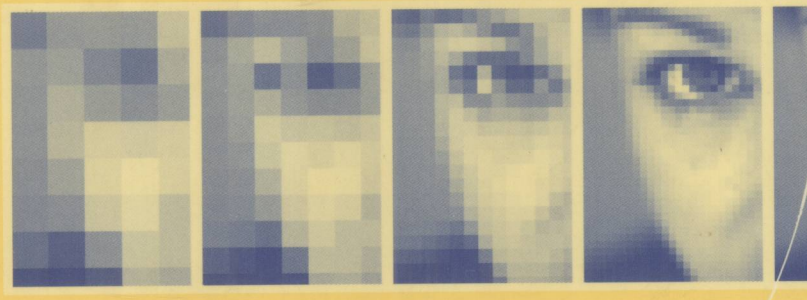
Advances in the Evolutionary Synthesis of Intelligent Agents

edited by Mukesh J. Patel, Vasant Honavar, and Karthik Balakrishnan



Advances in the Evolutionary Synthesis of Intelligent Agents

edited by Mukesh Patel, Vasant Honavar, Karthik Balakrishnan

A Bradford Book
The MIT Press
Cambridge, Massachusetts
London, England

Library of Congress Cataloging-in-Publication Data

Patel, Mukesh.
Advances in the evolutionary synthesis of intelligent agents / Mukesh Patel, Vasant Honavar, Karthik Balakrishnan
p. cm.
"A Bradford Book."
Includes bibliographical references and index.
ISBN 0-262-16201-6 (hc. : alk. paper)
1. Intelligent agents (computer software). 2. Evolutionary programming (Computer science). I. Honavar, Vasant, 1960-. II. Balakrishnan, Karthik, 1970- . III. Title.

QA76.76.I58 P78 2001
006.3—dc21 00-052691

Advances in the Evolutionary Synthesis of Intelligent Agents

To my parents and Leonard Uhr – my mentor and friend.

Vasant Honavar

To the truly wonderful women in my life – my wife, my mother, and my sisters.

Karthik Balakrishnan

Karthik Balakrishnan
Decision Analytics Group
Obongo Inc.
Redwood City, USA
balak_k@yahoo.com

Egbert Boers
Artificial Intelligence Group
National Aerospace Laboratory
Amsterdam, The Netherlands
boers@nlr.nl

Brian Carse
Faculty of Engineering
University of the West of England
Bristol, UK
Brian.Carse@uwe.ac.uk

Frederick Crabbe IV
Computer Science Department
University of California
Los Angeles, USA
cardo@cs.ucla.edu

Melania Degeratu
Program in Applied Mathematical and Computational Sciences
University of Iowa
Iowa City, USA
mdegerat@math.uiowa.edu

Michael Dyer
Computer Science Department
University of California
Los Angeles, USA
dyer@cs.ucla.edu

Dario Floreano
Department of Computer Science
Swiss Federal Institute of Technology
Lausanne, Switzerland
Dario.Floreano@epfl.ch

Terence Fogarty
Department of Computer Studies
Napier University
Edinburgh, UK
tcf@dcs.napier.ac.uk

Frederic Gruau
Laboaratoire d'Informatique Robotique et Microelectronique de Montpellier
Montpellier, France
gruau@lirmm.fr

Vasant Honavar
Department of Computer Science
Iowa State University
Ames, USA
honavar@cs.iastate.edu

Yong Liu
Computer Science Division
Electrotechnical Laboratory
Ibaraki, Japan
yliu@etl.go.jp

Bruce MacLennan
Computer Science Department
University of Tennessee
Knoxville, USA
MacLennan@cs.utk.edu

Thilo Mahnig
RWCP Theoretical Foundation
GMD Laboratory
Sankt Augustin, Germany
thilo.mahnig@gmd.de

Filippo Menczer
Department of Management Sciences
University of Iowa
Iowa City, USA
filippo-menczer@uiowa.edu

J. J. Merelo
Departamento de Electrónica y
Tecnología de los Computadores
Universidad de Granada
Granada, Spain
jmerelo@kal-el.ugr.es

