Population-based Adaptive Systems: concepts, issues, and the platform NEW TIES

A.E. Eiben[1], A.R. Griffioen[2], and E. Haasdijk[3]

[1] Free University Amsterdam gusz@few.vu.nl
[2] Free University Amsterdam argriffi@few.vu.nl
[3] Tilburg University E.W.Haasdijk@uvt.nl

Summary. In this paper we introduce the notion of Population-based Adaptive Systems (PAS) with 3-tier adaptation by evolutionary, individual, and social learning (EL, IL, SL). We discuss some important aspects of using combinations of these adaptive mechanisms in general. We pay special attention to using natural reproduction in EL and the consequences of this for EL+IL combinations. We also present details on the NEW TIES system, partly as a running example, partly for its own sake as a concrete realisation of a 3-tier PAS that is available as a research platfrom for experimental studies.

1 Introduction: Population-based Adaptive Systems

The umbrella term that best describes the subject matter of this paper is *Population-based Adaptive Systems* (PAS). Such systems can be characterised by two essential features:

- They consist of a group of basic units that can perform actions, e.g., computation, communication, interaction, etc. By performing actions these units exhibit behaviour – individual behaviour per unit, as well as collective behaviour as the group all together.
- The ability to adapt at individual and/or group level. If the exhibited behaviour is generated through behavioural rules[4] inside the units, adaptation implies that these rules change. For instance, a change can take place inside a unit by replacing an existing rule by a new one, or a change can take place at population level by creating a new unit with a new set of rules.

There is a large diversity of PAS with quite different examples. For instance, a peer-to-peer computer system where each node (peer) is able to improve

[4] We do not mean a set of IF-THEN rules, but any representation, including, for instance neural nets, decision trees, etc.

its workings through experience, a genetic algorithm seeking an optimal solution to the travelling salesman problem, a group of learning robots collectively gathering red rocks on Mars, or a simulation of socio-economic processes by means of adaptive agent society. This kind of systems receives more and more interest over the last years with an increasing number of related papers. However, because of the lack of a common underlying framework, the terminology and the presentation of related problems and solutions show a large (application dependent) variation. This hinders the identification of similar concepts, problems, solutions, etc. over various publications. It is therefore possible that "the wheel is reinvented" by individual researchers. A common conceptual framework describing a large class of PAS forms a helpful stepping stone towards futher developments in the area.

The goal of this paper is to introduce the notion of Population-based Adaptive Systems and to discuss related concepts and research issues. We concentrate on a certain type of PAS, where adaptation takes place via three fundamental adaptation mechanisms: evolution, individual learning, and social learning. Part of our treatment will be in the context of a particular realisation of such PAS: the NEW TIES[5] [9] system which was created to investigate the interplay of the three adaptation mechanisms. Instantiating generic notions using NEW TIES as an example allows us better to illustrate certain matters and at the same time provide information on this research platform.

The main contributions of this paper can be summarised as follows.

- We present a conceptual framework that captures a wide class of adaptive systems.
- We identify research issues of general relevance in PAS.
- We introduce a software platform with which one can research such systems.

1.1 Population-based Adaptive Systems with 3-tier adaptation

In the remainder of this paper, we use an agent-based metaphor; the group of basic units is perceived as a population of agents who control their own behaviour – subject to environmental constraints, of course. We assume that each agent has a controller that takes observations of the environment and the agent's internal state as input and generates actions to be executed by the agent as output. Furthermore, we assume the following properties.

1. Changes at population level are possible. That is, it is possible to delete existing agents from and add new agents to the population. In common parlance, this amounts to birth and death in the system.

[5] New and Emerging World models Through Individual, Evolutionary and Social learning (NEW TIES), EU FP6 Project, `http://www.new-ties.org`

2. Changes at agent level are possible. That is, the controllers of the agents can change. (Clearly, this causes changes in the population, but that is just a secondary effect.)

