UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

NURTURING PROMOTES THE EVOLUTION OF LEARNING IN CHANGING ENVIRONMENTS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

SYED NAVEED HUSSAIN SHAH
Norman, Oklahoma
2015

NURTURING PROMOTES THE EVOLUTION OF LEARNING IN CHANGING ENVIRONMENTS

A DISSERTATION APPROVED FOR THE
SCHOOL OF COMPUTER SCIENCE

BY

Dr. Dean F. Hougen, Chair

Dr. Ingo B. Schlupp

Dr. Andrew H. Fagg

Dr. John K. Antonio

Dr. Rickey P. Thomas

Thank to Almighty ALLAH Subhanahu Wa Ta'ala, our creator
and Holy Prophet Muhammad (P.B.U.H), mercy to mankind.

Acknowledgements

I would like to thank numerous people for their support during my dissertation.

Sincere thank to my research adviser Dr. Dean F. Hougen for his excellent supervision, guidance, encouragement, deligent suggestions, and making my doctoral research a wonderful experience. To me, he is an institute of learning. I thank him for awakening my interest in the various areas of machine learning and for his insightful discussions with me over the course of my research. I also thank him for providing me such a detailed feedback on improving this document. Further, I thank him for being a great support especially during my rough times. I could not have asked for a better adviser.

My doctoral committee including Dr. Antonio and Dr. Andrew Fagg for their useful insights to improve the quality of this work, Dr. Rick Thomas for his encouragement and useful feedback, and Dr. Ingo Schlupp together with Dr. Dean F. Hougen for providing the foundations of this work.

The University of Oklahoma (OU), Dr. Sridhar Radhakrishnan, and Dr. Abousleiman Younane for supporting my doctoral studies with continuous graduate assistantship positions. I also thank Dr. Sridhar Radhakrishnan for his guidance, encouragement, and trust in my abilities, during the past six years.

Big thanks to OU Supercomputing Center for Education and Research (OSCER) and especially Dr. Henry Neeman for being so generous in letting me collect the data for the experimental work of this dissertation. I want to thank Patrick Calhoun for making it seamless to run source code, related to this dissertation, on OSCER machines. I also thank Joshua Alexander and rest of the staff for their cooperation. I also want to thank Dr. Henry Neeman for being a very kind and supportive supervisor during my teaching assistantship.

All Robotics, Evolution, Adaptation, and Learning (REAL) Laboratory members especially Mark Woehrer, Benjamin Carlson, Juhee Suh, Ryan Phillips, Seyed Ali Hajimirza, and Nicholas Pequeno for listening to my talks and their extremely valuable feedback over the years.

A very especial gratitude to my late father, Syed Maqbool Shah and my mother, Taj Sultana for their unconditional love, support, and for being such great parents. With this dissertation, I fulfill the dream of my late father. I would also like to thank my brothers for supporting my education throughout my educational career. I acknowledge my sisters, Syeda Shagufta Maqbool, Syeda Kanwal Maqbool, and Syeda Kokab Maqbool to be my inspiration and for teaching me the basics, 20 years ago, of what turned into a Ph.D. I owe them a lot for my success.

A very heartening thanks to my wife Asma Farooq for being a wonderful and loving lifelong partner, her immense patience during my doctoral studies, and intelligent suggestions during my research and dissertation writing. I also thank my nephew Syed Adil Hussain Shah for his kind support during my Ph.D.

My friend Mohammad Naved Khan for his unconditional support and friendship especially during my rough times.

My fellow Ph.D. students Dr. Thomas Palmer for being a great friend, Dr. Amlan Chatterjee, Dr. Khondker Hassan, Asif M. Adnan, Tristin Forbus, and Dr. Maryam Nafari for providing a wonderful company during this journey for the past six years.

Chyrl Yerdon, Barbara Bledsoe, Emily Pierce, Virginie Perez Woods, Jaime Hoots, and Sara Vaughan for their amazing administrative support.

Mamta Yadav and Yiming Xu besides other Computer Science Graduate Student Association (CSGSA) members for their support in various activities in the department of Computer Science.

Irfan Nazir for inspiring me to the field of computer science and in particular computer programming and artificial intelligence. To me, he is an inspiring personality for countless reasons. Other mentors Kashif Bilal, Naeem Sakhawat, Azeem Khan, Dr. Waqas Anwar, Dr. Sheikh Muhammad Azam, Mubasshar Ahmad, Rashid Mehmood, and Khalid Shah for laying the foundations of my educational career. Also I want to say a very special thank to all my friends (Ameer Abbas, Ahsan Shehzad, Umair Javed Sundhu, Farrukh Tareen, Abdul Waseh, Babar Aziz, Saad Masood, Sardar Usman, and Mohammad Shoaib) in my undergraduate studies for their discussions on various areas of Computer Science that triggered my interest in this amazing field of study. I owe all these people a lot.

Contents

List of Tables

List of Figures

List of Algorithms

Abstract

An agent may interact with its environment and learn complex tasks based on evaluative feedback through a process known as reinforcement learning. Reinforcement learning requires exploration of unfamiliar situations, which necessarily involves unknown and potentially dangerous or costly outcomes. Supervising agents in these situations can be seen as a type of nurturing and requires an investment of time usually by humans. Nurturing, one individual investing in the development of another individual with which it has an ongoing relationship, is widely seen in the biological world, often with parents nurturing their offspring. There are many types of nurturing, including helping an individual to carry out a task by doing part of the task for it. In artificial intelligence, nurturing can be seen as an opportunity to develop both better machine learning algorithms and robots that assist or supervise other robots. Although the area of nurturing robotics is at a very early stage, the hope is that this approach can result in more sophisticated learning systems. This dissertation demonstrates the effectiveness of nurturing through experiments involving the evolution of the parameters of a reinforcement learning algorithm that is capable of finding good policies in a changing environment in which the agent must learn an episodic task in which there is discrete input with perceptual aliasing, continuous output, and delayed reward. The results show that nurturing is capable of promoting the evolution of learning in such environments.

Chapter 1

Introduction

This research aims at contributing to the novel yet important research area of robot-to-robot (R2R) nurturing. The machine learning (ML) research community has so far not invested much in this area, with a few exceptions (Leonce et al., 2012; Eskridge & Hougen, 2012; Woehrer et al., 2012). As originally pointed out in a call to research community, nurturing plays a central role in the development of biological individuals (Woehrer et al., 2012). Similar behaviors in our robots might play an important role in their development but we need to investigate these areas of ML research.

The larger research agenda to which this dissertation contributes is to evolve nurturing robots, use R2R nurturing to promote the evolution of learning, then have the learning robots learn to be better nurturers. If nurturing promotes the evolution of learning, and learning enables greater nurturing, then this will start a virtuous cycle that eventually results in intelligent systems. The end goal of this research is to make substantially better ML systems for autonomous robots that can adapt to changing environments. Toward this end goal, this research contributes by demonstrating that nurturing promotes the evolution of learning in changing environments.

1.1 Motivation

The main objectives for this research are tied to the integration of AI and evolutionary biology. We hope to contribute to the evolution of machine learning and better understanding of animal behavior. For AI, our goal is to develop substantially better machine learning systems for autonomous robots and to make extensive robot learning practical through robot oversight of robot learning. We also hope to contribute to biology a better understanding of how nurturing and learning evolved in nature[1]. Both of these goals are supported by investigating the virtuous cycle of nurturing and learning.

1.1.1 Virtuous Cycle of Nurturing

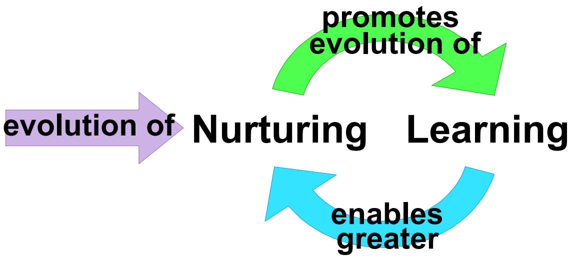

Figure 1.1: Virtuous Cycle

A virtuous cycle in which the evolution of nurturing enables greater evolution of learning and the evolution of learning enables greater nurturing (see footnote for credit of this figure).

Figure 1.1 presents the big picture of our research agenda at the Robotics, Evolution, Adaptation, and Learning Laboratory (REAL Lab) at the University of Oklahoma. The virtuous cycle is a positive (self-reinforcing) feedback loop with desirable outcomes (the evolution of nurturing and the evolution of learning) (Woehrer et al., 2012). *Nurturing* is the contribution of resources by one individual to the development of another individual with which it has an ongoing relationship (Leonce et al., 2012). Woehrer et al. (2012) define nurturing as “the contribution of time, energy, or other resources by one individual to the expected physical, mental, social, or other development of another individual with which it has an ongoing relationship”. Nur-

[1] Credit for the background work and establishing the vision goes to Dr. Dean F. Hougen and Dr. Ingo B. Schlupp

turing can be of various types such as safe exploration and social learning (Eskridge & Hougen, 2012). In this dissertation, nurturing is operationalized as helping an individual by completing a part of its task. Learning, on the other hand, is acquiring knowledge or skills through experience. There can be various types of learning such as learning to avoid predators and learning to find better food sources over time. In this dissertation, learning is operationalized as maximizing reward (favorable environmental feedback) in an initially unknown or partially known environment. In biology, nurturing (parental investment in offspring) and learning are studied separately. However, in machine learning, we may integrate the two concepts and can possibly answer the questions such as, "how does nurturing impact the evolution of learning in living organisms?" In order to accomplish that, robots (artificial representations of living organisms) can be used as suitable entities. This virtuous cycle depicts the nurturing loop which starts by claiming that nurturing can be evolved in Robot-to-Robot (R2R) collaborations. Then this evolved nurturing can be further seen as promoting the evolution of learning which in turn enables greater learning, i.e., learning to nurture.

1.1.2 Robot-to-Robot (R2R) Nurturing

In R2R collaborations, nurturing means one robot providing for the development of another robot. The robot that nurtures the other robot can be called *nurturer* while the robot being nurtured can be termed as *nurturee*. This caring (development) could mean that the nurturer provides resources to the nurturee, protects it from hazards, or helps it to learn about its environment. Nurturing shares with altruism and cooperation the idea of one individual contributing to another (Woehrer et al., 2012). However, in both of these concepts, it is not essential to have an ongoing relationship between the individuals. Thus, nurturing is related to these concepts; however, it is distinct. In R2R nurturing, both the contributing and the beneficiary individuals are robots. There has been limited research in the area of R2R nurturing except in a few related directions such as a robot imitating another robot or a robot demonstrating for another robot (Nicolescu & Mataric, 2001; Demiris & Hayes, 2002). Note that some of these examples may not be considered nurturing as, for instance, the robot imitating the other robot might be getting a benefit by imitating; however, the robot being imitated may be performing the task for its own benefit. R2R nurturing is also related to developmental robotics including both morphogenetic and epigenetic robotics for the development of their physical and mental capacities (Lungarella & Metta, 2003; Zlatev & Balkenius, 2001; Berthouze & Metta, 2005; Jin & Meng, 2011). As discussed previously, through nurturing a robot may invest in the development of another robot. This development can be in terms of either physical or mental capabilities of the nurturee.

1.1.3 Nurturing and Self-Care

Previous work on the evolution of nurturing shows that it is possible for a robot to gain resources for itself by carrying out a complete task. This is referred as *self-care*. It is also possible for another robot in the environment to do a part of the task and thereby simplify the task remaining for the first robot. The first robot then only performs the remaining part of the task. This kind of task assistance is considered nurturing where the second robot is nurturing the first robot. Accordingly, these two successful strategies are termed self care and nurturing (Leonce et al., 2012).

In this dissertation, there are two basic types of environments. In the first, a robot needs to carry out a full task in order to gain resources. In the second, a part of the task has already been accomplished for the robot and it only needs to carry out the remainder of the task to gain resources. These two types of environments can be thought of as environmental niches to which an organism could evolve adaptations. Accordingly, these two environments are here termed the *self-care* niche (or *no-nurturing niche*) and the *nurturing niche*. Correspondingly, we will refer to individuals evolved in the self-care niche as *self-care individuals* and to individuals evolved in the nurturing niche as *nurtured individuals*. We use the terms no-nurturing and self-care interchangeably in this dissertation.

1.1.4 Evolution of Nurturing

In biological organisms we see numerous examples of the evolution of nurturing in species. In various parent-child relationships in nature, parents often risk their own lives to nurture their offspring. The offspring then become parents and nurture their children and the cycle continues. During this generational cycle, we often see individuals and their offspring gradually become better in their nurturing capabilities over time. In R2R nurturing, we expect individual robots to become better at nurturing over time and then pass on their successful genes to offspring, the same way biological organism do. The concept of the evolution of nurturing is vital before we start looking into the idea that, nurturing promotes the evolution of learning. The co-evolution of nurturing and learning is an important link between the idea of the evolution of nurturing leading to the evolution of learning. Previous experiments show that nurturing can be evolved in R2R parent/child collaborations (Leonce et al., 2012). Our work, at the REAL Lab, shows that the evolution of nurturing in siblings, and between grandparents, parents, and children are all possible in R2R setups (forthcoming publications). All these results indicate that nurturing is evolved naturally in related individuals.

1.1.5 Evolution of Learning

For instincts to be effective, the environment must not change much between generations. This is because evolution adapts organisms to a particular environment and if that environment is changed in crucial ways, the organisms might not perform well in that environment any more. Conversely, if the environment is stationary during the lifetime of an individual, a good instinctive individual may be capable of performing well during its life. This is because the instinctive knowledge passed from parents to offspring over generations is sufficient for individuals to survive and gain high fitness during their lifetimes. However, if the world around the individuals is changing, they must learn in order to thrive. Depending on the rate of change in the environment, individuals have to adjust their pace of learning according to the changes happening around them. If the environment is changing slowly, an individual that learns slowly may survive and perform well during its lifetime. However, if the rate of change in the environment is high, the individual must learn quickly enough to keep pace with the changes. If, however, the individual is not able to learn quickly, it may die off and thus have no chance of passing on its genes to the next generation. We claim that nurturing can help individuals cope with rapidly changing environments. If an offspring is nurtured such that it gets enough time and resources to learn, it may survive in a rapidly changing environment and pass its genes to the next generation.

Further, the evolution of learning over generations would enable individuals in the later generations to evolve better learning mechanisms, i.e., we will observe more and better learning within the nurturing niche than within the self-care niche, which is the work proposed in this dissertation. It is important to note that, while it is interesting to see what will happen if those evolved learning mechanisms are tested in different niches, that is not a primary focus in this research.

Therefore, the central claim of this dissertation is that nurturing promotes the evolution of learning in changing environments. The main theme behind this research is to address this claim which is an important link in the virtuous cycle between the evolution of nurturing and learning to be a better nurturer.

1.1.6 Evolution of Learning and Instincts

Learning is an essential trait of all the intelligent systems that need to be adaptive to changing environments. Further, the evolution of learning provides more robust and scalable solutions to the learning problems found in nature. The evolution of learning is an important dimension in neuroevolutionary research. The evolution of instincts, on the other hand, play an important role in survival of the fittest in stationary environments. However, by combining the two together in a non-stationary environment, if the quality of learning is sufficient that it outperforms purely the instinctive individuals, then learning is said to be evolved when instincts are possible in changing environments. If this happens more often in the case of nurturing than in the case of self-care, then we can claim that nurturing is beneficial to the evolution of learning when instincts are possible. This is an important concept as it demonstrates how generally applicable and important nurturing is to the evolution of learning.

1.2 Contents of the Dissertation

In this dissertation, the usefulness of nurturing is investigated in the course of the evolution of learning. The hypothesis is that nurturing promotes the evolution of learning in changing environments. To investigate the hypothesis, nurturing is compared to a no-nurturing (self-care) case in various categories/sub-hypotheses. In order to accomplish that, an environment is designed in which learning is advantageous. An agent in the form of a simulated robot is used to demonstrate learning in this environment. This robot is controlled by an artificial neural network that uses reinforcement learning with eligibility traces to learn the dynamics of the environment in a terminal reward, episodic task scenario. Once the learning is established, an experiment is designed to evolve learning parameters using genetic algorithms. Further, an extension of this experiment by introducing the evolution of instincts together with learning is performed. The results related to all the sub-hypotheses are compared and contrasted.

As in the evolution of nurturing part of the virtuous cycle, Leonce et al. (2012) use a simple light switching experimental setup based on Floreano & Urzelai (2000), it makes sense to continue with that setup for the sake of combining the evolution of nurturing with the evolution of learning later. In the current light switching arena, a robot starts from the center of the arena and its goal is to get to the best rewarding light via a (near) optimal path. Once the robot reaches the light source, the task is considered complete for that trial (episode) and it receives the reward. The trial may also end if time expires and in that case the robot receives a penalty. At the end of the trial, the reinforcement learning algorithm uses the collected reward or penalty to give evaluative feedback to the ANN. The algorithm is described in detail in Chapter 3. It makes sense to use reinforcement learning for the proposed problem, as the only feedback from the environment that the algorithm is going to receive is the reward or penalty. The ANN uses discrete input and continuous valued output units. It calculates values stochastically, sampling from a normal distribution for each synapse, and uses on-policy learning to update its synapses.

The rest of this dissertation is organized as follows: **Chapter 2** discusses background material on artificial neural networks, evolutionary computation, reinforcement learning, nurturing robotics, nurturing niche construction, reward shaping and chaining, nurturing as task simplification, and evolution of learning. Next, **Chapter 3** introduces a class of ANNs and proposes a reinforcement learning algorithm, then validates the algorithm and its implementation. The last section in this chapter discusses the results of the implementation. Next is **Chapter 4** on experimental design, which introduces the hypotheses list, translates each hypothesis into experiments, and discusses the evolution of learning and the evolution of learning and instincts, covering design and implementation. **Chapter 5** discusses the evaluation criteria for the results, all the data collected from the proposed experiments in the previous chapter, and their statistical analysis. This chapter also briefly discusses the best evolved individuals in the cases of nurturing and self-care and related typical results in both cases. **Chapter 6** is a discussion chapter where the best individuals found in the nurturing and self-care niches are highlighted. This is followed by a discussion on some of the results that do not exactly fit the proposed objective criteria; however, these arguments further support the hypotheses. Next is **Chapter 7** that concludes contributions of this research to our larger research agenda, the virtuous cycle, and to ML and R2R nurturing research in general. **Chapter 8** is on future work and highlights a few potential directions, such as moving forward in the virtuous cycle (Figure 1.1) to use the evolved learning algorithms in the parents to become better nurturers, in which the work from this dissertation can be expanded and built on further. At the end is **Appendix A** that includes detailed tables of average reward data collected from the evolved individuals for analysis on performance continuum results. This is followed by **Appendix B** that shows the acronym list.

Chapter 2

Background and Related Work

This chapter introduces and discusses machine learning sub-domains including artificial neural networks; evolutionary computation, particularly genetic algorithms; and learning paradigms, in particular reinforcement learning with a focus on temporal and structural credit assignment. It also discusses related work in the areas of nurturing robots, niche construction, reward shaping and chaining, and the evolution of learning.

2.1 Artificial Neural Networks

Artificial neural networks (ANN) can be viewed as parallel and distributed processing systems which consist of a huge number of simple and massively parallel connected processors (Jain et al., 1996). ANNs can be considered models to solve computational problems. These models are used to approximate solutions to various computational problems related to recognition, prediction, optimization, associative memory, and control (Jain et al., 1996). There are several types of neural networks used for approximation, one of them is *feed-forward neural networks* in which data from a set of input units is operated on as it passes through the network to its output units, with data always being fed in the same direction through the network with no lateral or backward connections. These artificial neural structures generally consist of neurons interconnected using synapses. The strengths of these connections, known as *synaptic weights*, are used to store knowledge (I. W. Sandberg & Haykin, 2001).

Harvey (1997) discusses an interesting approach to cognition where he talks about neural networks as dynamical systems and suggests the use of dynamic recurrent artificial neural networks (DRNNs). DRNNs are capable of doing everything that a formal computational system can do plus, importantly, they provide the dynamical systems element that, in this view, are deemed crucial for cognition. Unfortunately, DRNNs are quite complex and difficult to analyze and understand (Harvey, 1997).

Organisms having significantly different neural structures differ in their behavior when dealing with their corresponding environments. However, computational models of the nervous systems show that intricate computations can be done as a result of simple neural circuits (Fellous & Linster, 1998). This leads to a very important indication that for complex tasks in our environments, even simple feed-forward neural networks may suffice. As described by Leonce et al. (2012) in their work on the evolution of nurturing, a simple, fully connected, feed-forward neural network is not only enough to drive the robot around in the light switching arena, a moderately complex task, but also to evolve nurturing in this stationary environment. There are no recurrent connections and thus no memory is retained, yet this simple neural network performs well enough to adapt to a moderately difficult problem space.

2.2 Evolutionary Computation

Evolution is a powerful natural force. One may observe species of animals including both vertebrates and invertebrates that evolve over time with sophisticated instinctive and learning mechanisms that include survival, hunting, and many more, thus evolution naturally addresses their problems. A general computational model in machine learning inspired by evolution and known as *evolutionary computation* (EC), offers variety of methods and tools to address computational problems. Some of the sub-fields of EC include genetic algorithms (GAs), evolution strategies, evolutionary programming, genetic programming (GP), classifier systems, and combinations or hybrids thereof (Bäck et al., 1997). Next is a brief look at the two most typically used evolutionary computation methods, i.e., genetic algorithms and genetic programming.

Genetic algorithms offer a method to iterate from one population of chromosomes to a new population using selection and genetic operators such as cross-over, mutation, and inversion (Mitchell, 1998). Similar to GAs, *genetic programming* follows a similar methodology to compute solutions; however, genetic programming is different in a sense that it does not explicitly require the user to specify the structure of the solution in advance (Poli et al., 2008). GPs are generally used to evaluate a single function using variable-sized tree data structure of functions and values (Sivanandam & Deepa, 2007).

The choice of technique depends on the type of problem being addressed. For instance, the evolution of the structure of a learning rule is typically implemented using a tree-based genetic programming approach because generally a learning rule consists of operands and arithmetic operators and a tree data structure can appropriately represent such structure. Therefore, a genetic programming approach suits this situation well. On the other hand, a typical implementation of the evolution of the learning rule parameters (such as learning and decay rate values) involves genetic algorithms. Here genes of a chromosome represent those values that can be evolved over generations and can be evaluated at every generation. It is important to note that there are many variations and hybrid models that are used by researchers; what is discussed above are very broad typical approaches.

An interesting take on evolutionary methods suggests that some problems should not be treated as optimization problems and that, in the long term, much of the evolutionary robotics will require a different framework (Harvey, 1997). The claim is that the main properties of an optimization problem are that it is one specific problem and that the search space of possible solutions is well defined in terms of a fixed number of parameters whereas natural evolution is different in terms of the properties explained and that we need a different class of GAs that work on the principle of incremental adaptive improvements.

GAs can be implemented in several ways depending on the class of problems being addressed. There are a number of characteristics of GAs that are useful to be considered when designing a GA. Selection and reproduction are the two main components of GAs.

The selection scheme is analogous to survival of the fittest in nature. There are various types of selection schemes used in GAs such as fitness-proportional, rank-based, tournament, and steady-state selection each with different characteristics (Mitchell, 1998). Tournament selection is efficient for both parallel and non-parallel architectures (Miller & Goldberg, 1995). In *tournament selection*, a tournament is held between b individuals from the population where b is the tournament bracket size. The fittest individual based on the evaluation criteria is selected as the winner of the tournament and that individual reproduces to produce offspring. This tournament is held number of times until a whole new population equal to the size of the current population is created.

Reproduction may take place via elitism, cloning, or mating. Retaining unaltered copies of the most fit individuals to exploit the best individuals found at a given generation is known as *elitism*. *Clones* are copies of individuals that may not be the most fit from the population and may be altered through mutation (see below). Finally, through *mating*, copies of the genes of two or more individuals are combined to make one or more offspring. These offspring may be mutated. *Mutation* alters one or more genes in an individual depending on the rate of mutation. All aspects of selection and reproduction must keep the balance between retaining the good genes and introducing diversity (exploitation vs. exploration) to find global optima in the search space.

Reproduction through mating relies on crossover. *Crossover* determines how the genes of the parents are combined to form the offspring. Crossover is of different types such as one-point, two-point, and uniform crossover. *Uniform crossover* means each gene for each offspring is equally likely to come from each parent. On average, one can expect to get half of the genes from parent one and the other half from parent two. If the parents are very similar, this will produce very similar offspring and crossover will be essentially exploiting the values of the parents. However, if the parents are very different from one another, then the offspring that result are likely to be very different from either parent. Hence, when diversity in the population is high, crossover will be likely to jump to new regions of the search space and can be seen as a global exploration operator, as opposed to mutation which is generally a local exploration operator. Note that crossover actually combines exploitation with exploration. It uses existing alleles and it does not generate new alleles the way mutation does. However, it combines these existing alleles into potentially novel chromosomes and hence it explores, at least when the population is diverse. In fact, mutation generally just explores locally, since it typically changes only one gene (or some small number of genes) and even then the change is generally by some small amount.

For detailed implementation of the GA used in this dissertation, see the Chapter 4.

2.3 Reinforcement Learning

Learning is a natural trait of many biological organisms. In machine learning, learning methods can be divided into three main learning paradigms: (1) Unsupervised or self-supervised learning, (2) reinforcement learning, and (3) supervised learning.

Unsupervised learning may be used for clustering problems where the goal is to group a particular data point with similar data points. This type of learning does not require any external teacher or evaluative feedback. However, one can argue that it has a built-in supervisor that determines what constitutes similarity among points and what defines a good cluster. An example of this type of learning is k-means clustering.

Supervised learning methods work on the principle of an external teacher. The teacher is used to teach the learning system the desired behavior. In machine learning, a typical representation of a learning system involves an ANN with a learning rule adjusting its weights to estimate the desired output. In the case of supervised learning, the external teacher is represented by the training data. Already known input-output relationships are used to find the difference between the actual output and the desired output. Usually, the difference is then taught to the neural network by propagating the error to correct the weights of the connections. Examples of such system include pattern recognition algorithms.

The other type of learning paradigm, *reinforcement learning*, is similar to supervised learning except that the external teacher is more evaluative than instructional (Gullapalli, 1990). In this type of learning, desired input-output relationships are not known in advance, thus the actual output generated for a given input is evaluated based on some performance measure. This evaluative feedback is generally in the shape of a reward or a penalty. Further, the reward or penalty is usually proportional to the performance of the system. The evaluative feedback is then used to tune weights of the network, also called as adjusting the policy, and the process may take many trials before system learns a good policy.

Reward situations that reinforcement learning systems face are either immediate reward or delayed reward (Kaelbling et al., 1996). *Immediate reward* situations are the ones in which the reward is received immediately after an action is performed i.e., there is no delay in the reinforcement. This reward can be stochastic i.e., a risk can be involved or the reward can be a deterministic amount. However, if a robot needs to take a series of actions before it is able to collect the reward, such as playing soccer until it scores a goal, the task would be considered a *delayed reward* scenario.

Furthermore, an *episodic task* is one in which the environment is reset when some condition is satisfied, such as when a particular state is reached or after some number of time steps have passed. Each period between resets is known as an *episode* or a *trial*. A delayed reward that is only given at the end of an episode is known as a *terminal reward*. An example of a terminal reward scenario is where a bee starts each trial (episode) above a flower patch, either goes straight or orients randomly, and then takes a series of steps downward before landing on a flower and receiving a reward (Niv et al., 2002). A delayed reward scenario is complex as compared to an immediate reward situation mainly because of the credit assignment problem (discussed later) and requires more insight when designing the algorithms.

Reinforcement learning algorithms are either off-policy or on-policy (Poole & Mackworth, 2010). An *off-policy* learner learns the value of the optimal policy independent of what actions an agent takes, as long as it explores enough. An example of such learning is Q-learning. On the other hand, an *on-policy* learner learns the value of the policy that agent is carrying out including the exploration so that the policy can be iteratively improved. An example of such learning algorithms is state-action-reward-state-action (SARSA) (Poole & Mackworth, 2010). Various authors, such as Sutton et al. (2009), have shown successful implementations of both on-policy and off-policy learning in the computational domains.

Reinforcement learning faces many challenges in dealing with robotic problems. Problems in robotics are often related to high-dimensional, continuous states and actions (Kober, 2013; Powell, 2012). To deal with continuous domains, a learning algorithm must be specially designed. (Peters et al., 2003) show that while using on-policy and off-policy learning, policy improvement is guaranteed in discrete problem domains; however, it is not guaranteed in continuous domains or function-approximation-based policy representations.

Williams (1992) provide an excellent overview of general reinforcement learning algorithms for stochastic units using temporal credit assignment for delayed reward scenarios. Further, Gullapalli (1990) presents a reinforcement learning algorithm for learning real valued functions. There are two main functions of the stochastic learning units that produce continuous output. One is to estimate the correct value of the output for a given input. This estimation is denoted by a mean of a normal distribution of the unit's activation values. The other is to determine the exploration/exploitation behavior the units should exhibit, which is controlled by a standard deviation parameter for the normal distribution of the unit's activation values (Gullapalli, 1990). In this learning algorithm, standard deviation is based on the known maximum reward in the environment.

Another crucial aspect of delayed reward reinforcement learning systems is credit assignment. There are two different aspects of credit assignment: temporal and the structural (Gullapalli, 1992). If an agent takes several actions over some period of time before a reward value is obtained, how can it know which action(s) during that period of time were responsible for the reward value obtained? This would be considered a *temporal credit assignment problem*. On the other hand, if there are several inputs and thus multiple synapses that are responsible for the reward collected from the environment at a given time step, then each synapse should be given credit for its participation accordingly. This is a *structural credit assignment* problem. Consider again the example of robots playing soccer — a robot takes a series of steps before it receives any feedback from the environment, as discussed by Riedmiller et al. (2009). This feedback in the form of a reward or penalty is then used by the robot to improve its behavior. As this type of situation is a delayed reward scenario, one of the problems that this kind of system faces is temporal credit assignment. How does the system know which actions of the robot were good or bad? Ideally good actions should be encouraged and bad actions should be discouraged to gain maximum efficiency out of the system. Further, such a system usually involves multiple inputs at the same time, thus it would be facing a structural credit assignment problem as well.

Temporal credit assignment is the process of crediting the previous actions and the credit assigned should be a monotonically decreasing function of the time between action and reinforcement (Sutton, 1984). One basic approach is to consider those elements responsible for the actions taken to be eligible for change (encouragement or discouragement) based on the reward or penalty received. This can be done by keeping track of participation in the process with *eligibility traces* of one of three basic types: replacing traces, saturating traces, and accumulating traces. Typically, in discrete state-action spaces there is a constant base eligibility value that is replaced or accumulated for a unit whenever it is active on a given timestep. Note that in discrete state-action spaces, a given (base) value for any action taken in a given state can be added because a discrete action is either taken or not taken. However, with continuous actions chosen by sampling from a probability distribution, the value added to a synapse's eligibility varies with the current weight sampled from the distribution. An *accumulating trace* adds an eligibility value to the previous value instead of replacing the old value, as with a *replacing trace*; thus an accumulating trace pays attention to recent actions but also credits non-recent actions by considering their eligibilities. Thus, an accumulating trace is a recency and frequency heuristic. There is no limit to the accumulation in such a trace, in contrast to a saturating trace (Hougen et al., 2000).

2.4 Nurturing Robotics

Nurturing robotics is a relatively new area of research in artificial intelligence (AI). Like many other sub domains of AI, it is inspired by evolutionary biology. In intelligent physical robots, learning in unknown environments is often reward based, relying on trial and error. With trial and error, generally some kind of supervision is required to look after the robots. This supervision is usually done by people. It would be beneficial for robots to sense and act in changing or uncertain environments without constant human supervision (Hougen et al., 2000; Bekey & Goldberg, 1993; Connell & Mahadevan, 1993). We would like robots to supervise (nurture) other robots. This calls for our research community to progress in exploring this sub area of developmental robotics (Woehrer et al., 2012). Further, via progress in developmental robotics by evolving more sophisticated algorithms for robots nurturing other robots, we may contribute to better machine learning.

Studies such as Leonce et al. (2012) investigate the evolution of nurturing using a simple task of light switching based on the Floreano & Urzelai (2000) light arena setup. Further, Eskridge & Hougen (2012) use an abstract environment to observe the relationship between nurturing and the evolution of learning in a preliminary study. Several other evolution of nurturing experiments are on-going and are explained in the next sections. An important question is why do we need to evolve nurturing? Woehrer et al. (2012) discusses why the evolution of R2R nurturing is important for both scientific and practical reasons. The scientific reason is that the evolution of R2R nurturing will help biologists understand the reasons behind the evolution of nurturing in nature. The practical reason is what was discussed in the previous paragraph: our robots need to sense and act in uncertain environments without human supervision. Thus, this supervision can be handed over to robots that can nurture other robots. Further, nurturing should be evolved rather than hard coded because the evolutionary process is well suited to find unexpected and/or difficult to find solutions to problems. Furthermore, an evolutionary process offers more flexible, scalable, and robust solutions as compared to a hard coded approach.

Previous research, on which this dissertation is partially modeled, shows that nurturing can successfully evolve in a R2R nurturing setup (Leonce et al., 2012).

The authors use a light switching arena where a parent robot nurtures its child by turning on a light switch. The child robot thus spends most of its time under the light instead of worrying about self care for turning on the switch. In the evolution of nurturing experiments, *self care* means to complete the task of resource collection unaided, that is, to turn on the light switch first and then move to sit under the light for the maximum amount of time.

The authors first validate their experimental setup by evolving robots capable of self care in the light switching arena. A single individual simulated robot from each generation is placed in the arena. The individual is allowed to perform self care and gain fitness by first turning on the switch and then sitting under the light. The total time spent under the light by the individual is used as that individual's fitness and successful genes are passed on to the next generation using a fitness-based GA. Self care is seen to be evolved and thus the setup is validated.

The authors then show that adding a development state to the neural controller capable of evolving self care allows for R2R nurturing and nurturability. In their setup, the authors use an individual to create a possibly mutated copy of itself and place both the original (parent) and the offspring (the possibly mutated copy) into the arena. The developmental state is specified to both the parent and the child using two additional input neurons specifying if the individual is currently in the role of parent or child. The only important fitness measure in the arena is the time spent by the child under the light source after the light switch has been turned on. The results demonstrate that nurturing and nurturability significantly outperform self care.

Similarly, the authors show that parental nurturing is more likely to evolve if parents have great capabilities than offspring. Also, Leonce et al. (2012) highlight that nurturing is more likely to evolve between parents and offspring than between unrelated individuals. Furthermore, there are various other directions in which nurturing research, using a similar experimental setup, have been investigated at the REAL

Lab, some of which includes sibling nurturing and grandparent, parent, and offspring nurturing.

2.5 Nurturing Niche Construction

"The capacity of organisms to construct, modify, and select important components of their local environments" can be referred as *niche construction* (Day et al., 2003). Laland (2004) gives an analogy between niche construction theory and extended phenotype theory and gives a description of linear versus cyclical causation, arguing that niche construction and natural selection are cyclically causal. This idea could be seen as similar to the idea proposed in this dissertation that the evolution of nurturing and the evolution of learning are causally cyclical (whether or not one accepts niche construction theory, see below). The authors also give an example of how niche construction can influence evolution: people first kept dairy cows and later genes for lactose tolerance in adults spread through dairying populations but not through other populations. According to Laland (2004), after natural selection, niche construction is a second major participant in evolution. Based on Wolf et al. (1998, 2000), Mousseau & Fox (1998), Odling-Smee et al. (2003), and Laland et al. (1996, 1999, 2001), the author suggests that niche construction changes the dynamics of the evolutionary process. This is an important remark as the nurturing and self-care niches in the experiments proposed in this dissertation are also hoped to reflect that. On the other hand, Dawkins (2004) argues that some kinds of niche construction do exist and do influence evolution in the manner of extended phenotypes — he gives an example of beaver dams that help the survival of genes for dam building among beavers — however, he argues extended phenotype theory covers those cases and that niche construction theory both isn't necessary for those real effects and also brings in many ideas that can't be considered extended phenotypes and can actually interfere with our understanding of evolution.

Scott-Phillips et al. (2014) gives a critical analysis of niche construction theory, presenting arguments both for and against it. Without taking a position on this controversy, I note that the environment constructed by a species can influence the evolutionary course of that species, a point on which both niche construction theory and standard evolutionary theory agree.

2.6 Reward Shaping and Chaining

Shaping can be seen as reinforcement of a series of successive approximations (Gullapalli, 1992). “Shaping by successive approximations is considered as an important animal training technique in which behavior is gradually adjusted in response to strategically timed reinforcements” (Saksida et al., 1997). For example, a service dog can be trained through shaping to respond to multiple verbal commands to assist a disabled person (Saksida et al., 1997).

Reward shaping is explained as a technique that provides localized feedback based on prior knowledge to guide the learning process (Laud, 2004). On the other hand, *chaining* “is a method that formalizes this intuition to create chains of options to reach a given target event by repeatedly creating options to reach options created earlier in the chain” (Konidaris & Barreto, 2009). Note that the authors did not mention skill chaining as being conceptually derived from what is called chaining in the animal learning literature. Nonetheless, their concept of chaining seems to be very similar to the animal learning approach (Konidaris & Barreto, 2009). The authors also describe chaining as breaking the solution into subtasks and learning lower-order option policies for each one.

Gullapalli (1992) discusses two types of training that he calls “shaping.” The first is “shaping through differential reinforcement of behavior over time,” which is what is used to train animals, even more so than the typical "reward shaping" used in RL. The second is "shaping through incremental development of the learning system" which borrows many concepts from the planning or problem solving AI literature and decomposes the overall task into subtasks. The subtasks are then learned independently. Once all of the subtasks can be accomplished independently, a higher-level controller was added to the system and it learned to generate sequences of commands to the original (bottom-level) controller to accomplish the task as a whole by having the bottom-level controller carry out the subtasks for it in whatever order it commanded.

Norouzzadeh (2010) uses a very broad definition of shaping to mean anything that simplifies the task for a reinforcement learning (RL) system. This includes: "modifying the dynamics" of the task (e.g., physically simplifying the overall task, learning on that simplified version, then moving to the full task), "modifying the initial state" (keeping the environment the same but starting closer to the goal initially, then moving back gradually to the original starting point), "modifying the action space" (limiting some of the choices of the agent), "modifying the internal parameters" (tuning parameters such as the learning rate), and "extending the time horizon" (initially giving the agent a longer time to learn). Reward shaping or shaping (modifying the initial state) is very much akin to the examples of animal parents teaching their offspring by bringing them dead prey, then wounded prey, etc., which itself is similar to reward chaining, particularly if we are dealing with discrete states and distinct behaviors for moving from one state to another. Modifying the dynamics is somewhat similar to the work in this dissertation, except that the nurtured offspring does not move on to the full (non-nurtured) task after learning a simpler task which is more toward the proposed furture work. Norouzzadeh (2010) also makes a distinction between (1) permanantly changing the task and (2) starting with an easier version of the task then switching to the full task. Interestingly, Norouzzadeh (2010) does not consider a version of modifying the action space that initially limited the choices of the agent but then gradually allowed the agent to have more options.

Both reward shaping (by providing partial rewards) and chaining (by building next steps on previous steps), simplify a complex task by providing intermediate rewards. The type of nurturing that is the focus of this dissertation simplifies a complex task by solving a part of the task for the nurturee. Therefore, the effect is same but the approaches are different. In nurturing, the task is simplified to observe if learning is evolved more often.

In RL, shaping and chaining are both used to simplify the learning task so that system performance improves. Similarly, in the proposed experiments, it is expected that the easier task in case of nurturing, i.e., moving to the light while the switch is already turned on, will be performed better than the self-care task, i.e., turn on the switch and move to the light.

2.7 Nurturing as Task Simplification

Ziemke et al. (2004) talk about the idea of cognitive scaffolding in robotics. *Scaffolding* is an idea analogous to instructional scaffolding from education. In *instructional scaffolding*, the basic idea is that the teacher provides some support to the student during the learning process and the support is gradually removed as the student learns (Orey, Michael, 2001). Ziemke et al. (2004) study how species during evolution and individuals during their lifetimes are able to modify their environment for their own (individual tasks) or other agents, (collaborative tasks) cognitive benefit. The experiments in their work provide simple examples of how changes to the environment, by individual agents, can impact a tasks behavioral complexity in individual, competitive, and collaborative task scenarios. The authors also relate the idea of scaffolding in evolutionary robotics to niche construction as providing a support to the agents is like altering their environments according to their survival needs. They also indicate that their work sheds light on other interesting questions along a new line of research in evolutionary robotic models of agent-environment interaction. In the work in this dissertation, nurturing as task simplification can be seen as an instructor doing a part of the task and thus fits into their proposed general framework. However, note that in the proposed nurturing/task simplification approach, scaffolding is not removed to have the nurturee do a complete task itself similar to Ziemke et al. (2004) approach. Interestingly, in their approach they also never have their robots learn, which further brings into question their use of the term scaffolding. However, their robots do evolve behaviors and they do so using simple feed-forward ANNs with two outputs which are the two wheel speeds of their robots, similar to our approach.

Further, Ziemke et al. (2004) also co-evolve robot behaviors, where a "scout" robot went through a T-maze (or a double T-maze) and drops lights to guide a "drone" robot. This isn't exactly nurturing, since the two robots are apparently paired up randomly, so there is no ongoing relationship, and neither robot is developing (whereas ours involves learning which is a form of development), and they both get fitness from the success of the other (unlike our evolution of nurturing experiments where the parent's own fitness is not affected by the action of the child). Still, it has some interesting parallels to our work.

Caro (1980) and Ewer (1969) discuss a mother cat nurturing her offspring, where the mother cat catches a prey, kills it and eats in front of her kitten. Thus she teaches her offspring how to eat a prey. In the next step, she kills the prey and lets her offspring attack and eat the already dead prey. Next, she brings a live prey and lets her offspring kill the prey while she communicates with them. In the final stage of nurturing this task, she lets them find the prey themselves and does not interfere with their efforts; however, in case the prey escapes, the mother cat brings it back. Note that the mother cat is making the task simpler for its offspring thus nurturing can be seen as a task simplification or task assistance. It is also important to note the sequence in which the mother cat nurtures, that is, it offers them some kind of reward for the task completion. The learning step where the offspring only eat an already killed prey tells them the importance of the reward as it fills their stomachs to remove hunger. In the next phase, they successfully learn to kill a prey brought by their mother, with the help of her communication. Finally, they learn to hunt the prey themselves. This is all driven because they expect a similar reward at the end of the task. Similar nurturing behaviors have been observed in female suricate and tigers. The study in this dissertation shows a single step task simplification where the effort is to understand the importance of nurturing (via task simplification) in the evolution of learning. As seen in these animals, they work from the terminal stages forward to teach their offspring. Turning on a light switch and letting the robot only worry about finding the light source and gain energy is analogous.

Another interesting idea, given by Caro & Hauser (1992), considers teaching as either opportunity teaching or coaching. *Opportunity teaching* is where offspring are provided with opportunities to practice skills. *Coaching* is where the behavior of young is either encouraged or punished by adults. Various adults such as whales and raptors instruct their offspring prey catching techniques. They emphasize that there are different forms of teaching found in animals unexplored by humans as they do not fit our human teaching/nurturing criteria. Opportunity teaching can also be seen as safe exploration and social learning (Eskridge & Hougen, 2012). Further, it can also be seen as offspring being provided with an opportunity to learn about the best light source.

The approach in this dissertation is based on task simplification as through nurturing, a partial task is completed for the offspring. The expectation is that this nurturing as task simplification promotes the evolution of learning in the nurtured niche as compared to the self-care niche.

2.8 Evolution of Learning

Ever since robotics research began, roboticists have aimed for fully autonomous robots. However, this is an extremely difficult problem (Lin, 1993). After decades of research, roboticists have only been able to design partially intelligent, mostly manually controlled, robots that can only work on a task or a set of tasks, rather than being given an entire mission. One of the main reasons that developing robotic intelligence is so difficult is that we do not yet fully understand the intelligence of biological organisms. Computational neuroscience, an interdisciplinary science, is one field that facilitates an understanding of intelligent structures. However, we are still far from implementing such complex structures in our artificial domains. Biology indicates that learning complex structures requires substantial time and energy investment (Reece & Campbell, 2011). Thus a transformative approach is required to evolve nurturing and learning in robots (Woehrer et al., 2012).

Nurturing is one of the important contributing factors to the evolution of learning (Woehrer et al., 2012; Eskridge & Hougen, 2012). Nurturing as both social learning (for example, a child imitating its parent) and safe exploration (for example, a child being provided for by its parent which gives it the opportunity to experience an uncertain environment without risk) has been explored by Eskridge & Hougen (2012) at an abstract level. Adapting to an uncertain environment requires learning. However, factors contributing to the evolution of learning are poorly understood. The experiments conducted by Eskridge & Hougen (2012) involve food patch estimation in uncertain environments. The results demonstrate that nurturing as both social learning and safe exploration promote the evolution of learning. After these preliminary results, the work in this dissertation evolves learning in a much more detailed and complex environment as compared to the one used by Eskridge & Hougen (2012). Further, the experimental design allows for future integration of the evolution of nurturing (as explored by (Leonce et al., 2012)) to the evolution of learning (work in this dissertation). Finally, this work involves the evolution of learning in both the absence and presence of instincts which suggests the generality of the approach in this dissertation.

Artificial neural networks (ANNs) are commonly used tools for learning in robots. ANNs alone are powerful computational tools to solve approximation problems. However, big questions remain as to how scalable, general, and robust these ANNs and learning systems are to different situations in a reasonable extended boundary on a slightly different task. If, at the end of the day, we are to design learning systems that deal with real world situations, we have to think about the general applicability of the designed learning algorithms. In our current context, we can take an example of R2R nurturing. If an individual being nurtured learns well in a designed environment, we may conclude that we solved a learning problem in our designed environment for a particular task. However, if we change the task or the environment slightly, how well our hand-designed algorithm is going to perform remains a question. Unfortunately, in most of ML research so far, we have seen limited instances of this issue being addressed. One possible answer to this question is evolution, which is a natural remedy. Nature has shown us that countless evolved species exhibit learning and that learning can be scalable and very adaptive to unknown situations. Evolution is an additional adaptive component that we need for our algorithms to be scalable. Thus artificial neural networks that are evolved refer to a special class of ANNs in which evolution is another fundamental form of adaptation in addition to learning (Yao, 1991, 1993a,b, 1994, 1995, 1999). Evolution, infact, is a powerful tool that can be used to find a scalable and more general solution to the problems by finding the best architecture of an ANN including the number of neurons in each layer and their connection types (feed-forward vs. recurrent and fully vs. partially connected). Further, the number of hidden layers can be evolved and the connectivity for those layers can be evolved as well. Activation functions can be evolved as well. Evolution of neuromodulatory connections is yet another possibility as shown by Niv et al. (2002) in their research on the evolution of reinforcement learning in uncertain environments. Neuromodulation is used by a neuron to regulate other neurons. Evolution of weight-update learning rules is immensely important to cope with the increasing network complexities and to find more general solutions. Chalmers (1990) shows that the delta learning rule can be evolved in certain situations for supervised networks. Similarly, other authors demonstrate the evolution of learning by successfully evolving the learning rules for unsupervised and reinforcement learning adaptive environments (Fontanari & Meir, 2009; Dasdan & Oflazer, 1993; Nolfi & Parisi, 1996; Niv et al., 2002; Di Paolo, 2003; Soltoggio et al., 2007).

2.9 Summary

In this chapter background and related work is discussed mostly from the perspective of the evolution of nurturing and learning. We also briefly looked at various types of artificial neural networks. We also glanced over evolutionary computational techniques and reinforcement learning paradigms. Finally, discussion on the area of nurturing robotics and recent developments in this potentially important research area of developmental robotics is highlighted.

Chapter 3

Stochastic Synapse Learning Algorithm

This dissertation considers the influence of nurturing on the evolution of parameters for a reinforcement learning (RL) algorithm for a class of artificial neural networks (ANNs). This chapter describes this class of ANNs and an RL algorithm and verifies that this algorithm is capable of learning the desired behaviors within the same arena that will be used during the evolution of learning rule parameters experiments. Toward this end, this chapter also describes the arena, shows the particular ANN used for verification and experimentation, and gives the parameter values used during the verification process.

