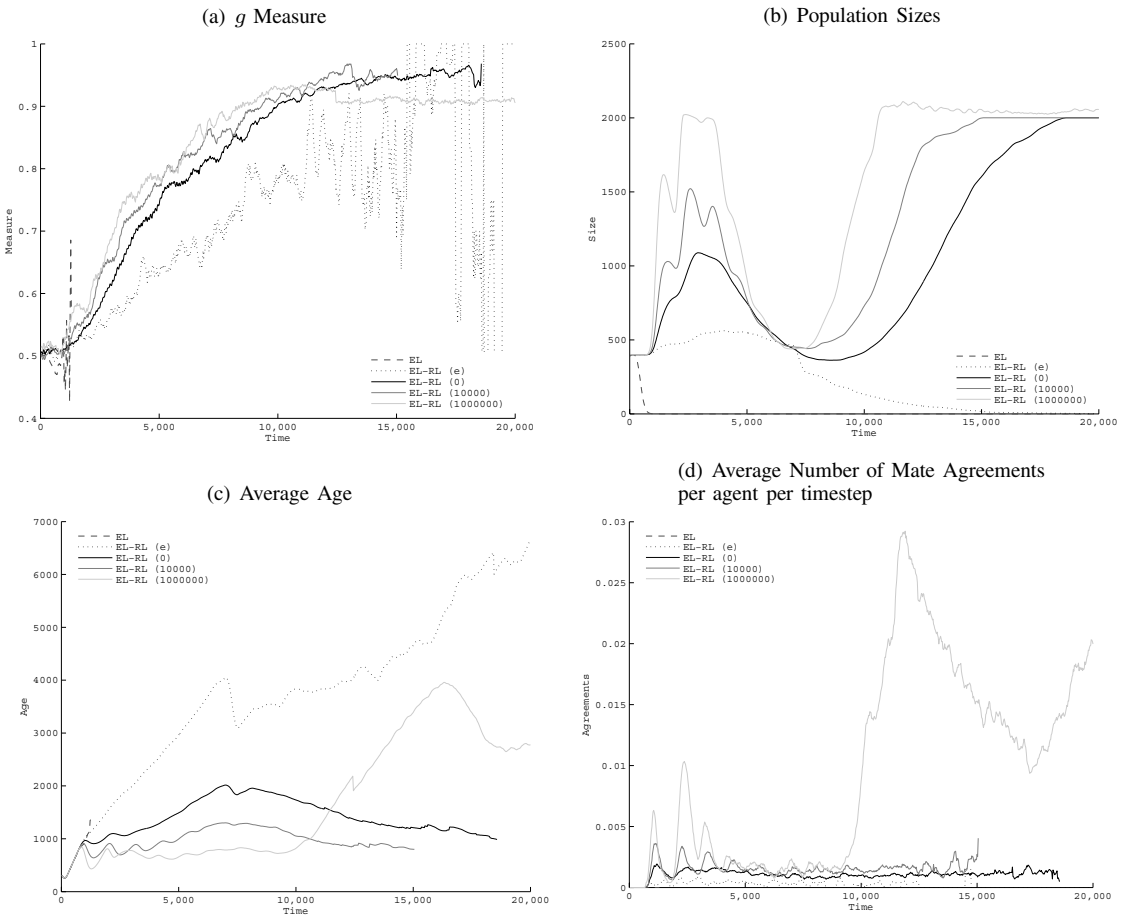

Fig. 1. Results Setup I

VI. Results: setup II

A. Viability

Both the combination and evolution only are viable in this setup. There is again strange behaviour close to the maximum. If the population size drops below the maximum (sometimes not visible in the graph, because it is an average of 10 runs), new agents are allowed to enter the population. Since there is no 'hard' limit, but a soft limit in which the mate-proposals and agreements fail, there can be overshoot over the maximum. In the section below we explain why this overshoot increases.