We envision adaptation as the change of controllers in a population of agents and distinguish three fundamentally different adaptation mechanisms. Denoting the set of all possible controllers by C, we can perceive adaptation mechanisms in PAS as search algorithms traversing the space C in a volume oriented manner – maintaining a population of controllers $P = \{c_1, \ldots, c_n\} \subset C$ simultaneously. Adaptation then amounts to taking search steps: moving from the present set P of controllers to some new P'. We distinguish adaptation at population level (cf. property 1) and adaptation at agent level (cf. property 2). We call these *evolutionary learning* (EL) and *lifetime learning* (LL), respectively. Furthermore, we make an additional distinction between two types of lifetime learning. If an agent adapts its controller autonomously through a purely internal procedure then we have *individual learning*. If agents adapt their controllers by communicating controller information to each other and incorporating (good) pieces of information they receive, then we have *social learning*. Figure 1 illustrates this taxonomy and the corresponding terminology. In this framework we have 3-tier adaptation (by evolutionary learning, individual learning, and social learning), therefore we call such systems *triple-A PAS* (AAA–PAS). Note, that the branches of the tree in Figure 1 correspond to the terminology as used throughout this paper. Here, learning is a synonym for adaptation, and the three types are distinguished by the adjectives evolutionary, individual and social. The dotted boxes illustrate another option, where learning is an alternative for evolution, including what we call IL and SL.

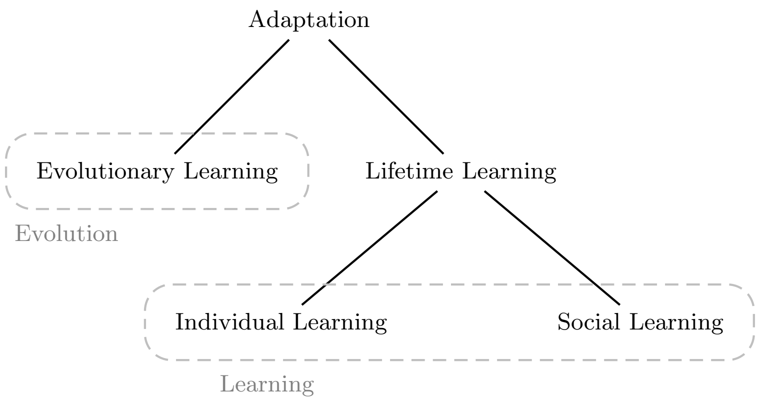

Fig. 1. Taxonomy of adaptation mechanisms in AAA–PAS

To explore the borders of this framework, let us consider a few examples. A genetic algorithm solving the Travelling Salesman Problem has birth and death, but the agents (individuals, candidate solutions) do not have a controller because they are not supposed to *do* anything except producing offspring. However, even reproduction is not something the individuals themselves actively control. They rather undergo it, arranged by an "oracle", the outer loop of the evolutionary algorithm procedure. Thus, in this example we have no contollers and only property 1), because changes at individual level are not possible here. The well-known artificial ant problem from genetic programming provides an example where agents do have controllers: each individual is a symbolic tree that represents the controller of an artificial ant and ants compete in "good behaviour" (how well they can walk the trail and collect all pieces of food). However, controllers, i.e., symbolic trees, cannot change during an individual's lifetime. Rather, a good individual collecting a lot of food using its superior controller will seed many offspring and disappear (in case of a generational replacement scheme) or perhaps survive into the next generation (in case of a steady-state scheme). As a third example, consider a single Web-agent serving one user by selecting news items every morning using a set of rules that is continuously improved through reinforcement learning. Here, the agents do have a controller (the rule set) that can change, but the population is a singleton and there is no death. Finally, let us consider the AEGIS artificial life system [5, 6] where a population of agents exists in an artificial habitat. The agents can move, eat, mate, fight, etc. as determined by their controllers and they undergo adaptation of their bodily characteristics (by evolution from generation to generation) and their controllers (by evolution from generation to generation or by learning during their lifetime). In this system, we have controllers and both porperties 1) and 2) are present.

In the following sections we discuss AAA–PAS. Throughout these sections we will present details regarding the NEW TIES system with a double purpose. On the one hand, these details constitute a (partial) description of the NEW TIES system, while on the other hand, they serve as a running example to illustrate matters. The paper is arranged as follows. In Section 2 we sketch the environment for our adaptive populations and the main features of NEW TIES agents. Next, in Section 3 we elaborate on the three main adaptation mechanisms separately. Thereafter, in Section 4 we discuss the relationships and interactions between these mechanisms and identify several research challenges raised in this context. The paper is concluded by a summary of the main issues and a discussion of how NEW TIES can be used as an experimental research platform. Note, that this paper does not present any experimental results – these will be presented in other publications.