The RL algorithm described here is inspired by the algorithms explained in Gullapalli (1990) and Williams (1992) based on the similarity of the problem domains. However, there are several notable differences in the proposed learning rule in this dissertation: stochastic synaptic units, the use of a sliding window to compute the average reinforcement from the previous episodes, and the use of standard deviation traces to consider past actions and reward exploration/exploitation strategies accordingly. Note that, in this learning rule, learning parameters are evolved rather than network topology and other possibilities.

3.1 Algorithm Description

To design a learning algorithm capable of learning in the cases of both nurturing and self-care, a terminal/delayed reward scenario in the realm of reinforcement learning is chosen. As with classic reinforcement learning systems, using the exploration/exploitation trade-off to maximize the reward received while learning an episodic task will be the main objective of the proposed algorithm design.

3.1.1 Artificial Neural Network Controller Representation

I present a class of ANNs suitable for delayed reward problems using simple feed-forward neural networks. In such networks, binary input units are fully and directly connected to real valued output units. The weights of each synapse are sampled from a continuous probability distribution. The output units' activations are computed as the hyperbolic tangent of the sum of the corresponding weighted inputs for all the synaptic units. Such an ANN is shown in Figure 3.1.

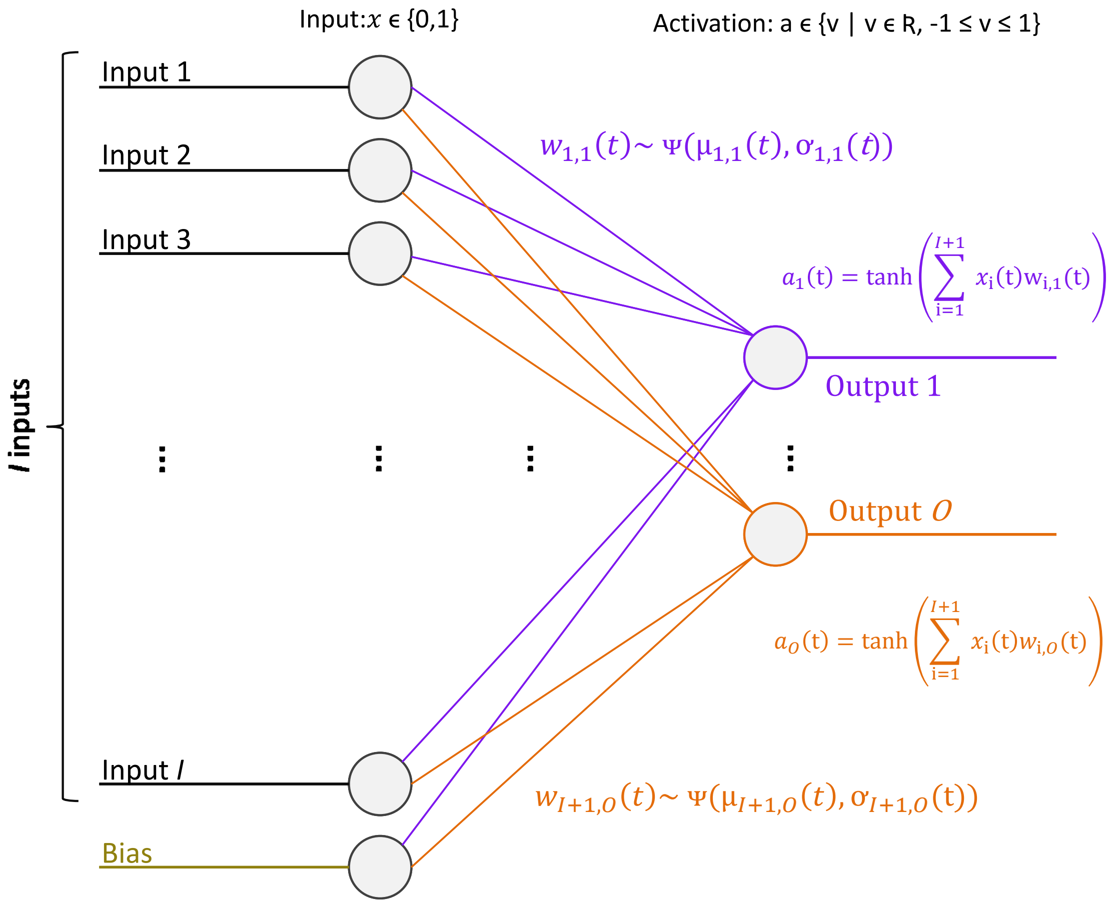
Input:$x \in \{0,1\}$
Activation: $a \in \{v \mid v \in \mathbb{R}, -1 \le v \le 1\}$
Input 1
Input 2
Input 3
Input I
Bias
I inputs
$w_{1,1}(t) \sim \Psi(\mu_{1,1}(t), \sigma_{1,1}(t))$
$a_1(t) = \tanh\left(\sum_{i=1}^{I+1} x_i(t) w_{i,1}(t)\right)$
Output 1
Output O
$a_O(t) = \tanh\left(\sum_{i=1}^{I+1} x_i(t) w_{i,O}(t)\right)$
$w_{I+1,O}(t) \sim \Psi(\mu_{I+1,O}(t), \sigma_{I+1,O}(t))$

Figure 3.1: General class of ANNs.

A fully connected feed-forward neural network with I input units, a bias unit, no hidden units, and O output units.

In Figure 3.1, I represents the number of binary inputs, O represents the number of real valued outputs ranging between -1 and 1, w represents a sampled weight from the weight distribution for each synapse, μ and σ represent the synaptic weight mean and the synaptic weight standard deviation of the weight distribution for each synapse, and a represents the activation value for each output unit.

3.1.2 Reinforcement Learning Algorithm for Real Valued Units

To devise a learning algorithm that works well in delayed reward situations for this class of ANNs, let us first look at the input/output patterns to study the properties and requirements of such a system. As Gullapalli (1990) and Williams (1992) demonstrate, there are two important aspects of stochastic learning units producing real valued outputs. One is estimating the mean value to output. The other is adjusting the standard deviation for calculating the activation for that unit. Gullapalli (1990) and Williams (1992) use a mean and standard deviation to calculate the activation of each unit, thus these stochastic units determine their output by sampling from a continuous probability distribution, such as a normal distribution. However, in the algorithm proposed in this dissertation, each mean and standard deviation of the weight distribution (for a particular synapse) is used to sample a synaptic weight value. The mean of each weight distribution corresponds to a noisy partial policy. When a certain presynaptic unit has a value of one, the synapse contributes to the activation of the postsynaptic unit a value that is likely to be close to the mean. Therefore, the approach taken here differs from that of Gullapalli (1990) and Williams (1992) which both talk about deterministic synaptic weights and stochastic activation/outputs. In the proposed approach, I use stochastic weights and deterministic activations/outputs based on the weights. Thus, the algorithm is allowed to be more or less exploratory at the synapse level as compared to the output level. Finally, using a weight mean update rule, the algorithm estimates all the synaptic weight mean values of the corresponding weight distributions for the next episode and this probabilistically leads to appropriate output values over the course of proceeding episodes. Similarly, a synaptic weight standard deviation update rule controls the exploration/exploitation trade-off. The synaptic weight mean μ is updated using Equation 3.4 and the synaptic weight standard deviation σ using Equation 3.10.

Algorithm

Algorithm 3.1 presents the pseudocode for an individual learning over an entire series of trials. The reinforcement learning algorithm proposed above is designed for a terminal reward episodic situation. Generally, there are several time steps involved in each of the several trials during the lifetime of an individual which is expected to perform learning. The algorithm iterates through each trial's time steps. At each time step, it calculates the output of the network using the inputs from the environment. At the end of each trial, the synaptic weight means and the synaptic weight standard deviations are updated and the process continues until the lifetime of the individual completes. The description of the above steps follows:

1. Initialize the number of trials τ, time steps t, means of the weight distributions μ, standard deviations of the weight distributions σ, learning rate for mean ${}_{\mu}\eta$, learning rate for standard deviation ${}_{\sigma}\eta$, decay rate for mean ${}_{\mu}d$, decay rate for standard deviation ${}_{\sigma}d$, average (expected) reward using a sliding window $\overline{r}$ (see the end of this section for description of the sliding window), the number of inputs, and the number of outputs.

2. For each episode (trial), iterate through all the time steps as a delayed reward is expected at the end of the trial.

3. Reset the synaptic weight means and the synaptic weight standard deviations of the corresponding weight distributions at the start of each episode.

4. At each time step, input from the environment is captured and the output is calculated. During this calculation step, weight w is sampled from the normal distribution of the corresponding synaptic weight. Also, the corresponding eligibilities for mean, shown in Equation 3.7, and standard deviation, shown in

Algorithm 3.1: Algorithm demonstrating an individual learning an episodic task by updating mean and standard deviation of each synaptic weight's continuous probability distribution after calculating the network output.

```
Algorithm LearnFromEpisodes()
    Initialize $\tau, t, \boldsymbol{\mu}, \boldsymbol{\sigma}, {}_{\mu}\eta, {}_{\sigma}\eta, {}_{\mu}d, {}_{\sigma}d, \overline{r}, sizeX, sizeY$
    for $\tau \leftarrow 0$ to $NumTrials$ do
        ResetToZero(${}_{\boldsymbol{\mu}}\boldsymbol{e}$, ${}_{\boldsymbol{\sigma}}\boldsymbol{e}$)
        for $t \leftarrow 0$ to $NumTimesteps$ do
            $\boldsymbol{x} \leftarrow$ Inputs from environment
            $\boldsymbol{y} \leftarrow$ CalcNetworkOutput($\boldsymbol{x}$, $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, ${}_{\boldsymbol{\mu}}\boldsymbol{e}$, ${}_{\boldsymbol{\sigma}}\boldsymbol{e}$, ${}_{\mu}d$, ${}_{\sigma}d$, $sizeX$, $sizeY$)
            Take action $\boldsymbol{y}$
            $TerminateTrial == true$ ? break : continue
        end
        $r(\tau) \leftarrow$ Reward from environment
        for $i \leftarrow 0$ to $sizeX$ do
            for $j \leftarrow 0$ to $sizeY$ do
                $\Delta\mu_{ij}(\tau) \leftarrow {}_{\mu}\eta(r(\tau) - \overline{r}(\tau))_{\mu}e_{ij}$
                $\mu_{ij}(\tau+1) \leftarrow \mu_{ij}(\tau) + \Delta\mu_{ij}(\tau)$
                $\Delta\sigma_{ij}(\tau) \leftarrow {}_{\sigma}\eta(r(\tau) - \overline{r}(\tau))_{\sigma}e_{ij}$
                $\sigma_{ij}(\tau+1) \leftarrow \sigma_{ij}(\tau) + \Delta\sigma_{ij}(\tau)$
            end
        end
        UpdateExpectedReward $(r(\tau), \overline{r})$
    end
Procedure CalcNetworkOutput($\boldsymbol{x}$, $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, ${}_{\boldsymbol{\mu}}\boldsymbol{e}$, ${}_{\boldsymbol{\sigma}}\boldsymbol{e}$, ${}_{\mu}d$, ${}_{\sigma}d$, $sizeX$, $sizeY$)
    for $i \leftarrow 0$ to $sizeX$ do
        for $j \leftarrow 0$ to $sizeY$ do
            $w_{ij} \sim \Psi(\mu_{ij}, \sigma_{ij})$
            $a_j \leftarrow a_j + (x_i * w_{ij})$
            ${}_{\mu}e_{ij} \leftarrow ({}_{\mu}e_{ij} * {}_{\mu}d) + x_i(w_{ij} - \mu_{ij})$
            ${}_{\sigma}e_{ij} \leftarrow ({}_{\sigma}e_{ij} * {}_{\sigma}d) + x_i(|w_{ij} - \mu_{ij}| \; - \sigma_{ij})$
        end
    end
    for $j \leftarrow 0$ to $sizeY$ do
        $a_j \leftarrow \tanh(a_j)$
        $y_j \leftarrow F(a_j)$
    end
    return $y$
Procedure UpdateExpectedReward($r(\tau), \overline{r}$)
    /* Replace the oldest reward with the newest                    */
    $\overline{r}.dequeue()$
    $\overline{r}.enqueue(r(\tau))$
```

Equation 3.13, are traced. Each weight is sampled using

$$w_{ij}(t) \sim \Psi(\mu_{ij}(t), \sigma_{ij}(t)), \tag{3.1}$$

where $w_{ij}(t)$ is the sampled weight value, $\mu_{ij}(t)$ is the mean of the synaptic weight's distribution, and $\sigma_{ij}(t)$ is the standard deviation of the synaptic weight's distribution, for the synapse between input neuron i and output neuron j at time step t.

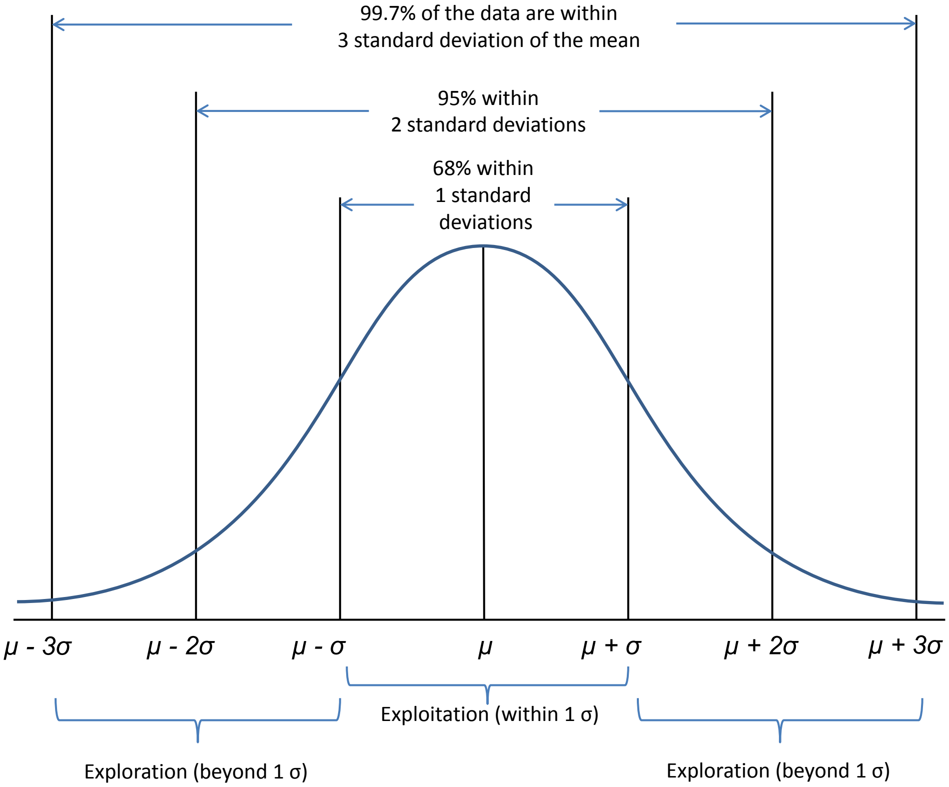

Figure 3.2: Normal Distribution with Exploration and Exploitation. Bell Curve showing Exploration and Exploitation Values. Values within 1σ are used by the algorithm to cause exploitatory behavior, whereas values outside 1σ are used to cause exploration. This figure is based on Wikipedia (2015).

Figure 3.2 shows various normal distribution values and their probabilities. The synapse weight randomly sampled from the normal distribution determines whether the policy for the current trial is more exploratory or exploitatory.

5. The activation value for each of the output units is computed using the weighted sum of the inputs connected to that output unit squashed using a hyperbolic tangent function, a specific type of sigmoidal function, to scale values down to

the range between -1 and +1 as given by

$$a_j(t) = \tanh(\sum_{i=1}^{I} x_i(t) w_{i,j}(t)), \tag{3.2}$$

where $a_j(t)$ is the activation value of output unit j at time step t, $x_i(t)$ is the input to neuron i at time step t, and I is the number of input units in the network.

6. The squashed real valued activations are then used to generate the output using some function F defined on [-1, 1]. The output function is

$$y_j(t) = F(a_j(t)), \tag{3.3}$$

where $y_j(t)$ represents the output of unit j at time step t and F represents some task-dependent function.

7. During each episode, the eligibility values of each synapse, which are based on the difference between the sampled weight and the mean (Eqs. 3.7 and 3.13), are accumulated based on recency as well as frequency heuristics using accumulating traces. This means that the algorithm pays attention to all the input units on the basis of how recently and frequently they had a binary input of 1 and the degree to which the sampled weight value differs from the mean during each time step. As mentioned previously, these eligibilities are reset at the beginning of each trial. The computation of eligibility traces for synaptic weight means μ and standard deviations σ are shown in Eqs. 3.9 and 3.15 in the next sections. Note that these discounted eligibilities are different from the ones used by Gullapalli (1990) and Williams (1992) which both referred to a single eligibility based on deterministic weights whereas in the proposed algorithm's eligibility values are on a per synapse basis and are based on stochastic weights.

Further, note that due to the possibility of multiple inputs (sensory data) being present at any given time and a delayed reward situation, both structural and temporal credit assignment problems need to be solved. This is unlike Gullapalli (1990)'s reinforcement learning stochastic units implementation where he updated weights every time step based on immediate rewards.

8. At the end of an episode, the means and the standard deviations of all the synapses are updated using Equations 3.4 and 3.10, which can also be considered as one learning step.

As with a typical reinforcement learning system, expected reward plays the role of a teacher to correct an individual's behavior over time. The use of a sliding window for recent rewards makes sense as changing policies based on too little experience (for example, by just looking at the last reward collected) causes individuals to make inappropriate reward estimations (unless future reward values are based entirely on recent reward values) and thus they do not perform well. Similarly, on the other extreme, paying attention to all the previous rewards collected causes the individual to change its policies based on the information that is likely to be too old considering that the environment change is expected during the lifetime of the individual which is why it needs to learn. Note that the concept of a sliding window for determining expected reward is a novel contribution compared to Gullapalli (1990) and Williams (1992), approach which leave the discussion on the computation of expected reward as an open question.

Learning through Weight Adjustment

At the end of each episode an individual gets a reward or a penalty depending on its actions in the arena (see Table 3.8). Using that reward or penalty, the algorithm updates the weight mean parameter for a particular synapse between input neuron i

and output neuron j using

$$\mu_{ij}(\tau+1) = \mu_{ij}(\tau) + \Delta\mu_{ij}(\tau), \tag{3.4}$$

where $\mu_{ij}(\tau+1)$ is the new value of the synaptic weight mean for the next trial $\tau+1$, $\mu_{ij}(\tau)$ is the value of the synaptic weight mean during the current trial τ, and $\Delta\mu_{ij}(\tau)$ is the change in value of the synaptic weight mean. $\Delta\mu_{ij}(\tau)$ is calculated using

$$\Delta\mu_{ij}(\tau) = {}_{\mu}\eta\,(r(\tau) - \overline{r}(\tau)) \sum_{k=1}^{t} {}_{\mu}e_{ij}(k)_{\mu}d^{(t-k)}, \tag{3.5}$$

where ${}_{\mu}\eta$ is the learning rate, $r(\tau)$ is the reward/penalty collected at the end of trial τ, $\overline{r}(\tau)$ is the average (expected) reward received so far (until this trial τ) using a sliding window, and $\sum_{k=1}^{t} {}_{\mu}e_{ij}(k)_{\mu}d^{(t-k)}$ is the sum of all the discounted eligibilities so far for a particular synaptic weight mean in this trial, and t denotes the time step in a given trial τ. Note that the synaptic weight eligibility values reset at the beginning of each trial. These eligibilities will be referred in this dissertation as *mean eligibility traces*[1].

Expanding the sum of the discounted eligibilities gives

$$\sum_{k=1}^{t} {}_{\mu}e_{ij}(k)_{\mu}d^{(t-k)} = {}_{\mu}e_{ij}(1)_{\mu}d^{(t-1)} + {}_{\mu}e_{ij}(2)_{\mu}d^{(t-2)} + {}_{\mu}e_{ij}(3)_{\mu}d^{(t-3)} + ... + {}_{\mu}e_{ij}(t)_{\mu}d^{(t-t)}, \tag{3.6}$$

where ${}_{\mu}e_{ij}(k)$ represents the eligibility of a given synaptic weight mean at time step k, ${}_{\mu}d$ is the discount rate (a constant), and t is the time step on which eligibility is being calculated (the time step on which the trial ends).

The eligibility of a synapse on a given time step k depends on the binary input value of the presynaptic unit and the difference between the sampled weight value on

[1] The leading subscript μ in various terms such as ${}_{\mu}\eta$ is to distinguish the terms related to the mean of the synaptic weight distribution μ from those related to the synaptic weight distribution σ.

that time step and the mean of the synapse's weight distribution, as follows

$$ {}_{\mu}e_{ij}(k) = x_i(k)(w_{ij}(k) - \mu_{ij}), \tag{3.7} $$

where $x_i(k)$ represents the binary input to a particular input neuron i at time step k, $w_{ij}(k)$ is the sampled weight value, and μ_{ij} is the mean of the weight distribution for the synapse between input neuron i and output neuron j at the corresponding time step k. Thus looking back at Equations 3.5 and 3.7, the weight adjustment rule can be summarized as follows (also shown in Table 3.1):

1. If $r - \overline{r} > 0$, then the individual gained a better than expected reward, which suggests that at least some of the sampled synaptic weight values were better than their corresponding synaptic weight means. In this case, if $w_{ij}(k) - \mu_{ij} > 0$ then a positive value for $r - \overline{r}$ times a positive value for $w_{ij}(k) - \mu_{ij}$ will cause $\Delta\mu$ to be positive. Thus, algorithm will shift μ in the direction of the sampled weight $w_{ij}(k)$.

2. If $r - \overline{r} > 0$, then the individual gained a better than expected reward, which suggests that at least some of the sampled synaptic weight values were better than their corresponding synaptic weight means. In this case, if $w_{ij}(k) - \mu_{ij} < 0$ then a positive value for $r - \overline{r}$ times a negative value for $w_{ij}(k) - \mu_{ij}$ will cause $\Delta\mu$ to be negative. Thus, algorithm will shift μ in the direction of the sampled weight $w_{ij}(k)$.

3. Contrary to the previous situations, if $r - \overline{r} < 0$, then the individual gained a lower than expected reward, which suggests that at least some of the sampled synaptic weight values were worse than their corresponding synaptic weight means. In this case, if $w_{ij}(k) - \mu_{ij} > 0$ then a negative value for $r - \overline{r}$ times a positive value for $w_{ij}(k) - \mu_{ij}$ will cause $\Delta\mu$ to be negative. Thus, algorithm

will shift μ in the direction opposite of the sampled weight $w_{ij}(k)$.

4. Finally, if $r - \overline{r} < 0$, then the individual gained a lower than expected reward, which suggests that at least some of the sampled synaptic weight values were worse than their corresponding synaptic weight means. In this case, if $w_{ij}(k) - \mu_{ij} < 0$ then a negative value for $r - \overline{r}$ times a negative value for $w_{ij}(k) - \mu_{ij}$ will cause $\Delta\mu$ to be positive. Thus, algorithm will shift μ in the direction opposite of the sampled weight $w_{ij}(k)$.

Reward	Eligibility	Eligibility Basis	Mean Adjustment	Resulting Change
$r - \overline{r} > 0$	$w - \mu > 0$	$w > \mu$	Increase Mean	Shift μ toward w
$r - \overline{r} > 0$	$w - \mu < 0$	$w < \mu$	Reduce Mean	Shift μ toward w
$r - \overline{r} < 0$	$w - \mu > 0$	$w > \mu$	Reduce Mean	Shift μ away from w
$r - \overline{r} < 0$	$w - \mu < 0$	$w < \mu$	Increase Mean	Shift μ away from w

Table 3.1: Summary of the learning algorithm—Mean adjustment.

Therefore, if the individual is performing better than expected, the algorithm shifts the synaptic weight means of the corresponding weight distributions in the direction of the sampled weight values that resulted in better performance. On the other hand, if it performed worse than expected, then the algorithm shifts the means in the direction opposite of the sampled weight values. This learning method worked well as shown in Section 3.2.7.

Equation 3.6 can be expanded using Equation 3.7 as shown below:

$$\sum_{k=1}^{t} {}_{\mu}e_{ij}(k)_{\mu}d^{(t-k)} = \sum_{k=1}^{t} x_i(k)(w_{ij}(k) - \mu_{ij})_{\mu}d^{(t-k)}. \tag{3.8}$$

Expanding the summation, we get the following equation:

$$\begin{aligned}\sum_{k=1}^{t} {}_{\mu}e_{ij}(k)_{\mu}d^{(t-k)} &= x_i(1)(w_{ij}(1) - \mu_{ij})_{\mu}d^{(t-1)} + x_i(2)(w_{ij}(2) - \mu_{ij})_{\mu}d^{(t-2)} \\ &+x_i(3)(w_{ij}(3) - \mu_{ij})_{\mu}d^{(t-3)} + ... + x_i(t)(w_{ij}(t) - \mu_{ij})_{\mu}d^{(t-t)}.\end{aligned} \tag{3.9}$$

Exploration/Exploitation Trade-off

Besides updating the means of the weight distributions for synapses of the neural network, the algorithm also updates the standard deviation values for the weight distributions for all of the connections between input neurons and output neurons that were active during the current trial. Updating the standard deviations is an important part of the proposed learning algorithm as updating these values effectively determines whether to explore or exploit during the next trial. This is in contrast to Gullapalli (1990) which uses expected reinforcement to compute both the mean and the standard deviation for each neural unit as a whole, rather than calculating a mean and a standard deviation value for each synapse. The equation to update the standard deviation values is

$$\sigma_{ij}(\tau+1) = \begin{cases} 0.05, & \text{if } \sigma_{ij}(\tau) + \Delta\sigma_{ij}(\tau) \leq 0.05 \\ 1, & \text{if } \sigma_{ij}(\tau) + \Delta\sigma_{ij}(\tau) \geq 1 \\ \sigma_{ij}(\tau) + \Delta\sigma_{ij}(\tau), & \text{otherwise,} \end{cases} \tag{3.10}$$

where $\sigma_{ij}(\tau+1)$ is the new value of the synaptic weight standard deviation for the next trial $\tau+1$, $\sigma_{ij}(\tau)$ is the current value of the synaptic weight standard deviation during this trial τ, and $\Delta\sigma_{ij}(\tau)$ is the change in value of the synaptic weight standard deviation for a particular synapse.

Equation 3.10 essentially adds $\Delta\sigma_{ij}(\tau)$ to $\sigma_{ij}(\tau)$ but ensures that $\sigma_{ij}(\tau+1)$ has a lower bound of 0.05 and an upper bound of 1. This helps control the amount of exploration. Even if the algorithm is very successful when exploiting some parts of the environment, it should still explore a little in case there are changes in other parts of the environment. Similarly, there should be an upper limit to the amount of exploration the algorithm permits. If exploration is not capped at the upper end, it becomes difficult for the algorithm to converge back to a reasonable exploration rate even if the sampling becomes conservative.

The change in the standard deviation of the weight distribution is calculated using

$$\Delta\sigma_{ij}(\tau) = {}_{\sigma}\eta\left(r(\tau) - \overline{r}(\tau)\right)\sum_{k=1}^{t} {}_{\sigma}e_{ij}(k)_{\sigma}d^{(t-k)}, \tag{3.11}$$

where ${}_{\sigma}\eta$ is the learning rate, $r(\tau)$ is the reward/penalty collected at the end of trial τ, $\overline{r}(\tau)$ is the average (expected) reward so far at trial τ, and $\sum_{k=1}^{t} {}_{\sigma}e_{ij}(k)_{\sigma}d^{(t-k)}$ is the sum of all the discounted eligibilities for the standard deviation from time step 1 to time step t for a particular synapse in this trial. t denotes the time step in a given trial τ. These eligibilities will be referred to in this dissertation as *standard deviation eligibility traces*. Note that the standard deviation eligibility values also reset to 0 at the beginning of each trial[2].

Expanding the calculation of the exploration traces gives

$$\sum_{k=1}^{t} {}_{\sigma}e_{ij}(k)_{\sigma}d^{(t-k)} = {}_{\sigma}e_{ij}(1)_{\sigma}d^{(t-1)} + {}_{\sigma}e_{ij}(2)_{\sigma}d^{(t-2)} + {}_{\sigma}e_{ij}(3)_{\sigma}d^{(t-3)} + ... + {}_{\sigma}e_{ij}(t)_{\sigma}d^{(t-t)}, \tag{3.12}$$

where ${}_{\sigma}e_{ij}(k)$ represents the exploration eligibilities of a given synapse at various time

[2] Again, the leading subscript σ in various terms such as ${}_{\sigma}\eta$ is used to distinguish the terms related to the mean of the synaptic weight distribution μ from those of the standard deviation of the synaptic weight distribution σ, as seen previously.

steps denoted by k, while ${}_{\sigma}d$ is the discount rate and is constant[3], and t is the time step on which eligibility is being calculated. As with the eligibility traces for the means of the weight distributions, those synapses that were active more frequently and more recently in the current trial are more eligible for adjustments to their synaptic weight standard deviations which is again different from the approach of Gullapalli (1990) where standard deviation is a monotonically decreasing non-negative function of the expected reward.

The eligibility for change to the standard deviation of a synapse's weight distribution is calculated using

$$ {}_{\sigma}e_{ij}(k) = x_i(k)(|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij}), \tag{3.13}$$

where $x_i(k)$ represents binary input to a particular input neuron i at time step k, $w_{ij}(k)$ is the sampled weight, μ_{ij} is the mean of the weight distribution, and σ_{ij} is the standard deviation for the weight distribution of the synapse between input neuron i and output neuron j at the corresponding time step k. Thus looking back at Equations 3.11 and 3.13, the exploration/exploitation rule can be summarized as follows (also shown in Table 3.2):

1. If $r - \overline{r} > 0$ then the individual gained a better than expected reward, which suggests that the exploration pattern exhibited by the individual was beneficial to it. In this case, if $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij} > 0$, the algorithm chose a value outside 1σ (see Figure 3.2), thus the individual can be considered to have explored. Since exploration resulted in a better than expected reward, the algorithm encourages exploration and thus increases the standard deviation of the synaptic weight distribution. Considering that a positive value for $r - \overline{r}$ times a positive value for $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij}$ will cause $\Delta\sigma$ to be positive, the algorithm increases

[3] Note that this discount rate value is independent of the one used for weight mean adjustment.

σ.

2. Again, if $r - \overline{r} > 0$ then the individual gained a better than expected reward, which suggests that the exploration pattern exhibited by the individual was beneficial to it. In this case, if $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij} < 0$, the algorithm chose a value inside 1σ (see Figure 3.2), thus the individual can be considered to have exploited (been conservative). Since exploitation resulted in a better than expected reward, the algorithm encourages exploitation and thus decreases the standard deviation of the synaptic weight distribution. Considering that a positive value for $r - \overline{r}$ times a negative value for $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij}$ will cause $\Delta\sigma$ to be negative, the algorithm decreases σ.

3. On the other hand, if $r - \overline{r} < 0$ then the individual gained a lower than expected reward, which suggests that the exploration pattern exhibited by the individual was not beneficial to it. In this case, if $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij} > 0$, the algorithm chose a value outside 1σ (see Figure 3.2), thus the individual can be considered to have explored. Since exploration resulted in a lower than expected reward, the algorithm discourages exploration and thus decreases the standard deviation of the synaptic weight distribution. Considering that a negative value for $r - \overline{r}$ times a positive value for $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij}$ will cause $\Delta\sigma$ to be negative, the algorithm decreases σ.

4. Finally, if $r - \overline{r} < 0$ then the individual gained a lower than expected reward, which suggests that the exploration pattern exhibited by the individual was again not beneficial to it. In this case, if $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij} < 0$, the algorithm chose a value inside 1σ (see Figure 3.2), thus the individual can be considered to have exploited (been conservative). Since exploitation resulted in a lower than expected reward, the algorithm encourages exploration and thus increases the standard deviation of the synaptic weight distribution. Considering that a

negative value for $r - \overline{r}$ times a negative value for $|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij}$ will cause $\Delta\sigma$ to be positive, the algorithm increases σ.

If the individual is performing better than expected, the algorithm encourages exploration or exploitation, whichever was being used. However, if it performs worse than expected, then the algorithm encourages the opposite of what it had been doing. Contrary to Gullapalli (1990) and Williams (1992) the algorithm controls which policy to follow and how exploratory it should be at the synapse level. These algorithmic strategies worked well as shown in Section 3.2.7.

Equation 3.12 can be expanded using Equation 3.13 to get:

$$\sum_{k=1}^{t} {}_{\sigma}e_{ij}(k)_{\sigma}d^{(t-k)} = \sum_{k=1}^{t} x_i(k)(|w_{ij}(k) - \mu_{ij}| \ - \sigma_{ij})_{\sigma}d^{(t-k)}. \tag{3.14}$$

The summation can then be expanded to get:

$$\begin{aligned} \sum_{k=1}^{t} {}_{\sigma}e_{ij}(k)_{\sigma}d^{(t-k)} = x_i(1)(|w_{ij}(1) - \mu_{ij}| \ - \sigma_{ij})_{\sigma}d^{(t-1)} + \\ x_i(2)(|w_{ij}(2) - \mu_{ij}| \ - \sigma_{ij})_{\sigma}d^{(t-2)} + \\ x_i(3)(|w_{ij}(3) - \mu_{ij}| \ - \sigma_{ij})_{\sigma}d^{(t-3)} + \\ ... + \\ x_i(t)(|w_{ij}(t) - \mu_{ij}| \ - \sigma_{ij})_{\sigma}d^{(t-t)}. \end{aligned} \tag{3.15}$$

3.2 Algorithm Validation

The purpose of this section is to validate that the algorithm is capable of learning appropriate ANN weights in the setup that will later be used for the evolutionary experiments. In the coming subsections, the various types of learning environments are discussed before introducing the problem followed by the validation experiment's design and its implementation. Implementation of the formerly introduced ANN and

Reward	Eligibility	Eligibility Basis	Standard Deviation Adjustment	Resulting Change
$r - \overline{r} > 0$	$\|w - \mu\| \ - \sigma > 0$	Exploration	Increase Std. Dev.	More Exploration
$r - \overline{r} > 0$	$\|w - \mu\| \ - \sigma < 0$	Exploitation	Reduce Std. Dev.	More Exploitation
$r - \overline{r} < 0$	$\|w - \mu\| \ - \sigma > 0$	Exploration	Reduce Std. Dev.	More Exploitation
$r - \overline{r} < 0$	$\|w - \mu\| \ - \sigma < 0$	Exploitation	Increase Std. Dev.	More Exploration

Table 3.2: Summary of the learning algorithm—Standard deviation adjustment.

learning algorithm are discussed as well. The last section talks about the results of the validation experiment.

3.2.1 Learning Environments

There are various types of learning environments as shown in Table 3.3 including

1. *Learning neutral environment*: An environment in which learning and instincts perform the same. An example of such environment could be one in which there is constant reassignment of reward values i.e., the change is random enough that it does not help to learn about the environment as current experience cannot predict future reward and there is no opportunity to act on learned knowledge.

2. *Learning positive environment*: An environment in which learning has the potential to outperform instincts. An example of this type of environment is one in which there is infrequent change in reward values. An individual can learn from experience and then exploit the knowledge gained; there is enough opportunity to explore, learn, and then act.

3. *Learning negative environment*: An environment in which learning is disadvantageous compared to instincts. An example of such an environment is one in which there are no changes in reward values. In such an environment inherited instincts can define an optimal policy and the exploration for learning deviates from the optimal policy.

		Learning Environment Types		
		Positive	Neutral	Negative
Behavior	Learning	✓	—	×
	Instincts	×	—	✓

Table 3.3: Various Learning Environments. Learning positive, neutral, and negative environments and their characteristics. ✓ means this behavior is advantaged, × means this behavior is disadvantaged, and — means this behavior is neither advantaged nor disadvantaged for the given environment type.

3.2.2 Problem Definition

The hypotheses (see Section 4.1) are focused on comparisons between nurturing and self-care when learning evolves. To address these hypotheses, I first need to establish an experiment that promotes learning (in a learning positive environment) then build an evolutionary environment around that learning experiment to evolve learning later. In the context of the proposed hypotheses, immediate reward (Vermorel & Mohri, 2005), terminal/delayed reward (Niv et al., 2002), or extended delayed reward (Leonce et al., 2012) scenarios could be implemented for learning. However, considering the hypotheses we are examining, a terminal/delayed reward scenario is appropriate due to being neither too simple nor too complicated so as to spend most of the attention on the actual questions to be answered.

Further, realizing various important features of the experimentation environment is important as well. The basic setup of the experiments are inspired by the light switching arena of Floreano & Urzelai (2000) while in implementation and experimental design it is a modification of the setup by Leonce et al. (2012) where at one end is the light source and at the other end is the light switch. An agent moves across the arena and its goal is to get to a light source in minimal time. In the original experiments of Floreano & Urzelai (2000), the robot needs to turn on the switch in order to collect energy from the light. In the experiments of Leonce et al. (2012), the light switch can be turned on by the robot itself (in which case the entire behavior is known as self care) or it can be turned on for the robot by a second robot that is also present in the arena (in which case the second robot is said to nurture the first robot). In this dissertation, there is only one robot present in the arena for each trial as in Floreano & Urzelai (2000) but, inspired by Leonce et al. (2012), the light switch is either turned on for the robot prior to each trial (to provide the nurturing treatment case) or turned off at the beginning of each trial (to provide the non-nurturing or self-care control case).

In addition to the above setup, the experiments require an additional component of a changing environment that is essential to observe learning in effect. It is important to consider building a learning positive environment. Niv et al. (2002) show in a bee foraging experiment that the evolution of learning in a terminal reward scenario can be accomplished using hebbian and antihebbian learning mechanisms. The two important aspects of the bee foraging experiments that are not present in the previous light-switching experiments by Leonce et al. (2012) are multiple possible targets (which are different colored flowers in the bee experiments) with different reward values and the fact that the rewards of these targets change both between and within generations, so an individual needs to learn to perform well. Thus, the proposed experimental design in this dissertation is a fusion and extension of these two experiments.

Next is an abstract discussion of the setup designed with the two cases of nurturing and self-care in the arena followed by a detailed description of the arena.

The setup consists of an individual robot that starts from the center of the arena and aims to find a (near) optimal path to the best rewarding light source present in the arena. There are three lights of different colors with different reward values: high, medium, and low. The reward values change during the lifetime of the individual, thus the robot has to learn in order to acquire a high level of reward from the environment.

In the case of nurturing, the switch and thus the lights are already turned on for the robot. The robot is being nurtured externally. The robot starts each trial of its life looking for the best rewarding light source. In the case of self-care, the switch is turned off at the start of every trial. Thus the robot has to first travel to the switch and then look for the best rewarding light source.

In both these cases, half way through the lifetime of the robot the highest reward value is swapped with the lowest reward value to change the environment and encourage learning. It is important to note that a successful instinctive individual with no learning capability visits the same light source over and over again, so the maximum reward it is able to collect in its lifetime is a moderate reward by following one of the following three strategies: (1) visiting a light that provides it a medium reward throughout its lifetime, (2) visiting a light that provides it a high reward during the first half of its lifetime but a low reward during the second half of its lifetime, or (3) visiting a light that provides a low reward during the first half of its lifetime but a high reward during the second half of its lifetime.

Having a difference of the nurturing (treatment) and non-nurturing (control) conditions while keeping everything else being the same, the expectation is that the data collected will highlight the nurturing vs. self-care performance differences.

3.2.3 Arena Setup

In the experimental setup, the arena consists of a square 50 by 50 environment surrounded by walls as used by Leonce et al. (2012). The arena contains a colored light switch on the wall at one end and three colored lights equally spaced on the wall at the other end of the arena. The lights and the switch can be turned on and off and they appear differently based on their state using various colors as shown in Table 3.4 and also as shown in Figure 3.3.

Light/Switch Color	State	RGB Color
Red	OFF	(0.2, 0.0, 0.0)
Red	ON	(1.0, 0.0, 0.0)
Green	OFF	(0.0, 0.2, 0.0)
Green	ON	(0.0, 1.0, 0.0)
Blue	OFF	(0.0, 0.0, 0.2)
Blue	ON	(0.0, 0.0, 1.0)
Switch	OFF	(0.4, 0.9, 0.0)
Switch	ON	(0.7, 0.7, 0.0)

Table 3.4: The lights and the switch in the arena, their states, and their corresponding color representations.

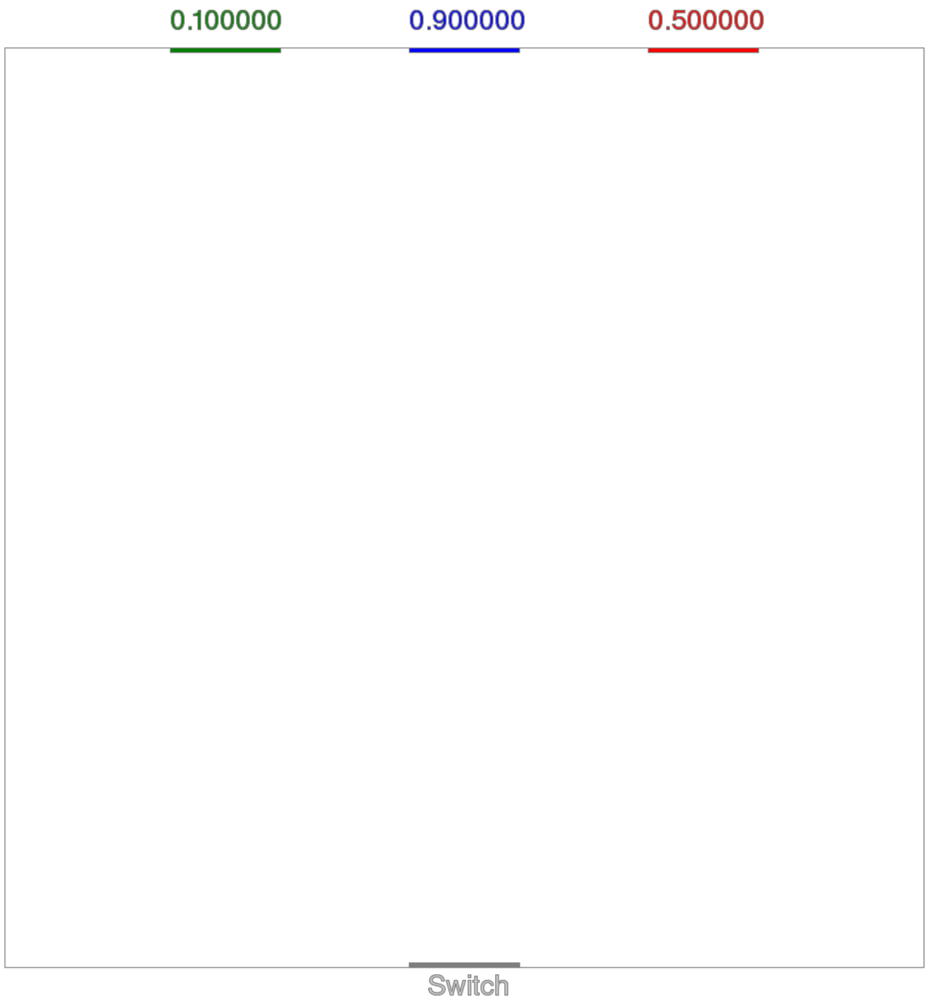

Figure 3.3: Empty Arena. Switch and red, green, and blue light sources with randomly assigned positions and randomly assigned reward values.

The three lights are assigned positions randomly on the wall, as shown in Figure 3.3. This means that any of these three lights can take any position randomly as shown in Table 3.6. Once the lights are positioned, the reward values of high, medium, and low are assigned to them randomly. The reward possibilities are shown in Table 3.5.

First Light Reward	Second Light Reward	Third Light Reward
High	Medium	Low
High	Low	Medium
Medium	High	Low
Medium	Low	High
Low	High	Medium
Low	Medium	High

Table 3.5: Various rewards in the arena and their possible positions.

First Light	Second Light	Third Light
Red	Green	Blue
Red	Blue	Green
Green	Red	Blue
Green	Blue	Red
Blue	Red	Green
Blue	Green	Red

Table 3.6: Various lights in the arena and their possible positions.

This simulated arena is explored by an e-puck robot. The robot uses two differential wheels using a left and a right motor. The speed for each wheel ranges from -15 to +15. The robot also uses a front-facing linear color camera, 60 pixels wide, that has a range capable of seeing walls and objects (the lights and the switch) across the arena from one end to the other. A further explanation of how this camera is used by the robot to sense the world using an ANN is given in the Section 3.2.4 as it is more relevant to the neural network discussion. The robot starts each new trial of its life by facing a neutral direction toward the wall (westbound), as shown in Figure 3.4. For details regarding trials, refer to Section 3.2.6.

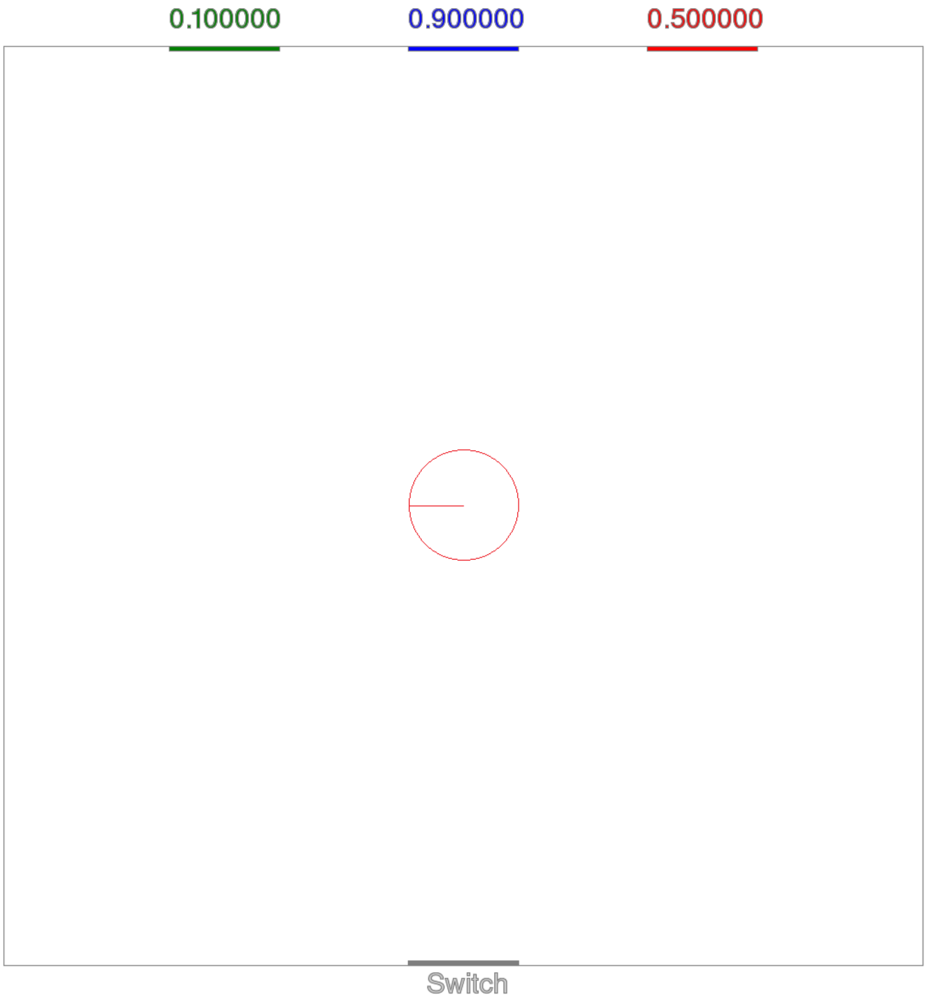

Figure 3.4: Arena with a Robot. E-puck robot represented by red circle; switch; and red, green, and blue light sources with randomly assigned positions and randomly assigned reward values.