Olivier Michel
LAMI
Swiss Federal Institute of Technology
Lausanne, Switzerland
Olivier.Michel@epfl.ch

Risto Miikkulainen
Department of Computer Sciences
The University of Texas at Austin
Austin, USA
risto@cs.utexas.edu

Francesco Mondada
K-Team SA
Switzerland
mondada@k-team.com

F. Moran
Departamento de Bioquimica y
Biología MolecularI
Universidad Complutense de Madrid
Madrid, Spain
fmoran@solea.quim.ucm.es

David Moriarty
Information Sciences Institute
University of Southern California
Marina del Rey, USA
moriarty@isi.edu

Heinz Muehlenbein
RWCP Theoretical Foundation
GMD Laboratory
Sankt Augustin, Germany
muehlenbein@gmd.de

Stefano Nolfi
Institute of Psychology
C.N.R.
Rome, Italy
nolfi@ip.rm.cnr.it

Domenico Parisi
Institute of Psychology
C.N.R.
Rome, Italy
parisi@ip.rm.cnr.it

Mukesh Patel
Applitech Solution Limited
Ahmedabad, India
mukesh@applitechsolution.com

A. Prieto
Departamento de Electrónica y
Tecnología de los Computadores
Universidad de Granada
Granada, Spain

Ida Sprinkhuizen-Kuyper
Computer Science Department
Universiteit Maastricht
Maastricht, The Netherlands
kuyper@cs.unimaas.nl

William Street
Department of Management Sciences
University of Iowa
Iowa City, USA
nick-street@uiowa.edu

John Sullivan
Faculty of Engineering
University of the West of England
Bristol, UK
John.Sullivan@uwe.ac.uk

Xin Yao
School of Computer Science
The University of Birmingham
Birmingham, UK
x.yao@cs.bham.ac.uk

Preface

The emergence of Artificial Intelligence and Cognitive Sciences concerned with understanding and modeling intelligent systems, both natural and synthetic, is perhaps one of the most significant intellectual developments of the twentieth century.

From the earliest writings of India and Greece, understanding minds and brains has been a fundamental problem in philosophy. The invention of the digital computer in the 1940s made this a central concern of computer scientists. Among the first uses of the digital computer, in the 1950s, was the development of computer programs to model perception, reasoning, learning, and evolution. Ensuing developments in the theory of computation, design and analysis of algorithms and the synthesis of programmable information processing systems, provided a new set of tools with which to study appropriately embodied minds and brains – both natural as well as artificial – through the analysis, design, and evaluation of computers and programs that exhibit aspects of intelligent behavior.

This systems approach to understanding intelligent behavior has been driven largely by the working hypothesis of Artificial Intelligence, which simply states that *thought processes can be modeled by computation or information processing*. The philosophical roots of this hypothesis can be traced at least as far back as the attempts of Helmholtz, Leibnitz, and Boole to explain thought processes in terms of mechanical (or in modern terms, algorithmic or computational) processes. The advent of molecular biology, heralded by the discovery of genetic code, has ushered in a similar information processing approach to the study of living systems. This has resulted in the emergence of computational molecular biology, synthetic biology (or artificial life), and related disciplines, which are driven by the working hypothesis that *life processes can be modeled and/or understood by computation or information processing*.

Systems whose information processing structures are fully programmed to their last detail, despite being viewed as the *ideal* in many areas of computer science and engineering, are extremely difficult to design for all but the simplest and the most routine of applications. The designers of such systems have to be able to anticipate in advance all possible contingencies such a system could ever encounter, and program appropriate procedures to deal with each of them. Dynamic, open, real-world world environments call for *adaptive and learning systems* that are equipped with processes that allow them to modify their behavior, as needed, by changing their information processing structures. Long-term collective survival in changing environments calls for systems that can *evolve*. Biological intelligence has emerged over the millennia and depends

crucially on evolution and learning – or rather, it is hard to conceive of a God so intelligent and yet so stupid, so mundane at detail yet so brilliant at the fantastic, as to – like a watchmaker – tediously craft each living creature to its last detail.