2 The NEW TIES Environment and Agents

The NEW TIES 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. The world 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 (and in NEW TIES we do) distinguish body and brain properties of the agents. Body properties are defined by features like colour, shape, sex and weight. These mostly remain fixed during the lifetime of an agent. Some can change, e.g., its weight, but they do not undergo adaptation. We will focus on the adaptation of controllers because this poses the most challenging questions concerning the adaptation of controllers.

2.1 Decision making and agent controllers

Each time-step the agent performs the following three activities:

1. processing the incoming information,
2. deciding what action to take,
3. performing the chosen action.

1) Categorisation and conceptualisation Categorisation and conceptualisation reduce the dimensionality of the observation space (the raw data where attributes are the elementary attributes of all possible entities in the world) by mapping it into another space, where the attributes form the so-called concepts. Raw data are aggregated in two steps. First, raw data are aggregated to form categories that are further aggregated to concepts. The incoming information is processed by categorising the raw data-bundle of features. Each feature of objects in the world, like colour or shape, can be regarded as an axis in the feature space; a category is defined by a range of possible values within the whole range of a feature. For example, for the feature `colour` everything between $1, \ldots, 10$ could be considered `green`. Concepts are (more dimensional) entities composed from (one dimensional) categories.To define a concept not all features have to be used. For instance, plants could be `green` and `triangular` objects, agents could be `pink` and `circular`, while `pink, circular` and `big` could correspond to the concept of “enemy”. Concepts are stored in an agent's ontology and used to provide a characterisation of a given situation at a higher lavel than the original raw data.

2) Decision making NEW TIES agents use a special kind of decision tree – a Decision Q-Tree (DQT) – as a controller. A DQT contains two types of intermediary nodes. The first type is a test node: it represents a test, expressed in terms of the available concepts, that characterises a situation. For example, `my-energy-low, food-ahead` or `female-nearby` are possible test nodes with obvious interpretations. The other type is a bias node: it forms a branch-off based on the agent's preferences in the situation specified along the path on the DQT leading to this node. The probablities of choosing among the branches under a bias node are determined by relative weights of its sub-trees. The leaf-nodes of the tree represent the actions that the controller decides to take. Decision making amounts to traversing this tree.

3 Adaptation Mechanisms: AAA in Detail

In general, we envision adaptation as the change of controllers in a population of agents as outlined in Section 1. In NEW TIES, this amounts to modifying DQTs. In this section we discuss how the general EL-IL-SL trinity of adaptation is instantiated in NEW TIES. To begin with, we note that all three adaptation mechanisms work in the same search space – that of all possible DQTs.

3.1 Evolutionary Learning

In NEW TIES we deliberately adopt a non-Lamarckian notion of evolution: inheritable material cannot change during an agent's lifetime. This means that an agent created with a controller c seeds its descendants by exactly this controller c, regardless of any changes brought about by lifetime learning. In effect, the agents carry an unmodifiable copy of their initial controller – descendants are created using this copy.

The two pillars of evolution are selection and variation. Variation is realised by straighforward tree-crossover and tree-mutation operators, much like in genetic programming [2]. If we view adaptation as a search through the space of controllers, one elementary search step in this context amounts to combining two existing controllers c_1 and c_2 into a new one c_3.

An essential aspect of evolution in NEW TIES is that selection is not based on some objective function to be maximised [12, 13]. Survivor selection is strongly environmental: agents die when they run out of energy or reach the maximum age. As for parent selection, an agent can decide any time to mate (subject to some constraints). If the controller chooses the action `mate` then the agent selects itself as a would-be parent. To procreate, it needs to find and "convince" another agent of the opposite sex. To do this, it sends a special message, a mate proposal, whose code and interpretation are hard-wired and the same for all agents. Only if the other agent accepts this mate proposal do the two agents become actual parents and produce a child. To make the child viable, each parent donates a portion of its current energy.

The NEW TIES evolutionary system differs from usual evolutionary algorithms in a number of essential aspects.

1. Fitness is not an a priori utility measure that determines the number of offspring. One could say there is no notion of fitness at all, or that in NEW TIES fitness is a secondary, derived measure (an observable), such as the number of offspring – a truly Darwinian notion.
2. Reproduction is not orchestrated by some central authority. Individuals autonomously and asynchronously decide to mate.
3. Reproduction is detached from survivor selection. Newly produced individuals are added to the population without removing old ones. Likewise, an individual can die without being replaced by a new one. As a side-effect, there is no clear definition of a generation here.