3.2.4 Artificial Neural Network Controller

The simulated e-puck robot is controlled by an artificial neural network controller. This controller belongs to the same class of ANNs shown in Figure 3.1. The controller consists of a simple feed-forward neural network having 43 binary inputs and two outputs. 42 out of 43 inputs represent camera data from the environment while the 43^{rd} input is a bias unit. The camera data from the environment includes detection (that is a, binary value indicating presence or absence) of the various colors of the lights and the switch in their on and off states. The robots's sensing setup is robocentric rather than being world-centric. That means that the robot does not know its own x, y, or θ world coordinates or the locations of the lights, walls or the switch in world coordinates. Rather, it just knows what it can sense, for example; it might see the green light in its on state in the far right visual field of its camera. The robot might be very close to the green light or very far away from it or anywhere in between and still receive that same visual input. This results in perceptual aliasing. A complete description of what the robot senses from the environment, which is also input to the ANN, is shown below in Table 3.7.

Camera region	Light/Switch representations
Far Left (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON
Left (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON
Near Left (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON
Near Right (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON
Right (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON
Far Right (7 inputs)	Switch OFF, Red OFF, Red ON, Green OFF, Green ON, Blue OFF, Blue ON

Table 3.7: The 42 inputs to the ANN that are based on the camera data.

The neural network takes these 42 inputs together with the bias unit using 43 input neurons which are fully connected to two output motor neurons using 86 weighted connections. The structure of ANN is shown in Figure 3.5.

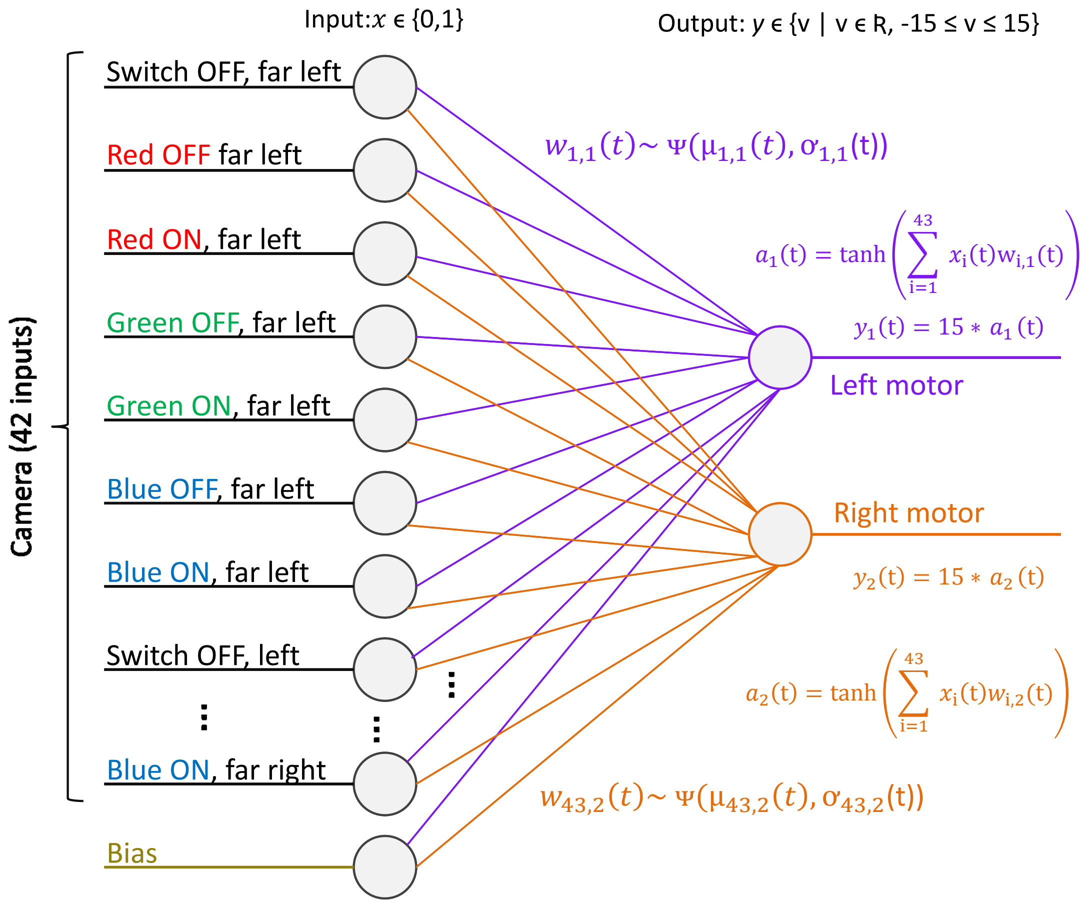
Input:$x \in \{0,1\}$
Output: $y \in \{v \mid v \in \mathbb{R}, -15 \le v \le 15\}$
Camera (42 inputs)
Switch OFF, far left
Red OFF far left
Red ON, far left
Green OFF, far left
Green ON, far left
Blue OFF, far left
Blue ON, far left
Switch OFF, left
Blue ON, far right
Bias
$w_{1,1}(t) \sim \Psi(\mu_{1,1}(t), \sigma_{1,1}(t))$
$a_1(t) = \tanh\left(\sum_{i=1}^{43} x_i(t) w_{i,1}(t)\right)$
$y_1(t) = 15 * a_1(t)$
Left motor
Right motor
$y_2(t) = 15 * a_2(t)$
$a_2(t) = \tanh\left(\sum_{i=1}^{43} x_i(t) w_{i,2}(t)\right)$
$w_{43,2}(t) \sim \Psi(\mu_{43,2}(t), \sigma_{43,2}(t))$

Figure 3.5: ANN Implementation. Fully connected feed-forward neural network with 42 input units, a bias unit, no hidden units, and two output units.

The robot senses the world around it using a linear color camera, 60 pixels wide, facing forward. In order to interpret this input into something meaningful for the proposed neural network, these 60 pixels are divided into 6 subgroups of 10 pixels each. These 6 subgroups represent far left, left, near left, near right, right, and far right regions of the camera field of view. Inside each camera region, the robot looks for seven different color values as shown in Table 3.4. Note that the robot does not recognize the Switch ON color as it does not need to pay attention to the switch once it has been turned on for any particular action. Thus these seven color values include:

1. the turned OFF switch color
2. the turned OFF red light color
3. the turned ON red light color
4. the turned OFF green light color
5. the turned ON green light color
6. the turned OFF blue light color
7. the turned ON blue light color

If 5 or more out of 10 of the color pixels for one of the seven colors are found to be present in a given camera region then an input of 1 is given to the corresponding neuron in the neural network as shown in Figure 3.5. Conversely, if 4 or fewer of pixels are of any particular color, an input of 0 is passed to the corresponding input neuron. Thus, considering six regions for each of the seven possibilities, there are a total of 42 binary inputs to the neural network. If the robot does not see any of the above mentioned objects and is, for instance, just facing the walls, a bias unit is used to input a boolean 1 value to the ANN in order to keep the robot moving in the arena. This makes the total count of binary input units 43 and thus the number of connection weights between the input and output units is 86 as this is a fully connected feed-forward neural network. The binary input values may change at every time step as the robot moves around the arena. The camera input depends upon what is in its field of view. The robot is determined to have reached the light (thus ending the trial), when all six of its visual input regions register the presence of the same light source. Similarly, an individual is determined to have reached the switch and have turned it on when all six of its visual input regions register the presence of the switch. This is robo-centric. Further, the robot's output units consist of two neurons each representing one of the robot's differential wheels, the left and the right motor. The method to calculate the motors output is described in Section 3.2.6 as it is more relevant to the reinforcement learning algorithm implementation.

3.2.5 Implementation and Tools

To implement the experiments, the Enki 2D robot simulator (Magnenat et al., 2007) is used. The development language used was C++. Message Passing Interface (MPI) using the master slave model (Rajan & Nguyen, 2004) is incorporated to execute parallel code while running generational evolutionary algorithms for the evolution of learning and the evolution of learning and instincts (see Chapter 4 for details).

3.2.6 Hand-Designed Reinforcement Learning Algorithm

To test the proposed hypotheses discussed in Section 4.1, an experiment should be designed in such a way that it can be extended to add a layer of the evolution of learning later. However, to accomplish that, determination of a learning algorithm appropriate for the environment for the cases of both nurturing and self-care is required. As discussed in Section 3.2.2, the terminal/delayed reward scenario in the light switching arena is suitable for reinforcement learning.

Reinforcement Learning

This section talks about a specific implementation of a general class of the reinforcement learning algorithm for real valued units presented as Algorithm 3.1.

The robot starts its lifetime, comprised of multiple trials, in the arena. Each trial is multiple time steps long. In the beginning of each trial, the robot starts from the center of the arena, shown in Figure 3.4 facing toward a neutral wall with no objects on it. The robot moves according to the control signals from its neural network. If the robot arrives at a light that is on before the time step limit is reached, the robot collects a reward and the trial ends. If all of the time steps for a given trial pass before the robot reaches a light that is on, the robot receives a penalty instead. In the nurturing (treatment) case, each trial starts with the switch, and therefore the lights, on. In the non-nurturing (control) case, each trial starts with the switch, and therefore the lights, off, which means that the robot will only collect a reward if it arrives at the switch and then at a light. The task of the robot in both cases is to maximize its reward. It is important to note that, although the robot is able to see lights that are off, it does not get any reward if it reaches a light in its off state. Half way through the lifetime of the individual, the reward values are swapped between the highest rewarding light source and the lowest rewarding light source as discussed in Section 3.2.2. This swap ensures that instinctive but non-learning individuals will never gain more than a moderate reward while learning individuals may outperform non-learners and random individuals. Thus, a good learning individual will look for the best rewarding light source in the first half of its lifetime and once discovered, will exploit that resource. Similarly, in the second half of its lifetime when the same light source no longer provides the best reward, the individual will explore again to find the new best rewarding light source. Once found, it will again exploit that resource. The reward values are given in Table 3.8 while specific environment variables and their values are shown in Table 3.9.

Reward source	Value
High reward	0.9
Medium reward	0.5
Low reward	0.1
Penalty	-0.25

Table 3.8: Various rewards and their values.

Environment Variable	Value
1 lifetime	4000 trials
1 trial	1000 time steps
Reward swap	2000 trials

Table 3.9: Various environment variables and their values.

Considering three light sources in the arena with constant rewards, the robot needs to explore in order to find out the best rewarding source. After that it should be conservative and exploit that resource until it no longer benefits from that. The reward collected by the robot at the end of each trial is a function of how quickly it reaches the light source. The positive reward value for each trial is calculated using

$$r = \frac{t_m - t_c}{t_m} r_v, \tag{3.16}$$

where r stands for the scaled reward calculated, t_m is max time step, t_c is the current time step on which individual reaches the light source, and r_v is the raw reward value of the light source (i.e., 0.1, 0.5, or 0.9). If the robot does not reach any light source before the trial ends, it gets a penalty of -0.25.

An example of a typical path in the nurturing arena is shown in Figure 3.6, a good path in the nurturing arena is shown in Figure 3.7, and a bad path for either arena (failing to reach any light) is shown in Figure 3.8.

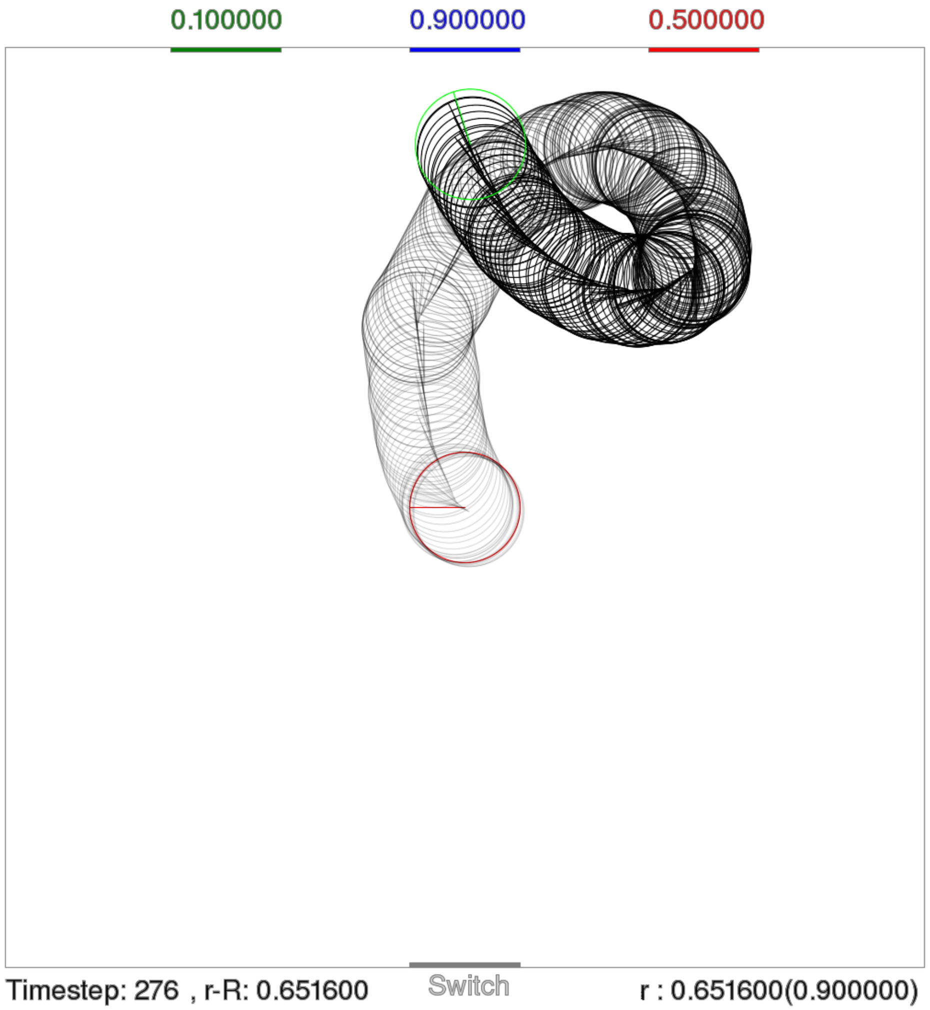

Figure 3.6: Demonstration of a Robot's typical path. Example of a typical path in the nurturing arena. Circles show the robot's position while lines show where its heading on each time step. Time step shows the total number of time steps out of 1000 taken by the robot to reach a light source. $r - R$ shows the current scaled reward minus the average reward and r shows the current scaled reward received and, parenthetically, the raw reward for the light reached (0.9 in this case). The robot's initial position is represented by a red circle and its final position is represented by a green circle. All the steps in between are represented by lighter gray (earlier steps) to darker gray (later steps).

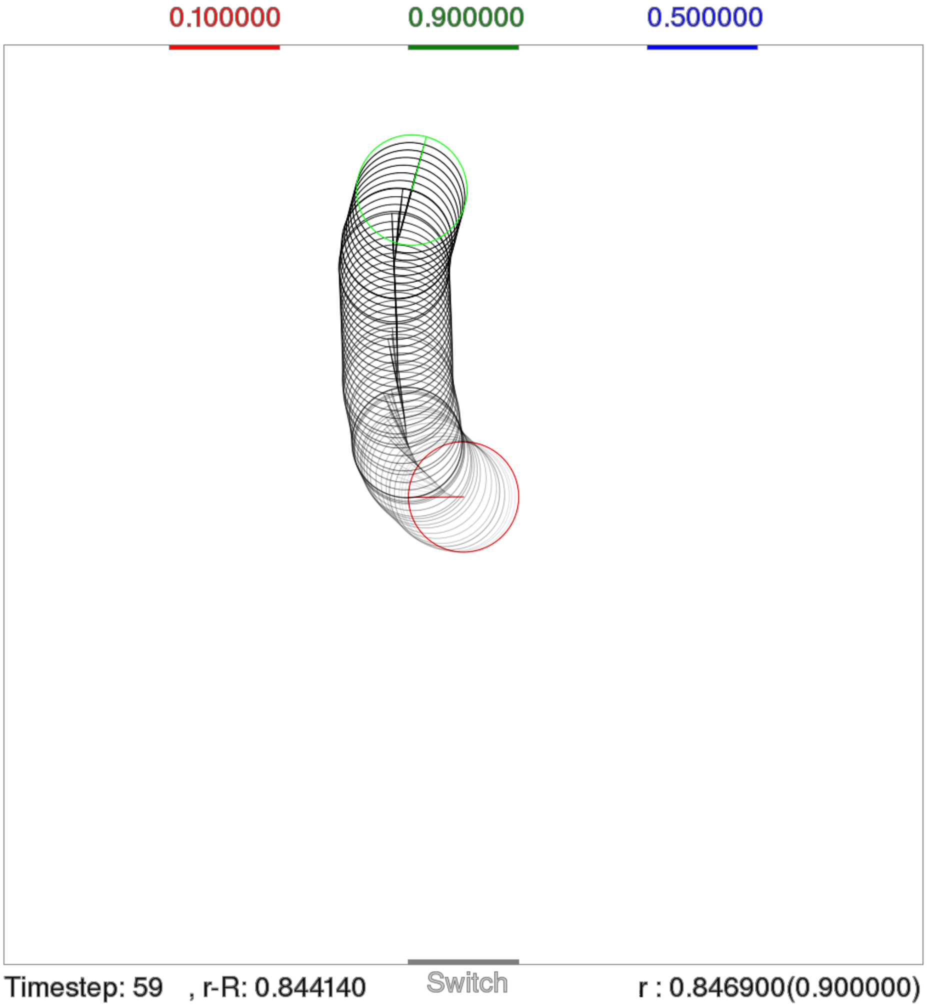

Figure 3.7: Demonstration of a Robot's near-optimal path. Example of a good path in the nurturing arena. The robot took 59 time steps to reach the high rewarding light. $r - R$=0.84 shows that the reward collected is far better than expected.

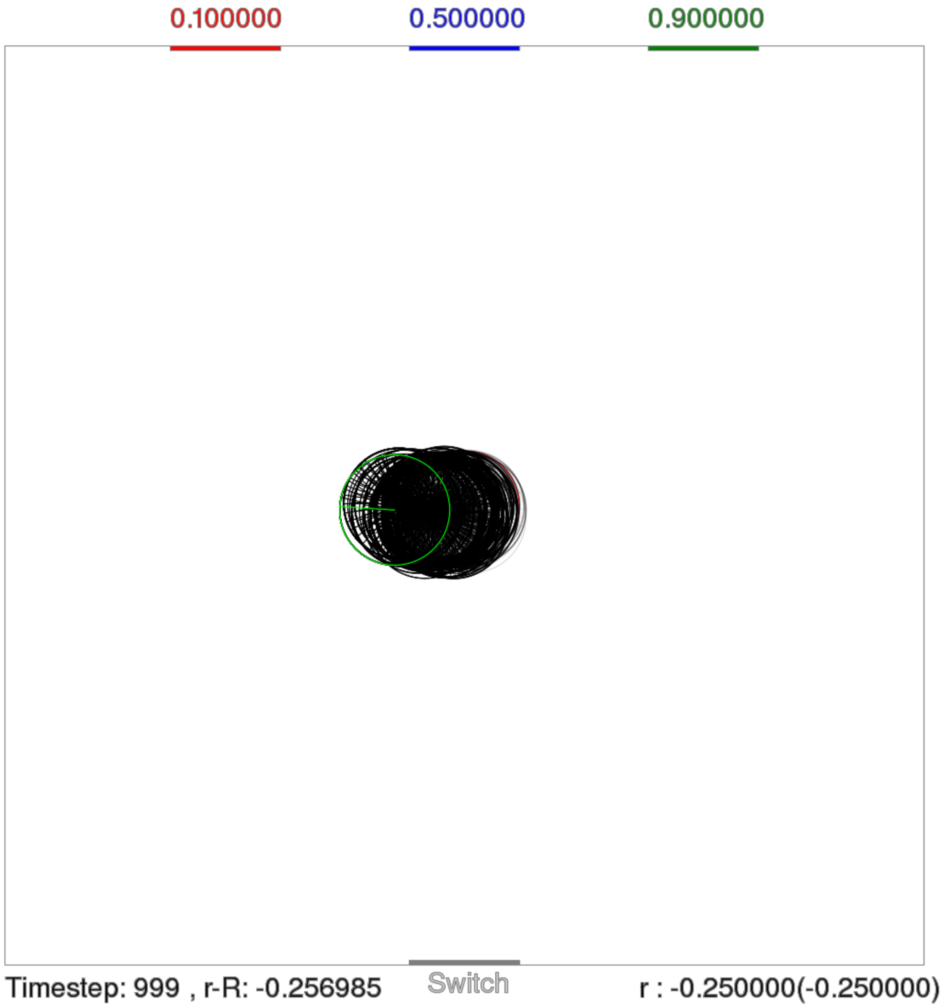

Figure 3.8: Demonstration of a Robot's failure. Example of a bad path in the arena. The robot spins in circles, fails to make substantial progress, and collects a penalty of -0.25.

In the case of both nurturing and self-care, at the end of each trial, the synaptic weight means (μ) and the synaptic weight standard deviations (σ) of the weight distributions from which the weights are sampled are updated using Equation 3.4 and Equation 3.10, respectively.

Algorithm

This section lists a step by step algorithmic implementation, based on Algorithm 3.1, designed to solve the problem defined in Section 3.2.2. It also make sense to refer to Figure 3.5, the neural network, which acts as a brain for the robot. It is important to note that the only difference between the nurturing and self-care conditions is the state of the switch at the start of each trial. In the case of nurturing, the switch is always on and in the case of self-care, the switch is initialized to off at the start of each trial. The algorithm's constant parameters are summarized in Table 3.10. Now let us examine the implementation details of Algorithm 3.1:

1. The lifetime of each robot is comprised of several trials. Each trial consists of several time steps.

2. At the beginning of each lifetime, all the neural network synaptic weight means are initialized randomly between 0 and 1. The range between -1 to 0 is avoided initially as they correspond to the robot moving backward, which is generally ineffective since the camera points forward. These means are denoted μ. Note that in the random case (used for baseline comparison), regardless of nurturing or self-care, the synaptic weight means are randomly initialized at the beginning of every single trial.

3. At the beginning of each lifetime all the neural network synaptic weights standard deviations, denoted σ, are initialized to a constant value of 0.9. It is important to note that the robot should start aggressively exploring the arena, thus a high initial value for each σ makes sense. Further, again for the random case, the synaptic weight standard deviations are randomly initialized at the beginning of every single trial.

4. The robot starts from the center of the arena as depicted in Figure 3.4 and

scans, using its camera, the environment its sees. This is passed as the input to the neural network on each time step t. The input units are binary and more than one unit may be active on a given time step as the robot might see one or more lights or the switch at any given time step (see Figure 3.5). Further, the robot has two continuous values as outputs (the two wheel speeds, which are independently determined).

5. After some or all of the input units become active, the weight w for each synapse is sampled from a normal distribution of that synapse's mean (μ) and standard deviation (σ) of the weight distribution as shown in Equation 3.1.

6. The activation value for each of the two output motor units is computed using the weighted sum of the inputs connected to that output unit squashed using a hyperbolic tangent function as shown in Equation 3.2.

7. The squashed real valued activations are then scaled up to generate the output motor speed in the range -15 to +15. The output function is simply

$$y_j(t) = 15\, a_j(t). \tag{3.17}$$

8. Using accumulating traces, as shown in Equation 3.7 and Equation 3.13, referred to in Algorithm 3.1, eligibility values for synaptic weight mean and synaptic weight standard deviation are computed. Equations 3.9 and 3.15 show details of how these values are calculated.

9. When a trial ends, either due to the robot reaching one of the light sources or 1000 time steps being completed, whichever comes first, the means and the standard deviations of all the synapses are updated using Equations 3.4 and 3.10.

Note that the binary vector of inputs representing the existence of various arena objects (the lights and the switch) and the continuous outputs are both different from the setup of Niv et al. (2002). For the input, the setup differs from Niv et al. (2002) in that the input includes (egocentric) directional information whereas Niv et al. (2002) only included relative quantity information, i.e., how much blue, yellow, and neutral color is found within the bee's visual field. On the other hand, compared to the continuous output used here, Niv et al. (2002) only had two output possibilities (go straight or orient randomly).

In the implementation of Equations 3.5 and 3.11, r(τ) is a scaled reward at the end of a trial τ (see Equation 3.16) or a penalty collected. Further, $\overline{r}(\tau)$ is the average reward so far until the current trial τ using a sliding window.

A sliding window of 20% of the total number of trials is used to compute the average of the most recent rewards collected. At the beginning of the lifetime of the individual, all the entries in the window are initialized to 0. This sliding window acts as a queue (FIFO). After each trial is over, the scaled reward collected is inserted into the queue while the oldest reward value drops out, working on the principle of first in first out. Thus, over several trials, this queue builds up recent rewards. The average reward is always computed over all the values in the queue. Thus, during the beginning phase of the individual's lifetime, it's expectation from the surrounding world is very low. As the number of trials progresses, the robot's reward expectation depends more on its past experiences. The main benefit of using this window is to ensure that frequent bad experiences in the beginning about the surrounding world should not unduly influence the robot's understanding of the world as its expectation initially will always be better than failure (-0.25 vs. close to 0). Similarly, if it happens to collect a positive reward (any of the three rewards) it is encouraged by positive experiences. Thus, this window gives a reasonably large opportunity for the simulated robot to learn about the environment before it decides to exploit a particular policy.

In order to understand the sliding window concept better, lets walk through the following example. If the robot fails on its first trial, it gets a penalty of -0.25. At that point, the average reward of the 800 values in the sliding window is 0, which means that $r(0) - \overline{r}(0)$ is also -0.25. This tells, for instance, Equation 3.5 to shift the policy by a moderate amount for those synapses for which there is a non-zero value coming from the eligibilities. At the end of the next trial, assume the robot fails again and gets a penalty of -0.25. At this point, the first trial's failure is already included in the average thus the expected reward $\overline{r}(1)$ is -0.000313, which is the average of one trial at -0.25 and the other 799 at 0. Similarly, at the end of third trial, if the robot fails again and gets a penalty of -0.25, at this point the first two trial's penalty values are already included in the average. Thus the expected reward $\overline{r}(2)$ is -0.000625, which is the average of the first two trials at -0.25 and the other 798 at 0. Conversely, at the end of the second trial, if the algorithm only considers the first trial's penalty as the average value so far, $r(1) - \overline{r}(1)$ would be 0 at this point and thus there would be no change in policy despite the repetition of the failure. Further, as can be noted, that despite consecutive failures, the algorithm still shifts the policy by the amount of the difference between $r(2) - \overline{r}(2)$, for instance. This shows that the robot will get numerous trials to learn about the environment.

The robot follows the policies described in Tables 3.1 and 3.2. Furthermore, all the algorithm constant parameters, their description and the initial values are summarized in Table 3.10.

Parameter Name	Symbol	Values
Learning rate for mean	${}_{\mu}\eta$	0.5
Learning rate for standard deviation	${}_{\sigma}\eta$	0.5
Decay rate for mean eligibility	${}_{\mu}d$	0.5
Decay rate for standard deviation eligibility	${}_{\sigma}d$	0.5
Minimum Sigma	${}_{min}\sigma$	0.05
Maximum Sigma	${}_{max}\sigma$	1
Initial Sigma	${}_{init}\sigma$	0.9
Sliding Window Size	s	20% of (4000) = 800

Table 3.10: Learning Algorithm parameter summary. Learning parameter constant symbols with their descriptions and values used in the hand-designed learning algorithm.

3.2.7 Validation Results

As discussed in the previous sections, designing and validating a learning algorithm that works for both nurturing and self-care is an important step toward the development of the neuro-evolutionary algorithms. These results demonstrate that the proposed algorithm works well in the target environment. To compare both nurturing and self-care learning algorithm results, it is important to set a baseline first with which results should be compared to. For this purpose, a random algorithm is executed and its results are collected. The following sections will highlight this comparison.

Nurturing — Learning vs. Random Behavior

In this section, we look at results from 30 repetitions for learning vs. random neural weight means and standard deviations in the nurturing condition. A particular interest is in the number of instances in which learning outperforms random (the baseline).

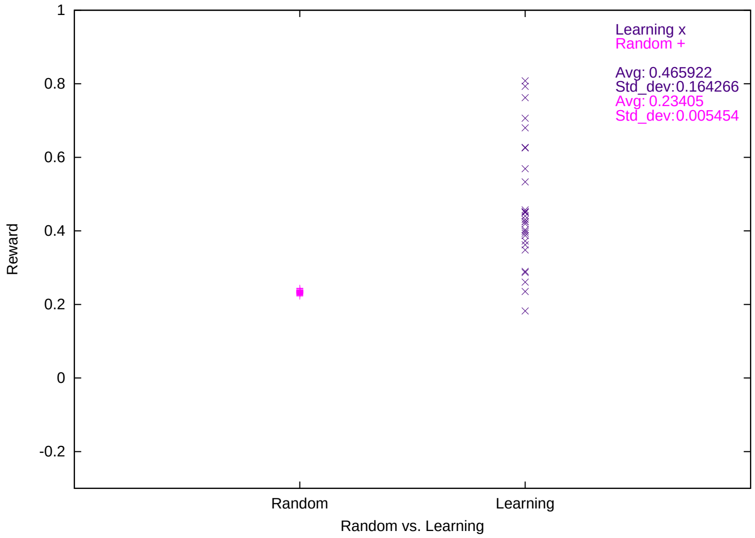

Figure 3.9: Validation Results for Nurturing Condition (Summary). Average rewards collected for 30 repetitions of 4000 trials each for the learning algorithm and for random ANN weights, both under the nurturing condition.

Starting with an overall summary of learning versus random, Figure 3.9 summarizes the results of learning versus random behavior in the nurturing condition. It shows that the learning algorithm outperformed random neural weights in approximately 93% of repetitions and that the average of 30 repetitions for learning (0.465) is better than that for random (0.23). These results are also statistically significant (t-test, p <0.0001). This gives us confidence that evolution will have sufficient opportunity to arrive at reasonable learning parameters in the evolutionary experiments.

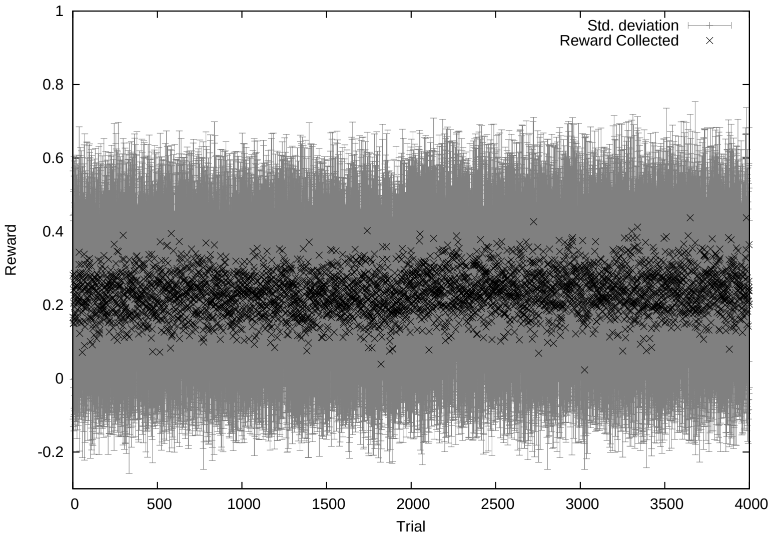

Figure 3.10: Validation Results for Nurturing Random Condition (Average). Mean and standard deviation of the reward collected by 30 individuals across a lifetime (4000 trials).

Looking at the results for random individuals, Figure 3.10 shows the average reward collected across 30 repetitions of each trial. As expected, the graph demonstrates poor average performance without discernible improvement.

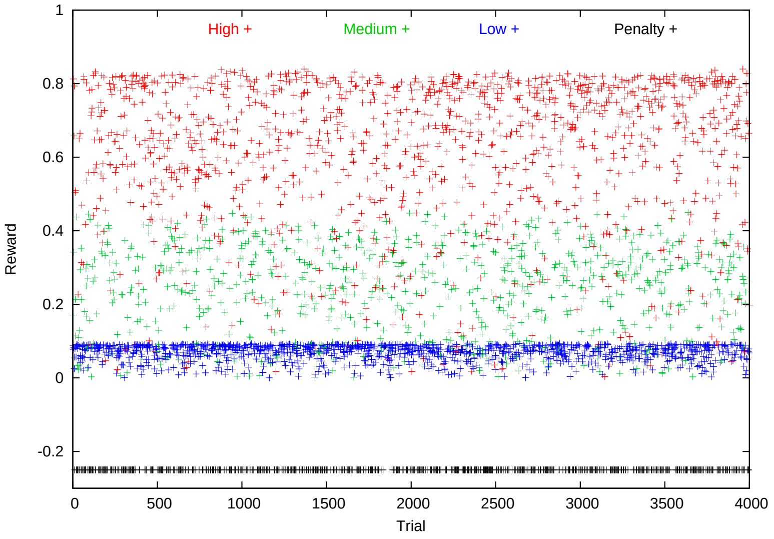

Figure 3.11: Validation Results for Nurturing Random Condition (Typical Individual). Reward collected in each of 4000 trials by an individual with random weights.

Figure 3.11 shows typical results for a random individual in the arena. It almost equally tries all the light sources throughout its lifetime regardless of reward received. Likewise, it frequently fails to reach any light before a trial ends, resulting in the individual receiving many penalties throughout its lifetime. As the weight means and standard deviations are initialized to random values every trial, these behaviors are expected[4].

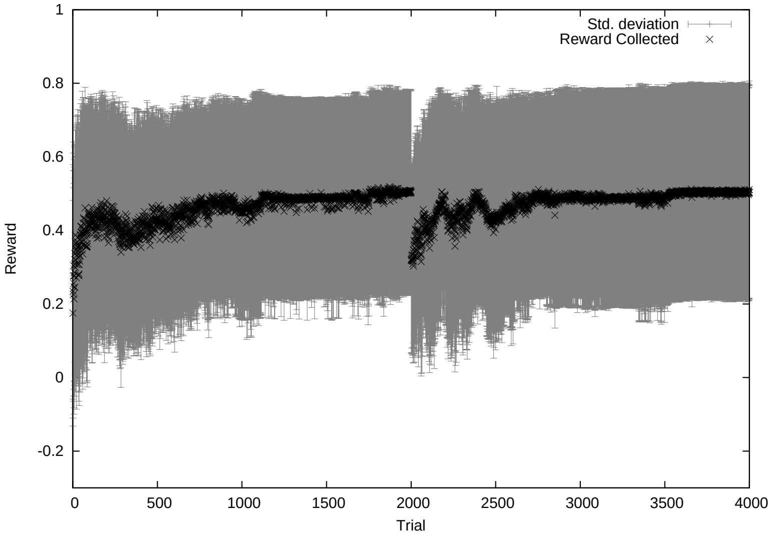

Figure 3.12: Validation Results for Nurturing Learning Condition (Average). Mean and standard deviation of the reward collected by 30 individuals across a lifetime (4000 trials).

Moving on to results for learning individuals, Figure 3.12 shows the average reward received across 30 repetitions of each trial. The graph shows that there is an upward trend of learning in both halves of the lifetime of the individuals. A drop in average reward collected can be noticed at trial 2000 and immediately following, due to the switch in the high and low rewarding lights. Note that, this drop was expected.

[4] Note that in all the individual graphs: (1) red represents reward received from the high rewarding light source (theoretical max = 0.9), (2) green represents reward received from the medium rewarding light source (theoretical max = 0.5), (3) blue represents reward received from the lowest rewarding light source (theoretical max = 0.1). This explanation should avoid any confusion in the color of the lights in the arena and the graph color representations as they are distinct pieces of information.

Unlike the random individuals that all behave very similarly to one another, there are a variety of behaviors exhibited by the individuals that use the learning algorithm.

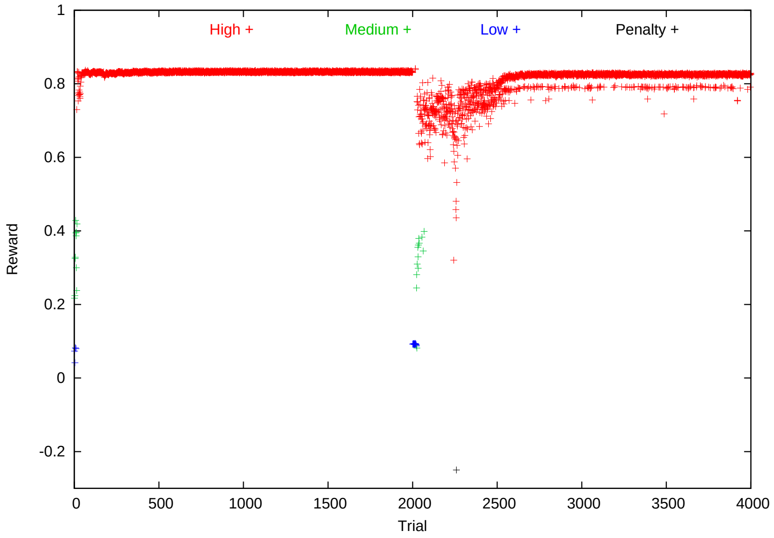

Figure 3.13: Validation Results for Nurturing Learning Condition (Typical Good Individual). Reward collected in each of 4000 trials.

Figure 3.13 presents a typical good learning individual that explores the arena a little in the beginning of its lifetime and then quickly focuses on the best rewarding light source. Half way through the lifetime at 2000 trials, when the reward for light source being exploited is switched from high to low, it again quickly adjusts to the new high rewarding light source and updates its path to get there.

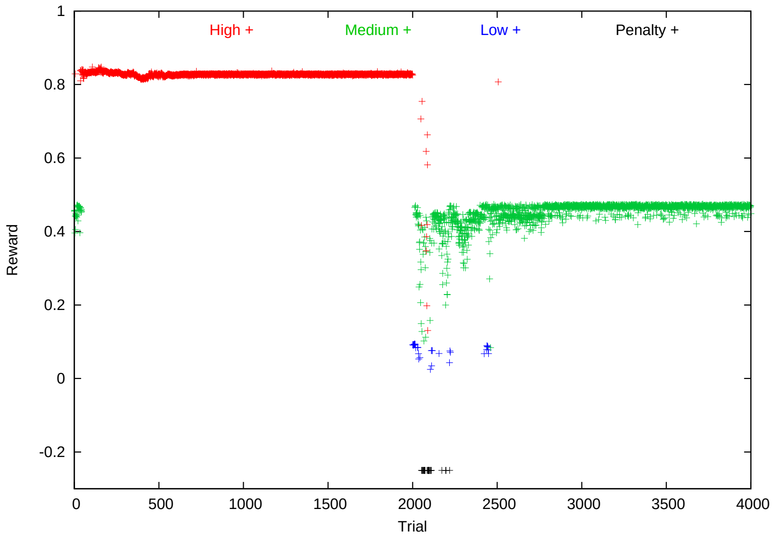

Figure 3.14: Validation Results for Nurturing Learning Condition (Typical Moderate Individual). Reward collected in each of 4000 trials.

Figure 3.14 depicts a typical moderate learning case where the individual has good initial random weights for going to the medium rewarding light source yet quickly finds the high rewarding light and changes its path to exploit that. However, once the change in reward happens at 2000 trials, the individual fails for a few trials before actually learning to get to a better light source. In this case, the individual never exploits the high rewarding light source during the second half of its lifetime even though it does encounter that light source during that period; however, the individual finds the next best rewarding light source and exploits that.

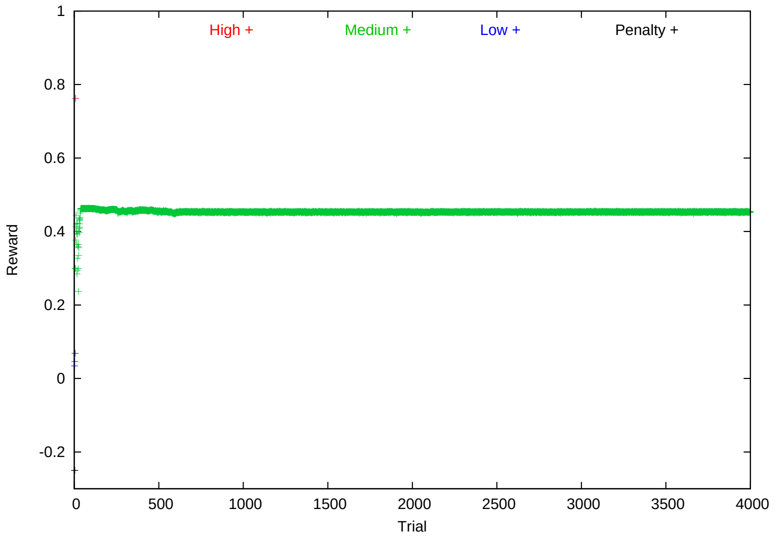

Figure 3.15: Validation Results for Nurturing Learning Condition (Typical Moderate Individual). Reward collected in each of 4000 trials.

Figure 3.15 shows another typical moderate learning case but of a different type. This individual explores all three light sources in the beginning of its lifetime and then chooses the medium rewarding light. Since the medium rewarding light offers a consistent reward throughout the lifetime of the individual, the individual performs moderately throughout its lifetime and does not alter its behavior when the low and high rewarding lights swap values at Trial 2000.

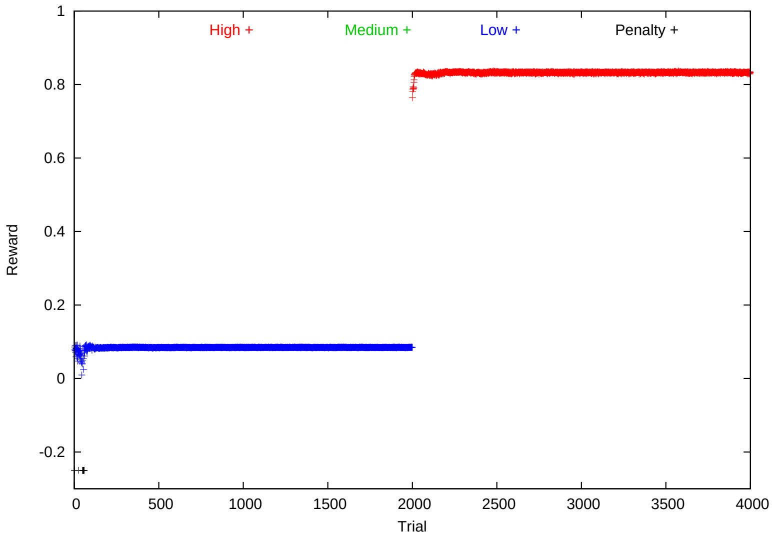

Figure 3.16: Validation Results for Nurturing Learning Condition (Typical Non-Substantial Individual). Reward collected in each of 4000 trials.

Figure 3.16 shows a typical non-substantial learning case where the individual experiences a low rewarding light as well as failures and learns to go to the low rewarding light. However, it never explores sufficiently to discover the medium or high rewarding lights and continues to exploit the low reward until the change in the environment makes the low rewarding light a high rewarding source.

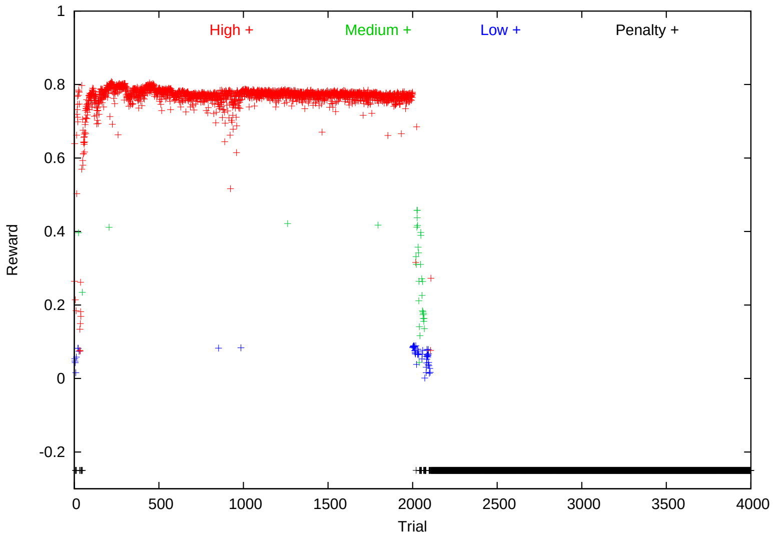

Figure 3.17: Validation Results for Nurturing Learning Condition (Typical Failed Individual). Reward collected in each of 4000 trials.

Finally, Figure 3.17 shows one of the two individuals that did not exhibit much learning. Here early exploration of all the lights and exploitation of the high rewarding light source in the first half of its life makes this individual a good learner during the first half of its lifetime. However, this individual is not able to cope with the change and its weight adjustments result in consistently poor behavior very soon after the high rewarding light source it was utilizing becomes the low rewarding light.

Self-Care — Learning vs. Random Behavior

This section presents results from 30 repetitions for learning vs. random for the non-nurturing (self-care) condition. Again, the intent is to find out the number of learning cases that outperform random behavior and also to determine if the learning algorithm works well enough for the self-care condition so that evolution can be introduced next.

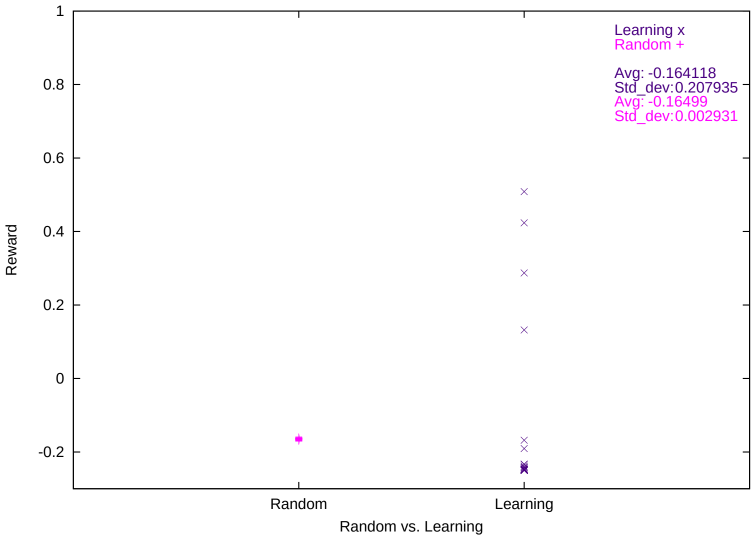

Figure 3.18: Validation Results for Self-Care Condition (Summary). Average rewards collected for 30 repetitions of 4000 trials each for the learning algorithm and for random ANN weight means and standard deviations, both under the self-care condition.

Figure 3.18 shows that 4 of the learning cases outperformed random behavior. This supports the idea that evolution will have sufficient opportunity to arrive at reasonable learning parameters in the evolutionary experiments using the spread of the learning cases shown. These results are not statistically significant (t-test, $p = 0.98$).

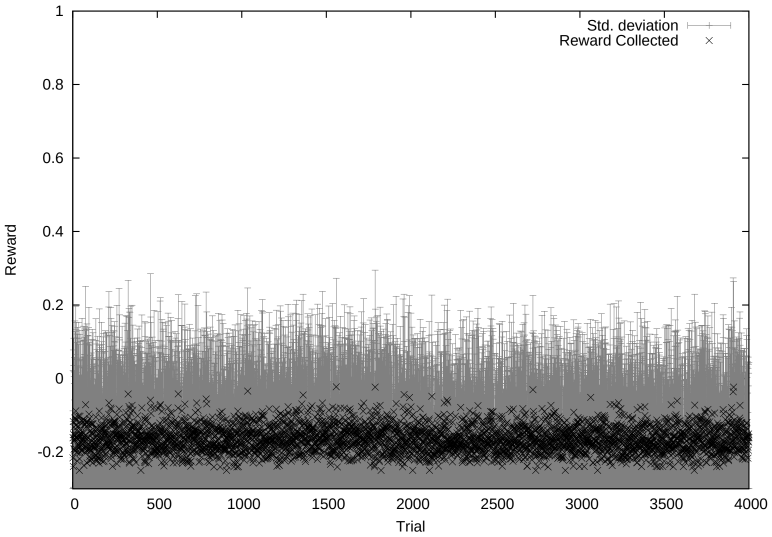

Figure 3.19: Validation Results for Self-Care Random Condition (Average). Mean and standard deviation of the reward collected by 30 individuals across a lifetime (4000 trials).