B. Performance of the population

The main result is that there is a clear hiding effect. The combination hides the ill-adapted nature of non-optimal agents that did not learn so fast the task as evolution only (Figure 2(a)) and therefore did not accumulate as much energy as evolution alone (Figure 2(f)).

That g and the total amount of energy are decreasing with evolution alone at some point in the simulation, has again to do with the maximum of the population size. This together with the fact that agents have accumulated an enormous amount of energy has changed the evolutionary pressure from eating (correctly) towards reproduction. This is evolving agents doing actions only involved with reproduction.

C. Evolutionary engine

As can be easily seen from 2(d), the average number of mate-agreements is much lower with the combination than evolution alone. This explains why evolution is impeded by reinforcement learning. However, this does not explain why there are fewer mate-agreements with the combination. The difference in number of mate-agreements already appears within the first 5000 time steps. In this period, there is no difference in g value or population size explaining the difference. This suggests that something else is going on. One possible explanation is that the combination is creating another type of agent that is reproducing less. This explanation is a very likely one, since we know that evolution is mainly focussed on reproduction, while the agents with reinforcement learning try to balance between both eating and reproduction, in order to maximize their rewards.

VII. Conclusions

Over the years there has been research towards combinations of learning and evolution, in particular to their costs, their benefits [7], [10], [12], [13] and factors that influence