These properties have two prominent consequences. The first is related to item 1. In the absence of an explicit objective function the selection probabilities (that embody the system bias for quality) must be based on indirect quality indicators. In general, the age and the energy level of agents can be used here: an agent that survives for a long period and/or has accumulated much energy must be well adapted, hence worthy of being reproduced. In this respect, PAS of this kind are closer to natural selection than, for instance, Genetic Algorithms where selection probabilities are calculated using an objective function.

The second effect is related to items 2) and 3) above. This kind of reproduction – *natural reproduction* – implies that the population size inherently changes over time. Users of such systems face a tough challenge: to calibrate the system so that unlimited population growth (explosion) or complete extinction (implosion) are avoided. In a particular system, such as NEW TIES or AEGIS, ad hoc solutions can work, based on balancing energy supply (number of plants, energy of plants, reproduction rate of plants) and energy consumption (costs of actions). From a general evolutionary point of view, population size can be controlled by tuning the selection mechanisms. For instance, the parameters specifying the minimum age or energy required for mating. At the moment there are no guidelines or design heuristics available to cope with this problem.

3.2 Individual Learning

In systems such as NEW TIES, supervised learning is impractical, because agents typically find themselves in an environment where the optimal (set of) action(s) is unknown. Unsupervised learning would waste any information present in the environment because it cannot be used as feedback for learning. Therefore, rewards based learning is the most appropriate option for individual learning.

In NEW TIES, reward based IL is implemented as reinforcement learning (RL) [10, 16]. RL changes the DQT by policy change. A *policy* of the agent is represented by the agent's DQT; any path in the DQT leads to an action – the result of the policy. A policy can be altered by changing the weights of the edges that govern the likelihood of taking a specific path. We use one of the variants of Q-learning, Sarsa.

In the current state of the NEW TIES agent the reward is based solely on energy. Indirect indicators of quality as discussed in 3.1 could also be used. Obviously, the components of any indicator have to be accessible to the agent, or the agent would not be able to use it for computing rewards. Mixtures of different measures can also be used to compute rewards. Such a mixture will probably be needed to tackle the problem described in Section 4.1: agents can unlearn reproduction if the reward is based only on energy.

An important challenge for reinforcement learning is the huge state-space in NEW TIES, even with the categorisation mechanism (Section 2) reducing dimensionality from the raw perceptual input to conceptual input. To illustrate, the conceptual state-space for the visual field is x^y, where x is the number of concepts and y is the number of grid cells of the agent's visual field. Given that we have at least 3 types of objects and that the visual field is 50 grid cells, the state space is obiously very large. Moreover, the state space is further extended by inputs for auditory, self-perception and reproduction stimuli. To cope with the size of this state-space, it is partitioned by the concepts present in the DQT's test nodes. For instance, a node that tests for `green agent` divides the state space into agents that are green on the one hand and all other coloured objects on the other. The test-node uses the conceptual input, but only tests for particular aspects of the environment. The success of reinforcement learning very much depends on whether the tests in the tree properly partition the state space – in other words, how relevant they are for a particular learning task or for survival.

3.3 Social Learning

Using SL, an agent modifies its controller by incorporating a piece of knowledge it receives from another agent. SL requires at least two agents a_1 and a_2 with controllers c_1 and c_2; one search step amounts to changing c_1 into c_1' (assuming that a_1 learns from a_2), where c_1' is some combination of c_1 and c_2.

Exchanging knowledge in this manner – through communication – implies a multi-faceted set of features and parameters that govern issues such as (social) networks of knowledge exchange, levels of trust and relative merit of knowledge, etc. In general, they concern:

- when and with whom to exchange knowledge;
- the selection of knowledge to send or elicit;

- when and how to accept offered knowledge.

Obviously, a general consideration when designing these features is including a bias for quality. In other words, at least some of the choices involved in importing a "knowledge nugget" from another agent must favor learning from a better agent. Similar to introducing a bias for quality in EL (cf. Section 3.1) the age and the energy level of agents can be used as quality indicators here. Apart from any specific quality-driven SL scenario, there is always qualitative pressure as described in Section 3.1: agents with poor controllers die sooner and therefore cannot participate in SL exchanges ("teach") as often as agents with good controllers.

Note that communication introduces a "social dimension"; an overlay network, in technical terms. The properties of this network depend on the implementation, but in general, the network will change over time (if only because some of the agents in it will die). In NEW TIES this is realised by a protocol similar to gossiping in peer-to-peer systems[8]. Every agent maintains a (fixed length) list of acquaintances – agents it has seen and talked to before. This list is updated with new observations (encounters with other agents) using a FIFO policy. The construction and maintenance of this social network can also be influenced by quality indicators of peers.