Looking at the results for randomly weighted individuals, Figure 3.19 shows the average reward collected across 30 repetitions of each trial. As expected, the graph demonstrates very poor average behavior with no discernible improvement. As can be seen, the average reward values in each trial here are lower as compared to the nurturing random case (Figure 3.10) due to the fact that in the non-nurturing case the lights in the arena are initialized to off at the start of each trial.

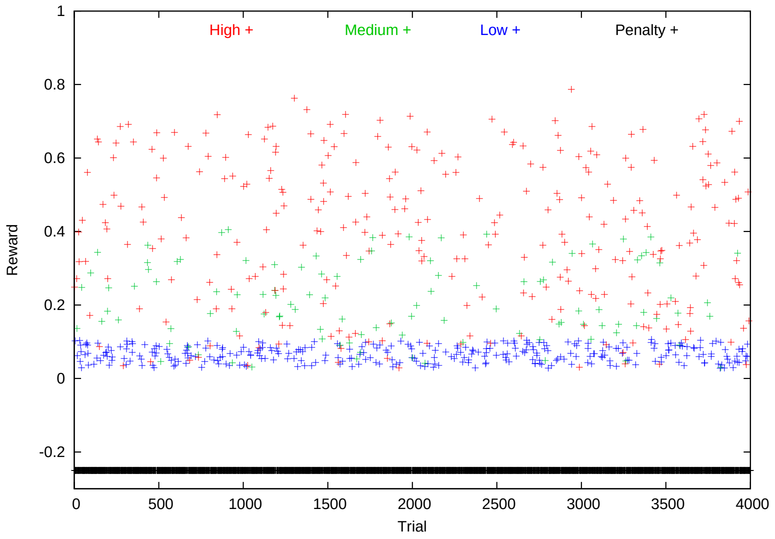

Figure 3.20: Validation Results for Self-Care Random Condition (Typical Individual). Reward collected in each of 4000 trials.

Figure 3.20 shows typical behavior by a random individual in the arena. As with random individuals in the nurturing condition, this individual almost equally tries all the light sources throughout its lifetime regardless of the reward received. However, in contrast to the results for the nurturing condition, the random individuals here receive penalties far more often due to the fact that the full task is more difficult than the partial task required in the nurturing niche. As the initial weight means and standard deviations are initialized to random values every trial, this behavior is expected.

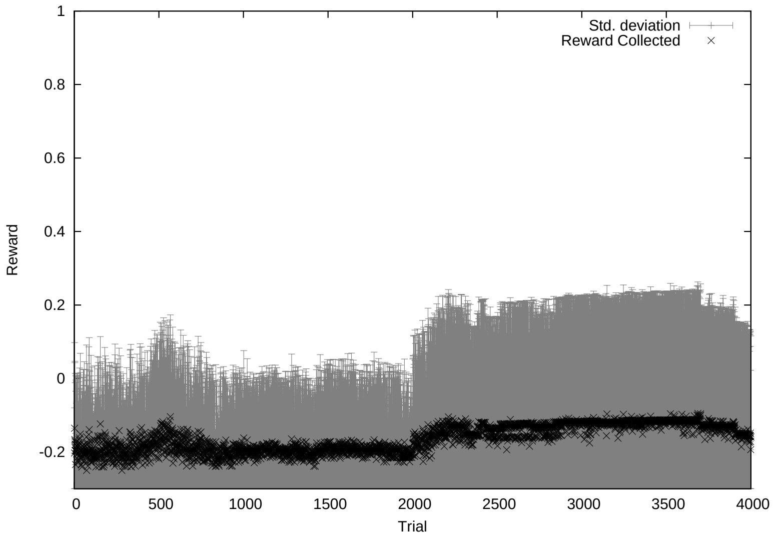

Figure 3.21: Validation Results for Self-Care Learning Condition (Average). Mean and standard deviation of the reward collected by 30 individuals across a lifetime (4000 trials).

Moving on to results for learning individuals, Figure 3.21 shows the mean and standard deviation across 30 repetitions of each trial. The graph appears to show a slight average upward learning trend especially in the second half of the lifetimes although the mean fitness/reward collected is quite low. This low value is expected as the individuals either exhibit self care — carrying out the full task with no assistance — which takes more time than the partial task present when being nurtured, or fail to exhibit self-care, which results in receiving a penalty.

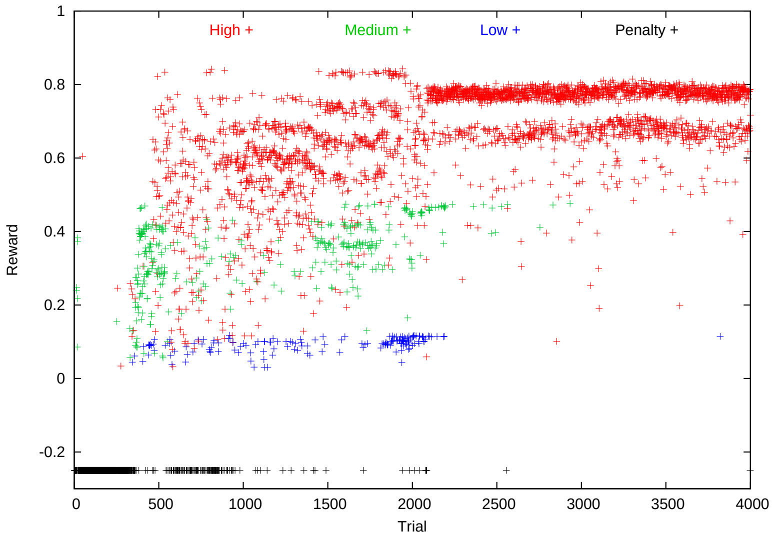

Figure 3.22: Validation Results for Self-Care Learning Condition (Typical Moderate Individual). Reward collected in each of 4000 trials.

Figure 3.22 shows a moderate learning individual that does not perform very well initially but eventually learns to go to the high rewarding light source during the first half of its lifetime. In the second half of its lifetime it quickly explores and finds the high rewarding light and becomes mostly conservative after that.

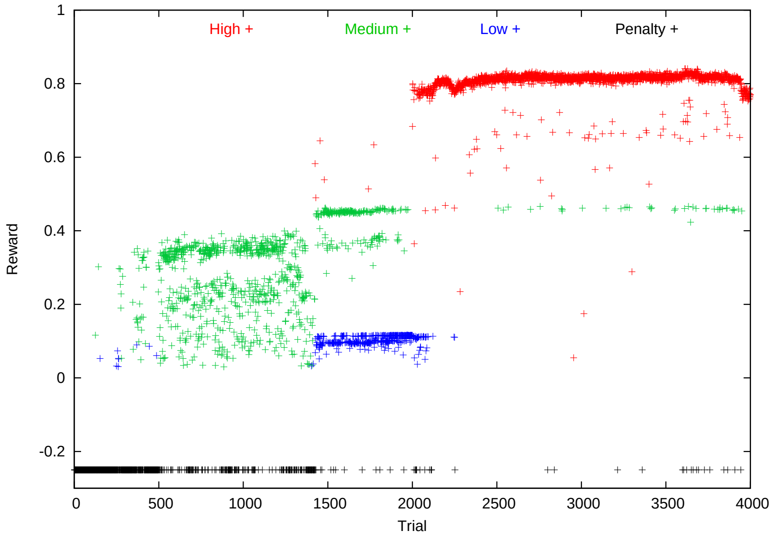

Figure 3.23: Validation Results for Self-Care Learning Condition (Typical Non-Substantial Individual). Reward collected in each of 4000 trials.

Figure 3.23 is a typical example of a poor learning case where the individual recovers from initial failures and explores all three lights sources. In the last 500 or so trials in the first half of its lifetime, the individual mostly settles on a low rewarding light which is considered poor behavior. When the low rewarding light becomes a high rewarding source in the second half of the lifetime, the individual keeps going to the same light and occasionally explores other sources with some failures.

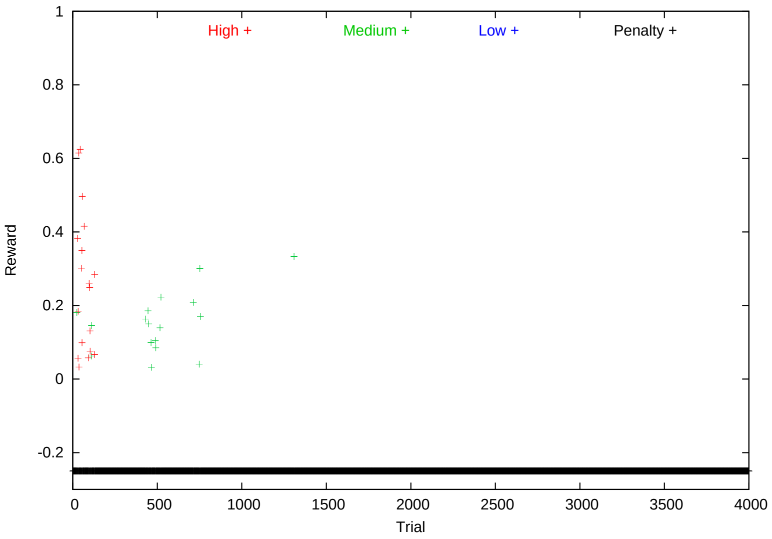

Figure 3.24: Validation Results for Self-Care Learning Condition (Typical Failed Individual). Reward collected in each of 4000 trials.

Figure 3.24 shows a typical example of an individual that mostly receives penalties throughout its lifetime even after exploring the high rewarding light source initially followed by the medium rewarding light.

3.3 Summary

This chapter discusses the design of the proposed RL algorithm in detail, followed by its validation for the task to be learned in the evolutionary experiments. The results suggest that it is reasonable to move forward with the validation of the hypotheses in the next chapter.

Chapter 4

Experimental Design

In this chapter, the hypothesis will be formally introduced. Next, the hypothesis will be translated into the experimental design in two main directions i.e., the evolution of learning and the evolution of learning and instincts. Further, the design of a genetic algorithm as an evolutionary computation method will be discussed followed by its implementation in both directions mentioned above.

4.1 Hypotheses

This study hypothesizes that nurturing promotes the evolution of learning. What this means is that in the nurturing niche, learning is more likely to be useful and therefore apparent than it is in the non-nurturing niche. If this hypothesis is true it could manifest itself in two primary ways: First, nurturing might improve the likelihood of evolving worthwhile learning. Secondly, performance of the evolved learning might be better in the nurturing niche than it is in the non-nurturing niche. We can further think of this either categorically, (with several possible categories of learning system performance) or in terms of reward received (a performance continuum) and also whether the individual's behavior is entirely learned or could be influenced by instincts. Considering the various possible combinations of each of these aspects of the hypothesis gives numerous possible sub-hypotheses. This section briefly introduces the learning performance categories used in this dissertation and then presents the sub-hypotheses considered.

4.1.1 Categories of Learners

To be able to objectively classify learning performance, I define two major categories of learners, substantial and non-substantial learners. I also define two sub categories of substantial learners, good learners and moderate learners[1].

1. A *substantial learner* is an individual who collects a lifetime average reward higher than that of the theoretical best instinctive individual. Within substantial learning there are two further categories:

    1.1. A *good learner* is a substantial learner whose average reward in each half of its life is higher than the average lifetime reward of the theoretical best instinctive individual.

    1.2. A *moderate learner* is a substantial learner whose average reward in exactly one half of its life is higher than the average lifetime reward of the theoretical best instinctive individual.

2. A *non-substantial learner* is an individual whose lifetime average reward is equal to or lower than that of the theoretical best instinctive individual. This includes an individual whose average reward in atmost one of its halves is higher than and overall lower than the average lifetime reward of the theoretical best instinctive individual.

4.1.2 Sub-Hypotheses

All of the hypotheses are tested using reward data, which is continuous data. However, for the first set of hypotheses (category likelihood) the data is discretized into the listed categories and counting of the number of occurrences of each category is performed. So, the first set of hypotheses is based on categorical data; this data can

[1] Note that the terms found in this list are operationalized in Section 5.1

also be understood as ordinal data. In contrast, for the second set of hypotheses, the plan is to do the statistical comparisons based on the continuous data itself. Still, almost all of the sub-hypotheses here look only at subsets of the data, and those subsets are based on the categories. Moreover, the categories come from discretizing continuous data. Thus these two types of hypotheses/results are called *category likelihood* (or *category frequency*) and *performance continuum.*

The sub-hypotheses are summarized in Table 4.1 for those hypotheses related to category likelihood and Table 4.2 for those hypotheses related to the performance continuum.

		Instincts		
		Absent	**P**resent	**E**ither/Both
Comparisons	**S**ubstantial	**CL-SA** (1.1.1)	**CL-SP** (1.2.1)	**CL-SE** (1.3.1)
	Good	**CL-GA** (1.1.2)	**CL-GP** (1.2.2)	**CL-GE** (1.3.2)
	Better	**CL-BA** (1.1.3)	**CL-BP** (1.2.3)	**CL-BE** (1.3.3)

Table 4.1: Category Likelihood Hypothesis summary — **CL** stands for **C**ategory **L**ikelihood hypotheses. **S**ubstantial comparison: (2-way Substantial vs. Not Substantial). **G**ood comparison: (2-way Good vs. Not Good). **B**etter comparison: (3-way Good vs. Moderate vs. Non-Substantial). Abbreviations **A**, **P**, and **E** stand for Absent, Present, and Either instincts respectively.

		Instincts		
		Absent	**P**resent	**E**ither/Both
Comparisons	**O**verall	**PC-OA** (2.1)	**PC-OP** (2.2)	**PC-OE** (2.3)
	Good	**PC-GA** (2.1.1)	**PC-GP** (2.2.1)	**PC-GE** (2.3.1)
	Substantial	**PC-SA** (2.1.2)	**PC-SP** (2.2.2)	**PC-SE** (2.3.2)
	Moderate	**PC-MA** (2.1.3)	**PC-MP** (2.2.3)	**PC-ME** (2.3.3)
	Non-Substantial	**PC-NA** (2.1.4)	**PC-NP** (2.2.4)	**PC-NE** (2.3.4)

Table 4.2: Performance Continuum Hypothesis summary — **PC** stands for **P**erformance **C**ontinuum hypotheses. **O**verall comparison, **G**ood comparison, **S**ubstantial comparison, **M**oderate comparison, **N**on-Substantial comparison. Abbreviations **A**, **P**, and **E** stand for Absent, Present, and Either instincts respectively.

Writing out these sub-hypotheses in list form gives the following:

1. Learning is more likely to evolve with nurturing than without nurturing.

    1.1. Learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

        1.1.1. **CL-SA** Substantial learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

        1.1.2. **CL-GA** Good learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

1.1.3. **CL-BA** Better learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

1.2. Learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

1.2.1. **CL-SP** Substantial learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

1.2.2. **CL-GP** Good learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

1.2.3. **CL-BP** Better learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

1.3. Learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

1.3.1. **CL-SE** Substantial learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

1.3.2. **CL-GE** Good learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

1.3.3. **CL-BE** Better learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

2. Nurturing promotes the evolution of higher performing behaviors.

2.1. **PC-OA** Nurturing promotes the evolution of higher performing behaviors in the absence of instincts.

2.1.1. **PC-GA** Nurturing promotes the evolution of higher performing behaviors within good learners in the absence of instincts.

2.1.2. **PC-SA** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the absence of instincts.

2.1.3. **PC-MA** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the absence of instincts.

2.1.4. **PC-NA** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the absence of instincts.

2.2. **PC-OP** Nurturing promotes the evolution of higher performing behaviors in the presence of instincts.

2.2.1. **PC-GP** Nurturing promotes the evolution of higher performing behaviors within good learners in the presence of instincts.

2.2.2. **PC-SP** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the presence of instincts.

2.2.3. **PC-MP** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the presence of instincts.

2.2.4. **PC-NP** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the presence of instincts.

2.3. **PC-OE** Nurturing promotes the evolution of higher performing behaviors regardless of instincts.

2.3.1. **PC-GE** Nurturing promotes the evolution of higher performing behaviors within good learners regardless of instincts.

2.3.2. **PC-SE** Nurturing promotes the evolution of higher performing behaviors within substantial learners regardless of instincts.

2.3.3. **PC-ME** Nurturing promotes the evolution of higher performing behaviors within moderate learners regardless of instincts.

2.3.4. **PC-NE** Nurturing promotes the evolution of higher performing behaviors within non-Substantial learners regardless of instincts.

4.1.3 Multiple Comparisons

Any time a study involves multiple comparisons, it is sensible to ask whether a multiple comparison problem is present (and, if so, how to address it). The *multiple comparison problem* is the problem of taking a statistical method that is appropriate for the analysis of a single comparison and naively applying it to multiple comparisons. There are many types of multiple comparison problems but they share the common feature that the statistical result found will not be appropriate for the data considered unless it is adjusted or interpreted for the multiple comparisons case. Typical examples of the multiple comparisons problem in machine learning are comparing multiple algorithms on the same data set, comparing two algorithms across multiple data sets, and sub sampling results in multiple ways when there is no effect found at the aggregate level. In these situations, the primary concern is generally that by using the selected statistical hypothesis test(s) with multiple comparisons, one will reject the null hypothesis unjustifiably (that is, that one will make a Type I error).

In the present study, we have a single main hypothesis that can be manifested in one of two ways (category likelihood and performance continuum), may be influenced by the absence or presence of instincts, and may be looked for in subsets of the data as well as at the aggregate level.

Considering category likelihood versus performance continuum results, these are complementary ways of looking at the same data. This means that results that comport with one another across hypothesis categories would tend to give support to different aspects of the main hypothesis while mixed results (rejecting a null hypothesis related to category data but not for corresponding performance continuum data or vice versa) would tend to provide additional insight into the influence of nurturing on the evolution of learning, rather than simply resulting in false positives in the conclusions.

In more detail, it would be logically possible for there to be the same number of instances of each learning category for the nurturing and self-care niches while still showing performance differences within each category and/or overall. This is because the category likelihood data is coarse grained (with only three categories, at most), whereas the performance continuum data is fine grained (a continuum). There might, for example, be a ceiling (or floor) effect with regard to the categorical data, where all or most of the samples are in the good (or non-substantial) category, yet the effect might not be so overwhelming as to entirely obscure the performance continuum results. In this case, if the hypothesis were supported in the performance continuum results, this would suggest that nurturing does promote the evolution of learning but that the extent of that influence is unclear due to the ceiling (or floor) effect and that additional experiments in a more difficult (or easier) environment should be conducted to avoid the ceiling (or floor) effect in order to determine the extent of the influence. Alternately, there might be neither a ceiling nor a floor effect, yet the category membership counts might be roughly equal between the niches. In this case, a result rejecting the null hypothesis for the performance continuum data but not for the corresponding category likelihood data would support the main hypothesis; however, because it doesn't push the performance of the learners across category boundaries it would suggest that the extent of the influence is not large.

Likewise, it would be possible to have no average performance difference, whether overall or within categories, while still having differences in instance counts for each category between the two niches. In the overall case, this would require a difference in standard deviations between the distributions. In this case, if the test results tend to support the hypothesis for the categorical data, then it would appear that nurturing influences the diversity of the results in such a way that more repetitions are classified into more favorable learning categories but that this outcome might be idiosyncratic because with different category boundaries greater or lesser diversity might result in more favorable counts. Regarding the within-category results, this could indicate that there are thresholds in the evolutionary environment that correspond to the category thresholds established for objective classification purposes, and that until a lineage breaks into a higher category, nurturing has little effect. However, if the null hypothesis were rejected in the categorical case, that would suggest that the effect of nurturing is to help the evolutionary process to cross those evolutionary boundaries.

Considering next the results with respect to instincts, we have expectations for how instincts may influence the outcomes. If these expectations are violated in the statistical hypothesis test results, that is an indication that the results may be spurious. Otherwise, there is little cause for concern.

In more detail, we expect that the addition of instincts will provide more benefit in the self-care niche than in the nurturing niche. That is because almost all aspects of the nurturing niche vary both between and within each lifetime — the positions and reward values of the lights are randomly determined at trial 0 and again at half way through the lifetime (maximum-trial-size/2) for each individual. That means that inherited instincts related to these features are unlikely to be helpful. One of the only constants in the nurturing niche is that the lights are always to the individual's right at the start of the trial, so an instinct to turn right when no lights are visible might be helpful. In contrast, the self-care niche requires the individual to carry out another (partial) task in the environment — it needs to turn on the switch before going to the lights. The switch is always the same color (until it is turned on) and in the same position and thus it is amenable to being handled instinctively. Indeed, Leonce et al. (2012) showed that instincts for turning on the light can be easily and effectively evolved for a similar (albeit non-learning) neural controller. Thus, if we see the null hypothesis rejected in a case involving evolved instincts but not for the corresponding case where instincts cannot be evolved, that would violate our expectations and raise concerns about spurious results. On the other hand, if the null hypothesis is rejected in a case where instincts cannot be evolved but is not rejected in the corresponding case involving evolved instincts, this would point to instincts as a possible promoter of the evolution of learning, at least where the environment involves an important constant component.

Finally, considering the hypotheses that use only subsets of the data, if the null hypothesis cannot be rejected at the higher aggregate level and we continue to subdivide the data and conduct hypothesis tests on the smaller subsets, that would present a classic multiple comparisons problem because smaller data sets are more subject to the effects of random noise. For this reason, if the null hypothesis is not rejected at the higher aggregate level, we will flag all tests of the sub-hypotheses involving the subsets of that data as likely false positives.

4.2 Translating Hypotheses into Experimental Design

To answer the hypotheses, it is essential to introduce an evolutionary process through which the learning algorithm parameters can be evolved. Further, an option is needed to evolve instincts together with learning parameters to demonstrate that nurturing promotes the evolution of learning both with and without instincts. *Instincts* are innate patterns of behavior that manifest themselves in response to certain stimuli. In the ANN control systems used in this dissertation, instincts correspond to the initial mean values of the synapse weights, as these are the primary determinants of an individual's behavior unless and until they are adjusted based on experience. In the experiments in which only learning rule parameters may be evolved, the synapse weight mean values are randomly initialized. This means that an individual cannot inherit its instincts from its ancestors, which means that instincts cannot be evolved. In these experiments, only the learning rule parameters are encoded in each individual's chromosome, so only learning can be evolved. In contrast, for those experiments in which we want to allow learning and/or instincts to evolve, the initial mean values of each synapse weight are also encoded in each individual's chromosome.

This section describes the general evolutionary algorithm used to evolve learning rule parameters and (optionally) instincts, then describes the experiments for evolution of learning rule parameters and evolution of learning rule parameters and instincts.

4.2.1 Genetic Algorithm (GA)

A generational genetic algorithm in which fitness is defined to be the total reward collected in the arena by an individual during its lifetime is shown in Algorithm 4.1. Figure 4.1 also shows the general workings of the class of GAs used in this dissertation. Here is how the GA works:

1. Chromosomes for all individuals in the starting population (generation 0) are randomly initialized.

2. All individuals are evaluated independently.

3. After all individuals are evaluated, selection is performed to determine the composition of the next generation.

    3.1. First, zero or more individuals are copied without changes to the next generation in order of fitness. These unaltered copies of the most fit individuals are known as *elites*.

    3.2. Next, *clones* are added to the new generation. Clones differ from elites in that clones are not necessarily the most fit individuals from the population and they may undergo mutation. For each clone, a tournament bracket of size b is formed and b individuals are selected at random (with replacement) from the population to fill it. The individual in the tournament

Algorithm 4.1: Genetic Algorithm for the evolution of learning with optional instincts.

```
Algorithm Evolve(population, poolSize, popSize, eliteSize, crossoverSize)
    Init (population)
    for gen ← 0 to NumGenerations do
        EvaluateFitness (population)
        Sort (population)
        /* Elites: Copy over the best individuals                  */
        for indv ← 0 to eliteSize do
            newPop[indv] ← population[indv]
        end
        /* Clones: Select and mutate                               */
        for indv to (popSize - crossoverSize) do
            winner ← Tournament (popSize, poolSize)
            newPop[indv] ← MutateGenes (population[winner])
        end
        /* Reproduction: Select, crossover, and mutate             */
        for indv to popSize, indv ← indv + 2 do
            winner1 ← Tournament (popSize, poolSize)
            winner2 ← Tournament (popSize, poolSize)
            newPop[indv], newPop[indv + 1] ← UniformCrossOver
                (population[winner1], population[winner2])
            newPop[indv] ← MutateGenes (newPop[indv])
            newPop[indv + 1] ← MutateGenes (newPop[indv + 1])
        end
        population ← newPop
    end
    return
```

Listing 4.2: Genetic Algorithm helper functions — selection, crossover, mutation.

```
Procedure Tournament($popSize$, $poolSize$)
    $winner \leftarrow popSize$
    for $i \leftarrow 0$ to $poolSize$ do
        $randSelection \leftarrow$ rand () MOD $popSize$
        /* Lower index means higher fitness                        */
        if $randSelection < winner$ then
            $winner \leftarrow randSelection$
        end
    end
    return $winner$
Procedure MutateGenes($chromosome$)
    for $i \leftarrow 0$ to $chromosome.length$ -1 do
        if (rand () MOD 100) <$mutationRate$ then
            $newGene$ = SampleDistribution ($chromosome[i]$, $mutationSigma$)
            $chromosome[i]$ += TruncateToLimits ($newGene$, $min$, $max$)
        end
    end
    return $chromosome$
Procedure UniformCrossOver($chromosome1$, $chromosome2$)
    if $chromosome1.length \neq chromosome2.length$ then
        return ERROR
    end
    /* For each gene                                               */
    for $i \leftarrow 0$ to $chromosome1.length$ -1 do
        /* Flip a coin and see if chromosome1 wins                 */
        if (rand () MOD 100) <50 then
            /* Gene from chromosome1 copies into newChromosome1    */
            /* Gene from chromosome2 copies into newChromosome2    */
            $newChromosome1[i] \leftarrow chromosome1[i]$
            $newChromosome2[i] \leftarrow chromosome2[i]$
        else
            /* Gene from chromosome2 copies into newChromosome1    */
            /* Gene from chromosome1 copies into newChromosome2    */
            $newChromosome1[i] \leftarrow chromosome2[i]$
            $newChromosome2[i] \leftarrow chromosome1[i]$
        end
    end
    return $newChromosome1$, $newChromosome2$
```

with the highest fitness is selected as the winner. The winner is cloned, possibly with mutation, and the clone is placed into the new generation. This process is repeated until the desired number of clones has been added to the new generation.

3.3. Finally, non-clonal offspring are added to the generation. These offspring are generated by performing two tournaments to find two winners and then using uniform crossover on the two winners to produce two offspring. Each offspring then has a chance of undergoing mutation. The process repeats until the size limit of the new population is reached. Note that the same individual can win multiple tournaments, thus it can crossover with itself to generate two offspring.

4. During mutation, there is a small chance that a given gene will be mutated. If selected for mutation, a normal distribution with zero mean is used to select the value to be added to the mutated gene. If the mutation would result in an allele outside the gene range limits (if any), the allele is set to be equal nearest limit value.

5. The algorithm runs for a fixed number of generations.

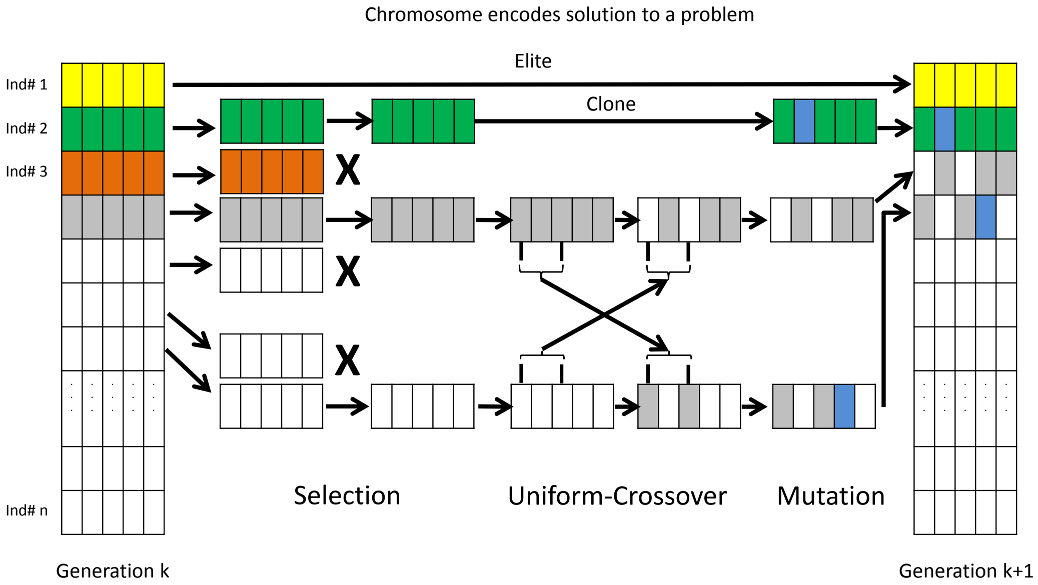

Figure 4.1: Visual Representation of Genetic Algorithm. Genetic algorithm used as an evolutionary process to find multiple optima.

4.2.2 Evolution of Learning

The first experiment will study the evolution of learning when evolved instincts are not possible. The learning rule parameters to be evolved are

1. ${}_{\mu}\eta$, learning rate for synaptic weight mean, μ,
2. ${}_{\sigma}\eta$, learning rate for synaptic weight standard deviation, σ,
3. ${}_{\mu}d$, discount rate for synaptic weight mean eligibility traces,
4. ${}_{\sigma}d$, discount rate for synaptic weight standard deviation eligibility traces,
5. ${}_{init}\sigma$, initial exploration rate (initial synaptic weight standard deviation),
6. ${}_{min}\sigma$, minimum exploration rate allowed (lower bound),
7. ${}_{max}\sigma$, maximum exploration rate allowed (upper bound), and

8. s, sliding window size.

These eight learning rule parameters are encoded in each chromosome. Each parameter, a gene in the chromosome, is a randomly generated value between 0 and 1 (inclusive). However, before the beginning of an individual's lifetime scaling is required for some learning parameters to make them algorithmically plausible in the context of learning. The scaling details are shown in Table 4.3.

Name	Symbol	Calculation
Mean (μ) Learning Rate	${}_{\mu}\eta$	${}_{\mu}\eta = g_0$
Standard Deviation (σ) Learning Rate	${}_{\sigma}\eta$	${}_{\sigma}\eta = g_1$
Minimum Sigma	${}_{min}\sigma$	${}_{min}\sigma = 0.5g_2$
Maximum Sigma	${}_{max}\sigma$	${}_{max}\sigma = 0.5g_3 + 0.5$
Initial Sigma	${}_{init}\sigma$	${}_{init}\sigma = g_4({}_{max}\sigma - {}_{min}\sigma) + {}_{min}\sigma$
Sliding Window Size	s	$s = g_5(TrialSize)$
Mean (μ) Decay	${}_{\mu}d$	${}_{\mu}d = 10^{2g_6-1}$
Standard Deviation (σ) Decay	${}_{\sigma}d$	${}_{\sigma}d = 10^{2g_7-1}$

Table 4.3: Learning parameter symbols and descriptions. Learning parameter symbols and their calculated scaled values. Here g_l is the gene at locus l.

In the table above, learning rates ${}_{\mu}\eta$ and ${}_{\sigma}\eta$ do not need any scaling as a number between 0 and 1 is a valid learning rate for both μ and σ. Minimum sigma is scaled to be in the range [0, 0.5] and maximum sigma is scaled to be in the range [0.5, 1]. This ensures that the minimum exploration rate is in the lower half of the range of possible exploration rates while the maximum exploration rate is in the upper half of possible exploration rates. Initial sigma is scaled to make sure that it is between the minimum and maximum scaled sigma values. Sliding window size s is multiplied by the trial size to ensure that minimum size of the sliding window is zero and the maximum size is the length of an entire episode. The final two parameters are decay rates for μ and σ. Both of these parameters are scaled the same way and are described using

$$d = 10^{2g_l - 1}, \tag{4.1}$$

where g_l is the appropriate gene with a value sampled from a uniform distribution in [0, 1]. (This scales the value of d to [0.1, 10].). A normalization factor ν is derived from d using

$$\nu = 1/d^0 + d^1 + d^2 + ... + d^{T-1}, \tag{4.2}$$

where T is the maximum time steps in a trial (The normalization factor is the sum of a geometric series).

The value of d in the above set of equations is the value that is considered the decay rate and is used to calculate eligibility values at all the time steps. To apply the normalization factor ν, assuming that the above calculations are performed for ${}_{\mu}d$ for the sake of example, then the normalization factor can be applied as follows:

$$\Delta\mu_{ij}(\tau) = {}_{\mu}\eta \left(r(\tau) - \overline{r}(\tau)\right) \sum_{k=1}^{t} {}_{\mu}e_{ij}(k) {}_{\mu}d^{(t-k)} \nu. \tag{4.3}$$

In Equation 4.3, the normalization factor ν keeps the total of the discount factors applied to the eligibilities at less than or equal to one. A gene value g used in the above equations will function as follows:

$g < 0.5$ means give more importance to the recent actions in this trial,

$g = 0.5$ means give equal importance to all the actions in this trial,

$g > 0.5$ means give more importance to the earlier actions in this trial.

GA Implementation

Now that we are familiar with all the learning parameters represented by genes in the individual's chromosome, the discussion on the genetic algorithm parameters can proceed. Various parameters chosen for the genetic algorithm are listed in Table 4.4. All of the learning algorithms[2] are compared and the successful ones are passed to the next generation. Ten generations were determined to be sufficient for the evolutionary courses to diverge in the two niches (nurturing and self-care). The GA Algorithm 4.1 is implemented as follows:

1. Chromosomes for all learning algorithms in the starting population (generation 0) are randomly initialized to be in [0, 1].

2. The initialized gene values in the chromosomes are scaled using Table 4.3.

3. All individual learning algorithms are evaluated independently using Algorithm 3.1.

4. After all individuals are evaluated, selection is performed as follows:

   4.1. Elites, in the order of best fitness (accumulated reward collected) are copied to the next generation to ensure that the evolutionary process keeps the best algorithms found so far in the solution's landscape.

   4.2. Cloned learning algorithms, after possible mutation, are added to the new population.

   4.3. Finally, with uniform crossover to generate non-clonal offspring with possible slight mutations are added to the next generation. That should result

[2] Each learning algorithm is represented by an individual in the population.

in a diverse population that explores the undiscovered areas of the solution's landscape.

5. The algorithm runs until the number of generations is reached.

GA Parameter	Description/Value
Population Size	30
Number of Generations	10
Chromosome Length	8
Fitness	Total Reward Collected
Selection Method	Tournament with Replacement
Tournament Bracket Size	3
Crossover Type	Uniform
Crossover Percentage	73%
Reproduction Method	1 Elite, 7 Clones, 22 Crossed-over
Gene Mutation Rate	5% per Gene
Gene Mutation Standard Deviation	0.1
Mutation Method	Normal Distribution

Table 4.4: Genetic Algorithm Parameters used and their Descriptions/Values.

4.2.3 Evolution of Learning and Instincts

In the evolution of learning experiment Section 4.2.2, only learning rule parameters are evolved and initial weights of the ANN are initialized randomly. In this experiment, initial weights of the ANN are part of the chromosome together with the learning rule parameters. As the initial weights are passed from the parent population's successful individuals to the offspring with little or no change, they can be considered instincts. With the evolution of learning, the proposed learning algorithm is used in the evolutionary process to evaluate individuals. Similarly, in this experiment, instincts are added into the chromosome of each individual in the evolution of learning setup. The objective of this experiment is not only to answer the hypotheses' related to the evolution of learning and instincts but also to see if the main hypothesis still holds true after letting instincts to evolve together with learning. This would help to indicate the generality of this approach.

In the experiments where only learning is evolved (no instincts), the nurtured individual only has to learn one thing, i.e., to go to the high-rewarding light source whereas the non-nurtured individual has to learn two things, i.e., to go to the switch and then to the high-rewarding light. However, in the experiment where the evolution of both learning and instincts is allowed, both nurtured and non-nurtured individuals only need to learn one thing, i.e., to go to the high-rewarding light. This is because nothing changes about the switch either within or between lifetimes, i.e., switch position and behavior are constants. This means that a lineage could evolve instincts to turn on the switch and then individuals in that lineage would only need to learn about the lights. Allowing individuals to evolve instincts (for the non-changing parts of the environment) should aid in the evolution of learning in the non-nurtured niche. Nonetheless, the nurtured niche is still distinct from the non-nurtured niche. In the nurtured niche, the individual only needs to carry out one action, whereas in the non-nurtured niche the individual needs to carry out two actions. This still provides an advantage to individuals in the nurtured niche and therefore I expect useful learning to appear more often in the nurtured niche.

To evolve instincts together with learning parameters, 86 more genes are added to each individual's chromosome. These additional genes represent the initial mean values of the synaptic weights of the neural network. A set of 14 genes represent the synaptic weights between the 7 inputs and 2 outputs, i.e., Switch Off, Red Off, Red On, Green Off, Green On, Blue Off, Blue On, each connected to the left and the right motor. These 14 genes available for each of the 6 camera regions (see Table 3.7 for more details on those regions) makes a total of 14×6 = 84 genes to cover instinctive responses to stimuli. Further, a pair of genes to connect the bias unit to the two output units provides baseline instincts and thus adds two more to make the total 86. Finally, each chromosome also contains a gene representing each of the learning parameters, as before, also described in Table 4.3, which makes up 8 of the total 94 genes. All together it makes the chromosome length 94.

4.3 Summary

This chapter talks in depth about the proposed hypotheses and how these hypotheses are translated into a feasible design. Further, the discussion of the experimental design followed in conjunction with the hypotheses. Furthermore, discussion on a careful design together with the details on the proposed GA and the experiments is provided.

Chapter 5

Results

Although 4000 trials were used to verify the stochastic synapse learning algorithm, due to computational expense a smaller yet equally effective number of trials were found to use in the evolutionary experiments. Thus the results of 4000, 2000, and 1000 trials were compared. There was no notable loss of performance by going from 4000 to 2000; however, 1000 trials showed far less learning. Therefore, for both experiments, the evolution of learning and the evolution of learning and instincts, the option with 2000 trials is selected. All the results shown in this chapter are over 30 repetitions and for each repetition only the most fit individual is presented here.

5.1 Evaluation Criteria

It is important to recall (from Section 4.1.1) the boundary line drawn between acceptable and not acceptable learning is based on the performance possible with instinctive behaviors alone. Any learning that performs better than the best theoretically possible instinctive behavior on average at the end of an agent's lifetime falls under the category of *substantial learning*. In contrast, performance that is lower than or equal to that of the best theoretically possible instinctive performance will be called *non-substantial*. Further, recall that the category of substantial learning is further divided into *good* and *moderate* learning, for those individuals who outperform the best theoretically possible instinctive individual in both halves of their lifetimes and those substantial learning individuals who outperform the best theoretically possible instinctive individual in exactly one half of their lifetimes, respectively. To operationalize these terms with respect to these experiments, recall that the reward value for each trial is calculated using Equation 3.16. A search is performed for the lowest final time step value of any individual in the arena for both the nurturing and self-care niches. These values represent good approximations of the minimum amount of time in which an individual can complete the task(s) in each environment. That number is taken and 10% is added to that value to consider the possibility that the highest rewarding light might be located in the farthest corner[1] of the arena in order to calculate a fair value for each niche. Further, these numbers are used to calculate the maximum reward value using Equation 3.16 for the high, medium and low rewarding lights for both niches. The numeric comparison is given below:

Minimum estimated time steps	$47 + 10\% \approx 52$
Max possible high-reward	$((999 - 52)/999)*0.9 = 0.85$
Max possible medium-reward	$((999 - 52)/999)*0.5 = 0.47$
Max possible low-reward	$((999 - 52)/999)*0.1 = 0.09$

Table 5.1: Nurturing—Maximum theoretical rewards summary—After 10% addition.

[1] While the left and the right light positions are the same distance from the robot's starting position in the arena, the robot's starting orientation means that the minimum time to the right light position is longer than the minimum time to the left light position. (See Figure 3.4 for robot orientation.)

Minimum estimated time steps	$158 + 10\% \approx 174$
Max possible high-reward	((999 - 174)/999)*0.9 = 0.74
Max possible medium-reward	((999 - 174)/999)*0.5 = 0.41
Max possible low-reward	((999 - 174)/999)*0.1 = 0.082

Table 5.2: Self-Care—Maximum theoretical rewards summary—After 10% addition.

Table 5.1 and Table 5.2 show that the best instinctive individual that always goes to the same light in both halves of its life should achieve a maximum of 0.47 on average in the nurturing niche and 0.41 in the self-care niche. Note that this is true whether the instinctive individual goes to the non-changing, medium-rewarding light throughout its lifetime or goes to a light that switches rewards at the halfway point of the individual's life such that the individual receives the high reward in one half of its lifetime and the low reward in the other half. Thus any individual that gains a fitness higher than the cuttoff value for its corresponding niche belongs in the substantial learning category while an individual with lower or equal fitness has performance that is poorer than or equal to the theoretical best instinctive performance and thus belongs in the non-substantial category. Moreover, a substantial learning individual that gains a fitness higher than the corresponding substantial value for both halves of its lifetime belongs in the good learning subcategory, whereas a substantial learning individual that exceeds the substantial learning value in only one half of its lifetime (and overall) belongs in the moderate subcategory, as shown in Tables 5.3 and 5.4.

Sub category name	First half	Operator	Second half
Good	>0.47	AND	>0.47
Moderate	>0.47	XOR	>0.47

Table 5.3: Subcategories of substantial learning in the nurturing niche.

Sub category name	First half	Operator	Second half
Good	>0.41	AND	>0.41
Moderate	>0.41	XOR	>0.41

Table 5.4: Subcategories of substantial learning in the self-care niche.

5.1.1 Data Scaling

In order to fairly compare the data between the nurturing niche and the self-care niche, it is important to have them on the same scale. While the base reward values of the lights are the same in both niches (0.1, 0.5, and 0.9), the fact that the optimal route to each light in the self-care niche is longer than the corresponding optimal route in the nurturing niche means that the best possible earned reward for each light is lower in the self-care niche and therefore normalization of earned reward is necessary. However, because some behaviors result in rewards (which have positive values) while others result in penalties (which have negative values), it seems unintuitive to normalize the data from both niches to [0, 1], as is typical in normalization. Instead, data are normalized throughout this dissertation by converting the self-care data to the nurturing scale, which does not change the sign of the data, it simply rescales it. This can be accomplished by normalizing the self-care data to [0, 1] in the standard way, then using the inverse process with the nurturing data coefficients.

Consider data set D with known minimum d_{min} and maximum d_{max}. To normalize data in this set to [0, 1] we would use

$$n_i = \frac{d_i - d_{min}}{d_{max} - d_{min}} \tag{5.1}$$

on each datum d_i from D to arrive at its normalized value n_i. This means that to go the other way we would use

$$d_i = n_i(d_{max} - d_{min}) + d_{min}. \tag{5.2}$$

Thus, with two data sets $D1$ and $D2$ with their respective minima and maxima $d1_{min}$, $d2_{min}$, $d1_{max}$, and $d2_{max}$, to convert the $D2$ data to the $D1$ scale we use

$${}_1n2_i = \frac{d2_i - d2_{min}}{d2_{max} - d2_{min}}(d1_{max} - d1_{min}) + d1_{min} \tag{5.3}$$

on each datum $d2_i$ from $D2$ to arrive at its normalized value ${}_1n2_i$ on the $D1$ scale.

Given our maximum theoretical rewards of 0.85 for the nurturing niche and 0.74 for the self-care niche, and the penalty score of -0.25 that is common to both niches, Equation 5.3 simplifies to

$${}_ns_i = (s_i + 0.25) * 1.\bar{1} - 0.25 \tag{5.4}$$

where s_i is a datum from the self-care niche and ${}_ns_i$ is its value normalized to the nurturing niche. Therefore, all data reported for the self-care niche, both in text and in graphs, is normalized using Equation 5.4.

5.2 Evolution of Learning

5.2.1 Statistical Analysis

The first set of hypotheses considered are those concerning category likelihood. These results are analyzed using Fisher's exact tests (Agresti, 1992) for those hypotheses that consider two category comparisons (for example, substantial vs. non-substantial learning) across the nurturing and self-care niches and chi-squared tests (Foster, 2006) for those hypotheses that consider three category comparisons (good, moderate, and non-substantial learning).

Category Likelihood

1.1 Learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

1.1.1 **CL-SA** Substantial learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

		Learning Category		
		Substantial	Non-Substantial	Total
Niche	Nurturing	29	1	30
	Self-Care	19	11	30

Table 5.5: Evolution of Learning (Nurturing vs. Self-Care)—Substantial vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.5 shows the results for hypothesis **CL-SA** (1.1.1). The results for the nurturing niche are 29 substantial learners and 1 non-substantial learner. The results for the self-care niche are 19 substantial learners and 11 non-substantial learners. These results are statistically significant (Fisher's exact test, two-tailed $p = 0.0025$).

1.1.2 **CL-GA** Good learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

		Learning Category		
		Good	Not-Good	Total
Niche	Nurturing	27	3	30
	Self-Care	8	22	30

Table 5.6: Evolution of Learning (Nurturing vs. Self-Care)—Good vs. Not-Good Learning Category Likelihood Statistics.

Table 5.6 shows the results for hypothesis **CL-GA** (1.1.2). The results for the nurturing niche are 27 good learners and 3 not-good learners. The results for the self-care niche are 8 good learners and 22 not-good learners. These results are statistically significant (Fisher's exact test, two-tailed p <0.0001).

1.1.3 **CL-BA** Better learning is more likely to evolve with nurturing than without nurturing in the absence of instincts.

		Learning Category			
		Good	Moderate	Non-Substantial	Total
Niche	Nurturing	27	2	1	30
	Self-Care	8	11	11	30

Table 5.7: Evolution of Learning (Nurturing vs. Self-Care)—Good vs. Moderate vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.7 shows the results for hypothesis **CL-BA** (1.1.3). The results for the nurturing niche are 27 good learners, 2 moderate learners, and 1 non-substantial learner. The results for the self-care niche are 8 good learners, 11 moderate learners, and 11 non-substantial learners. These results are statistically significant (chi-squared test, p <0.00001).

Performance Continuum

The remaining hypotheses belong to the performance continuum category. These results are analyzed using t-tests (Ha & Ha, 2011). A score termed *relative success* is calculated for each repetition. The *relative success* is a measure of how close the best individual in the final generation of that repetition is to the theoretical best omniscient individual in that niche. Note that this is not the same as the theoretical best instinctive individual, which goes to the same light every trial and receives (on average) the reward for moving quickly to the moderate rewarding light. Instead, this theoretical best omniscient individual moves quickly to the high rewarding light on every trial (or to the switch and then to the high rewarding light for the self-care niche), regardless of which light gives which reward, and does not need to spend time exploring. This relative success is compared between all repetitions for each niche (for Hypothesis 2.1) as well as within the learning categories of each niche (for the other performance continuum hypotheses).

2.1 **PC-OA** Nurturing promotes the evolution of higher performing behaviors in the absence of instincts.

The relative success for the 30 nurturing niche individuals has a mean of 81.6 and standard deviation of 9.55 while the relative success for the 30 self-care niche individuals has a mean of 60.3 and a standard deviation of 9.15 (see Table 5.8). These results were statistically significant (t-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	81.5890	60.3410
SD	9.5486	9.1460
SEM	1.7433	1.6698
N	30	30

Table 5.8: Evolution of Learning (Nurturing vs. Self-Care)—Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.1.1.

2.1.1 **PC-GA** Nurturing promotes the evolution of higher performing behaviors within good learners in the absence of instincts.

The relative success for the 27 nurturing niche good learners has a mean of 83.6 and standard deviation of 7.33 while the relative success for the 8 self-care niche good learners has a mean of 69.8 and a standard deviation of 7.42 (see Table 5.9). These results are statistically significant (*t*-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	83.6433	69.8225
SD	7.3297	7.4173
SEM	1.4106	2.6224
N	27	8

Table 5.9: Evolution of Learning (Nurturing vs. Self-Care)—Good Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.1.2.

2.1.2 **PC-SA** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the absence of instincts.