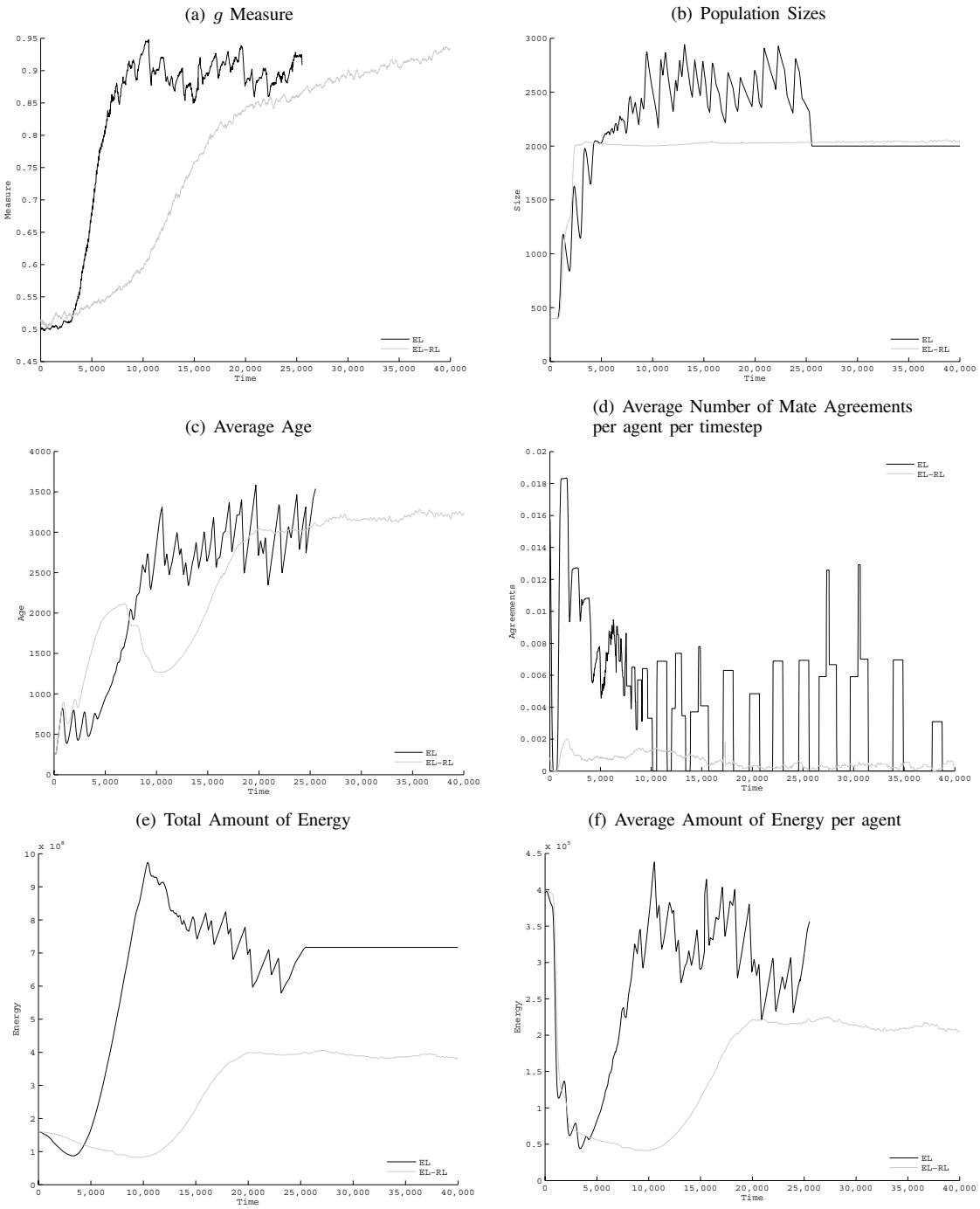

Fig. 2. Results Setup II

this relationship [10], [13]. In this paper we consider these issues. We perform our investigation in a new kind of system that features:

- Natural reproduction, where agents can decide autonomously if/when they reproduce (this implies shrinking/growing populations).
- An adaptable bias in the agents for the mating action (this implies the possibility of unlearning mating).

Our experiments show that in such systems learning can counteract evolution. To be concrete: Using reinforcement learning with the straightforward reward system based on energy, the agents will loose their interest in mating because of the high individual costs. Hereby the group benefits (maintaining the evolving population) are lost. This effect can be counteracted by introducing a specific reward for the mating action that gives positive feedback to the agents, regardless the related costs in terms of energy. One could argue that this

trick is known in nature, commonly called an orgasm. All in all, here we identify the reward for reproduction as another factor that influences the effect of learning on evolution in addition to the list of Mayley [10].

Regarding the effect of learning on the viability/performance of the population we observed that it can literally be a matter of life and death. In our first scenario, evolution only was not powerful enough to keep the population alive. Adding reinforcement learning changed this and we observed that populations survived and prospered until the end of the simulations. Simply put: Learning keeps the population alive. It does so by creating agents that minimize their energy spendings. This behaviour is non-optimal, in the sense that such agents do not learn to eat the correct plant type. As we mentioned in Section II-C reinforcement learning is able to modify the initial controller to eat correctly, but the probability is very low. So, instead of learning the optimal behaviour they learn the behaviour of minimizing energy.

We also found evidence of costs of learning. Learning causes a clear hiding effect as is clear from setup II, because it keeps non-optimal agents alive that do not accumulate much energy. In contrast, evolution alone optimizes by cutting much rougher the bad agents away, but with the risk that there is no population left. In a system allowing a changing population size this can be lethal. It is interesting that the hiding effect, keeping non-optimal agents alive, which usually has a negative connotation, has also a benefit from the perspective of system in which the population size is able to change.

Further research could show whether there is an optimal value for the reproduction reward (i.e., the extent of "pleasure" during mating). A good value would not frustrate evolution and still be able to make a population viable when needed. One possibility to find this optimal value is to put this value into the genome so that evolution can find this value by itself. This would also give evolution the possibility to tune itself.

Another possible subject is the study of an inverse system, where evolution can switch off learning. As it happens in our present system, learning has no explicit costs and it is enforced on the agents – unlike mating that is under their control. Studying such a system, or rather, a bidirectional one where both adaptation mechanisms can influence the working of the other one woule be highly interesting.