A knowledge nugget in our system is represented by a sub-DQT (extracted from the sender's controller). In the current implementation, this sub-DQT is included in the tree of the receiving agent by inserting –at some appropriate location in the DQT– a bias node that has two children: the foreign sub-DQT and the already existing native sub-DQT. These alternatives are weighted by newly defined biases based on the ratio between the sender and recipient's age and energy levels.

4 Relationships Between Adaptation Mechanisms

To position EL, IL, and SL it is helpful to consider them from the perspective of knowledge transfer, where knowledge is seen as (good) parts of the agent controllers. From this point of view, knowledge is transferred vertically by evolutionary learning, down along the line of descendants. (Recall the note from Section 3.1 that we do not have a clear notion of generations here, because agents residing on different levels of the family tree can live at the same time in the same population.) Individual learning, on the other hand, is a sink: in the absence of social learning, individually accumulated knowledge simply disappears when the agent carrying it dies. Social learning can alleviate this problem because it amounts to horizontal knowledge transfer, passing knowledge nuggets within the population. In this respect SL makes the population a knowledge reservoir, preventing (at least potentially) that knowledge must be rediscovered over and over again.

4.1 Evolutionary and Lifetime Learning

A marked distinction between EL and LL is that evolutionary operators do not change the controllers of agents during their lifetime, while LL operators obviously do. If evolution were the only adaptation mechanism, agents would die with the controller they were born with. Hence, evolutionary learning does not take place on an individual, but strictly on a population level. From this perspective, the death of an agent represents a contribution to the evolutionary learning process, because the population adapts with each death.[6] This is particularly not the case for IL, where the death of an agent disastrously destroys the results of the learning process.

In the current NEW TIES implementation, EL also differs from LL in the entity that initiates a learning step. IL and SL steps are initiated outside the influence of the agent's controller – by an oracle, or subconsciously, if you will. This is not the case for EL search steps, because the agent itself has to decide to reproduce by selecting the `mate` action or accepting a mate proposal. As a compelling consequence, agents can unlearn reproduction through LL because the individual reward for mating is negative: it costs energy without any mitigating personal benefit. To counteract such tendencies, one can introduce some specific reward for mating (orgasm), make mating a subconscious process or take population-level benefits into account in LL.[7] This represents an obvious opportunity for further research.

Concerning positive interactions between EL and LL, we refer to memetic algorithm research that has shown that combinations of EL and IL are particularly beneficial[11]. An interesting and promising interaction between EL and LL is described in [3]. This study finds that the chance of finding the optimal solution is much greater with LL and EL combined than with either in isolation.

4.2 Individual and Social Learning

As noted above, the non-Lamarckian nature of evolution in NEW TIES entails that knowledge that an agent acquires through IL will not affect inheritable material, and therefore be lost when that agent dies. By proliferating knowledge over the population of agents, SL preserves such knowledge pieces that would otherwise disappear. Thus, SL turns the population into a reservoir of (individually acquired) knowledge.

A system that combines IL and SL can be considered as having division of labour: IL generates novel knowledge nuggets and SL disseminates these. SL can also be seen as an accelerator making the system more efficient. Think, for instance, of agent a_1 learning x, agent a_2 learning y and a_1 and a_2 telling

[6] Supposedly changing for the better, cf. survival of the fittest.

[7] Taking a learning step in both IL and SL could also be made into a conscious action, in which case similar considerations would apply.

x and y to each other, rather than having to learn these knowledge pieces themselves. In general, efficiency improves if the costs of, respectively time needed for, learning through communication are lower for the agents than the costs/time of acquiring knowledge individually – an assumption that holds in a great many systems. As a net effect, combining SL and IL allows agents to possess knowledge regarding situations they never encountered themselves, acquired at greater speed and at lower costs. Such constellations have been shown to outperform either adaptation mechanism by itself, e.g., in [4].

4.3 Individual and Social Learning as Evolution

Recall from Section 3.3 that knowledge nuggets are sub-DQTs. Incorporating such sub-DQTs into an agent's controller amounts to an operation similar to crossover in GP. Similarly, one can see an analogy between a learning step in IL and a GP mutation operator: both change a given controller c into c'. From this perspective it is quite natural to see the combination of IL and SL as an evolutionary process. Such cultural evolution has been described in e.g., [4], [15] and [14].