The relative success for the 29 nurturing niche substantial learners has a mean of 82.5 and standard deviation of 8.23 while the relative success for the 19 self-care niche substantial learners has a mean of 65.7 and a standard deviation of 6.85 (see Table 5.10). These results are statistically significant (*t*-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	82.5159	65.6837
SD	8.2305	6.8547
SEM	1.5284	1.5726
N	29	19

Table 5.10: Evolution of Learning (Nurturing vs. Self-Care)—Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.1.3.

2.1.3 **PC-MA** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the absence of instincts. The relative success for the 2 nurturing niche moderate learners has a mean of 67.3 and a standard deviation of 1.49 while the relative success for the 11 self-care niche moderate learners has a mean of 62.7 and a standard deviation of 4.73 (see Table 5.11). These results are not statistically significant (t-test, $p = 0.2118$).

Group	Nurturing	Self-Care
Mean	67.2950	62.6736
SD	1.4920	4.7323
SEM	1.0550	1.4268
N	2	11

Table 5.11: Evolution of Learning (Nurturing vs. Self-Care)—Moderate Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.1.4.

2.1.4 **PC-NA** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the absence of instincts. The relative success for the 1 nurturing niche non-substantial learner is 54.7 while the relative success for the 11 self-care niche non-substantial learners has a mean of 51.1 and a standard deviation of 3.03 (see Table 5.12). Having a single data point for the nurturing niche means that it is not possible to run a *t*-test on this data set.

Group	Nurturing	Self-Care
Mean	54.71	51.112727
SD	-	3.029192
SEM	-	0.957914
N	1	11

Table 5.12: Evolution of Learning (Nurturing vs. Self-Care)—Non-Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.1.5.

5.2.2 Averages and Exemplars

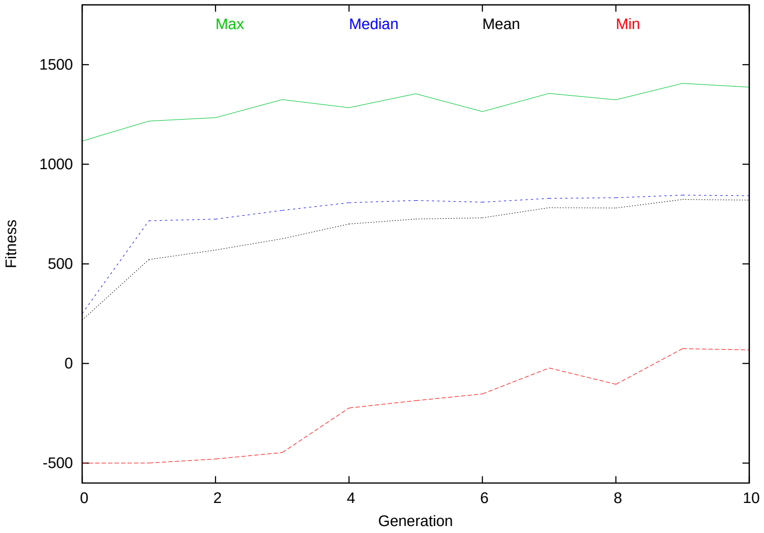

Figure 5.1: Evolution of Learning—Nurturing Statistics. Max fitness, median fitness, mean fitness, and min fitness across 10 generations averaged over 30 repetitions.

Figure 5.1 shows that in case of nurturing with the evolution of learning, on average (over 30 repetitions), the worst individual at generation 0 has a fitness (reward collected) of -500 which improves to 68.07 by the end of generation 10. Similarly, on average, the mean fitness of the whole population at generation 0 is 217.93 which improves to 819.88 by the end of generation 10. Further, on average, the median fitness of the whole population at generation 0 is 250.04 which improves to 842.38 by the end of generation 10. Finally, on average, the best individual has a fitness of 1116.10 at generation 0 which improves to 1386.87 by the end of generation 10.

The evolution of learning in the self-care niche can be compared at the same cutoff point as shown in Figure 5.2.

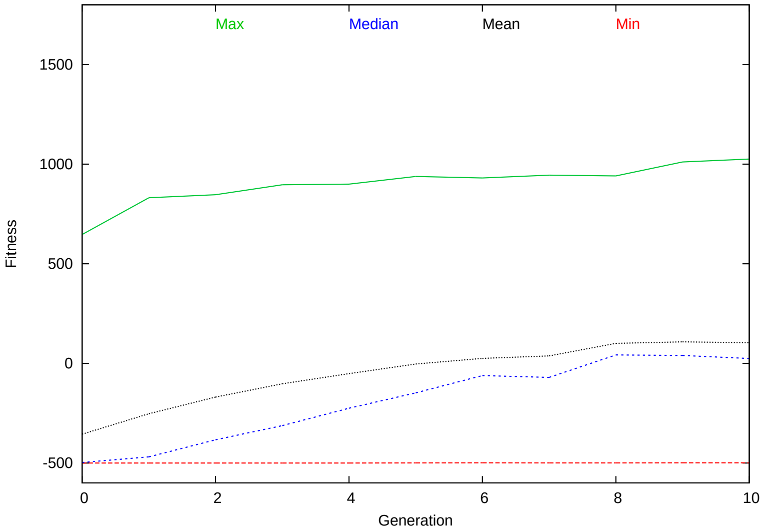

Figure 5.2: Evolution of Learning—Self-Care Statistics. Max fitness, median fitness, mean fitness, and min fitness across 10 generations averaged over 30 repetitions.

Figure 5.2 shows that in case of self-care with the evolution of learning, on average (over 30 repetitions), the worst individual at generation 0 has a fitness (reward collected) of -500 which remains almost the same (-499.03) at generation 10. Similarly, on average, the mean fitness of the whole population at generation 0 is -355.28 which improves to 103.90 by the end of generation 10. Further, on average, the median fitness of the whole population at generation 0 is -497.36 which improves to 24.55 by the end of generation 10. Finally, on average, the best individual has a fitness of 647.09 at generation 0 which improves to 1025.87 by the end of generation 10.

Evolved Learning Exemplars in the Nurturing Niche

This section presents examples of reward patterns for the most fit individuals from the final generation in the nurturing niche to give a feel for the behaviors evolved.

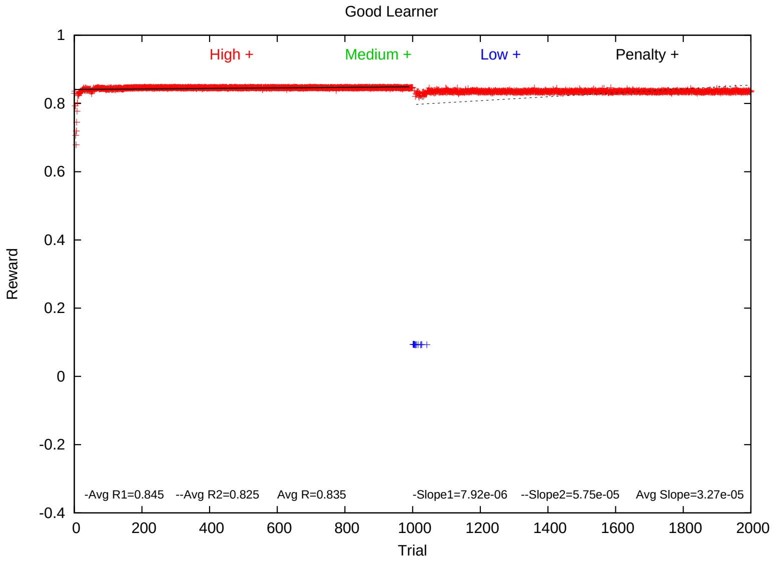

Figure 5.3: Evolution of Learning—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.3 represents a typical good learning individual[2]. The first light encountered by this individual is the high-rewarding light and it learns within a few trials to move very quickly to that light. When that light becomes the low-rewarding light at Trial 1000, the individual receives a lower than expected reward for a few trials, then tries a different light that turns out to be the current high-rewarding light and quickly learns to prefer that light, moving to it quickly on each subsequent trial.

[2] Note that linear regression lines and slopes in the graphs are presented solely to highlight trends in the data. They are not used to determine to which category an example repetition belongs.

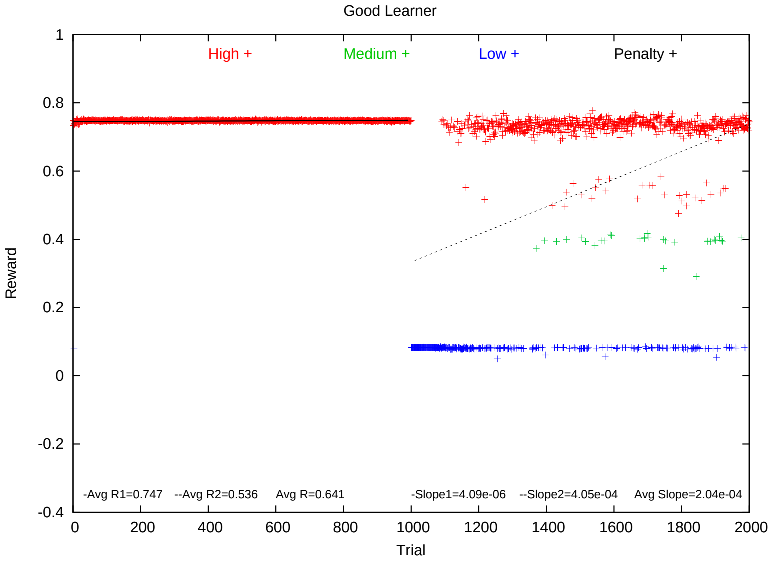

Figure 5.4: Evolution of Learning—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.4 is another example of a typical good learner. This individual moves quickly to the high rewarding light initially, adjusts its weights in such a way as to slightly reduce its travel time, and so performs extremely well in the first half of its lifetime. In the second half, of its lifetime, when the light it had been seeking becomes the low rewarding light, it adjusts its weights and finds the new high rewarding light. It adjusts its weights so that it moves quickly to this light on most trials, thereby earning an average reward of greater than 0.47 in the second half of its lifetime as well, and thus it is categorized as a good learner.

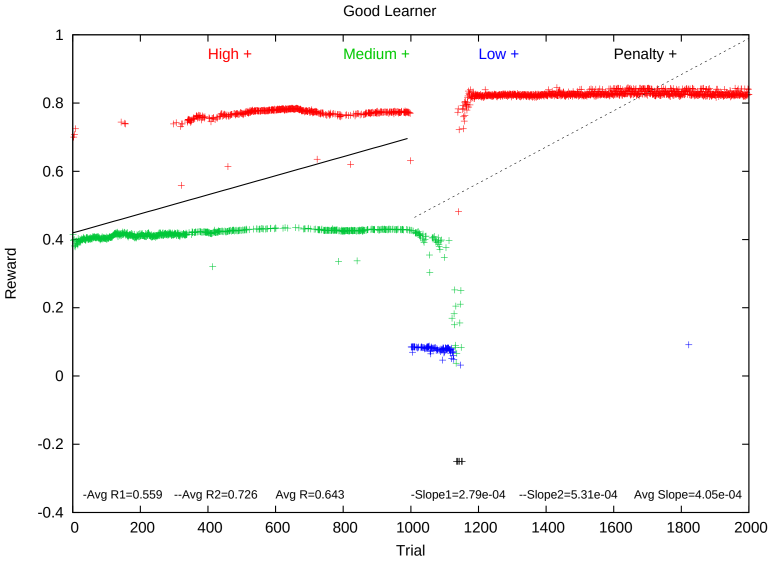

Figure 5.5: Evolution of Learning—Good (Substantial) Learning example in nurturing niche.

Figure 5.5 is a contrasting example of a good learner where the individual shifts its focus from the medium rewarding light source to the high rewarding light in the first half of its lifetime. In the second half of its lifetime, when the high rewarding light that it had been targeting becomes the low rewarding light, it quickly adjusts its weights such that it moves to the high rewarding light source quickly and then it becomes conservative; that is, it keeps exploiting its knowledge of this resource.

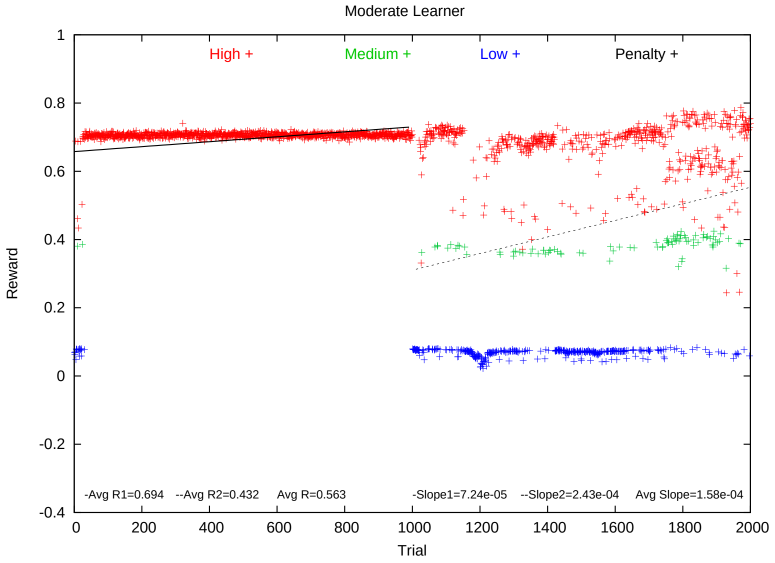

Figure 5.6: Evolution of Learning—Moderate (Substantial) Learning example in nurturing niche.

Figure 5.6 shows a moderate learning case, where the individual exhibits learning in both halves of its lifetime. However, in the second half of its lifetime it learns more slowly and therefore its average reward in the second half of its lifetime stays moderate, thus it is categorized as a moderate learner. Note the similarity in the behavior pattern between this case and the individual shown in Figure 5.4. The difference in this case is that this individual retains its focus more on the low rewarding light during the second half of its lifetime and thus ends up receiving an average lifetime reward that categorizes it as a moderate learner.

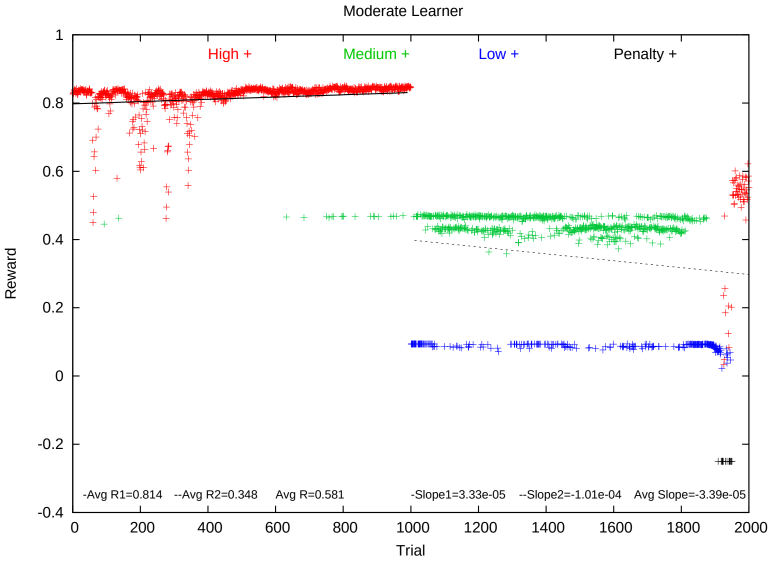

Figure 5.7: Evolution of Learning—Moderate (Substantial) Learning example in nurturing niche.

Figure 5.7 depicts an individual that improves its performance in the first half of its lifetime, primarily by speeding its travel to the high rewarding light, while its behavior is more difficult to characterize in the later half of its lifetime. When the high rewarding light that had been favored by this individual becomes the low rewarding light, it increases its frequency of visits to the medium rewarding light, which it had encountered very rarely during the first half of its lifetime. However, its movement to the medium rewarding light is not consistent. Then, during the final 100 trials of this individual's lifetime, it seems to be returning its focus to the low rewarding light only to shift again, fail a few times and end its lifetime moving repeatedly to the high rewarding light. In total, the individual does gain sufficient fitness to fit the moderate learner criteria.

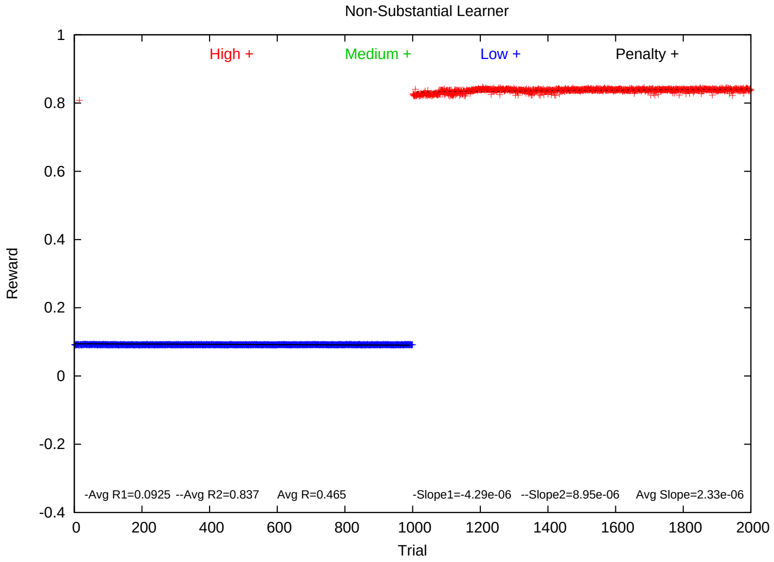

Figure 5.8: Evolution of Learning—Non-Substantial Learning example in nurturing niche.

Finally, Figure 5.8 depicts a non-substantial learner found among the results in the nurturing niche. It uses the initial weights that it was randomly assigned at the start of its lifetime and never improves on them. It exploits the low rewarding light during the first half of its lifetime and keeps going to that same light even after the change in the environment half way through the lifetime. It is interesting to note that it does see the highest rewarding light once early in its lifetime and finds quite an excellent path to it but never shifts its policy toward those actions. It is likewise interesting to note that this individual's overall behavior and fitness scores are very close to those of the theoretical best instinctive individual.

Evolved Learning Exemplars in the Self-Care Niche

This section presents examples of reward patterns for the most fit individuals from the final generation in the self-care niche to give a feel for the behaviors evolved.

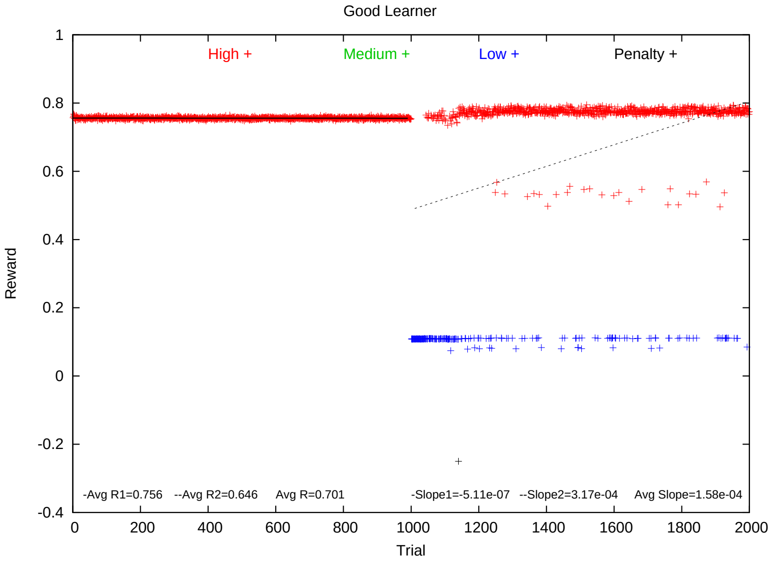

Figure 5.9: Evolution of Learning—Good (Substantial) Learning typical example in self-care niche.

Figure 5.9 is an example of a typical good learner which happens to have good initial weights for going to the high rewarding light source although not it does not follow the optimal path. In the second half of its lifetime when the high rewarding light it had been visiting becomes the low rewarding light, it clearly shifts from that light to the new high rewarding light and mostly exploits that resource with an approximately equally rewarding path as the one used in the first half of its lifetime.

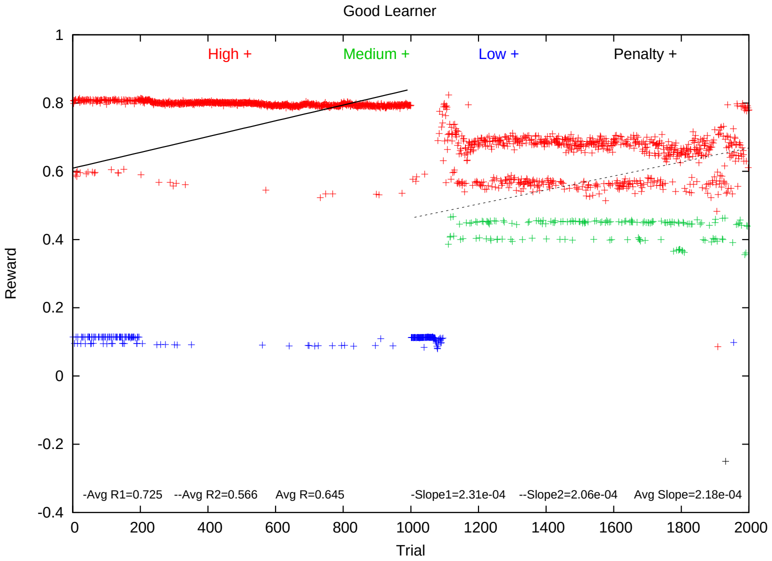

Figure 5.10: Evolution of Learning—Good (Substantial) Learning typical example in self-care niche.

Figure 5.10 is another example of a typical good learner that shows learning and moves to the high rewarding light during most trials in both halves of its lifetime. Although, it does not follow the optimal path but performs sufficiently to fit the criteria of a good learner.

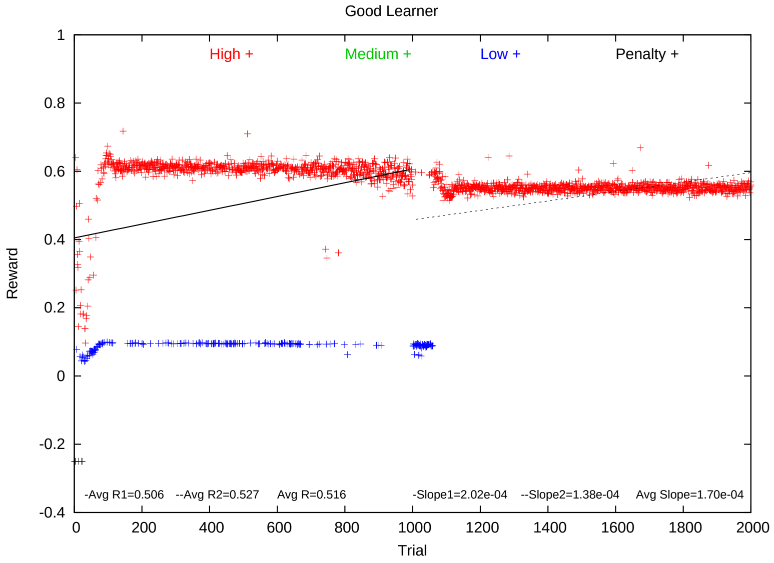

Figure 5.11: Evolution of Learning—Good (Substantial) Learning example in self-care niche.

Figure 5.11 is another interesting example of a good learner where the overall trend of the individual shows learning in both halves of its lifetime. However, while it learns to go to the high rewarding light source it slows down its travel there throughout its lifetime.

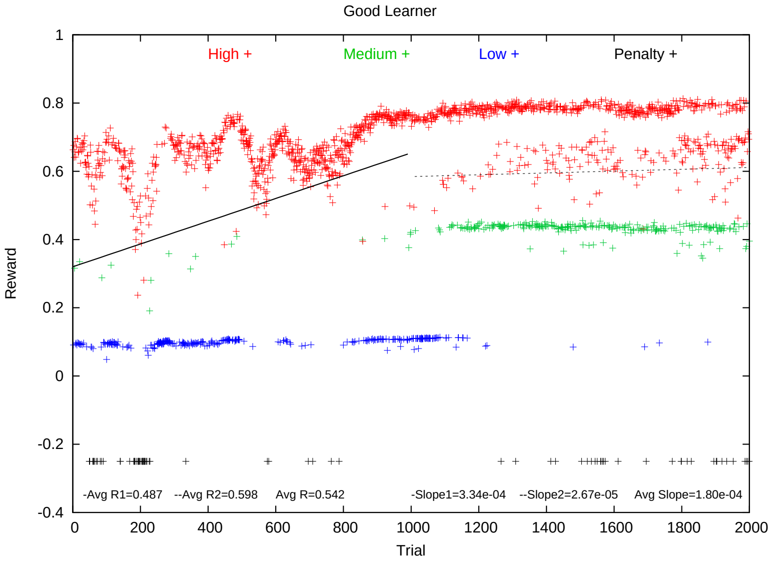

Figure 5.12: Evolution of Learning—Good (Substantial) Learning example in self-care niche.

Finally, Figure 5.12 is an example of a good learner which marginally exceeds the criterion for substantial learning in the first half of its lifetime with an average reward of 0.413. Nonetheless, it shows a positive trend to the rewards it receives in both halves of its lifetime and it travels to the high rewarding light more frequently than it engages in other behaviors. However, it does fail to get to any light on several trials and also explores somewhat more than required. Thus, while it is not necessarily the best algorithm, it performs sufficiently to be called a good learner.

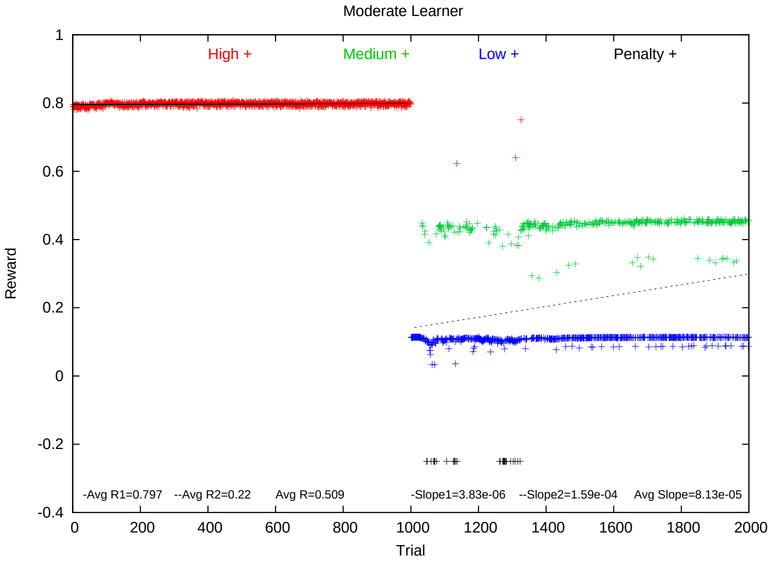

Figure 5.13: Evolution of Learning—Moderate (Substantial) Learning typical example in self-care niche.

Figure 5.13 shows an example of a typical moderate learner that happens to have good initial weights to start with and thus never explores in the first half of its lifetime. In the second half of its lifetime though, when the high rewarding light that it had been targeting becomes the low rewarding light, it shifts its focus from that light to a medium rewarding light even though it explores the best light source on three trials.

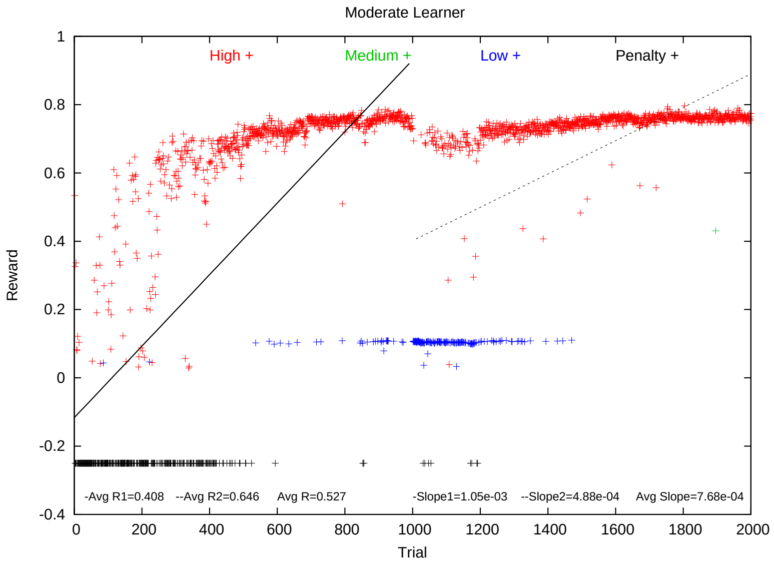

Figure 5.14: Evolution of Learning—Moderate (Substantial) Learning example in self-care niche.

Figure 5.14 is an interesting case of moderate learning where the individual fails frequently for more than one fifth of its lifetime and then shifts its path to the high rewarding light. In the second half of its lifetime, it responds to the change in environment well and gets a decent amount of fitness. The failures in the first half of the lifetime mostly contribute to this individual being categorized as a moderate learner and not a good learner.

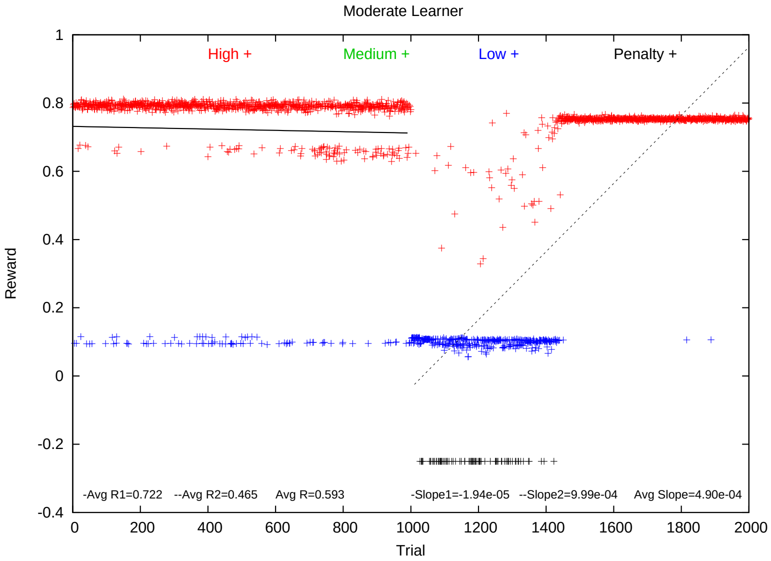

Figure 5.15: Evolution of Learning—Moderate (Substantial) Learning example in self-care niche.

Finally, Figure 5.15 is an example of a moderate learner that declines slightly in performance during the first half of its lifetime and responds slowly to the change in the environment that occurs at trial 1000. It is categorized as a moderate learner based on the average fitness it obtains, which is above the substantial learning cutoff overall but slightly below the cutoff in the second half of its lifetime.

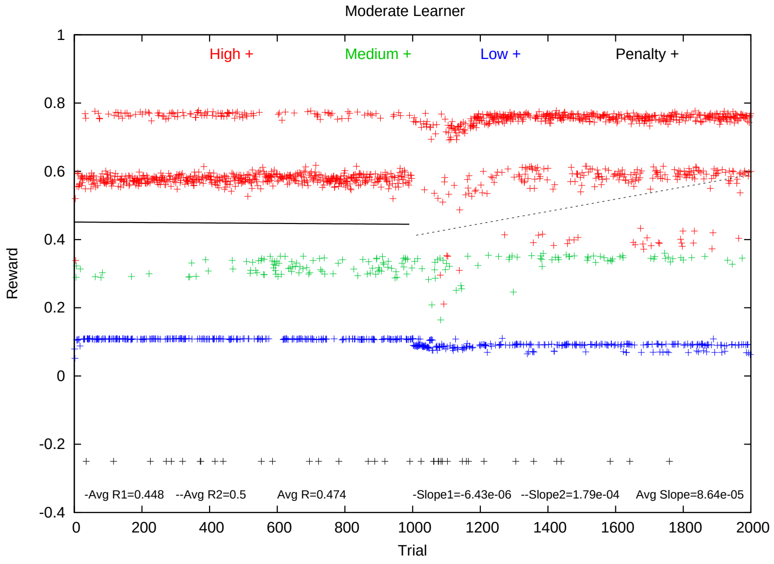

Figure 5.16: Evolution of Learning—Moderate (Substantial) Learning example in self-care niche.

Figure 5.16 shows an individual with a broad range of behaviors throughout its lifetime. During the first half of its lifetime, it visits all three lights frequently and fails infrequently. Moreover, it follows two distinct routes to the high rewarding light. There is a slight shift in behavior after trial 1000, when the high and low rewards are switched, but it settles into a pattern of behavior that can be characterized similarly to that of the first half of its lifetime. Other than a slight improvement right after the reward swap the individual does not demonstrate any major learning. However, its increased frequency of visiting the high rewarding light (as shown by the regression line) during the second half of its lifetime is a notable improvement. This result is classified as belonging to the moderate learning category as it performs slightly better than the theoretical best instinctive performance.

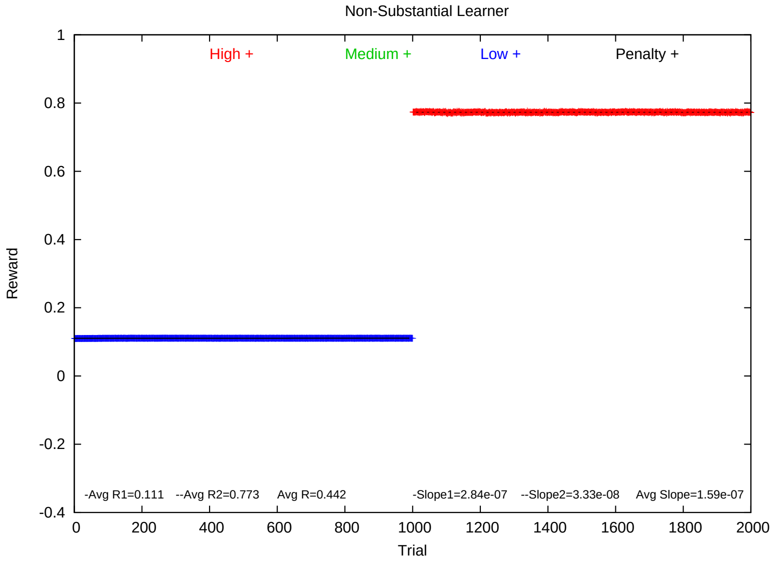

Figure 5.17: Evolution of Learning—Non-Substantial Learning example in self-care niche.

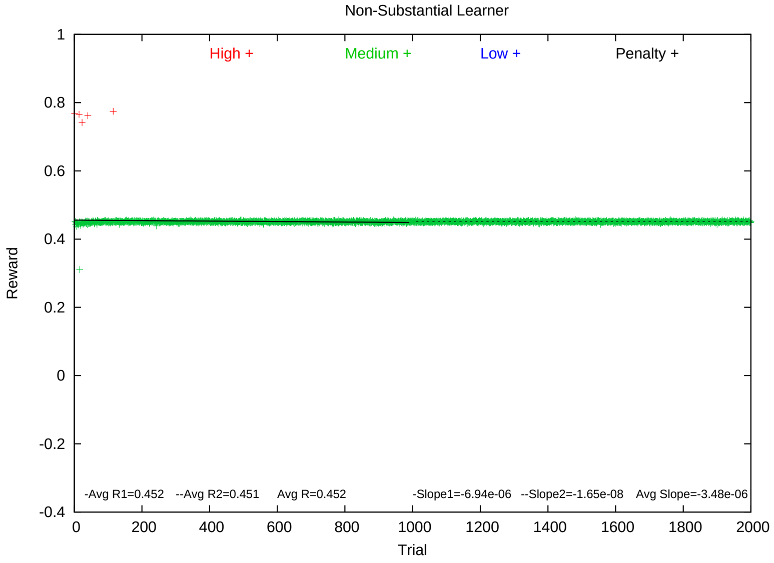

Figure 5.18: Evolution of Learning—Non-Substantial Learning typical example in self-care niche.

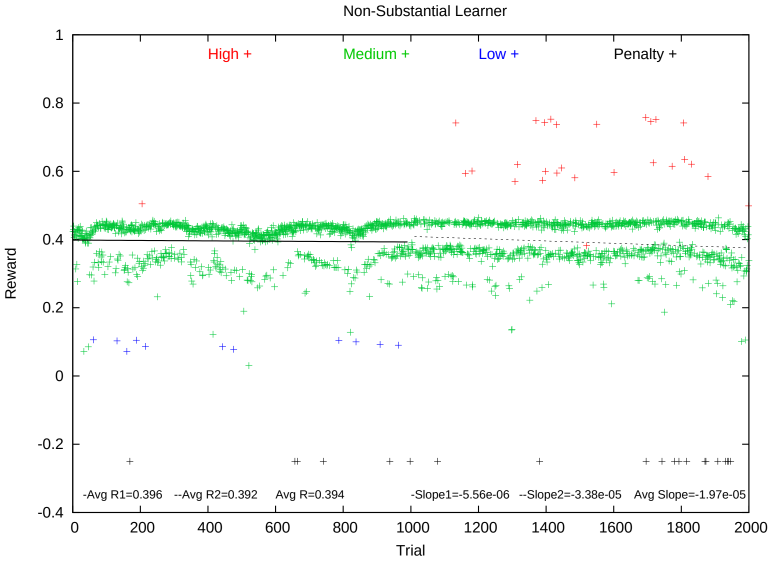

Figure 5.19: Evolution of Learning—Non-Substantial Learning example in self-care niche.

Figure 5.17 and Figure 5.18 show examples of individuals which do not seem to learn anything substantial throughout the course of their lifetimes. In both cases, the individuals do not deviate substantially from the behaviors corresponding to their initial (random) weight means. Note that the individual depicted by Figure 5.17 is quite similar to the one shown by Figure 5.8 as both the individuals act instinctively. Similarly, Figure 5.19 shows an individual which is a noisier version of the individual shown in Figure 5.18.

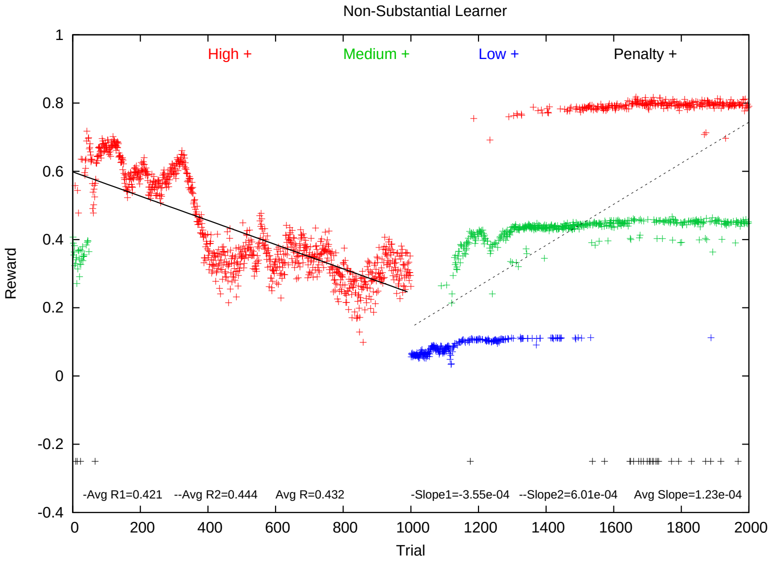

Figure 5.20: Evolution of Learning—Non-Substantial Learning example in self-care niche.

In Figure 5.20, we see an individual that initially samples the medium rewarding light, the high rewarding light, and failures, before focusing on the high rewarding light. It then keeps slowing down and taking longer routes to the high rewarding light until the environmental change switches the reward for that light from high to low. During the second half of the lifetime, though, this individual shows improvement in its exploration and exploitation of better and better resources. Nonetheless, this individual still does not do well enough to reach the threshold for substantial learning.

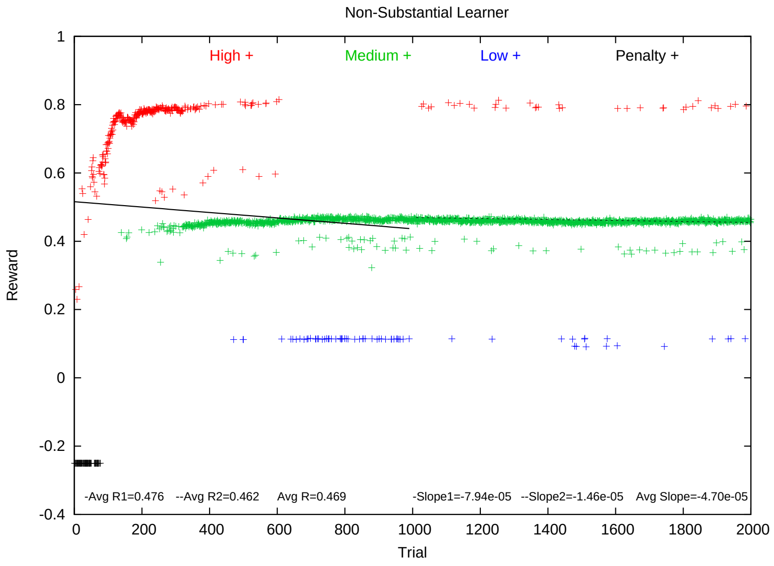

Figure 5.21: Evolution of Learning—Non-Substantial Learning example in self-care niche.

Figure 5.21, depicts a somewhat complex case. The individual mostly fails during the first 100 trials of its lifetime but shifts its behavior to moving to the high rewarding light and improves its time to that light such that by trial 200, it is receiving quite good rewards. However, it gradually shifts its behavior again, increasingly favoring the medium rewarding light and even experiences the low rewarding light repeatedly before the end of the first half of its lifetime. During the second half of its lifetime, the individual settles into a behavior pattern where it almost always moves to the medium rewarding light although it still encounters the high and low rewarding lights on occasion.

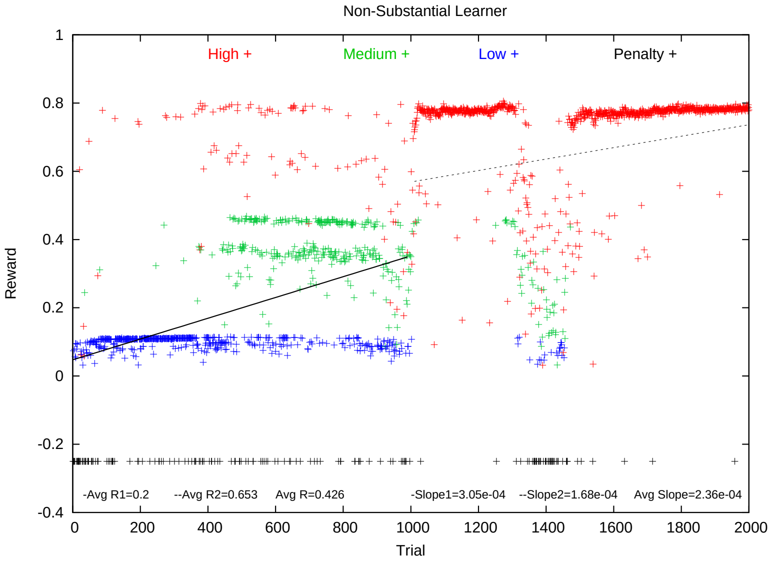

Figure 5.22: Evolution of Learning—Non-Substantial Learning example in self-care niche.

Figure 5.22 is another rather complex case. The individual depicted here starts off with many failures, frequent encounters with the low rewarding light, and rare encounters with the high and moderate rewarding lights. It improves its performance in the first half of its lifetime by decreasing failures and increasing visits to the high and, particularly, to the medium rewarding light. When the light that had been delivering the low reward becomes the high rewarding light, the individual, which had been visiting that light moderately frequently quickly shifts to visiting it almost exclusively. The individual keeps visiting that light and receiving a high reward almost exclusively for roughly 250 trials, at which time its performance falls off sharply. For approximately 150 trials the individual fails frequently and visits each of the lights, often through very slow routes. Finally, the individual begins exploiting a good route to the high rewarding light again and continues to do so almost exclusively for the final 500 trials of its lifetime.

5.3 Evolution of Learning and Instincts

5.3.1 Statistical Analysis

In the following section, we will now look at the statistical analysis of the results obtained. All the metrics related to the evolution of learning and instincts are answered using the data collected.

Category Likelihood

1.2 Learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

1.2.1 **CL-SP** Substantial learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

		Learning Category		
		Substantial	Non-Substantial	Total
Niche	Nurturing	30	0	30
	Self-Care	24	6	30

Table 5.13: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Substantial vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.13 shows the results for hypothesis **CL-SP** (1.2.1). The results for the nurturing niche are 30 substantial learners and no non-substantial learners. The results for the self-care niche are 24 substantial learners and 6 non-substantial learners. These results are statistically significant (Fisher's exact test, two-tailed $p = 0.0237$).

1.2.2 **CL-GP** Good learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

		Learning Category		
		Good	Not-Good	Total
Niche	Nurturing	26	4	30
	Self-Care	7	23	30

Table 5.14: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Good vs. Not-Good Learning Category Likelihood Statistics.

Table 5.14 shows the results for hypothesis **CL-GP** (1.2.2). The results for the nurturing niche are 26 good learners and 4 not-good learners. The results for the self-care niche are 7 good learners and 23 not-good learners. These results are statistically significant (Fisher's exact test, two-tailed p <0.0001).

1.2.3 **CL-BP** Better learning is more likely to evolve with nurturing than without nurturing in the presence of instincts.

		Learning Category			
		Good	Moderate	Non-Substantial	Total
Niche	Nurturing	26	4	0	30
	Self-Care	7	17	6	30

Table 5.15: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Good vs. Moderate vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.15 shows the results for hypothesis **CL-BP** (1.2.3). The results for the nurturing niche are 26 good learners, 4 moderate learners, and no non-substantial learner. The results for the self-care niche are 7 good learners, 17 moderate learners, and 6 non-substantial learners. These results are statistically significant (chi-squared test, p <0.00001).

Performance Continuum

Now the performance continuum results based on comparisons of *relative success.*
2.2 **PC-OP** Nurturing promotes the evolution of higher performing behaviors in the presence of instincts.
The relative success for the 30 nurturing niche individuals has a mean of 83.7 and standard deviation of 8.68 while the relative success for the 30 self-care niche individuals has a mean of 65.2 and a standard deviation of 9.43 (see Table 5.16). These results were statistically significant (t-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	83.7483	65.1717
SD	8.6803	9.4281
SEM	1.5848	1.7213
N	30	30

Table 5.16: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.2.1.

2.2.1 **PC-GP** Nurturing promotes the evolution of higher performing behaviors within good learners in the presence of instincts.

The relative success for the 26 nurturing niche good learners has a mean of 85.9 and standard deviation of 6.87 while the relative success for the 7 self-care niche good learners has a mean of 78.2 and a standard deviation of 5.33 (see Table 5.17). These results are statistically significant (t-test, $p = 0.0093$).

Group	Nurturing	Self-Care
Mean	85.9950	78.2014
SD	6.8734	5.3296
SEM	1.3480	2.0144
N	26	7

Table 5.17: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Good Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.2.2.

2.2.2 **PC-SP** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the presence of instincts.

The relative success for the 30 nurturing niche substantial learners has a mean 83.7 and standard deviation of 8.68 while the relative success for the 24 self-care niche substantial learners has a mean of 67.9 and a standard deviation of 8.48 (see Table 5.18). These results are statistically significant (*t*-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	83.7483	67.9404
SD	8.6803	8.4782
SEM	1.5848	1.7306
N	30	24

Table 5.18: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.2.3.

2.2.3 **PC-MP** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the presence of instincts. The relative success for the 4 nurturing niche moderate learners has a mean 69.1 and a standard deviation of 2.56 while the relative success for the 17 self-care niche moderate learners has a mean of 63.7 and a standard deviation of 5.26 (see Table 5.19). These results are not statistically significant (t-test, $p = 0.0622$).

Group	Nurturing	Self-Care
Mean	69.1450	63.7153
SD	2.5643	5.2577
SEM	1.2822	1.2752
N	4	17

Table 5.19: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Moderate Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.2.4.

2.2.4 **PC-NP** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the presence of instincts. There is no non-substantial learner in the nurturing niche while the relative success for the 6 self-care niche non-substantial learners has a mean of 54.1 and a standard deviation of 0.88 (see Table 5.20). Having no data point for the nurturing niche means that it is not possible to run a *t*-test on this data set.