ACKNOWLEDGEMENTS

This work was financially supported by the grant FP6-502386 of the European Commission. The authors gratefully acknowledge the valuable contributions of Judit Belcsik, Akos Bontovics, and the other members of the NEW TIES consortium.

REFERENCES

[1] W. Banzhaf, P. Nordin, R.E. Keller, and F.D. Francone. *Genetic Programming: An Introduction*. Morgan-Kauffmann, 1998.

[2] R. Belew, J. McInerney, and N. Schraudolph. Evolving networks: Using the genetic algorithm with connectionist learning. In C.G. Langton et al., editor, *Proceedings of the Second Conference on Artificial Life*, Reading, MA, 1990. Addison-Wesley.

[3] R.K. Belew and M. Mitchell. *Adaptive Individuals in Evolving Populations: Models and Algorithms*. Addison-Wesley, 1996.

[4] T. Buresch, A.E. Eiben, G. Nitschke, and M.C. Schut. Effects of evolutionary and lifetime learning on minds and bodies in an artifical society. In D. Corne, Z. Michalewicz, B. McKay, G. Eiben, D. Fogel, C. Fonseca, G. Greenwood, G. Raidl, K.C. Tan, and A. Zalzala, editors, *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 1448–1454. IEEE Press, 2005.

[5] Dara Curran and Colm O'Riordan. Increasing population diversity through cultural learning. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 14(4):315–338, 2006.

[6] A.E. Eiben, A.R. Griffioen, and E.Haasdijk. Population-based adaptive systems: concepts, issues, and the platform new ties. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Proceedings European Conference on Complex Systems (ECCS)*, pages 487–509. Addison-Wesley, 2007.

[7] G.E. Hinton and S.J. Nowlan. How learning can guide evolution. *Complex Systems*, 1:495–502, 1987.

[8] A.J. Ijspeert, J. Hallam, and D. Willshaw. From lampreys to salamanders: evolving neural controllers for swimming and walking. In R. Pfeifer, B. Blumberg, J.-A. Meyer, and S.W. Wilson, editors, *From Animals to Animats, Proceedings of the Fifth International Conference of The Society for Adaptive Behavior (SAB98)*, pages 390–399. MIT Press, 1998.

[9] N. Krasnogor. *Studies on the Theory and Design Space of Memetic Algorithms*. PhD thesis, University of the West of England, 2002. Supervisor: Dr. J.E. Smith.

[10] G. Mayley. Landscapes, learning costs, and genetic assimilation: Modeling the evolution of motivation. *Evolutionary Computation*, 4(3):213–234, 1996.

[11] P. Moscato. A gentle introduction to memetic algorithms. In D. Corne, F. Glover, and M. Dorigo, editors, *New Ideas in Optimisation*, page Chapter 14. McGraw-Hill,, 1999.

[12] S. Munroe and A. Cangelosi. Learning and the evolution of language: the role of cultural variation and learning costs in the baldwin effect. *Artif. Life*, 8(4):311–339, 2002.

[13] S. Nolfi and D. Floreano. Learning and evolution. *Auton. Robots*, 7(1):89–113, 1999.

[14] S. Nolfi and D. Parisi. Learning to adapt to changing environments in evolving neural networks, 1995.

[15] E. Ruppin. Evolutionary autonomous agents: A neuroscience perspective. *Nature Reviews Neuroscience*, 3:132–141, 2002.

[16] K. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, 2002.

[17] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[18] P. M. Todd and G. F. Miller. Exploring adaptive agency ii: simulating the evolution of associative learning. In *Proceedings of the first international conference on simulation of adaptive behavior on From animals to animats*, pages 306–315, Cambridge, MA, USA, 1990. MIT Press.

[19] P. Turney, D. Whitley, and R. Anderson (eds.). Evolution, learning, and instinct: 100 years of the baldwin effect. *Special Issue of Evolutionary Computation*, (4(3)), 1996.

[20] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, May 1999.