The selection components for this cultural evolutionary system consist of the mechanisms regulating when two agents engage in sending/receiving knowledge pieces (parent selection) and the policies to accept and incorporate a received piece of knowledge (survivor selection).

It should be noted that this constitutes an evolutionary process quite different from the one described in Section 3.1. The most visible difference lies in the replacement strategies: in the LL-based evolutionary process, reproduction and survivor selection *are* coupled: a new controller, whether made by mutation or crossover, immediately replaces an existing one: its parent and the population size remains unaffected. Another difference is that here, a new controller is created by either crossover (SL step) *or* mutation (IL step), while in EL this happens by crossover *and* mutation (which occurs sequentially in the reproduction procedure). Furthermore, we can note that here we *do* have an explicit fitness measure, used in at least some parts of the system. For the parent selection component this is not necessarily the case. An agent can perform a mutation (do an IL step) regardless of the quality of its present controller c – making c the parent of the new c' – and the same holds for an agent a_1 deciding to talk to a_2 – selecting their controllers c_1 and c_2 as would-be parents. Regarding the survivor selection component we can distinguish two cases. In case of mutation (an IL step), survivor selection does not involve fitness either: the old c (the controller being improved by IL) is simply deselected and replaced by c' (the improved controller). However, if a new controller is created by crossover (an SL step), a utility function is used to determine the relative merit of the received knowledge when integrating it with the already known c_1 to create the new c'_1. This utility is related to the relative ages and energy levels of the two agents involved.

Considering IL and SL in this light raises two prominent research questions. First, how does existing evolutionary computing knowledge, e.g., regarding variation, selection and their balance, translate into this context? Second, how do the two evolutionary processes, EL on the one hand, SL and IL on the other, interact in one system?

5 Discussion

So, what is this paper about: PAS, AAA–PAS, or NEW TIES? The main subject matter is AAA–PAS, including special cases not featuring all three adaptation mechanisms. For instance, a genetic algorithm is an A**–PAS, a group of learning software agents is a *A*–PAS, a Web-based system where (user made) ontologies at different sites are merged into one common ontology by gossiping is a **A–PAS [1], a memetic algorithm is an AA*–PAS, and a typical simulation in NEW TIES is a genuine AAA–PAS. Regarding the evolutionary component, we concentrate on systems with natural reproduction (not to be confused with natural selection). While this type of system is hardly considered in mainstream evolutionary computing, it does form an important class, for there are important (types of) applications, where technical constraints or modelling priorities require this. For instance in peer-to-peer evolutionary systems, or multi-agent simulations in a biological or social context.

Most of the technical details we introduced are merely illustrative in the sense that they do not restrict the generality of our discussions. Using trees to represent agent controllers one such detail. Our line of thought about variation operators in EL and merge operators in SL can be repeated for other data structures. A similar argument holds for the categorisation and conceptualisation mechanism to pre-process sensory input of the agents. The general point here is dimensionality reduction. This is critical when using reinforcement learning algorithms, because they scale very badly with the size of the state-space, but this aspect is likely to occur with most learning methods.

One of the main contributions of this paper is the definition of a system where three different adaptation mechanisms, EL, IL, and SL, work simultaneously, yet clearly distinctly. The separation of the learning mechanisms is based on a differentiation between inheritable and learnable agent characteristics.[8] Designating agent characteristics as inheritable or learnable is one of the major design decision behind AAA–PAS. Inheritable properties can undergo evolution through appropriate variation operators and environmental selection, learnable properties can undergo lifetime learning through individual and social learning. Because of the clear separation between EL, IL, and SL, particular adaptation mechanisms can be switched on and off independently, allowing research on their effects separately or in various combinations.

[8] In the present NEW TIES system these are the same, but our considerations are still valid if this is not the case.

Thus, we can gain insights in their effects on each other and on the adapting population. Research in this area offers great benefits by the high potential of "fully powered" adaptive systems. In this paper we observed that in general:

- SL can act as an accelerator for IL in each agent and can preserve the IL-discovered knowledge nuggets for the population that would otherwise be lost after the death of the individual that learned them.
- The combination of IL and SL can be seen as an evolutionary system, creating an opportunity to use existing knowledge in evolutionary computing when designing such combined systems.