Group	Nurturing	Self-Care
Mean	-	54.096667
SD	-	0.879634
SEM	-	0.393384
N	0	6

Table 5.20: Evolution of Learning and Instincts (Nurturing vs. Self-Care)—Non-Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.2.5.

5.3.2 Averages and Exemplars

A few typical cases and examples of each type of learning in both nurturing and self-care niches are discussed as follows.

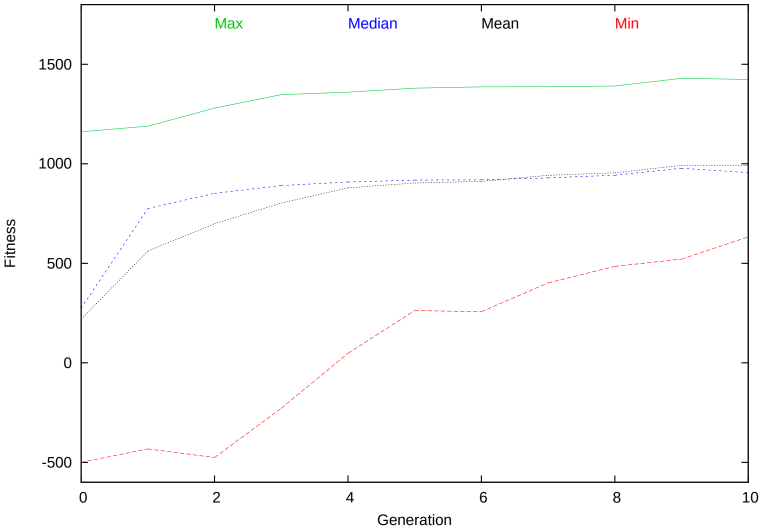

Figure 5.23: Evolution of Learning and Instincts—Nurturing Statistics. Max fitness, median fitness, mean fitness, and min fitness across 10 generations averaged over 30 repetitions.

Figure 5.23 shows that in the case of nurturing with the evolution of learning and instincts both, on average (over 30 repetitions), the worst individual at generation 0 has a fitness (reward collected) of -499.97 which improves to 632.90 by generation 10. Similarly, on average, the mean fitness of the whole population at generation 0 is 219.83 which improves to 990.86 by the end of generation 10. Further, on average, the median fitness of the whole population at generation 0 is 274.02 which improves to 955.55 by the end of generation 10. Finally, on average, the best individual has a fitness of 1160.75 at generation 0 which improves to 1423.84 by the end of generation 10.

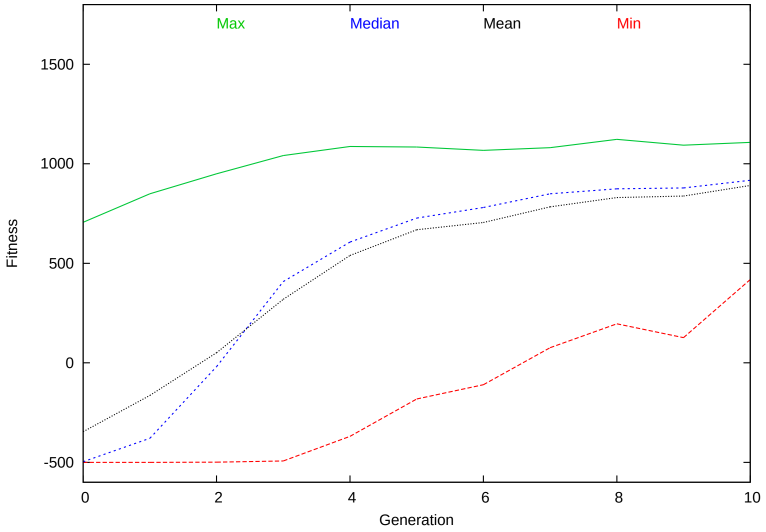

Figure 5.24: Evolution of Learning and Instincts—Self-Care Statistics. max fitness, median fitness, mean fitness, and min fitness across 10 generations averaged over 30 repetitions.

Figure 5.24 shows that in the case of self-care with the evolution of learning and instincts both, on average (over 30 repetitions), the worst individual at generation 0 has a fitness (reward collected) of -500 which improves to 418.25 by generation 10. Similarly, on average, the mean fitness of the whole population at generation 0 is -346.04 which improves to 891.33 by the end of generation 10. Further, on average, the median fitness of the whole population at generation 0 is -497.15 which improves to 916.96 by the end of generation 10. Finally, on average, the best individual has a fitness of 706.28 at generation 0 which improves to 1107.79 by the end of generation 10.

Evolved Learning and Instincts Exemplars in the Nurturing Niche

This section presents examples of reward patterns for the most fit individuals from the final generation in the nurturing niche.

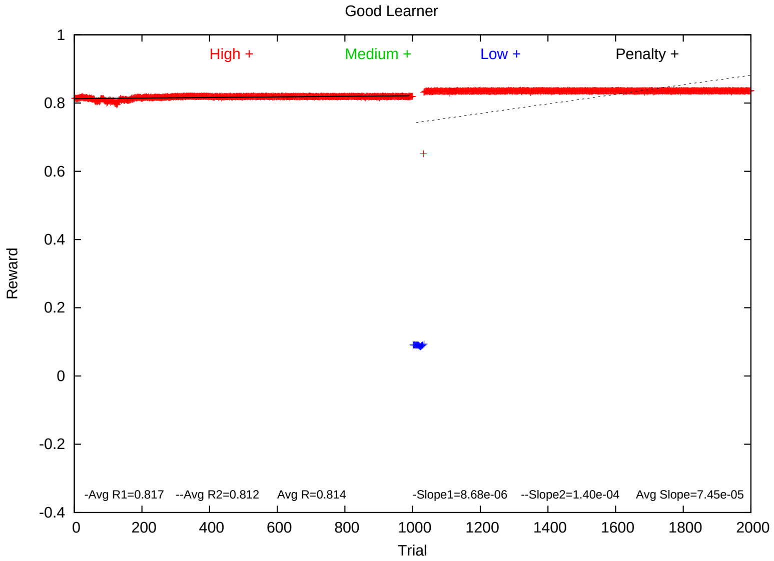

Figure 5.25: Evolution of Learning and Instincts—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.25 shows that the first light encountered by this individual is the high-rewarding light. Although it explores slightly different paths, it learns to move back to its original path. When that light becomes the low-rewarding light at Trial 1000, the individual receives a lower than expected reward for a few trials, then tries a different light that turns out to be the current high-rewarding light and quickly learns to prefer that light, moving to it quickly on each subsequent trial.

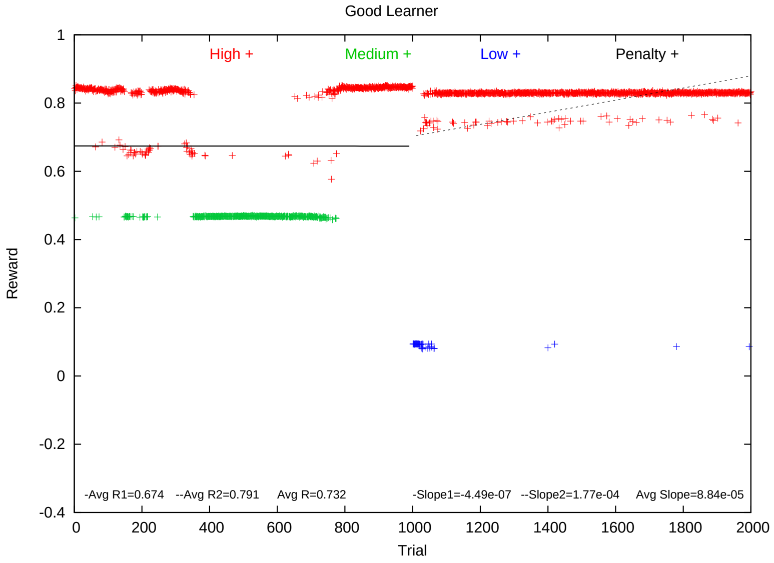

Figure 5.26: Evolution of Learning and Instincts—Good (Substantial) Learning example in nurturing niche.

Figure 5.26 is a contrasting example of a good learner. This individual, during the first half of its lifetime, explores two different paths to the high rewarding light including one near-optimal path. It also explores a near-optimal path to the medium rewarding light; however, it shifts its focus from the medium rewarding light source to the high rewarding light during the last 200 trials of the first half of its lifetime. In the second half of its lifetime, when the high rewarding light that it had been targeting becomes the low rewarding light, it quickly adjusts its weights such that it moves to the high rewarding light source quickly and then it mostly becomes conservative; that is, it keeps exploiting its knowledge of this resource. Note that it still rarely keeps exploring the low rewarding light.

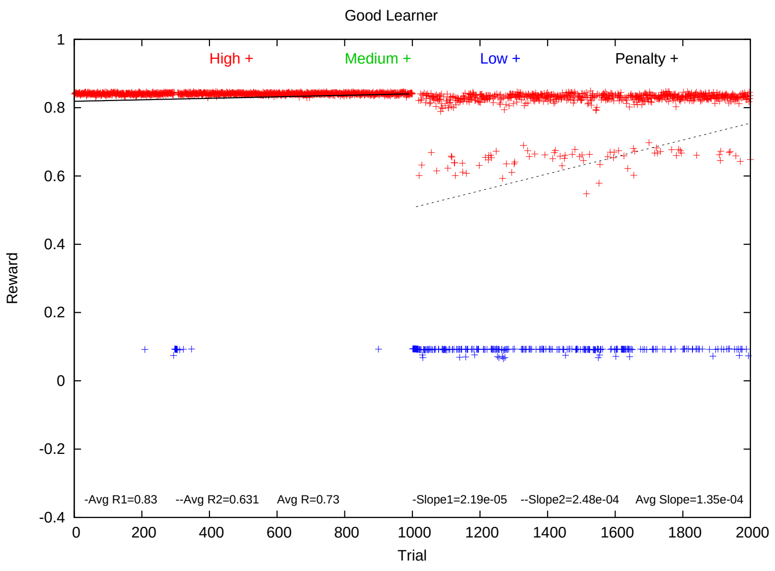

Figure 5.27: Evolution of Learning and Instincts—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.27 is another example of a typical good learner. This individual moves quickly to the high rewarding light initially, adjusts its weights in such a way as to slightly reduce its travel time, and so performs extremely well in the first half of its lifetime. In the second half of its lifetime, when the light it had been seeking becomes the low rewarding light, it adjusts its weights and finds the new high rewarding light. It adjusts its weights so that it moves quickly to this light on most trials, thereby earning an average reward of greater than 0.47 in the second half of its lifetime as well, and thus it is categorized as a good learner. This individual is a better version of the one shown in the evolution of learning results (Figure 5.4).

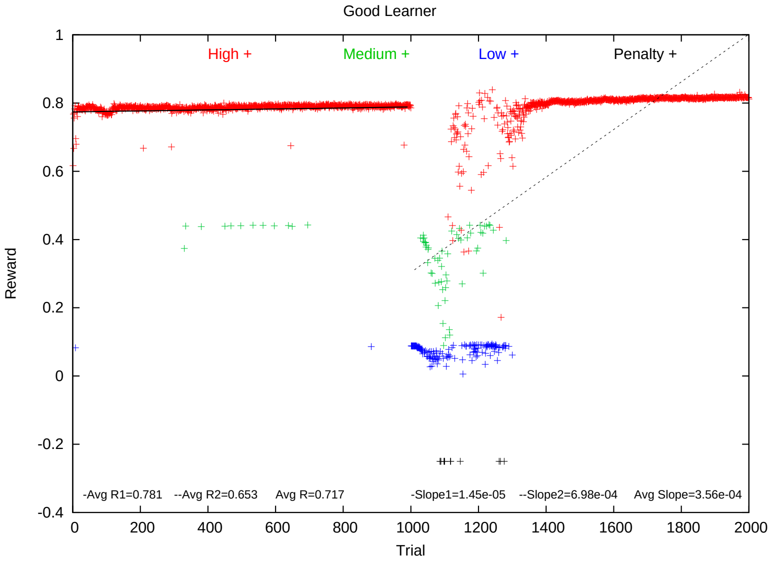

Figure 5.28: Evolution of Learning and Instincts—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.28 shows another typical example of a good learning individual which starts with good initial weights. It clearly becomes conservative and exploits the high rewarding light during the first half of its lifetime. In the second half of its lifetime, it spends about 200 trials exploring and finding the high rewarding light. This individual experiences penalties and rewards from all three lights before it learns to be conservative and starts exploiting the high rewarding light during the second half of its lifetime.

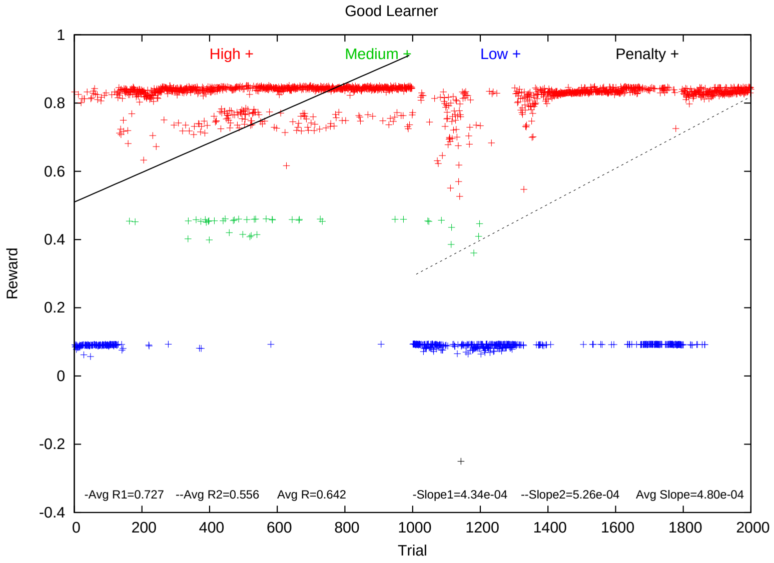

Figure 5.29: Evolution of Learning and Instincts—Good (Substantial) Learning typical example in nurturing niche.

Figure 5.29 shows another typical example that clearly depicts learning in both halves of its lifetime. On both occasions, i.e., the beginning of each half, the individual adjusts its weights in such a way that it leaves the lowest rewarding light source and finds the high rewarding light. Note that this individual is not born with instinctive knowledge of the high rewarding light source; however, it shows good learning behavior.

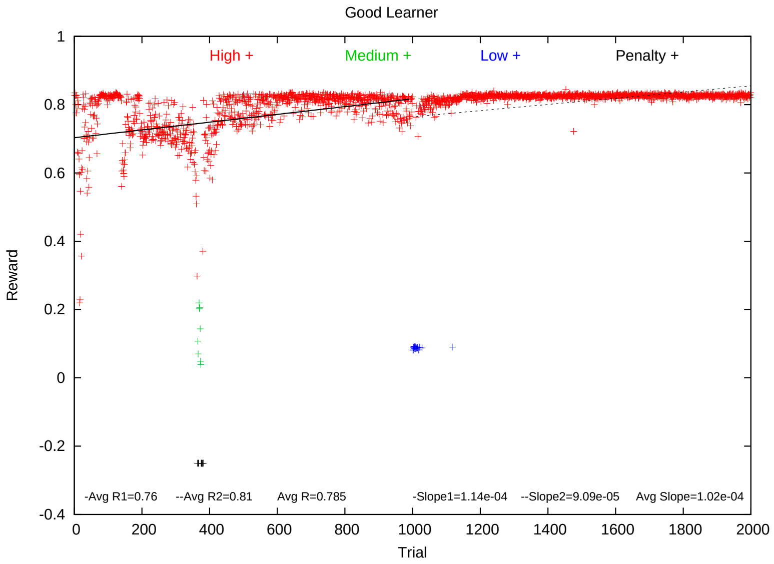

Figure 5.30: Evolution of Learning and Instincts—Good (Substantial) Learning example in nurturing niche.

Figure 5.30 shows an uncommon example of a good learning individual that is more exploratory during the first half as compared to the second half of its lifetime. Until the first quarter of its lifetime, this individual is more exploratory and experiences different paths to the high rewarding light. It also receives a penalty at the end of few trials and also finds the medium rewarding light; however, it shifts its focus to the medium and then high rewarding light soon after the penalty. In the second half of its lifetime, when the light it had been seeking becomes the low rewarding light, it adjusts its weights and finds the high rewarding light. It becomes conservative afterwards until the end of its lifetime.

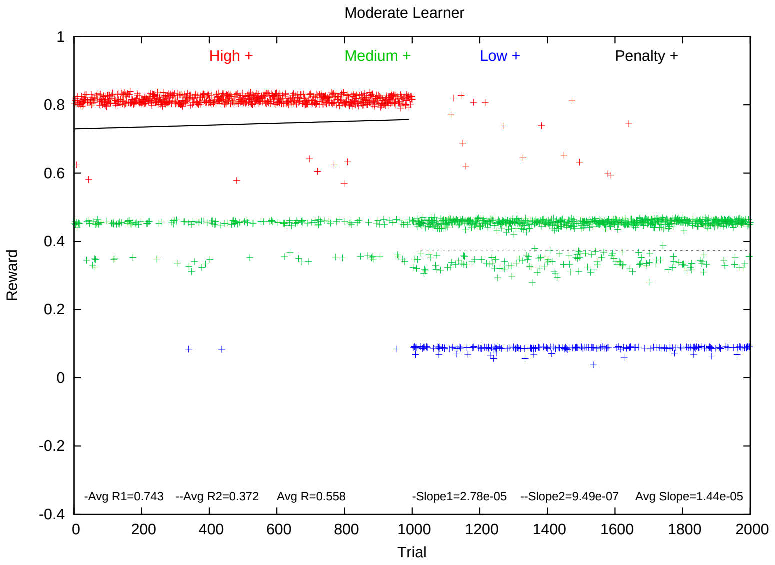

Figure 5.31: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in nurturing niche.

Figure 5.31 shows an individual that consistently finds the high rewarding light with continuous exploration of the medium rewarding light during the first half of its lifetime. Note that during the first half of its lifetime, this individual mostly learns to go to the high rewarding light. However, in the second half of its lifetime, when the light it had been seeking becomes the low rewarding light, it mostly exploits its knowledge of medium rewarding light and learns to choose a better option (i.e., choosing the medium reward compared to the low). Note that the frequency of visiting the medium rewarding light increases during the second half of its lifetime; however, it does not collect enough reward on average during this half to be categorized as a good learner. This individual is thus categorized as a moderate learner.

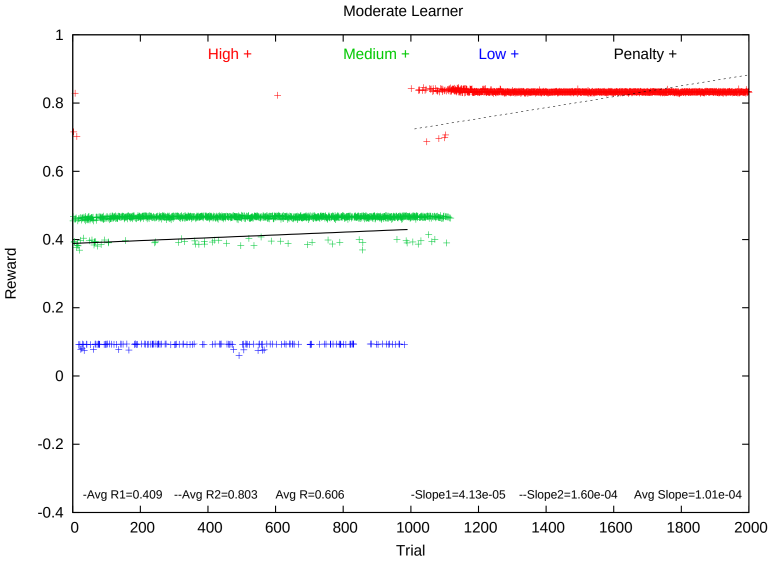

Figure 5.32: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in nurturing niche.

Figure 5.32 shows an individual that explores all three lights initially; however, it learns to find a medium rewarding light source more frequently instead of the high rewarding light during the first half of its lifetime. It also continuously improves on its path to the medium rewarding light during the first half of its lifetime. It also keeps going to the low rewarding light, although less frequently. In the second half of its lifetime, as it continues its pattern, the low rewarding light it has been visiting becomes the high rewarding light. The individual gradually shifts its weights in such a way that it starts exploiting the high rewarding light and completely stops going to the medium rewarding light. This individual shows learning in both halves of its lifetime; however, due to the lower average reward in the first half of its lifetime, it is categorized as a moderate learner.

There is no non-substantial learning case found in the nurturing niche.

Evolved Learning and Instincts Exemplars in the Self-Care Niche

The self-care individual learning cases are more diverse; therefore, more examples are presented.

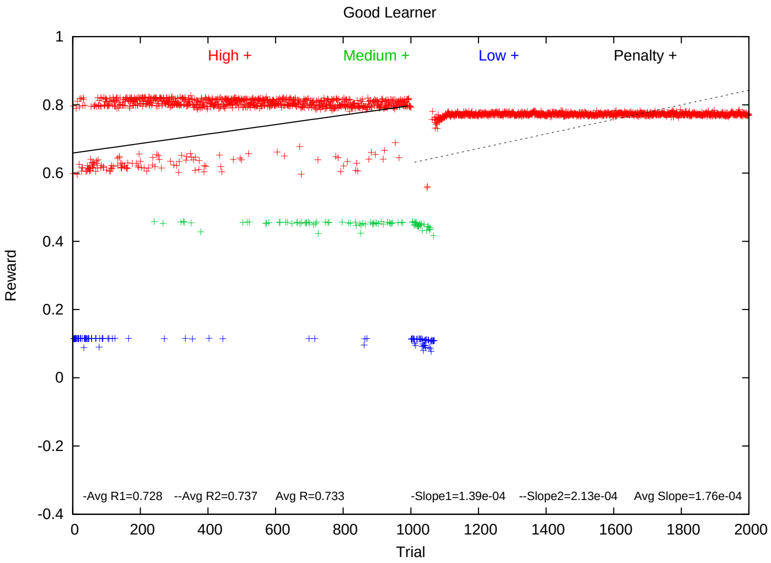

Figure 5.33: Evolution of Learning and Instincts—Good (Substantial) Learning example in self-care niche.

Figure 5.33 shows a good learning individual that starts by exploring two different paths to the high rewarding light. It also occasionally explores (initially) the low rewarding light and (later) the medium rewarding light. However, it finds a near-optimal path to the high rewarding light quickly. In the second half of its lifetime, once the light it was visiting becomes a low rewarding light, it quickly finds its path to the best rewarding light. Although it never finds the optimal path to the high rewarding light during the second half of its lifetime, it still performs sufficiently to be categorized as a good learner.

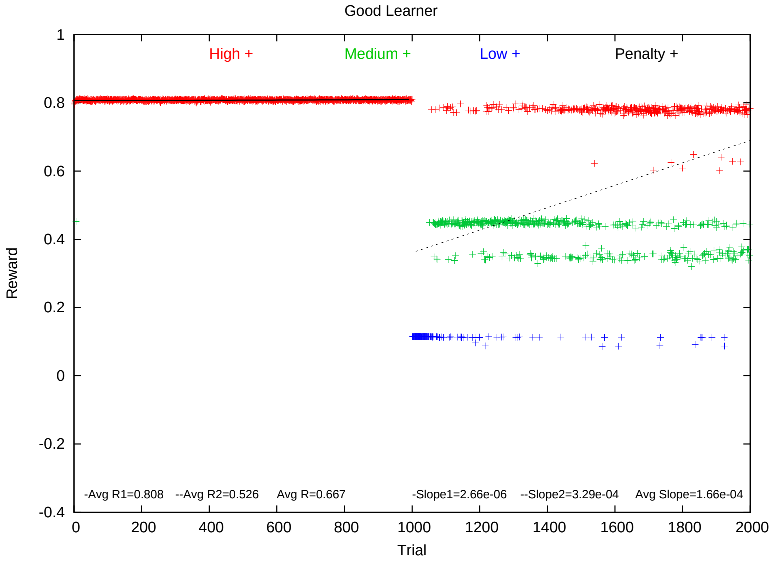

Figure 5.34: Evolution of Learning and Instincts—Good (Substantial) Learning example in self-care niche.

Figure 5.34 shows an individual that finds the high rewarding light very early in its lifetime. It also finds the medium rewarding light once but stays focused on visiting the high rewarding light during the first half of its lifetime. In the second half of its lifetime, once the high rewarding light becomes the low rewarding light, it explores to find both the medium and high rewarding lights. During the second half of its lifetime, this individual shifts its focus from low to medium and then eventually to high rewarding lights. Its overall average fitness (reward collected) is above the comparison threshold; therefore, this individual is categorized as a good learner.

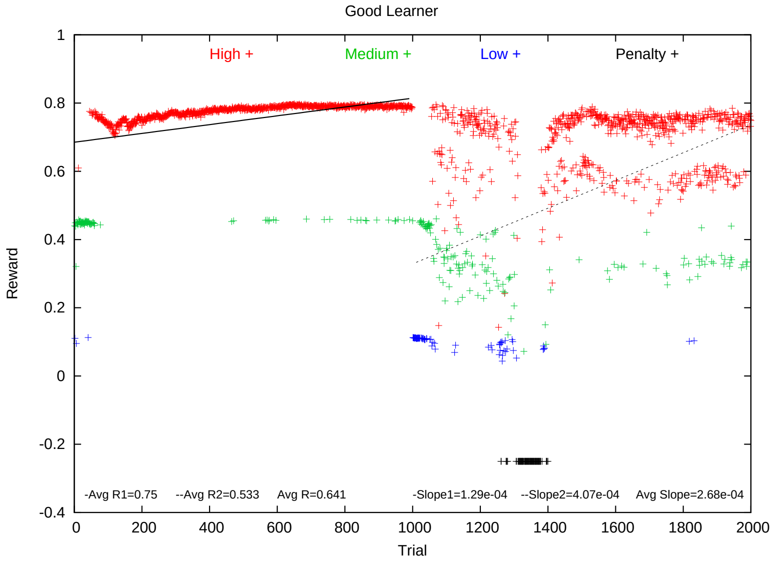

Figure 5.35: Evolution of Learning and Instincts—Good (Substantial) Learning example in self-care niche.

Figure 5.35 shows an example of a good learner that shows learning during both halves of its lifetime. During the first half of its lifetime, it learns to find the high rewarding light after visiting the low and the medium rewarding lights. Then it continuously improves on its path to the high rewarding light. During the second half of its lifetime, it finds the high rewarding light quickly; however, it changes its weights in such a way that it experiences consecutive failures. After some exploration, it eventually reduces the amount of exploration and gradually shifts its focus on visiting the high rewarding light more often.

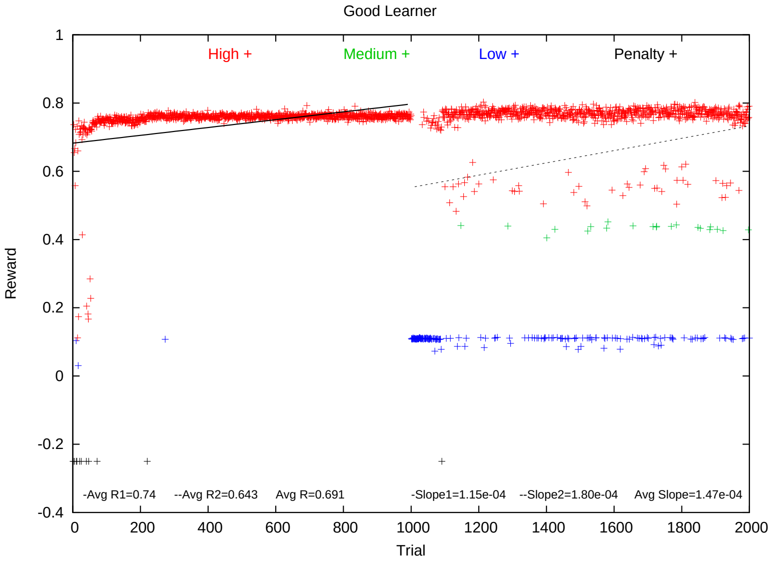

Figure 5.36: Evolution of Learning and Instincts—Good (Substantial) Learning example in self-care niche.

Figure 5.36 shows an individual that continuously improves on its path to the high rewarding light after initial failures and exploration during the first half of its lifetime. In the second half of its lifetime, this individual is more exploratory as compared to its earlier behavior as it visits the low rewarding light continuously and also finds a slower route to the high rewarding light and occasionally visits the medium rewarding light as well. Nonetheless, this individual performs well during its lifetime and is categorized as a good learner based on its average rewards during each half of its lifetime and overall.

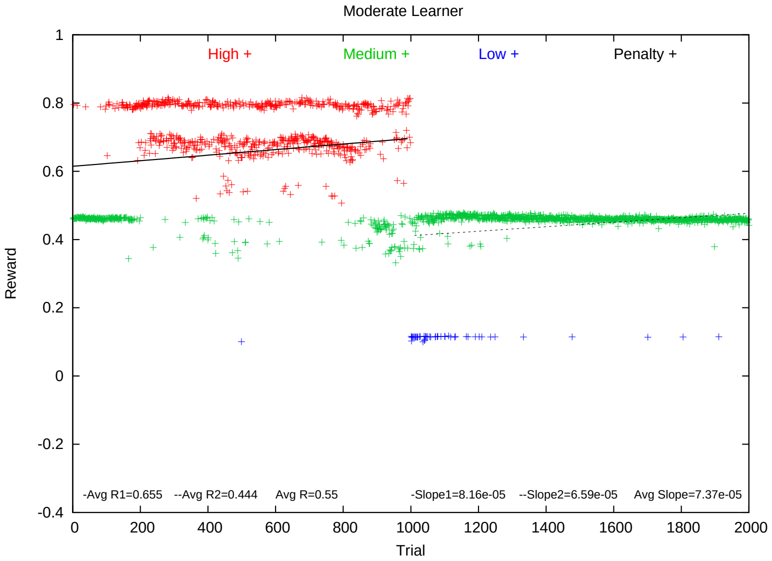

Figure 5.37: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.37 shows a moderate learning individual that visits the medium and the high rewarding lights during the first half of its lifetime. It also finds two different paths to the high rewarding light. Between the three experiences it stays exploratory however, and mostly keeps going to the high rewarding light. In the second half of its lifetime, when the high rewarding light it was visiting becomes the low rewarding light, it shifts its focus and chooses a better reward by going to the medium rewarding light.

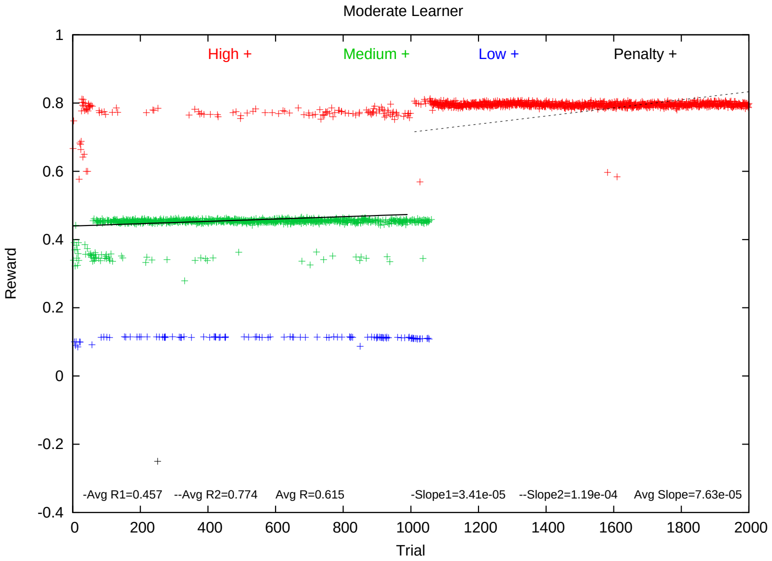

Figure 5.38: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.38 shows an individual that is similar to the one shown in Figure 5.37 in the sense that it shows similar behaviors but in a different half of its lifetime. This individual finds all three lights initially and goes to the medium rewarding light mostly during the first half of its lifetime. Note that it keeps exploring the other two lights as well. In the second half of its lifetime, once the low rewarding light it was exploring becomes the high rewarding light, it shifts its weights in such a way that it finds the high rewarding light and reduces its exploration. During the second half of its lifetime, it stays conservative.

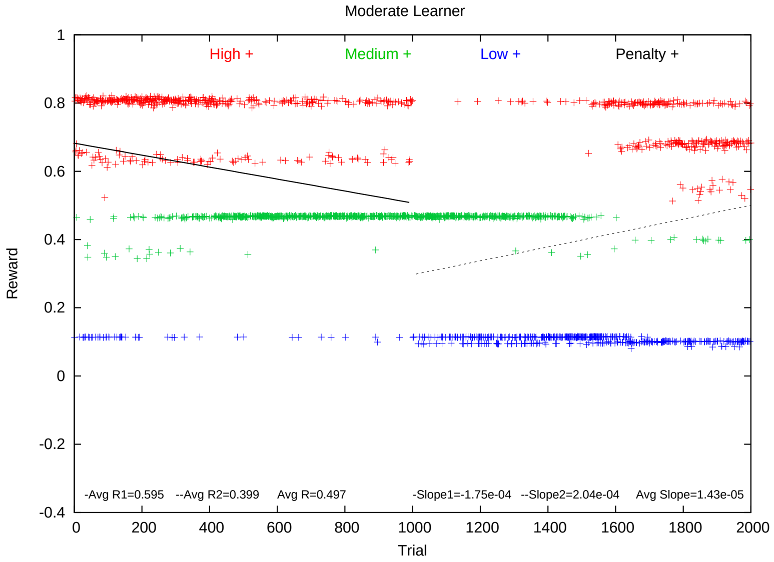

Figure 5.39: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.39 shows an individual that shifts its weights in such a way that it learns to go to the medium rewarding light after settling on the high rewarding light for quite some trials. In the second half of its lifetime, it initially explores the low rewarding light frequently and shifts its path to the low rewarding light; however, during the last one fifth of its lifetime, it finds the high rewarding light source again. During this time, it starts to reduce its frequency of visits to the low rewarding light and increases its frequency of visits to the high rewarding light. Based on the average rewards in each half of its lifetime and overall during its lifetime, this individual is categorized as a moderate learner.

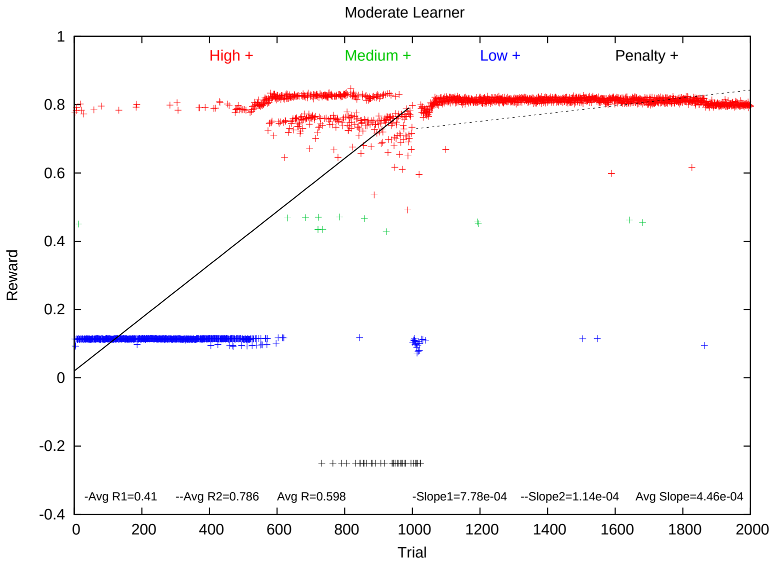

Figure 5.40: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.40 shows an individual that finds all three lights initially but becomes conservative and exploits the low rewarding light during most of the first half of its lifetime. Note that it keeps exploring the high rewarding light occasionally. After about 600 trials, it shifts its policy in such a way that it starts going to the high rewarding light more frequently. During this period, it often fails as well. The second half of this individual's lifetime shows that the policy shift helps it learn quickly about the new high rewarding light after the change. During this half, it becomes conservative and exploits the high rewarding light. This individual is categorized as a moderate learner.

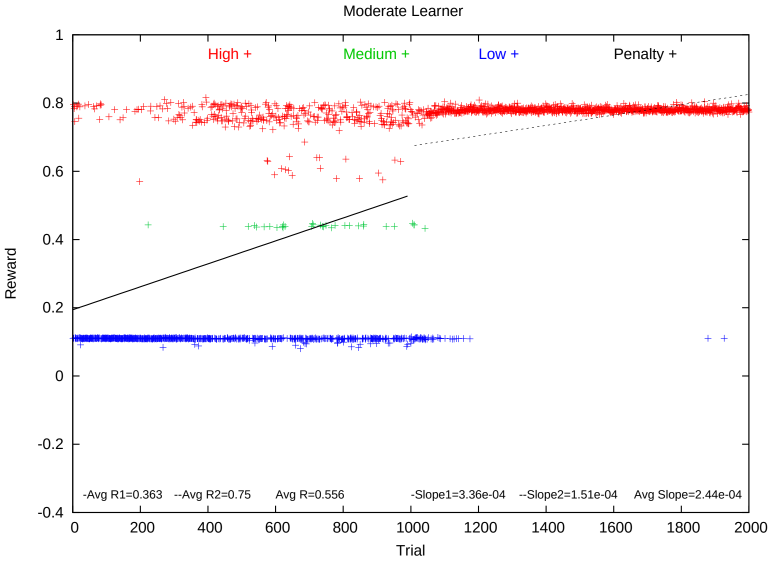

Figure 5.41: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.41 shows an individual that finds the low and the high rewarding lights initially but mostly exploits the low rewarding light during the first half of its lifetime. Note that it keeps exploring mostly the high rewarding light but also visits the medium rewarding light occasionally. In the second half of this individual's lifetime, once the low rewarding light becomes the high rewarding light, it becomes conservative and reduces the number of visits to the other lights. This individual is categorized as a moderate learner mostly due to the high average fitness during the second half of its lifetime.

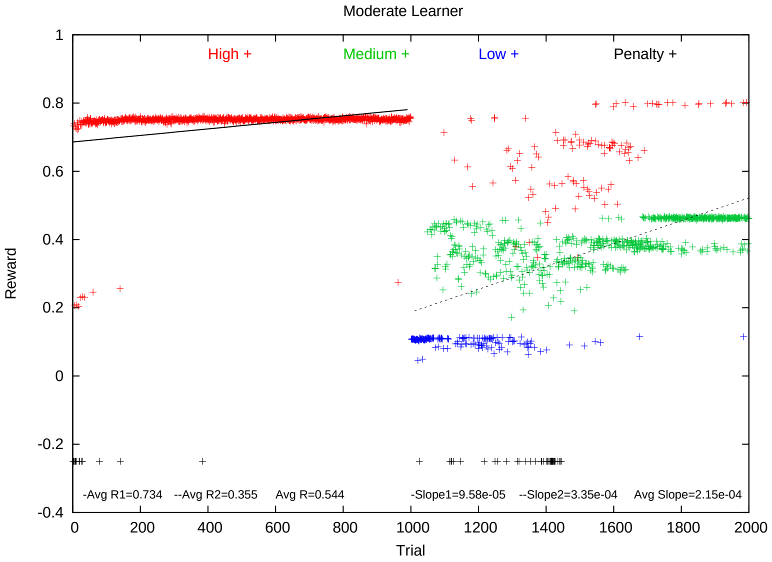

Figure 5.42: Evolution of Learning and Instincts—Moderate (Substantial) Learning example in self-care niche.

Figure 5.42 shows another moderate learning individual that initially learns to avoid failures and exploits the best of the paths experienced to the high rewarding light. Afterwards, it becomes conservative and keeps visiting the high rewarding light with very little exploration during the first half of its lifetime. In the second half of its lifetime, it learns from its failures and gradually shifts its weights in such a way that it first focuses on the low rewarding light, followed by gradually changing its path to visit the medium rewarding light although by taking different routes. During this later half, it often explores the high rewarding light as well but never shifts it's policy to exploit the high rewarding light.

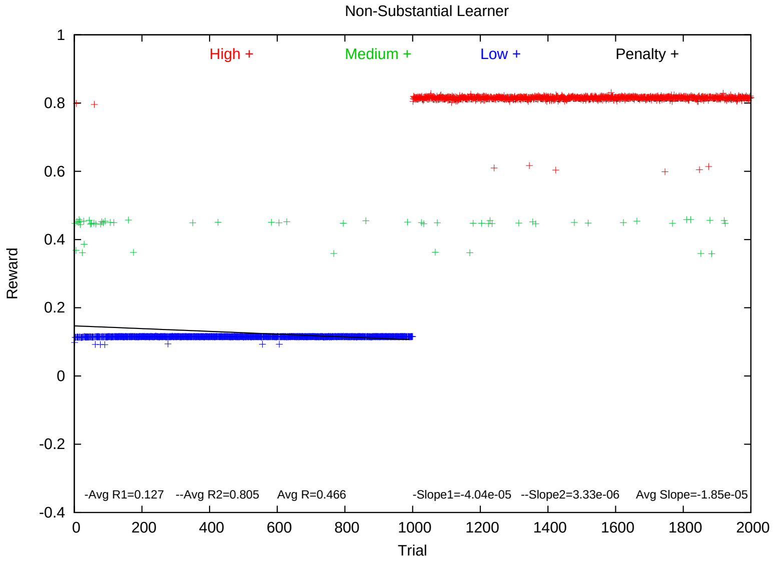

Figure 5.43: Evolution of Learning and Instincts—Non-Substantial Learning typical example in self-care niche.

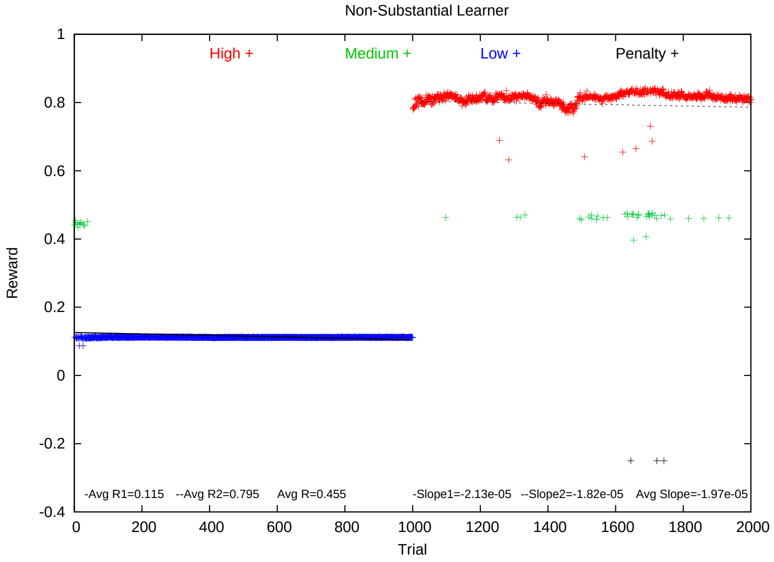

Figure 5.44: Evolution of Learning and Instincts—Non-Substantial Learning example in self-care niche.

Figures 5.43 and 5.44 show two of the typical non-substantial learning cases where both individuals start with the instincts of visiting the low rewarding light and never improve on that. Both these individuals also visit the medium rewarding light occasionally; however, they never show any learning to shift their focus to a better reward. Note that the individual in Figure 5.44 explores and slightly improves during the second half of its lifetime by attempting different paths to the highest rewarding light. However, both these individuals can be seen as instinctive individuals.

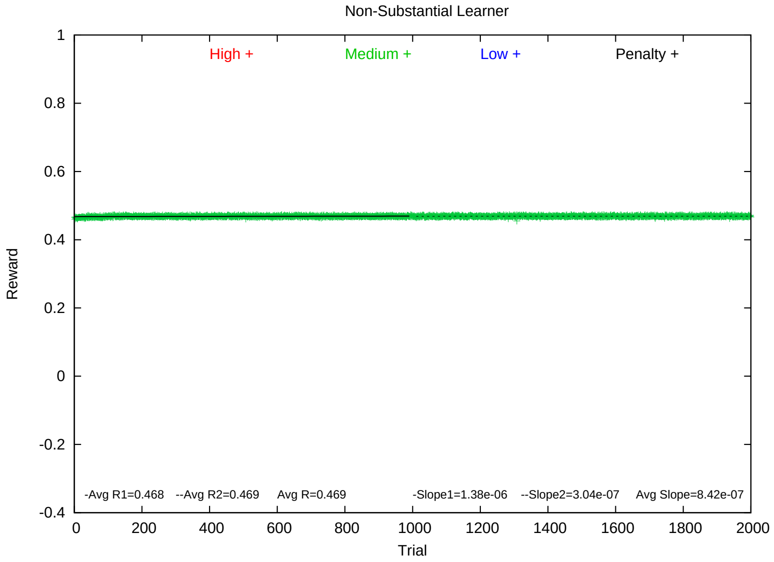

Figure 5.45: Evolution of Learning and Instincts—Non-Substantial Learning another typical example in self-care niche.

Figure 5.45 shows a typical individual that behaves instinctively. Note that this individual is similar to the ones shown in Figures 5.43 and 5.44 on the basis that there is no evidence of learning in all three cases. This individual happens to have instincts (initial weights) for visiting the medium rewarding light and it keeps doing that for all of its lifetime.

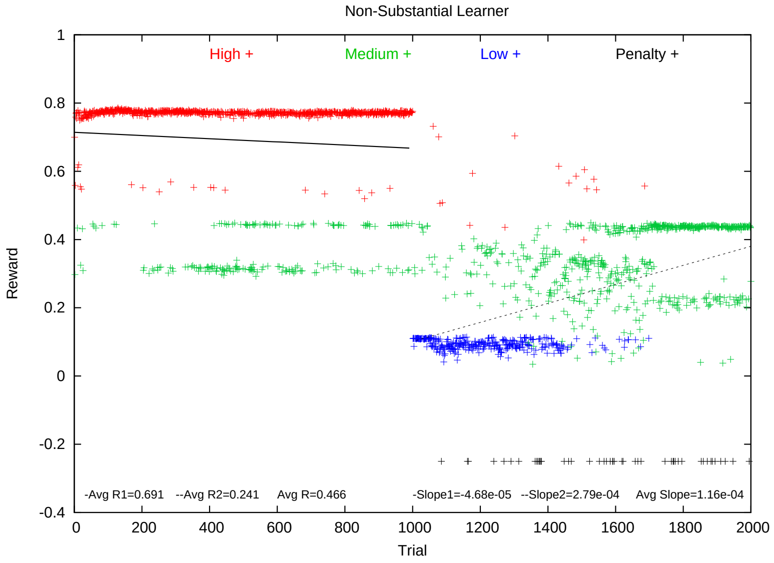

Figure 5.46: Evolution of Learning and Instincts—Non-Substantial Learning another example in self-care niche.

Finally, Figure 5.45 demonstrates an individual that happens to show learning with good instincts initially and exploits the high rewarding light. In the second half of its lifetime, it learns to avoid failures and then low rewarding light and shifts its weights in such a way that it finds the medium rewarding light consistently during the last one fifth of its lifetime. This individual still does not get substantial reward in the second half of its lifetime and is thus categorized as a non-substantial learner based on the criteria defined.

5.4 Evolution of Learning Regardless of Instincts

In this section the results of both of the previous experiments are combined and compared to observe any significance. These combined results will be used to answer the last set of hypotheses that represent the overall conclusion, i.e., does nurturing promote the evolution of learning in changing environments (with or without instincts).

In this section, statistical analysis is performed on the combined data. Following are the sub-hypothesis that can be answered using the related analysis. The order of the tables and placement of the larger tables in an appendix is similar to the previous two sections.

5.4.1 Statistical Analysis

Category Likelihood

1.3 Learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

1.3.1 **CL-SE** Substantial learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

		Learning Category		
		Substantial	Non-Substantial	Total
Niche	Nurturing	(29+30) = 59	(1+0) = 1	60
	Self-Care	(19+24) = 43	(11+6) = 17	60

Table 5.21: Evolution of Learning and the Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Substantial vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.21 shows the results for hypothesis **CL-SE** (1.3.1). The results for the nurturing niche are 59 substantial learners and 1 non-substantial learner. The results for the self-care niche are 43 substantial learners and 17 non-substantial learners. These results are statistically significant (Fisher's exact test, two-tailed p <0.0001).

1.3.2 **CL-GE** Good learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

		Learning Category		
		Good	Not-Good	Total
Niche	Nurturing	(27+26) = 53	(3+4) = 7	60
	Self-Care	(8+7) = 15	(22+23) = 45	60

Table 5.22: Evolution of Learning and the Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Good vs. Not-Good Learning Category Likelihood Statistics.

Table 5.22 shows the results for hypothesis **CL-GE** (1.3.2). The results for the nurturing niche are 53 good learners and 7 not-good learners. The results for the self-care niche are 15 good learners and 45 not-good learners. These results are statistically significant (Fisher's exact test, two-tailed p <0.0001).

1.3.3 **CL-BE** Better learning is more likely to evolve with nurturing than without nurturing, regardless of instincts.

		Learning Category			
		Good	Moderate	Non-Substantial	Total
Niche	Nurturing	(27+26) = 53	(2+4) = 6	(1+0) = 1	60
	Self-Care	(8+7) = 15	(11+17) = 28	(11+6) = 17	60

Table 5.23: Evolution of Learning and the Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Good vs. Moderate vs. Non-Substantial Learning Category Likelihood Statistics.

Table 5.23 shows the results for hypothesis **CL-BE** (1.3.3). The results for the nurturing niche are 53 good learners, 6 moderate learners, and 1 non-substantial learner. The results for the self-care niche are 15 good learners, 28 moderate learners, and 17 non-substantial learners. These results are statistically significant (chi-squared test, $p < 0.00001$).

Performance Continuum

Next is the list of performance continuum hypotheses answered based on *relative success.*

2.3 **PC-OE** Nurturing promotes the evolution of higher performing behaviors regardless of instincts.

The relative success for the combined 60 nurturing niche individuals has a mean of 82.7 and standard deviation of 9.11 while the relative success for the combined 60 self-care niche individuals has a mean of 62.8 and a standard deviation of 9.53 (see

Table 5.24). These results were statistically significant (t-test,p <0.0001).

Group	Nurturing	Self-Care
Mean	82.6687	62.7563
SD	9.1124	9.5257
SEM	1.1764	1.2298
N	60	60

Table 5.24: Evolution of Learning and Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.3.1.

2.3.1 **PC-GE** Nurturing promotes the evolution of higher performing behaviors within good learners regardless of instincts.

The relative success for the 53 nurturing niche good learners has a mean of 84.8 and standard deviation of 7.14 while the relative success for the 15 self-care niche good learners has a mean of 73.7 and a standard deviation of 7.64 (see Table 5.25). These results are statistically significant (t-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	84.7970	73.7327
SD	7.1403	7.6422
SEM	0.9808	1.9732
N	53	15

Table 5.25: Evolution of Learning and Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Good Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.3.2.

2.3.2 **PC-SE** Nurturing promotes the evolution of higher performing behaviors within substantial learners regardless of instincts.

The relative success for the 59 nurturing niche substantial learners has a mean of 83.1 and standard deviation of 8.41 while the relative success for the 43 self-care niche substantial learners has a mean of 66.9 and a standard deviation of 7.80 (see Table 5.26). These results are statistically significant (t-test, p <0.0001).

Group	Nurturing	Self-Care
Mean	83.1425	66.9433
SD	8.4120	7.7965
SEM	1.0952	1.1890
N	59	43

Table 5.26: Evolution of Learning and Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.3.3.

2.3.3 **PC-ME** Nurturing promotes the evolution of higher performing behaviors within moderate learners regardless of instincts.

The relative success for the 6 nurturing niche moderate learners has a mean 68.5 and a standard deviation of 2.30 while the relative success for the 28 self-care niche moderate learners has a mean of 63.3 and a standard deviation of 4.99 (see Table 5.27). These results are statistically significant (t-test, $p = 0.0185$).

Group	Nurturing	Self-Care
Mean	68.5283	63.3061
SD	2.3029	4.9944
SEM	0.9402	0.9439
N	6	28

Table 5.27: Evolution of Learning and Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Moderate Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.3.4.

2.3.4 **PC-NE** Nurturing promotes the evolution of higher performing behaviors within non-Substantial learners regardless of instincts.

The relative success for the 1 nurturing niche non-substantial learner is 54.7 while the relative success for the 17 self-care niche non-substantial learners has a mean of 52.2 and a standard deviation of 2.87 (see Table 5.28). Having a single data point for the nurturing niche means that it is not possible to run a *t*-test on this data set.

Group	Nurturing	Self-Care
Mean	54.71	52.165882
SD	-	2.871222
SEM	-	0.7178
N	1	17

Table 5.28: Evolution of Learning and Evolution of Learning and Instincts combined (Nurturing vs. Self-Care)—Non-Substantial Learning Performance Continuum Statistics.

The detailed numbers from which these statistics are calculated are shown in Appendix Table A.3.5.

5.5 Summary

This chapter presents detailed exemplar results from the experiments and shows their statistical analysis. The statistical analysis shows that nurturing niche outperforms self-care niche by significant margin. The exemplars show learning trends of the individuals from both niches. Chapter 6 discusses these results in detail.

Chapter 6

Discussion

This chapter discusses various aspects of the results, in particular what those results suggest regarding the validation of the hypothesis. A few interesting cases are discussed as well, including those that can be argued against the proposed evaluation criteria introduced in Section 5.1.

During the evolutionary process, ten generations were determined to be sufficient for the evolutionary courses to diverge in the two niches. The data shows (see Figures 5.1, 5.2, 5.23, and 5.24) that the evolutionary process improves the overall population fitness during the course of 10 generations and is also able to find near-optimal individuals.

6.1 Validation of Hypothesis

Both types of results, i.e., category likelihood and performance continuum, indicate that the nurturing niche favors the evolution of learning as compared to the self-care niche by overwhelmingly better performance. This indicates that nurturing plays an important role in nonstationary environments where maximum success for individuals comes from learning about the changing environment. Note that none of the possible multiple comparisons issues see Section 4.1.3 arose in the results.

There are nine hypotheses in category likelihood, all of which are found to be supported by the statistical hypotheses tests as shown in Table 6.1. In the performance continuum results, there are 15 sub-hypotheses, out of which ten are found to be supported by the statistical hypothesis tests as shown in Table 6.2. Three out of the remaining five sub-hypotheses, all relating to non-substantial learners (Hypothesis 2.1.4 PC-NA, Hypothesis 2.2.4 PC-NP, and Hypothesis 2.3.4 PC-NE), lacked sufficient samples for nurturing to carry out statistical hypothesis tests. This is expected as nurturing overwhelmingly outperforms self-care so very few non-substantial cases are found in the nurturing niche. The only sub-hypotheses that are not supported by the statistical tests are the moderate learning comparisons in the absence of instincts and in the presence of instincts (Hypothesis 2.1.3 PC-MA and Hypothesis 2.2.3 PC-MP). Again the number of moderate learning instances in the nurturing niche are far fewer than in the self-care niche. A small number of samples means that a t-test will lack power, so it isn't surprising that the null hypothesis cannot be rejected in these cases, even though the means appear to be higher for the nurturing condition for both hypotheses[1]. Note that when the data from these two hypotheses are pooled (as they are for Hypothesis 2.3.3 PC-ME), the results are statistically significant.

Altogether, the statistical comparisons very strongly indicate that the main hypothesis is supported and that nurturing promotes the evolution of learning in changing environments, i.e., nurturing outperforms self-care.

[1] Note that these means are not being used to base any conclusion due to insufficient data

		Instincts		
		Absent	**P**resent	**E**ither/Both
Comparisons	**S**ubstantial	✓ **CL-SA** (1.1.1)	✓ **CL-SP** (1.2.1)	✓ **CL-SE** (1.3.1)
	Good	✓ **CL-GA** (1.1.2)	✓ **CL-GP** (1.2.2)	✓ **CL-GE** (1.3.2)
	Better	✓ **CL-BA** (1.1.3)	✓ **CL-BP** (1.2.3)	✓ **CL-BE** (1.3.3)

Table 6.1: Category Likelihood Hypothesis Results summary — **CL** stands for **C**ategory **L**ikelihood hypotheses. **S**ubstantial comparison: (2-way Substantial vs. Not Substantial). **G**ood comparison: (2-way Good vs. Not-Good). **B**etter comparison: (3-way Good vs. Moderate vs. Non-Substantial). Abbreviations **A**, **P**, and **E** stand for Absent, Present, and Either instincts respectively. Check mark shows that the hypothesis is supported by the statistical tests.

		Instincts		
		Absent	**P**resent	**E**ither/Both
Comparisons	**O**verall	✓ **PC-OA** (2.1)	✓ **PC-OP** (2.2)	✓ **PC-OE** (2.3)
	Good	✓ **PC-GA** (2.1.1)	✓ **PC-GP** (2.2.1)	✓ **PC-GE** (2.3.1)
	Substantial	✓ **PC-SA** (2.1.2)	✓ **PC-SP** (2.2.2)	✓ **PC-SE** (2.3.2)
	Moderate	× **PC-MA** (2.1.3)	× **PC-MP** (2.2.3)	✓ **PC-ME** (2.3.3)
	Non-Substantial	— **PC-NA** (2.1.4)	— **PC-NP** (2.2.4)	— **PC-NE** (2.3.4)

Table 6.2: Performance Continuum Hypothesis Results summary — **PC** stands for **P**erformance **C**ontinuum hypotheses. **O**verall comparison, **G**ood comparison, **S**ubstantial comparison, **M**oderate comparison, **N**on-Substantial comparison. Abbreviations **A**, **P**, and **E** stand for Absent, Present, and Either instincts respectively. Check mark shows that the hypothesis is supported by the statistical tests. Cross mark shows that the hypothesis is not supported by the statistical tests. — indicates there is not enough data to perform a statistical comparisons.

6.2 Notable Cases

6.2.1 Evolution of Learning

This section presents the best learners from both the nurturing and self-care niches. Further, it discusses a few individuals from the self-care niche of the evolution of learning. It can be argued that these individuals may belong to different categories of learning than the ones to which they are assigned based on objective criteria.

Best Individuals

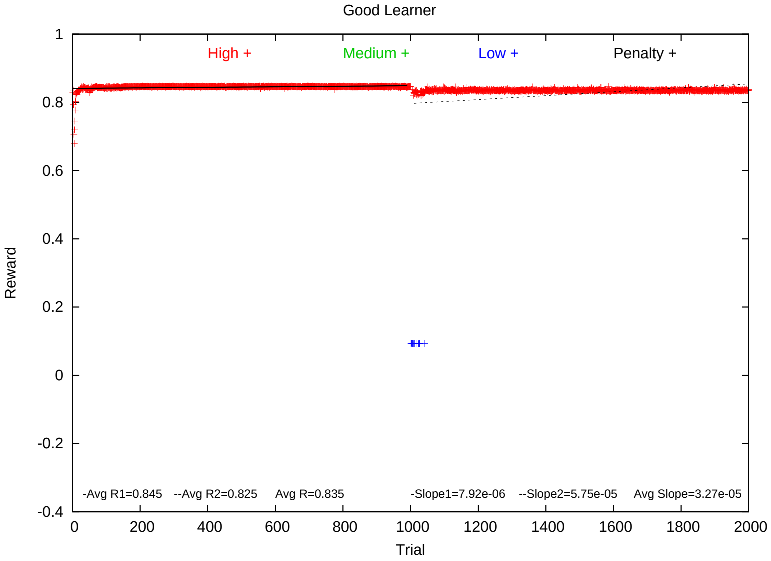

Figure 6.1: Evolution of Learning—Nurturing. Best learning individual found out of 30 runs.

Figure 6.1 shows the best individual found in this experiment in the nurturing niche. This individual starts with knowledge (initial random weights) for going to the best rewarding light source but not for using the optimal path. However, it very quickly improves its path and finds a near-optimal path. It becomes conservative afterwards until the change in the environment happens at trial 1000. Once it receives a lower reward from that light, this individual quickly shifts its policy again and finds the best rewarding light source again very quickly. It keeps exploiting the best rearding light afterwards until the end of its lifetime. This individual can be said to be the best of the best although such a learning category is not formally defined.

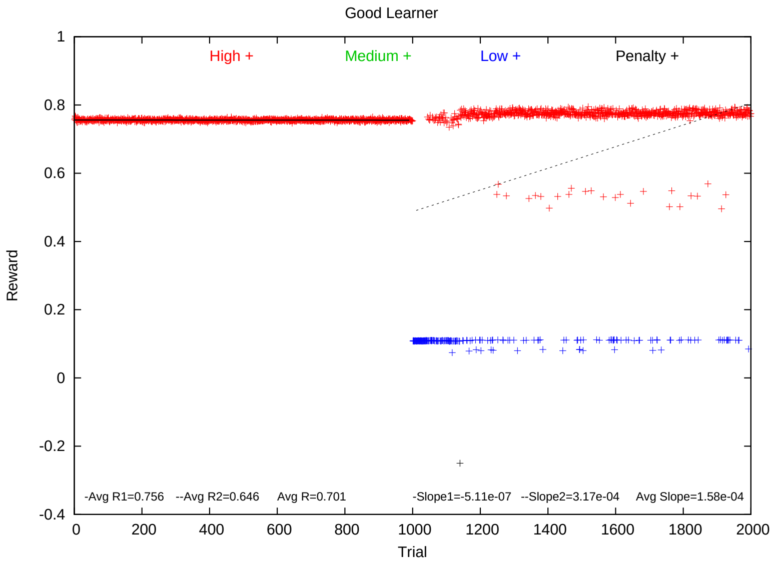

Figure 6.2: Evolution of Learning—Self-Care. Best learning individual found out of 30 runs.

Figure 6.2 shows the best individual from the self-care niche. This individual starts with good instincts to visit the highest rewarding light; however, it also does not follow the optimal path. It's exploration is minimal and thus it never learns a better path. After the reward switch happens and the environment changes, it performs well by only exploring a little and mostly exploiting the highest rewarding light. However, unlike the best individual in the nurturing case there is substantial space for improvement that is not utilized by this individual.

Self-Care Arguable Results

This section discusses the results that can be argued not to perfectly fit the evaluation criteria defined in Section 5.1. All of these cases come from the self-care niche.

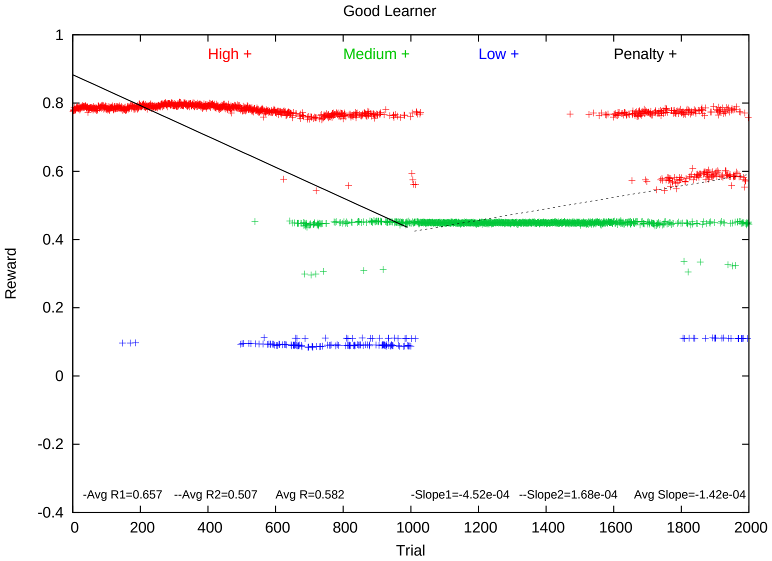

Figure 6.3: Evolution of Learning—Self-Care. A Good learning case that can be argued to be a Moderate learning individual.

Figure 6.3 shows an individual with quite decent initial weights. Most of the first half of its lifetime it exploits the highest rewarding light source. However, as the trials progress, the individual begins to seek out the medium and low rewarding lights more and more often, causing its average fitness to decline. During the second half of its lifetime, it focuses almost exclusively on the medium rewarding light for 500 trials or so. During the last portion of its life, it reduces its focus on the medium rewarding light and begins to focus on two distinct routes to the high rewarding light, one much slower than the other. It also samples the low rewarding light several times near the end of its lifetime. According to the objective criteria, this individual is a good learner, since it outperformed the lifetime average fitness of the theoretically best instinctive individual during both halves of its lifetime. However, the learning performance of this individual during the first half of its lifetime is rather poor, as it shifted from an initial good policy toward a worse policy up until trial 1000.

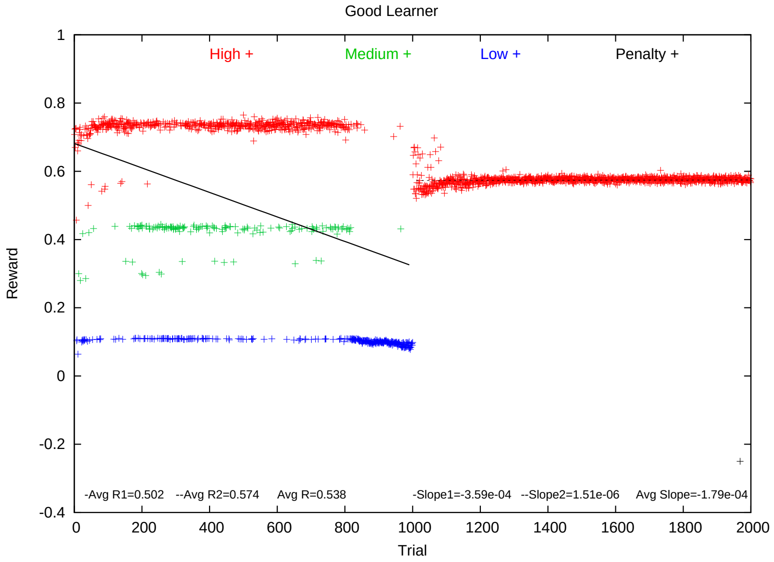

Figure 6.4: Evolution of Learning—Self-Care. A Good learning case that can be argued to be a non-substantial learning individual.

In Figure 6.4, the individual shown moves to the high rewarding light source frequently, although it also visits the medium and low rewarding lights fairly often. However, it mostly shifts over to visiting the low rewarding light by the end of the first half of its lifetime. After the change of reward happens in the environment, the earlier low rewarding light now yields the high reward; thus the individual never has to learn or explore anything new in order to receive a higher than expected reward. It can be argued that the performance in the first half of its lifetime is primarily due to its innate policy and that the primary effect of its weight adjustments during that portion of its life is the shift to the low rewarding light between trials 800 and 1000 — a shift from a good policy to a worse one. Moreover, it can be argued that the individual's performance in the second half of its lifetime was primarily due to continuing with a behavior that just happened to produce moderately good rewards, even though it was first adopted when it produced poor rewards and even though better rewards were available. Thus it can be argued that this individual should be categorized as a non-substantial learner rather than a good learner.

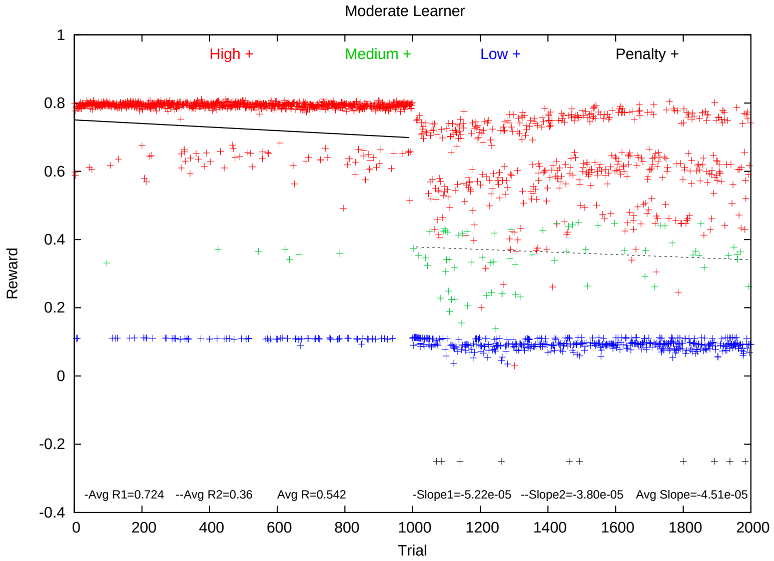

Figure 6.5: Evolution of Learning—Self-Care. A Moderate learning case that can be argued to be a non-substantial learning individual.

The individual shown in Figure 6.5 focuses primarily on the high rewarding light during the first half of its lifetime; however the regression line shows that it actually shifts its attention to the other lights and slower paths to the high rewarding light as the trials progress. In the second half, it continues to shift its focus to the low rewarding light resulting in a negative slope to its second regression line as well. Based on these arguments, one can argue that this individual might not be correctly classified as a moderate learner but is instead a non-substantial learner.

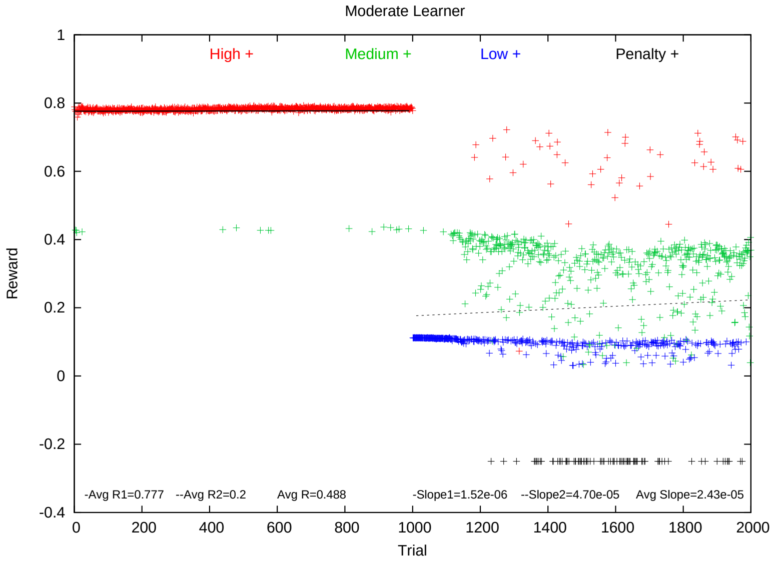

Figure 6.6: Evolution of Learning—Self-Care. A Moderate learning case that can be argued to be a non-substantial learning individual.

In Figure 6.6, the individual behaves mostly similar to an instinctive individual in the first half of its lifetime, while the learning in the second half of its lifetime is rather unimpressive. It does shift its attention somewhat to the medium rewarding light, but it also increases its frequency of failure and takes somewhat slower paths to the low and moderate rewarding lights as the trials progress. Its average reward in the second half of its lifetime is poor and its overall average is right on the borderline. Given the fact that most of its fitness comes from largely innate behaviors and the minimal amount of learning exhibited during the second half of its lifetime, this individual can be argued to be a non-substantial learner instead of a moderate learner.

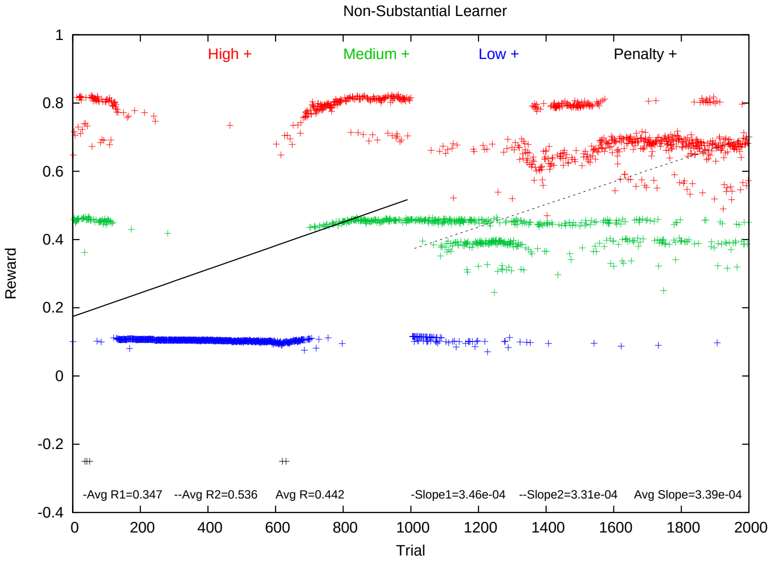

Figure 6.7: Evolution of Learning—Self-Care. A non-substantial learning case that can be argued to be a Moderate learning individual.

Figure 6.7, the individual exhibits learning in both halves of its lifetime. On the very first trial it moves to the low rewarding light. After that, it shifts its attention to the medium and high rewarding lights and visits those frequently during the first 100 trials. After that, it shifts its attention back to the low rewarding light (somewhat "unlearning" what it had learned) for several hundred trials. Then it shifts its attention again to the high and medium rewarding lights for the last 300 or 400 trials before the reward switch. When the high rewarding light becomes the low rewarding light, the individual finds itself splitting its attention between the low rewarding light and the medium rewarding light. It then shifts its attention away from the low rewarding light to the medium rewarding light and then to the high rewarding light. Despite learning in both halves of its lifetime, overall it falls into non-substantial learning category according to the objective criteria. Still, it can be subjectively argued that this individual deserves to be considered a moderate learner.

6.2.2 Evolution of Learning and Instincts

This section discusses the best learners from both the nurturing and self-care niches. Further, it discusses individuals that can be argued should be labeled categorically in a different way.

Best Individuals

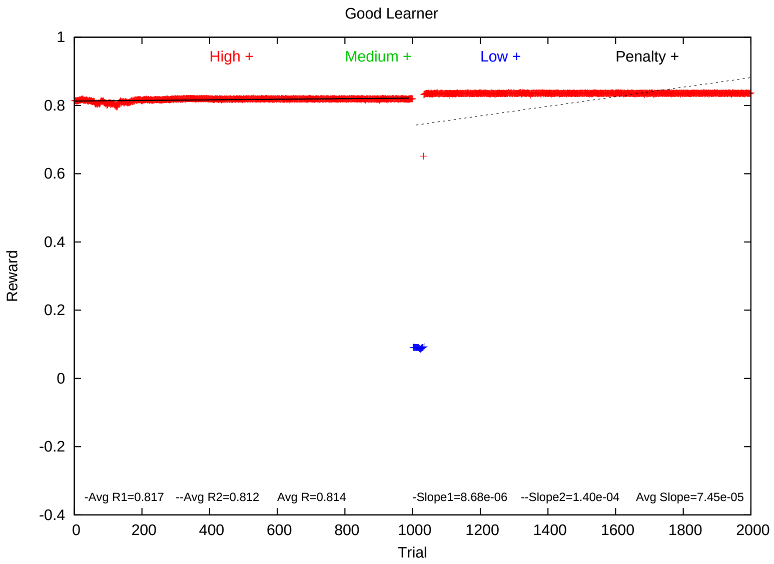

Figure 6.8: Evolution of Learning and Instincts—Nurturing. Best learning individual found out of 30 runs.

Figure 6.8 shows the best learner in this experiment in the nurturing niche. This individual inherits good instincts from its parents and uses those instincts to visit the high rewarding light source without any exploration. The path it finds can be called close to optimal but there is still room for improvement. This can also be seen by comparing Figure 6.8 with Figure 6.1. After the change in the environment at the midway point during the lifetime, it very quickly finds the high rewarding light again and exploits that resource over the remainder of its lifetime.

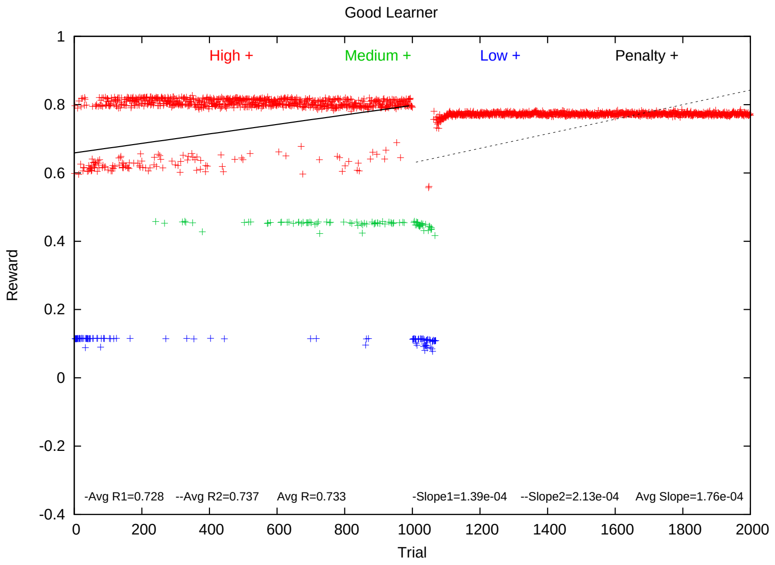

Figure 6.9: Evolution of Learning and Instincts—Self-Care. Best learning individual found out of 30 runs.

Figure 6.9 shows the best individual from the self-care niche. It inherits decent enough instincts to start successfully while a little exploration also helps it find the high rewarding light source. It continues to explore and during the second half of its lifetime finds the new high rewarding light source quickly, albeit using a less optimal path to it. During the second half it never explores after converging to the high rewarding light.

Self-Care Arguable Results

This section presents cases from the self-care niche that can be argued not to perfectly fit the evaluation criteria defined in Section 5.1.

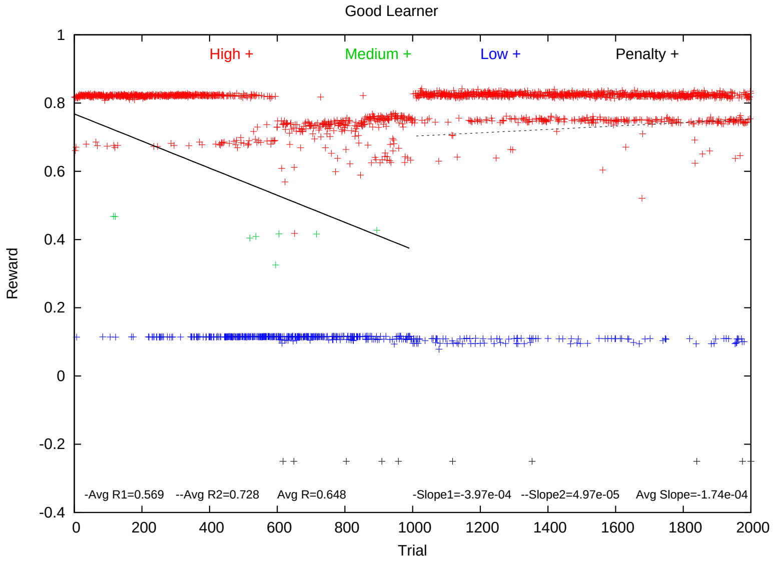

Figure 6.10: Evolution of Learning and Instincts—Self-Care. A Good learning case that can be argued to be a Moderate learning individual.

Figure 6.10 shows an individual that explores a little but mostly initially exploits the high rewarding light source. It gradually learns not to visit that light as frequently but to go to the low rewarding light source frequently. It also slows down in its path to the high rewarding light. By the end of the first half of its lifetime, it visits the low rewarding light quite often and then with the change in environment it keeps exploiting that light. Although it receives enough reward in both halves of its lifetime to be classified as a good learner, because of the decreasing performance in the first half of its lifetime, it can be argued that it should be subjectively categorized as a moderate learner instead.

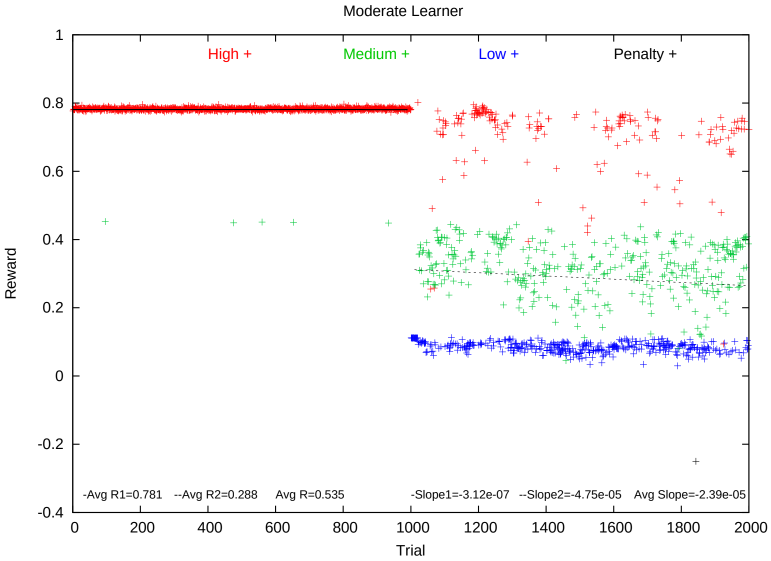

Figure 6.11: Evolution of Learning and Instincts—Self-Care. A moderate learning case that can be argued to be a non-substantial learning individual.

The individual in Figure 6.11 clearly starts with good instincts and never explores much. It also does not improve on its path to the high rewarding light source although it does have quite a lot of space for improvement. During the second half of its lifetime, it explores all the three lights but mostly continues to visit the low rewarding light. This individual can thus be argued to be a non-substantial learner instead of a moderate one.

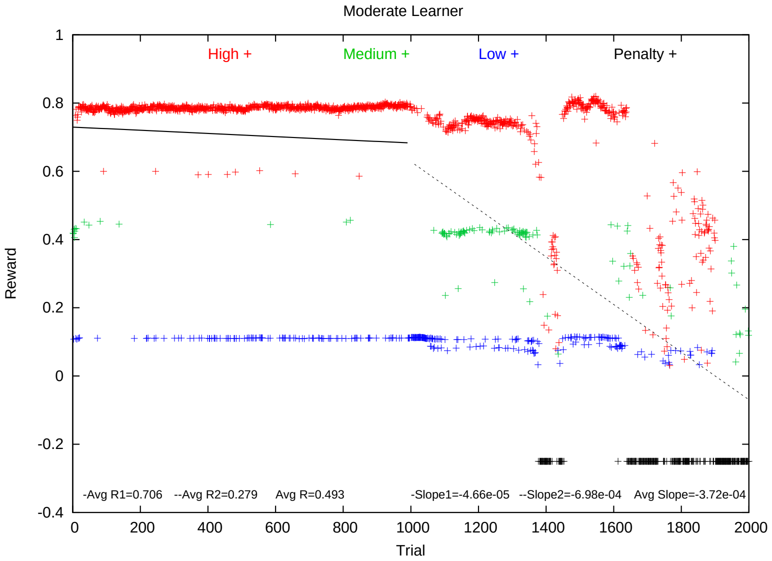

Figure 6.12: Evolution of Learning and Instincts—Self-Care. A moderate learning case that can be argued to be a non-substantial learning individual.

The individual in Figure 6.12 learns initially to find the high rewarding light and exploits it clearly. However, its frequency of visiting the low rewarding light source increases toward the end of the first half of its lifetime. During the second half of its lifetime, its performance is highly erratic and leaves the individual almost exclusively failing by the end of its lifetime. This calls us to look back at this individual critically and question whether it is actually a moderate learner. It can be argued instead that this individual is a non-substantial learner.

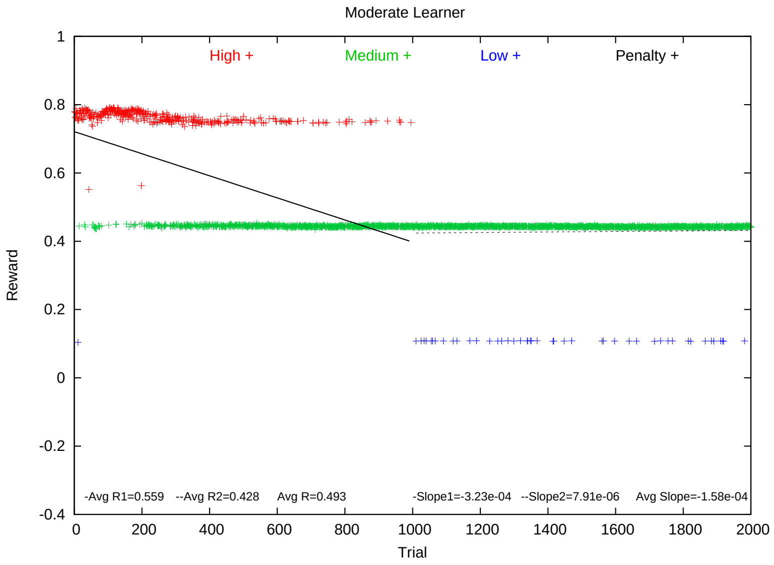

Figure 6.13: Evolution of Learning and Instincts—Self-Care. A Moderate learning case that can be argued to be a non-substantial learning individual.

Figure 6.13 shows a moderate learning individual that begins its lifetime with an innate tendency to visit the high rewarding light but shifts its attention to the medium rewarding light and retains its focus there for the rest of its life. In addition, the path its learns to take to the medium rewarding light is notably suboptimal. One can argue that for all these reasons this individual should be classified as a non-substantial learner.

Nurturing Arguable Results

Here is a case from the nurturing niche that can be argued not to perfectly fit the evaluation criteria defined in Section 5.1.

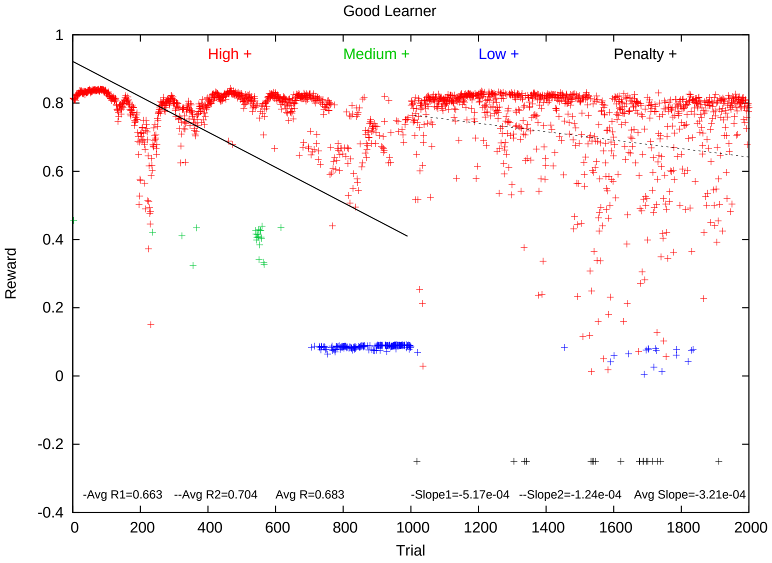

Figure 6.14: Evolution of Learning and Instincts—Nurturing. A Good learning case that can be argued to be a Moderate learning individual.

Figure 6.14 shows a case where the individual, although it obtains very high reward values and visits the best light source available throughout its lifetime, declines in performance in both halves of its lifetime. This individual is fortunate to begin its lifetime with the instinct to visit the light that happens to be the high rewarding light during the first half of its lifetime, yet it shifts its focus to the low rewarding light. That leaves the individual in a fortunate condition again when the environment changes at trial 1000 and the low rewarding light becomes the high rewarding light, yet it drifts to lower performance again as it takes slower routes to that light as well as occasionally failing or traveling to the low rewarding light. It's exploitation is good enough that it stays far above the threshold for the substantial learning category; however, due to the decrease in visits to the high rewarding light and the increasingly suboptimal routes to that light, it can be argued that this individual should be classified at best as a moderate learner and not a good learner.

6.2.3 Summary of Notable Cases

The discussion on various arguable results and their analysis shows that even if these cases move from one category to another, the difference in total is only going to strengthen support for the hypothesis because in almost every case, the result of the argument is to downgrade the category for the rule evolved in the self-care niche.

Further, the results indicate that the self-care individuals can benefit from the instincts as the parents can pass information about the switch from parents to offspring. (Recall that the switch location is the non-changing part of the environment.) This way, the offspring only have to worry about learning the location of the high rewarding light source, which is quite comparable to the nurturing task. However, the significance of the results in the case of evolved learning and instincts indicate that nurturing still outperforms self-care and does far more than instincts to promote the evolution of learning. Therefore, the nurtured instances evolve more and better learning rules than the self-care instances, even when instincts can be evolved. This is a clear indication of the importance of nurturing in the evolution of learning in changing environments.

6.3 Cross Niche Compatibility

It may be hypothesized that if nurturing does indeed promote the evolution of learning, that the learning rules evolved in the nurturing niche might not only allow for greater learning success to be observed in the nurturing niche but also that the learning rules evolved in that niche might be in some way better than the learning rules evolved in the self-care niche. Perhaps, for example, the learning rules from the nurturing niche might be better at cross-niche learning than those evolved in the self-care niche. The following experiments briefly examine this hypothesis.

1. Assign evolved learning rules from the nurturing niche to the self-care niche.

Thus the 30 evolved learning rules with the highest evolutionary fitness from the nurturing niche are executed in the self-care environment. For comparison, the 30 evolved learning rules with the highest evolutionary fitness from the self-care niche are also executed in the same self-care environment.

2. Conversely, assign evolved learning rules from the self-care niche to the nurturing niche. Thus the 30 evolved learning rules with the highest evolutionary fitness from the self-care niche are executed in the nurturing environment. For comparison, the 30 evolved learning rules with the highest evolutionary fitness from the nurturing niche are also executed in the same nurturing environment.

The results of the above experiments are compared using Fisher exact, chi-square, and t tests for analysis as in Chapter 5, considering both categorical likelihood and performance continuum versions of the hypothesis. No significant differences can be seen in the results either way. The reason for these inconclusive results appears to be a floor effect. Based on the objective criteria defined by comparing all learners to the theoretical best instinctive individual (with an average reward value of 0.47, see Section 5.1), the bar is set quite high. Due to time/resource limitations, only a single repetition of each previously evolved learning rule can be executed. Due to the stochastic nature of the problem being solved, this single repetition is probably not enough to conclude in favor or against this additional hypothesis. Thus, due to the high comparison bar and the single repetition of each learning rule, mostly non-substantial learners are found in these results and those individuals are not easily distinguishable (as shown in Table 6.3 and Table 6.4).

In Table 6.3, the count for both types (self-care and nurtured) of learning rules, in the self-care environment (full task), indicate the floor effect while both type of learning rules in nurtured environment (partial task) indicate that the nurtured learning rules count is higher for substantial learning as compared to the self-care individuals.

This is expected as nurturing rules evolved in their own environment show higher numbers than cross task learners. On the other hand, in the evolution of learning and instincts (see Table 6.4), in the self care environment (full task), the number of substantial learning cases for self-care niche are higher than that of the nurturing niche by only a single case. This might have happened due to the instincts of self-care individuals evolved for their own environment. Nonetheless, we see a floor effect in this case. Finally, similar to the evolution of the learning scenario in a cross task situation, the substantial learning count for the nurtured rules is higher than that of the self-care case in nurtured environment. Again this could be because of instincts.

In the evolution of learning experiments (see Section 4.1), a population of 30 individuals is initialized out of which the best evolved individual is selected from the final population. Similarly, running each evolved learning rule 30 times in a cross-niche situation would be likely to give results that are statistically meaningful but would also require more time and resources than are currently available. Due to this, further investigation will be carried out using more repetitions as future work.

			Learning Categories		
			Good	Moderate	Non-Substantial
Environment	**Self-Care**	Self-Care Rules	0	0	30
		Nurtured Rules	0	0	30
	Nurturing	Self-Care Rules	2	3	25
		Nurtured Rules	3	7	20

Table 6.3: **Evolution of Learning**—Cross Niche Compatibility Results. Likelihood count in each category for various cross niche compatibility scenarios.

			Learning Categories		
			Good	Moderate	Non-Substantial
Environment	**Self-Care**	Self-Care Rules	1	1	28
		Nurtured Rules	0	0	30
	Nurturing	Self-Care Rules	4	5	21
		Nurtured Rules	5	7	18

Table 6.4: **Evolution of Learning and Instincts**—Cross Niche Compatibility Results. Likelihood count in each category for various cross niche compatibility scenarios.

6.4 Summary

The analysis of all data presented for all of the sub-hypotheses strongly supports the overall hypothesis that nurturing promotes the evolution of learning. Also, the additional sub-hypotheses are possible but support for them is not available at this time and they will be investigated further as future work.

Chapter 7

Conclusions

The proposed approach is based on nurturing by task simplification. The overall impact of task simplification is similar to that of reward shaping in the sense that learning is observed more often if the task is simpler. The results in this dissertation confirm that niche construction changes the dynamics of the evolutionary process as seen by the nurturing niche outperforming the self-care niche in terms of the evolution of learning. The statistical tests indicate strongly that nurturing promotes the evolution of learning in changing environments.

The work in this dissertation mainly contributes to the fields of robotics, machine learning, evolutionary biology, and potentially computational neuroscience. These contributions in particular are:

1. This is the first study to demonstrate that nurturing promotes the evolution of learning in changing environments. In contrast to this work, Eskridge & Hougen (2012) used an abstract environment with no evolution of learning rule parameters. However, the results in this dissertation conform to their claim that "nurturing promotes the evolution of learning in uncertain environments in which learning would otherwise not be a viable strategy at statistically significant levels."

2. This work contributes to the larger research agenda of the Robotics, Evolution, Adaptation, and Learning Laboratory (REAL Lab), shown in the virtuous cycle (see Figure 1.1) by connecting the evolution of nurturing to the second half of the cycle, learning to be a better nurturer.

3. This dissertation provides an example of finding the right balance of task difficulty for evolution to perform effectively to evolve learning in a changing environment.

4. Chapter 3 introduces an effective stochastic synapse reinforcement learning algorithm that works well for an episodic task in a changing environment in which there is discrete input with perceptual aliasing, continuous output, and terminal/delayed reward.

5. This reinforcement learning algorithm demonstrates good performance using exploration/exploitation (synaptic weight standard deviation eligibility) traces which take into account past actions to refine the exploration/exploitation strategy for the next trial.

This work contributes to the area of nurturing robotics in machine learning. We believe that this area will prove to be highly important, yet it has been mostly overlooked so far. In biology, nurturing and learning are studied separately; however, machine learning provides us a platform to integrate the two with the objective to develop more robust algorithms that can solve arbitrarily complex tasks with more flexibility. In order to develop better machine learning algorithms, this work points out nurturing as an important part of the solution space where robots learn to perform complex tasks. In the work in this dissertation, it is shown that a simple fully connected feed-forward neural network with no hidden units is powerful enough to find a near-optimal path in a space with discrete input and continuous output where an individual robot's episodic task is to find the best light source in the arena. With a terminal/delayed reward scenario, a nurtured individual has to learn a near-optimal path to the most rewarding light source and cope with a change in environment half way through its lifetime. In the case of self-care, the individual's task is even more complex as it has to find a near-optimal path to a light switch first and then to the high rewarding light source in a changing environment.

The reinforcement learning algorithm for real valued units is novel and verified to work in the proposed experimental setup. This algorithm uses temporal-credit-assignment-based stochastic synaptic weight mean adjustment and stochastic synaptic weight standard deviation adjustment as a pair of tools to learn these complex tasks. The synaptic weight standard deviation adjustment acts as a control knob for the exploration/exploitation trade-off at the synapse level. Thus by controlling the knob through considering past actions in the state-action space, an algorithm capable of learning the aforementioned tasks with near-optimal solutions is devised. Further, variations of this algorithm are parameterized and evolution is allowed to take over and find maxima in the solution space using a carefully deigned fitness-based genetic algorithm. The overwhelmingly positive results show that the evolution of learning is promoted by nurturing with significant differences over self-care. In order to verify the generality of the solution, learning is evolved together with instincts to reduce the learning load required for self-care. With good instincts, self-care individuals have a simplified learning task because they only need to learn how to respond to the lights (as the nurtured individual does) as information about the switch, which does not change during the course of evolution, is allowed to pass to them through their genes from their parents. However, the results show that nurturing still outperforms self-care significantly in both category likelihood and on the performance continuum.