The specific choices we made for evolutionary learning in NEW TIES were reflected in our treatment of EL here. In particular, we focused on systems with natural reproduction, cf. Section 3.1. In many applications, e.g., ALife, social simulations, peer-to-peer systems, this is the obvious choice of reproduction scenario, so we can safely state that the subset of AAA–PAS with natural reproduction is large and interesting. Considering such systems we observed that:

- In an evolutionary process relying on natural reproduction the population size is inherently volatile. This represents a hard challenge for designers/users of such systems: to design (selection) mechanisms that prevent explosion and implosion of the population.
- While, in general, combining LL and EL is a powerful combination (cf. memetic algorithms), in PAS with natural reproduction LL can counteract EL by unlearning mating.

Apart from this conceptual framework, the NEW TIES system is available as a platform for experimental research in AAA–PAS with natural reproduction. The system is written in Java; the source-code and information on using and developing NEW TIES applications can be found at `http://www.new-ties.org`.

The present software is optimised for agent-based simulations with a biological, sociological flavour, since this is the application area of the NEW TIES project. Example applications include a world with edible and poisonous plants, where agents need to learn not to eat the latter, a world with two types of plants, where agents need to learn to trade them, and a world where plant location is seasonal and agents need to develop herding behaviour. In all such applications we are mainly interested in the emergent phenomena, particularly in emerging behaviour and emerging structures, such as the controllers of the agents (world models) or the social network generated by SL. It is essential for our approach that the experimenters can influence system properties only indirectly, via the adaptation mechanisms. That is, given a demanding world where agents only survive if they adapt to the particular challenges of that world, the experimenter's task is to engineer an appropriate mix of the adaptation mechanisms and these mechanisms will generate the required emergent behaviours and structures. It is this aspect that makes understanding the trichotomy of EL, IL, and SL crucial for NEW TIES and other AAA–PAS.

Acknowledgements

The authors gratefully acknowledge the helpful contributions of A. Lőrincz, M. Schut, S. Smit and P. Vogt and the financial support of the European Commission by the grant FP6-502386.

References

1. K. Aberer, P. Cudré-Mauroux, and M. Hauswirth. The Chatty Web: Emergent Semantics Through Gossiping. In *Proceedings of the 12th International World Wide Web Conference*, 2003.
2. W. Banzhaf, P. Nordin, R.E. Keller, and F.D. Francone. *Genetic Programming: An Introduction*. Morgan-Kauffmann, 1998.
3. M.L. Best. How Culture Can Guide Evolution: An Inquiry into Gene/Meme Enhancement and Opposition. *Adaptive Behavior*, 7(3-4):289, 1999.
4. L. Bull, M. Studley, A. Bagnall, and I. Whittley. Learning Classifier System Ensembles With Rule-Sharing. *IEEE Transactions on Evolutionary Computation (in press)*, 2007.
5. 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.
6. A. E. Eiben, D. Elia, and J. I. van Hemert. Population dynamics and emerging mental features in AEGIS. In W. Banzhaf, J. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M. Jakiela, and R.E. Smith, editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 2, pages 1257–1264, Orlando, Florida, USA, 13-17 July 1999. Morgan Kaufmann.
7. A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing*. Springer, 2003.
8. P. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Massoulié. Epidemic Information Dissemination in Distributed Systems. *IEEE Computer*, 37(5):60–67, May 2004.
9. N. Gilbert, M. den Besten, A. Bontovics, B.G.W. Craenen, F. Divina, A.E. Eiben, A.R Griffioen, G. Hévízi, A. Lörincz, B. Paechter, S. Schuster, M.C. Schut, C.K. Tzolov, P. Vogt, and L. Yang. Emerging artificial societies through learning. *Journal of Artificial Societies and Social Simulation*, 9(2), 2006.
10. L.P. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.
11. 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.
12. F. Menczer and R. Belew. From complex environments to complex behaviors. *Adaptive Behavior*, (4):317–363, 1996.

13. Melanie Mitchell and Stephanie Forrest. Genetic algorithms and artificial life. *Artificial Life*, 1(3):267–289, 1994.
14. P.J. Richerson and R. Boyd. *Not by genes alone: how culture transformed human evolution*. University of Chicago Press, 2005.
15. R.E. Smith, C. Bonacina, P. Kearney, and W. Merlat. Embodiment of Evolutionary Computation in General Agents. *Evolutionary Computation*, 8(4):475–493, 2000.
16. R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.