As discussed in Chapter 6, it can be argued that some results should be classified differently than they are according to the objective criteria (see categories defined in evaluation criteria in Chapter 5). However, placing them in the argued categories will only strengthen support for the proposed hypotheses.

This research, besides contributing to the field of machine learning also connects earlier research on the evolution of nurturing conducted here at the REAL lab (Leonce et al., 2012), to the later part of the virtuous cycle, learning to nurture. Evolutionary computation models are effective and provide us optimized solutions to problems that are not feasible to be computed in deterministic polynomial time. However, as most evolutionary computation techniques have a tremendous computational cost, it is cumbersome to evolve everything and anything in neural computational models. Due to this reason, it is important to carefully design the learning algorithm to the extent that it can be done and then let evolution do the optimization part. Therefore, the proposed reinforcement learning algorithm is designed by hand and then evolutionary methods are used to optimize and answer the questions asked in the proposed hypotheses.

Chapter 8

Future Work

This work is a contribution to the very early stages in the era of nurturing robotics. The future seems promising based on the results shown by Leonce et al. (2012) for the evolution of nurturing, the results shown by Eskridge & Hougen (2012) for the expanded benefits of learning in the presence of nurturing, and now in this work by demonstrating that nurturing promotes the evolution of learning in changing environments. Moving forward in the big picture, I would like to call the research community to recognize the importance of nurturing in developmental robotics and work towards the development of it. Narrowing down to the immediate future, there can be several parallel and sequential pathways explained below.

This chapter discusses the next major steps in the research to further move forward in the proposed virtuous cycle (Figure 1.1), considers an additional analysis of some of the data, and an notes an important variable that should be investigated in future work. Further, this chapter considers what can be added to the experimental setup to explore other interesting machine learning concepts.

8.1 Learning to be a Better Nurturer

The major step forward from this dissertation work can be to apply various successful learning algorithms evolved in these experiments to a parent to find out if it can learn to be a better nurturer for its offspring. This would mean, for example, that an arena could be designed in which there are several light switches on one end that activate a single light source on the other end. The reward value and variability of the light would be determined by which switch is turned on by the nurturer. The parent's (nurturer's) job would be to choose the right switch to turn on in order to provide maximum reward to its offspring. That would also mean that a communication mechanism between the child and the parent must exist and, preferably, be evolved. Based on the feedback from the child, the parent should improve on its behavior and make intelligent decisions over its lifetime. A reward switch between and during each individual's lifetime will be essential again to make this arena a non-stationary environment to encourage learning. This step forward should not only validate how general, robust, and scalable the evolved learning algorithms are but also will help us understand if the evolution of learning in turn enables an individual to be a better nurturer, thus completing the virtuous cycle.

8.2 Instincts and the Evolution of Learning

Instincts play an important role in helping individuals exploit useful resources. The following section considers how to further investigate the role of instincts in the evolution of learning.

8.2.1 Evolution of Learning vs. Evolution of Learning and Instincts (Nurturing)

Here an analysis is performed by comparing nurturing likelihood counts seen earlier from the evolution of learning and the evolution of learning and instincts experiments. It is interesting to determine whether instincts make any difference in the evolution of learning.

		Learning Category		
		Substantial	Non-Substantial	Total
Experiment	Nurturing (Evolution of Learning)	29	1	30
	Nurturing (Evolution of Learning and Instincts)	30	0	30

Table 8.1: Nurturing—Evolution of Learning vs. Evolution of Learning and Instincts—Substantial vs. Non-Substantial Learning Category Likelihood Statistics.

		Learning Category		
		Good	Not-Good	Total
Experiment	Nurturing (Evolution of Learning)	27	3	30
	Nurturing (Evolution of Learning and Instincts)	26	4	30

Table 8.2: Nurturing—Evolution of Learning vs. Evolution of Learning and Instincts—Good vs. Not-Good Learning Category Likelihood Statistics.

		Learning Category			
		Good	Moderate	Non-Substantial	Total
Experiment	Nurturing (Evolution of Learning)	27	2	1	30
	Nurturing (Evolution of Learning and Instincts)	26	4	0	30

Table 8.3: Nurturing—Evolution of Learning vs. Evolution of Learning and Instincts—Good vs. Moderate vs. Non-Substantial Learning Statistics.

There is no statistically significant difference found between the evolution of learning with or without instincts in the nurturing niche when comparing the number of substantial versus non-substantial learners (Fisher exact test, $p = 1.0$, see Table 8.1), good versus other learners (Fisher exact test, $p = 1.0$, see Table 8.2), and good versus moderate versus non-substantial learners (chi-squared test, $p = 0.43$, see Table 8.3).

Based on these results, there appears to be a ceiling effect as learning is evolved in almost all repetitions in the nurturing niche. However, if the environment is changed in such a way that it is harder for learning to evolve (even in the presence of nurturing), then we might see an impact of instincts and this might help to address the question of how important instincts are in the evolution of learning.

8.2.2 Evolution of Learning vs. Evolution of Learning and Instincts (Self-Care)

This section is a comparison using likelihood counts from the evolution of learning and the evolution of learning and instincts experiments in the self-care niche. It will be again interesting to determine whether instincts make a difference in the evolution of learning.

		Learning Category		
		Substantial	Non-Substantial	Total
Experiment	Self-Care (Evolution of Learning)	19	11	30
	Self-Care (Evolution of Learning and Instincts)	24	6	30

Table 8.4: Self-Care—Evolution of Learning vs. Evolution of Learning and Instincts—Substantial vs. Non-Substantial Learning Category Likelihood Statistics.

		Learning Category		
		Good	Not-Good	Total
Experiment	Self-Care (Evolution of Learning)	8	22	30
	Self-Care (Evolution of Learning and Instincts)	7	23	30

Table 8.5: Self-Care—Evolution of Learning vs. Evolution of Learning and Instincts—Good vs. Not-Good Learning Category Likelihood Statistics.

		Learning Category			
		Good	Moderate	Non-Substantial	Total
Experiment	Self-Care (Evolution of Learning)	8	11	11	30
	Self-Care (Evolution of Learning and Instincts)	7	17	6	30

Table 8.6: Self-Care—Evolution of Learning vs. Evolution of Learning and Instincts—Good vs. Moderate vs. Non-Substantial Learning Statistics.

There is no statistically significant difference found between the evolution of learning with or without instincts when comparing the number of substantial versus non-substantial learners (Fisher exact test, $p = 0.41$, see Table 8.4), good versus other learners (Fisher exact test, $p = 1.0$. see Table 8.5), and good versus moderate versus non-substantial learners (chi-squared test, $p = 0.49$, see Table 8.6).

Here most of the frequency counts both with and without instincts are similar and thus there is no significant difference found in these results as well. However, in this case, there is no ceiling or floor effect. If there is, in fact, a significant role for instincts to play in the evolution of learning in the self-care niche, a substantial experimental redesign appears to be necessary to uncover it. Investigating this parameter is not the main objective of this dissertation but it is worth pointing out for future investigation.

8.3 Evolving the Structure of the Learning Rule

Another interesting aspect to consider is the evolution of the structure of the learning rule (Char, 1997). This might help to explore another class of intelligent learning algorithms that can cope with changing environments better than structurally non-evolved learning algorithms. Additionally, it will be an interesting experiment to find out if nurturing promotes the evolution of these learning algorithms more than structurally non-evolved learning algorithms.

8.4 Fine Grained Control vs. Unit Level Control

As discussed in Section 3.1, the approach of stochastic synaptic weights introduced here contrasts with that of the stochastic activation units approach of Gullapalli (1990) and Williams (1992). In the future, various comparisons can be made between the two approaches to find out if one approach is better than the other for particular domains.

Similarly, the evolution of learning experiments involve the evolution of initial, minimum, and maximum standard deviation values. However, all the stochastic synapses within a given neural network use the same values. An interesting comparison in performance can be made by evolving individual values for each synapse. In fact, not only can these three parameters of the learning algorithm be fine grained but all the other parameters including learning, decay rates, and sliding window size can be evolved separately for each synapse rather than as a unit. It would be more computationally intensive, at least if implemented in standard sequential computers; however, it may result in neural network performance improvements in environments in which some features change more than others. Consider, for example, the self-care niche described in this dissertation. Here the behavior of the switch does not change but the values of the lights do. The synapses connecting switch-sensitive input units to the motors could evolve small values for genes related to standard deviation and learning rate so that evolved instincts related to the switch would not be forgotten during an individual's lifetime, while synapses connecting light-sensitive input units to the motors could evolve larger values for their matching genes to allow for quick learning of the current values of the lights.

8.5 Lamarckian Inheritance

Another area that can be explored is Lamarckian inheritance. *Lamarckian inheritance* is the inheritance of acquired characteristics (Kronfeldner, 2006). It will be interesting to find out if nurturing promotes the evolution of learning using Lamarckian inheritance as well. For example, an individual neural controller could be allowed to retain its learned synaptic weight means and pass them on to its offspring, which could be considered a form of Lamarckian inheritance.

Using this setup, another interesting question that can be investigated is how Lamarckian inheritance interacts with environmental change and the evolution of learning. Consider, for example, two contrasting environments. In the first, the environment changes toward the end of an individual's lifetime but not between generations. In the second, the environment likewise changes toward the end of an individual's lifetime, but then changes back to its original state between generations. In the first environment, Lamarckian inheritance might promote the evolution of learning because an individual's learning would benefit both itself (since its learning helps it to adapt to its environment after the change) and its child (since what it learns is passed on to its child, which begins its own lifetime in a similar environment). However, in the second environment, Lamarckian inheritance might interfere with the evolution of learning because, while an individual's learning would benefit the individual itself (again, it would help to adapt the individual to its environment after the change), it might hamper its child, which inherits behavior that is suboptimal for most of its lifetime. Here, an individual might be better off acting instinctively throughout its lifetime so long as those instincts served it well for most of its lifetime and likewise helped its offspring for most of its lifetime.

This could also be seen as a model of an alternative form of nurturing, as parental knowledge sharing with offspring (for example, through instruction or demonstration) is not so different from Lamarckian inheritance of learned knowledge except, of course, that cultural knowledge isn't passed through genes.

8.6 Risk Analysis

By introducing reward variability to the environment, risk analysis can be performed as in Niv et al. (2002). Consider a single switch, multi-light environment as used in this dissertation. In this environment, risk aversion can be analyzed using an experimental setup such as the following:

1. Very risky resource: A high rewarding light source with 1.0 reward 10% of the

time and 0 the other 90% of the time. Thus its mean is 0.1.

2. Risky resource: A medium rewarding light source with 0.5 reward 20% of the time and 0 the other 80% of the time. Thus its mean is also 0.1.

3. Non-risky resource: A low rewarding light source with 0.1 reward all the time consistently. Thus its mean is also 0.1.

Because risk aversion is a side effect of reinforcement learning (Niv et al., 2002), individuals should learn to visit the non-risky resource most often. One question that could be asked is whether reinforcement learning would be as readily evolved in an environment characterized by differences in reward risk (variability around a common mean reward) rather than by differences in mean reward. Another question that could be asked is whether nurturing promotes the evolution of learning in environments characterized by differences in reward risk. An element that can be added to each of these questions is whether reinforcement learning is evolved as readily when riskier options have higher mean rewards than less risky options.

Bibliography

Agresti, A. (1992). A survey of exact inference for contingency tables. *Statistical Science*, *7*, 131–153.

Bäck, T., Fogel, D. B., & Michalewicz, Z. (1997). *Handbook of Evolutionary Computation*. IOP and Oxford University Press, Bristol, UK.

Bekey, G., & Goldberg, K. Y. (1993). *Neural Networks in Robotics*. NY, USA: Springer Science & Business Media.

Berthouze, L., & Metta, G. (2005). Editorial: Epigenetic robotics: Modelling cognitive development in robotic systems. *Cognitive Systems Research*, *6*, 189–192.

Caro, T. (1980). Predatory behaviour in domestic cat mothers. *Behaviour*, *74*, 128–147.

Caro, T. M., & Hauser, M. D. (1992). Is there teaching in nonhuman animals? *Quarterly Review of Biology*, *67*, 151–174.

Chalmers, D. J. (1990). The evolution of learning: An experiment in genetic connectionism. In *Proceedings of the 1990 Connectionist Models Summer School* (pp. 81–90). Morgan Kaufmann.

Char, K. (1997). Evolution of structure and learning: A GP approach. In J. Mira, R. Moreno-Daz, & J. Cabestany (Eds.), *Biological and Artificial Computation: From Neuroscience to Technology* (pp. 510–517). Springer Berlin Heidelberg volume 1240 of *Lecture Notes in Computer Science*.

Connell, J., & Mahadevan, S. (1993). Introduction to robot learning. In J. Connell, & S. Mahadevan (Eds.), *Robot Learning* (pp. 1–17). Springer US volume 233 of *The Springer International Series in Engineering and Computer Science*.

Dasdan, A., & Oflazer, K. (1993). Genetic synthesis of unsupervised learning algorithms. In *Proceedings of the 2nd Turkish symposium on artificial intelligence and ANNs* (pp. 213–220). Department of Computer Engineering and Information Science, Bilkent University, Ankara.

Dawkins, R. (2004). Extended phenotype–but not too extended. a reply to Laland, Turner and Jablonka. *Biology and Philosophy*, *19*, 377–396.

Day, R. L., Laland, K. N., & Odling-Smee, F. J. (2003). Rethinking adaptation: the niche-construction perspective. *Perspectives in Biology and Medicine*, *46*, 80–95.

Demiris, J., & Hayes, G. M. (2002). Imitation in animals and artifacts. chapter Imitation As a Dual-route Process Featuring Predictive and Learning Components: A Biologically Plausible Computational Model. (pp. 327–361). Cambridge, MA, USA: MIT Press.

Di Paolo, E. A. (2003). Evolving spike-timing-dependent plasticity for single-trial learning in robots. *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, *361*, 2299–2319.

Eskridge, B., & Hougen, D. (2012). Nurturing promotes the evolution of learning in uncertain environments. In *2012 IEEE International Conference on Development and Learning and Epigenetic Robotics* (pp. 6 pages, unnumbered).

Ewer, R. F. (1969). The instinct to teach. *Nature*, *222*, 698.

Fellous, J.-M., & Linster, C. (1998). Computational models of neuromodulation. *Neural Computation*, *10*, 771–805.

Floreano, D., & Urzelai, J. (2000). Evolutionary robots with on-line self-organization and behavioral fitness. *Neural Networks*, *13*, 431–443.

Fontanari, J., & Meir, R. (2009). Evolving a learning algorithm for the binary perceptron. *Network Computation in Neural Systems*, *2*, 353–359.

Foster, J. J. (2006). Chi-square test. In V. Jupp (Ed.), *The SAGE Dictionary of Social Research Methods* (pp. 27–29). SAGE Publications, CA, USA.

Gullapalli, V. (1990). A stochastic reinforcement learning algorithm for learning real-valued functions. *Neural Networks*, *3*, 671–692.

Gullapalli, V. (1992). *Reinforcement learning and its application to control*. Ph.D. thesis University of Massachusetts.

Ha, R., & Ha, J. (2011). *Integrative Statistics for the Social and Behavioral Sciences*. SAGE Publications, CA, USA.

Harvey, I. (1997). Cognition is not computation; evolution is not optimisation. In *International Conference on Artificial Neural Networks* (pp. 685–690). Berlin, Springer, Verlag.

Hougen, D., Gini, M., & Slagle, J. (2000). An integrated connectionist approach to reinforcement learning for robotic control: The advantages of indexed partitioning. *Proceedings of the Seventeenth International Conference on Machine Learning*, (pp. 383–390).

I. W. Sandberg, C. L. F. J. C. P. S. K., James T. Lo, & Haykin, S. S. (2001). *Nonlinear dynamical systems: Feedforward neural network perspectives* volume 21 of *Adaptive and Learning Systems for Signal Processing, Communications and Control*. John Wiley & Sons.

Jain, A. K., Mao, J., & Mohiuddin, K. (1996). Artificial neural networks: A tutorial. *Computer*, *29*, 31–44.

Jin, Y., & Meng, Y. (2011). Morphogenetic robotics: An emerging new field in developmental robotics. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, *41*, 145–160.

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, *4*, 237–285.

Kober, J. A. P. J., Jens; Bagnell (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, *32*, 1238–1274.

Konidaris, G., & Barreto, A. S. (2009). Skill discovery in continuous reinforcement learning domains using skill chaining. In *Advances in Neural Information Processing Systems* (pp. 1015–1023).

Kronfeldner, M. E. (2006). Is cultural evolution Lamarckian? *Biology & Philosophy*, *22*, 493–512.

Laland, K. N. (2004). Extending the extended phenotype. *Biology and Philosophy*, *19*, 313–325.

Laland, K. N., Odling-Smee, F. J., & Feldman, M. W. (1996). The evolutionary consequences of niche construction: A theoretical investigation using two-locus theory. *Journal of Evolutionary Biology*, *9*, 293–316.

Laland, K. N., Odling-Smee, F. J., & Feldman, M. W. (1999). Evolutionary consequences of niche construction and their implications for ecology. *Proceedings of the National Academy of Sciences*, *96*, 10242–10247.

Laland, K. N., Odling-Smee, J., & Feldman, M. W. (2001). Cultural niche construction and human evolution. *Journal of Evolutionary Biology*, *14*, 22–33.

Laud, A. D. (2004). *Theory and application of reward shaping in reinforcement learning*. Ph.D. thesis University of Illinois at Urbana-Champaign.

Leonce, A., Hoke, B., & Hougen, D. (2012). Evolution of robot-to-robot nurturing and nurturability. In *2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL)* (pp. 7 pages, unnumbered).

Lin, L.-J. (1993). Hierarchical learning of robot skills by reinforcement. In *IEEE International Conference on Neural Networks* (pp. 181–186 vol.1).

Lungarella, M., & Metta, G. (2003). Beyond gazing, pointing, and reaching: A survey of developmental robotics. In *Proceedings of the Third International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems* (pp. 81–89). Lund University Cognitive Studies volume 101.

Magnenat, S., Waibel, M., & Beyeler, A. (2007). Enki, a fast 2d robot simulator. http://home.gna.org/enki/.

Miller, B. L., & Goldberg, D. E. (1995). Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, *9*, 193–212.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. MIT Press Cambridge Massachusetts, USA.

Mousseau, T. A., & Fox, C. W. (1998). *Maternal effects as adaptations*. Oxford University Press, NY, USA.

Nicolescu, M. N., & Mataric, M. J. (2001). Experience-based representation construction: Learning from human and robot teachers. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 740–745).

Niv, Y., Joel, D., Meilijson, I., & Ruppin, E. (2002). Evolution of reinforcement learning in uncertain environments: A simple explanation for complex foraging behaviors. *Adaptive Behaviour*, *10*, 5–24.

Nolfi, S., & Parisi, D. (1996). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behaviour*, *5*, 75–98.

Norouzzadeh, S. (2010). *Shaping Methods to Accelerate Reinforcement Learning: From Easy to Challenging Tasks*. Ph.D. thesis Delft University of Technology.

Odling-Smee, F. J., Laland, K. N., & Feldman, M. W. (2003). *Niche construction: The neglected process in evolution*. 37. Princeton University Press, Princeton, NJ.

Orey, Michael (2001). Emerging perspectives on learning, teaching, and technology. http://projects.coe.uga.edu/epltt. [Online; accessed 18-July-2015].

Peters, J., Vijayakumar, S., & Schaal, S. (2003). Reinforcement learning for humanoid robotics. In *Proceedings of the third IEEE-RAS international conference on humanoid robots* (pp. 1–20).

Poli, R., Langdon, W., McPhee, N., & Koza, J. (2008). *A Field Guide to Genetic Programming*. Lulu Enterprises Ltd, London, UK.

Poole, D. L., & Mackworth, A. K. (2010). *Artificial Intelligence: foundations of computational agents*. Cambridge University Press.

Powell, W. B. (2012). *AI, OR and control theory: A rosetta stone for stochastic optimization*. Technical Report Princeton University.

Rajan, S., & Nguyen, D. (2004). Design optimization of discrete structural systems using MPI-enabled genetic algorithm. *Structural and Multidisciplinary Optimization*, *28*, 340–348.

Reece, J. B., & Campbell, N., A. (2011). *Biology*. (9th ed.). Boston, Mass.; London: Pearson.

Riedmiller, M., Gabel, T., Hafner, R., & Lange, S. (2009). Reinforcement learning for robot soccer. *Autonomous Robots*, *27*, 55–73.

Saksida, L. M., Raymond, S. M., & Touretzky, D. S. (1997). Shaping robot behavior using principles from instrumental conditioning. *Robotics and Autonomous Systems*, *22*, 231–249.

Scott-Phillips, T. C., Laland, K. N., Shuker, D. M., Dickins, T. E., & West, S. A. (2014). The niche construction perspective: A critical appraisal. *Evolution*, *68*, 1231–1243.

Sivanandam, S., & Deepa, S. (2007). *Introduction to genetic algorithms*. Springer Science & Business Media.

Soltoggio, A., Dürr, P., Mattiussi, C., & Floreano, D. (2007). Evolving neuromodulatory topologies for reinforcement learning-like problems. In *IEEE Congress on Evolutionary Computation* (pp. 2471–2478).

Sutton, R. S. (1984). *Temporal Credit Assignment In Reinforcement Learning*. Ph.D. thesis University of Massachusetts.

Sutton, R. S., Maei, H. R., Precup, D., Bhatnagar, S., Silver, D., Szepesvári, C., & Wiewiora, E. (2009). Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th Annual International Conference on Machine Learning* (pp. 993–1000). ACM.

Vermorel, J., & Mohri, M. (2005). Multi-armed bandit algorithms and empirical evaluation. In *Machine Learning: European Conference on Machine Learning 2005* (pp. 437–448). Springer, Berlin, Heidelberg, volume 3720 of *Lecture Notes in Computer Science*.

Wikipedia (2015). Normal Distribution Emperical Rule. https://en.wikipedia.org/wiki/File:Empirical_Rule.PNG.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, *8*, 229–256.

Woehrer, M., Hougen, D. F., Schlupp, I., & Eskridge, B. E. (2012). Robot-to-robot nurturing: A call to the research community. In *International Conference on Development and Learning and Epigenetic Robotics* (pp. 2 pages, unnumbered).

Wolf, J. B., Brodie, E. D., & Wade, M. J. (2000). *Epistasis and the evolutionary process*. Oxford University Press, NY, USA.

Wolf, J. B., Brodie III, E. D., Cheverud, J. M., Moore, A. J., & Wade, M. J. (1998). Evolutionary consequences of indirect genetic effects. *Trends in Ecology & Evolution*, *13*, 64–69.

Yao, X. (1991). The evolution of connectionist networks. In T. Dartnall (Ed.), *Preprints of the Symposium on AI, Reasoning, and Creativity* (pp. 49–52). Griffith University Queensland, Brisbane, Australia.

Yao, X. (1993a). Evolutionary artificial neural networks. *International Journal of Neural Systems*, *04*, 203–222.

Yao, X. (1993b). A review of evolutionary artificial neural networks. *International Journal of Intelligent Systems*, *8*, 539–567.

Yao, X. (1994). The evolution of connectionist networks. In T. Dartnall (Ed.), *Artificial Intelligence and Creativity* number 17 in Studies in Cognitive Systems (pp. 233–243). Springer, Netherlands.

Yao, X. (1995). Evolutionary artificial neural networks. In A. Kent, & J. Williams (Eds.), *Encyclopedia of Computer Science and Technology* (pp. 137–170). Marcel Dekker Inc., volume 33.

Yao, X. (1999). Evolving artificial neural networks. *Proceedings of the IEEE*, *87*, 1423–1447.

Ziemke, T., Bergfeldt, N., Buason, G., Susi, T., & Svensson, H. (2004). Evolving cognitive scaffolding and environment adaptation: a new research direction for evolutionary robotics. *Connection Science*, *16*, 339–350.

Zlatev, J., & Balkenius, C. (2001). Introduction: Why epigenetic robotics? In *First International Workshop on Epigenetic Robotics* (pp. 1–4). Sweden: Lund University Cognitive Studies, volume 85.

Appendices

Appendix A

Performance Continuum Details

This appendix presents detailed numbers from the performance continuum results in the form of tables. Each table is related to the corresponding sections in the results chapter. For the sake of easy mapping, the hypothesis are stated before the table to which it relates.

A.1 Evolution of Learning

2.1 **PC-OA** Nurturing promotes the evolution of higher performing behaviors in the absence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.394	Non-Substantial	46.35
0.665	Good	78.24	0.512	Moderate	60.24
0.623	Good	73.29	0.509	Moderate	59.88
0.563	Moderate	66.24	0.442	Non-Substantial	52.00
0.746	Good	87.76	0.553	Moderate	65.06
0.791	Good	93.06	0.701	Good	82.47
0.641	Good	75.41	0.645	Good	75.88

0.643	Good	75.65	0.432	Non-Substantial	50.82
0.635	Good	74.71	0.527	Moderate	62.00
0.774	Good	91.06	0.593	Moderate	69.76
0.734	Good	86.35	0.468	Non-Substantial	55.06
0.759	Good	89.29	0.538	Good	63.29
0.835	Good	98.24	0.423	Non-Substantial	49.76
0.723	Good	85.06	0.384	Non-Substantial	45.18
0.648	Good	76.24	0.516	Moderate	60.71
0.698	Good	82.12	0.605	Moderate	71.18
0.713	Good	83.88	0.516	Good	60.71
0.733	Good	86.24	0.474	Moderate	55.76
0.754	Good	88.71	0.469	Non-Substantial	55.18
0.783	Good	92.12	0.542	Moderate	63.76
0.742	Good	87.29	0.452	Non-Substantial	53.18
0.757	Good	89.06	0.541	Moderate	63.65
0.729	Good	85.76	0.488	Moderate	57.41
0.682	Good	80.24	0.582	Good	68.47
0.633	Good	74.47	0.426	Non-Substantial	50.12
0.731	Good	86.00	0.635	Good	74.71

0.77	Good	90.59	0.589	Good	69.29
0.573	Good	67.41	0.542	Good	63.76
0.465	Non-Substantial	54.71	0.447	Non-Substantial	52.59
0.681	Good	80.12	0.442	Non-Substantial	52.00

Table A.1.1: Performance Continuum Data for all Repetitions—Evolution of Learning. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial. The data in above table is paired.

The summary statistics for the above table are shown in main results Table 5.8.

2.1.1 **PC-GA** Nurturing promotes the evolution of higher performing behaviors within good learners in the absence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.665	Good	78.24	0.701	Good	82.47
0.623	Good	73.29	0.645	Good	75.88

0.746	Good	87.76	0.538	Good	63.29
0.791	Good	93.06	0.516	Good	60.71
0.641	Good	75.41	0.582	Good	68.47
0.643	Good	75.65	0.635	Good	74.71
0.635	Good	74.71	0.589	Good	69.29
0.774	Good	91.06	0.542	Good	63.76
0.734	Good	86.35			
0.759	Good	89.29			
0.835	Good	98.24			
0.723	Good	85.06			
0.648	Good	76.24			
0.698	Good	82.12			
0.713	Good	83.88			
0.733	Good	86.24			
0.754	Good	88.71			
0.783	Good	92.12			
0.742	Good	87.29			
0.757	Good	89.06			
0.729	Good	85.76			

0.682	Good	80.24			
0.633	Good	74.47			
0.731	Good	86.00			
0.77	Good	90.59			
0.573	Good	67.41			
0.681	Good	80.12			

Table A.1.2: Performance Continuum Data for Good Learners—Evolution of Learning. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.9.

2.1.2 **PC-SA** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the absence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.512	Moderate	60.24

0.665	Good	78.24	0.509	Moderate	59.88
0.623	Good	73.29	0.553	Moderate	65.06
0.563	Moderate	66.24	0.701	Good	82.47
0.746	Good	87.76	0.645	Good	75.88
0.791	Good	93.06	0.527	Moderate	62.00
0.641	Good	75.41	0.593	Moderate	69.76
0.643	Good	75.65	0.538	Good	63.29
0.635	Good	74.71	0.516	Moderate	60.71
0.774	Good	91.06	0.605	Moderate	71.18
0.734	Good	86.35	0.516	Good	60.71
0.759	Good	89.29	0.474	Moderate	55.76
0.835	Good	98.24	0.542	Moderate	63.76
0.723	Good	85.06	0.541	Moderate	63.65
0.648	Good	76.24	0.488	Moderate	57.41
0.698	Good	82.12	0.582	Good	68.47
0.713	Good	83.88	0.635	Good	74.71
0.733	Good	86.24	0.589	Good	69.29
0.754	Good	88.71	0.542	Good	63.76
0.783	Good	92.12			

0.742	Good	87.29			
0.757	Good	89.06			
0.729	Good	85.76			
0.682	Good	80.24			
0.633	Good	74.47			
0.731	Good	86.00			
0.77	Good	90.59			
0.573	Good	67.41			
0.681	Good	80.12			

Table A.1.3: Performance Continuum Data for Substantial Learners—Evolution of Learning. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.10.

2.1.3 **PC-MA** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the absence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.512	Moderate	60.24
0.563	Moderate	66.24	0.509	Moderate	59.88
			0.553	Moderate	65.06
			0.527	Moderate	62.00
			0.593	Moderate	69.76
			0.516	Moderate	60.71
			0.605	Moderate	71.18
			0.474	Moderate	55.76
			0.542	Moderate	63.76
			0.541	Moderate	63.65
			0.488	Moderate	57.41

Table A.1.4: Performance Continuum Data for Moderate Learners—Evolution of Learning. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.11.

2.1.4 **PC-NA** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the absence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.465	Non-Substantial	54.71	0.394	Non-Substantial	46.35
			0.442	Non-Substantial	52.00
			0.432	Non-Substantial	50.82
			0.468	Non-Substantial	55.06
			0.423	Non-Substantial	49.76
			0.384	Non-Substantial	45.18
			0.469	Non-Substantial	55.18
			0.452	Non-Substantial	53.18
			0.426	Non-Substantial	50.12
			0.447	Non-Substantial	52.59
			0.442	Non-Substantial	52.00

Table A.1.5: Performance Continuum Data for Non-Substantial Learners—Evolution of Learning. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.12.

A.2 Evolution of Learning and Instincts

2.2 **PC-OP** Nurturing promotes the evolution of higher performing behaviors in the presence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.732	Good	86.12	0.615	Moderate	72.35
0.652	Good	76.71	0.521	Moderate	61.29
0.73	Good	85.88	0.497	Moderate	58.47
0.717	Good	84.35	0.562	Moderate	66.12
0.789	Good	92.82	0.55	Moderate	64.71
0.77	Good	90.59	0.454	Non-Substantial	53.41

0.814	Good	95.76	0.667	Good	78.47
0.774	Good	91.06	0.589	Good	69.29
0.752	Good	88.47	0.47	Moderate	55.29
0.585	Moderate	68.82	0.535	Moderate	62.94
0.686	Good	80.71	0.466	Non-Substantial	54.82
0.769	Good	90.47	0.648	Good	76.24
0.814	Good	95.76	0.574	Moderate	67.53
0.661	Good	77.76	0.598	Moderate	70.35
0.638	Good	75.06	0.641	Good	75.41
0.693	Good	81.53	0.466	Non-Substantial	54.82
0.809	Good	95.18	0.493	Moderate	58.00
0.658	Good	77.41	0.449	Non-Substantial	52.82
0.772	Good	90.82	0.604	Moderate	71.06
0.558	Moderate	65.65	0.455	Non-Substantial	53.53
0.785	Good	92.35	0.556	Moderate	65.41
0.606	Moderate	71.29	0.474	Moderate	55.76
0.642	Good	75.53	0.493	Moderate	58.00
0.683	Good	80.35	0.733	Good	86.24
0.602	Moderate	70.82	0.544	Moderate	64.00

0.785	Good	92.35	0.684	Good	80.47
0.775	Good	91.18	0.691	Good	81.29
0.703	Good	82.71	0.569	Moderate	66.94
0.755	Good	88.82	0.469	Non-Substantial	55.18
0.647	Good	76.12	0.552	Moderate	64.94

Table A.2.1: Performance Continuum Data for all Repetitions—Evolution of Learning and Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial. The data in above table is paired.

The summary statistics for the above table are shown in main results, Table 5.16.

2.2.1 **PC-GP** Nurturing promotes the evolution of higher performing behaviors within good learners in the presence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.732	Good	86.12	0.667	Good	78.47

0.652	Good	76.71	0.589	Good	69.29
0.73	Good	85.88	0.648	Good	76.24
0.717	Good	84.35	0.641	Good	75.41
0.789	Good	92.82	0.733	Good	86.24
0.77	Good	90.59	0.684	Good	80.47
0.814	Good	95.76	0.691	Good	81.29
0.774	Good	91.06			
0.752	Good	88.47			
0.686	Good	80.71			
0.769	Good	90.47			
0.814	Good	95.76			
0.661	Good	77.76			
0.638	Good	75.06			
0.693	Good	81.53			
0.809	Good	95.18			
0.658	Good	77.41			
0.772	Good	90.82			
0.785	Good	92.35			
0.642	Good	75.53			

0.683	Good	80.35			
0.785	Good	92.35			
0.775	Good	91.18			
0.703	Good	82.71			
0.755	Good	88.82			
0.647	Good	76.12			

Table A.2.2: Performance Continuum Data for Good Learners—Evolution of Learning and Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.17.

2.2.2 **PC-SP** Nurturing promotes the evolution of higher performing behaviors within substantial learners in the presence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.732	Good	86.12	0.615	Moderate	72.35

0.652	Good	76.71	0.521	Moderate	61.29
0.73	Good	85.88	0.497	Moderate	58.47
0.717	Good	84.35	0.562	Moderate	66.12
0.789	Good	92.82	0.55	Moderate	64.71
0.77	Good	90.59	0.667	Good	78.47
0.814	Good	95.76	0.589	Good	69.29
0.774	Good	91.06	0.47	Moderate	55.29
0.752	Good	88.47	0.535	Moderate	62.94
0.585	Moderate	68.82	0.648	Good	76.24
0.686	Good	80.71	0.574	Moderate	67.53
0.769	Good	90.47	0.598	Moderate	70.35
0.814	Good	95.76	0.641	Good	75.41
0.661	Good	77.76	0.493	Moderate	58.00
0.638	Good	75.06	0.604	Moderate	71.06
0.693	Good	81.53	0.556	Moderate	65.41
0.809	Good	95.18	0.474	Moderate	55.76
0.658	Good	77.41	0.493	Moderate	58.00
0.772	Good	90.82	0.733	Good	86.24
0.558	Moderate	65.65	0.544	Moderate	64.00

0.785	Good	92.35	0.684	Good	80.47
0.606	Moderate	71.29	0.691	Good	81.29
0.642	Good	75.53	0.569	Moderate	66.94
0.683	Good	80.35	0.552	Moderate	64.94
0.602	Moderate	70.82			
0.785	Good	92.35			
0.775	Good	91.18			
0.703	Good	82.71			
0.755	Good	88.82			
0.647	Good	76.12			

Table A.2.3: Performance Continuum Data for Substantial Learners—Evolution of Learning and Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.18.

2.2.3 **PC-MP** Nurturing promotes the evolution of higher performing behaviors within moderate learners in the presence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.585	Moderate	68.82	0.615	Moderate	72.35
0.558	Moderate	65.65	0.521	Moderate	61.29
0.606	Moderate	71.29	0.497	Moderate	58.47
0.602	Moderate	70.82	0.562	Moderate	66.12
			0.55	Moderate	64.71
			0.47	Moderate	55.29
			0.535	Moderate	62.94
			0.574	Moderate	67.53
			0.598	Moderate	70.35
			0.493	Moderate	58.00
			0.604	Moderate	71.06
			0.556	Moderate	65.41
			0.474	Moderate	55.76
			0.493	Moderate	58.00
			0.544	Moderate	64.00
			0.569	Moderate	66.94
			0.552	Moderate	64.94

Table A.2.4: Performance Continuum Data for Moderate Learners—Evolution of Learning and Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.19.

2.2.4 **PC-NP** Nurturing promotes the evolution of higher performing behaviors within non-substantial learners in the presence of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
			0.454	Non-Substantial	53.41
			0.466	Non-Substantial	54.82
			0.466	Non-Substantial	54.82
			0.449	Non-Substantial	52.82
			0.455	Non-Substantial	53.53
			0.469	Non-Substantial	55.18

Table A.2.5: Performance Continuum Data for Non-Substantial Learners—Evolution of Learning and Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.20.

A.3 Evolution of Learning Regardless of Instincts

2.3 **PC-OE** Nurturing promotes the evolution of higher performing behaviors regardless of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.394	Non-Substantial	46.35
0.665	Good	78.24	0.512	Moderate	60.24
0.623	Good	73.29	0.509	Moderate	59.88
0.563	Moderate	66.24	0.442	Non-Substantial	52.00
0.746	Good	87.76	0.553	Moderate	65.06
0.791	Good	93.06	0.701	Good	82.47
0.641	Good	75.41	0.645	Good	75.88
0.643	Good	75.65	0.432	Non-Substantial	50.82
0.635	Good	74.71	0.527	Moderate	62.00
0.774	Good	91.06	0.593	Moderate	69.76
0.734	Good	86.35	0.468	Non-Substantial	55.06
0.759	Good	89.29	0.538	Good	63.29
0.835	Good	98.24	0.423	Non-Substantial	49.76
0.723	Good	85.06	0.384	Non-Substantial	45.18
0.648	Good	76.24	0.516	Moderate	60.71
0.698	Good	82.12	0.605	Moderate	71.18
0.713	Good	83.88	0.516	Good	60.71

0.733	Good	86.24	0.474	Moderate	55.76
0.754	Good	88.71	0.469	Non-Substantial	55.18
0.783	Good	92.12	0.542	Moderate	63.76
0.742	Good	87.29	0.452	Non-Substantial	53.18
0.757	Good	89.06	0.541	Moderate	63.65
0.729	Good	85.76	0.488	Moderate	57.41
0.682	Good	80.24	0.582	Good	68.47
0.633	Good	74.47	0.426	Non-Substantial	50.12
0.731	Good	86.00	0.635	Good	74.71
0.77	Good	90.59	0.589	Good	69.29
0.573	Good	67.41	0.542	Good	63.76
0.465	Non-Substantial	54.71	0.447	Non-Substantial	52.59
0.681	Good	80.12	0.442	Non-Substantial	52.00
0.732	Good	86.12	0.615	Moderate	72.35
0.652	Good	76.71	0.521	Moderate	61.29
0.73	Good	85.88	0.497	Moderate	58.47
0.717	Good	84.35	0.562	Moderate	66.12
0.789	Good	92.82	0.55	Moderate	64.71
0.77	Good	90.59	0.454	Non-Substantial	53.41

0.814	Good	95.76	0.667	Good	78.47
0.774	Good	91.06	0.589	Good	69.29
0.752	Good	88.47	0.47	Moderate	55.29
0.585	Moderate	68.82	0.535	Moderate	62.94
0.686	Good	80.71	0.466	Non-Substantial	54.82
0.769	Good	90.47	0.648	Good	76.24
0.814	Good	95.76	0.574	Moderate	67.53
0.661	Good	77.76	0.598	Moderate	70.35
0.638	Good	75.06	0.641	Good	75.41
0.693	Good	81.53	0.466	Non-Substantial	54.82
0.809	Good	95.18	0.493	Moderate	58.00
0.658	Good	77.41	0.449	Non-Substantial	52.82
0.772	Good	90.82	0.604	Moderate	71.06
0.558	Moderate	65.65	0.455	Non-Substantial	53.53
0.785	Good	92.35	0.556	Moderate	65.41
0.606	Moderate	71.29	0.474	Moderate	55.76
0.642	Good	75.53	0.493	Moderate	58.00
0.683	Good	80.35	0.733	Good	86.24
0.602	Moderate	70.82	0.544	Moderate	64.00

0.785	Good	92.35	0.684	Good	80.47
0.775	Good	91.18	0.691	Good	81.29
0.703	Good	82.71	0.569	Moderate	66.94
0.755	Good	88.82	0.469	Non-Substantial	55.18
0.647	Good	76.12	0.552	Moderate	64.94

Table A.3.1: Performance Continuum Data for all Repetitions—Evolution of Learning Regardless of Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial. The data in above table is paired.

The summary statistics for the above table are shown in main results, Table 5.24.

2.3.1 **PC-GE** Nurturing promotes the evolution of higher performing behaviors within good learners regardless of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.665	Good	78.24	0.701	Good	82.47
0.623	Good	73.29	0.645	Good	75.88
0.746	Good	87.76	0.538	Good	63.29
0.791	Good	93.06	0.516	Good	60.71
0.641	Good	75.41	0.582	Good	68.47
0.643	Good	75.65	0.635	Good	74.71
0.635	Good	74.71	0.589	Good	69.29
0.774	Good	91.06	0.542	Good	63.76
0.734	Good	86.35	0.667	Good	78.47
0.759	Good	89.29	0.589	Good	69.29
0.835	Good	98.24	0.648	Good	76.24
0.723	Good	85.06	0.641	Good	75.41
0.648	Good	76.24	0.733	Good	86.24
0.698	Good	82.12	0.684	Good	80.47
0.713	Good	83.88	0.691	Good	81.29
0.733	Good	86.24			

0.754	Good	88.71			
0.783	Good	92.12			
0.742	Good	87.29			
0.757	Good	89.06			
0.729	Good	85.76			
0.682	Good	80.24			
0.633	Good	74.47			
0.731	Good	86.00			
0.77	Good	90.59			
0.573	Good	67.41			
0.681	Good	80.12			
0.732	Good	86.12			
0.652	Good	76.71			
0.73	Good	85.88			
0.717	Good	84.35			
0.789	Good	92.82			
0.77	Good	90.59			
0.814	Good	95.76			
0.774	Good	91.06			

0.752	Good	88.47			
0.686	Good	80.71			
0.769	Good	90.47			
0.814	Good	95.76			
0.661	Good	77.76			
0.638	Good	75.06			
0.693	Good	81.53			
0.809	Good	95.18			
0.658	Good	77.41			
0.772	Good	90.82			
0.785	Good	92.35			
0.642	Good	75.53			
0.683	Good	80.35			
0.785	Good	92.35			
0.775	Good	91.18			
0.703	Good	82.71			
0.755	Good	88.82			
0.647	Good	76.12			

Table A.3.2: Performance Continuum Data for Good Learners—Evolution of Learning Regardless of Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.25.

2.3.2 **PC-SE** Nurturing promotes the evolution of higher performing behaviors within substantial learners regardless of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.512	Moderate	60.24
0.665	Good	78.24	0.509	Moderate	59.88
0.623	Good	73.29	0.553	Moderate	65.06
0.563	Moderate	66.24	0.701	Good	82.47
0.746	Good	87.76	0.645	Good	75.88
0.791	Good	93.06	0.527	Moderate	62.00

0.641	Good	75.41	0.593	Moderate	69.76
0.643	Good	75.65	0.538	Good	63.29
0.635	Good	74.71	0.516	Moderate	60.71
0.774	Good	91.06	0.605	Moderate	71.18
0.734	Good	86.35	0.516	Good	60.71
0.759	Good	89.29	0.474	Moderate	55.76
0.835	Good	98.24	0.542	Moderate	63.76
0.723	Good	85.06	0.541	Moderate	63.65
0.648	Good	76.24	0.488	Moderate	57.41
0.698	Good	82.12	0.582	Good	68.47
0.713	Good	83.88	0.635	Good	74.71
0.733	Good	86.24	0.589	Good	69.29
0.754	Good	88.71	0.542	Good	63.76
0.783	Good	92.12	0.615	Moderate	72.35
0.742	Good	87.29	0.521	Moderate	61.29
0.757	Good	89.06	0.497	Moderate	58.47
0.729	Good	85.76	0.562	Moderate	66.12
0.682	Good	80.24	0.55	Moderate	64.71
0.633	Good	74.47	0.667	Good	78.47

0.731	Good	86.00	0.589	Good	69.29
0.77	Good	90.59	0.47	Moderate	55.29
0.573	Good	67.41	0.535	Moderate	62.94
0.681	Good	80.12	0.648	Good	76.24
0.732	Good	86.12	0.574	Moderate	67.53
0.652	Good	76.71	0.598	Moderate	70.35
0.73	Good	85.88	0.641	Good	75.41
0.717	Good	84.35	0.493	Moderate	58.00
0.789	Good	92.82	0.604	Moderate	71.06
0.77	Good	90.59	0.556	Moderate	65.41
0.814	Good	95.76	0.474	Moderate	55.76
0.774	Good	91.06	0.493	Moderate	58.00
0.752	Good	88.47	0.733	Good	86.24
0.585	Moderate	68.82	0.544	Moderate	64.00
0.686	Good	80.71	0.684	Good	80.47
0.769	Good	90.47	0.691	Good	81.29
0.814	Good	95.76	0.569	Moderate	66.94
0.661	Good	77.76	0.552	Moderate	64.94
0.638	Good	75.06			

0.693	Good	81.53			
0.809	Good	95.18			
0.658	Good	77.41			
0.772	Good	90.82			
0.558	Moderate	65.65			
0.785	Good	92.35			
0.606	Moderate	71.29			
0.642	Good	75.53			
0.683	Good	80.35			
0.602	Moderate	70.82			
0.785	Good	92.35			
0.775	Good	91.18			
0.703	Good	82.71			
0.755	Good	88.82			
0.647	Good	76.12			

Table A.3.3: Performance Continuum Data for Substantial Learners—Evolution of Learning Regardless of Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.26.

2.3.3 **PC-ME** Nurturing promotes the evolution of higher performing behaviors within moderate learners regardless of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.581	Moderate	68.35	0.512	Moderate	60.24
0.563	Moderate	66.24	0.509	Moderate	59.88
0.585	Moderate	68.82	0.553	Moderate	65.06
0.558	Moderate	65.65	0.527	Moderate	62.00
0.606	Moderate	71.29	0.593	Moderate	69.76
0.602	Moderate	70.82	0.516	Moderate	60.71

			0.605	Moderate	71.18
			0.474	Moderate	55.76
			0.542	Moderate	63.76
			0.541	Moderate	63.65
			0.488	Moderate	57.41
			0.615	Moderate	72.35
			0.521	Moderate	61.29
			0.497	Moderate	58.47
			0.562	Moderate	66.12
			0.55	Moderate	64.71
			0.47	Moderate	55.29
			0.535	Moderate	62.94
			0.574	Moderate	67.53
			0.598	Moderate	70.35
			0.493	Moderate	58.00
			0.604	Moderate	71.06
			0.556	Moderate	65.41
			0.474	Moderate	55.76
			0.493	Moderate	58.00

			0.544	Moderate	64.00
			0.569	Moderate	66.94
			0.552	Moderate	64.94

Table A.3.4: Performance Continuum Data for Moderate Learners—Evolution of Learning Regardless of Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.27.

2.3.4 **PC-NE** Nurturing promotes the evolution of higher performing behaviors within non-Substantial learners regardless of instincts.

Nurturing			Self-Care		
R_{avg}	Learning Category	Success	R_{avg}	Learning Category	Success
0.465	Non-Substantial	54.71	0.394	Non-Substantial	46.35
			0.442	Non-Substantial	52.00
			0.432	Non-Substantial	50.82
			0.468	Non-Substantial	55.06
			0.423	Non-Substantial	49.76
			0.384	Non-Substantial	45.18
			0.469	Non-Substantial	55.18
			0.452	Non-Substantial	53.18
			0.426	Non-Substantial	50.12
			0.447	Non-Substantial	52.59
			0.442	Non-Substantial	52.00
			0.454	Non-Substantial	53.41
			0.466	Non-Substantial	54.82
			0.466	Non-Substantial	54.82
			0.449	Non-Substantial	52.82
			0.455	Non-Substantial	53.53
			0.469	Non-Substantial	55.18

Table A.3.5: Performance Continuum Data for Non-Substantial Learners—Evolution of Learning Regardless of Instincts. R_{avg} means average reward collected during the lifetime of an individual. Learning Category indicates the category a particular individual is classified into using the objective criteria. Success indicates relative success an individual achieves compared to the best possible (near-optimal) value found in the data collected during any single trial.

The summary statistics for the above table are shown in main results, Table 5.28.

Appendix B

Acronyms

AI Artificial Intelligence

ANN Artificial Neural Networks

GA Genetic Algorithm

ML Machine Learning

R2R Robot-to-Robot

REAL Lab Robotics, Evolution, Adaptation, and Learning Laboratory