Cognitive and information structures and processes, embodied in living systems, offer us impressive existence proofs of many highly effective designs for adaptive, flexible, and creative biological intelligent agents. They are also a great source of ideas for designing artificially intelligent agents. Against this background, this book explores some of the central problems in artificial intelligence, cognitive science, and synthetic biology, namely the modeling of information structures and processes that create and adapt intelligent agents through evolution and learning.

Work on this book began in 1995 when two of us (Vasant Honavar and Mukesh Patel) independently started efforts to put together edited volumes on evolutionary approaches to the synthesis of intelligent systems. When we became aware of each other's work, it seemed logical to pool the efforts and produce a single edited volume. With encouragement from Harry Stanton of MIT press, we invited a team of authors to contribute chapters covering various aspects of evolutionary synthesis of intelligent systems. The original target date for publication of the book was late 1997. However, the sad demise of Harry Stanton interfered with this plan. Fortunately, Doug Sery, who inherited the project in 1998, was very supportive of the project and determined to see it through.

Karthik Balakrishnan joined the team in 1998. Honavar and Patel are grateful to Balakrishnan for taking over a large portion of the effort of putting together this volume between 1998 and 2000. During this period, all of the contributed chapters went through several rounds of reviews and revisions. The final product is the result of the collective work of the authors of the various chapters and the three editors.

The chapters included in this volume represent recent advances in evolutionary approaches to the synthesis of intelligent software agents and intelligent artifacts for a variety of applications. The chapters may be broadly divided into four categories. The first set of chapters demonstrates the raw power of evolution in determining effective solutions to complex tasks such as walking behaviors for 8-legged robots, evolution of meaningful communication in a population of agents, addressing of multi-objective tradeoffs inherent in the design of accurate and compact codebook vectors for vector quantization, and the design of effective box-pushing robot behaviors. The second set of chapters ex-

plores mechanisms to make evolutionary design *scalable*, largely through the use of biologically inspired *development* mechanisms. These include a recipe-like genetic coding based on L-systems (inspired by DNA coding of amino acids) and models of artificial neurogenesis (cell differentiation, division, migration, axonal growth, guidance, target recognition, etc.).

The third set of chapters largely deals with equivalents of *ontogenetic learning*, which encompass the use of powerful local learning algorithms to quickly improve and explore the promising solutions discovered by the underlying evolutionary search. These include gradient-descent based learning of radial-basis function network architectures designed by evolution, different flavors of Hebbian adaptation of evolved neural architectures, and the evolution of learning parameters and rules for second-order neural network architectures. The final set of chapters extends the principle of evolutionary search along novel and useful directions. These include the use of *local selection* as a powerful yet computationally inexpensive alternative to the more popular global selection schemes (e.g., proportional, tournament, etc.), *symbiotic evolution* for evolving and exploiting cooperating blocks (or units) of the neural network solution, treating the evolved population as an *ensemble of solutions* and exploiting the population diversity to produce enhanced and robust solutions, and developing a theoretically sound alternate interpretation of evolutionary search that operates over search distributions rather than recombination/crossover.

A summary of the key contributions of each chapter follows.

Evolutionary Synthesis of Complex Agent Programs

In Chapter 1, Balakrishnan and Honavar introduce the powerful agent synthesis paradigms of evolution and learning. Drawing inspiration from biological processes at work in the design of varied, and in many instances unique, adaptations of life forms, the authors present evolutionary synthesis of artificial neural networks as a computationally efficient approach for the exploration of useful, efficient, and perhaps even *intelligent* behaviors in artificial automata. Most important, the authors present a formal characterization of *properties* of genetic representations of neural architectures.

In Chapter 2, Gruau and Quatramaran address a challenging problem of evolving *walking gaits* for an 8-legged robot. This task is particularly difficult because the leg-controller must not only move the individual legs, but also coordinate them appropriately to ensure that the robot is balanced and stable. Using a genetic representation based on *cellular encoding*, they evolve programs for transforming a graph and decoding it into a neural network. Perhaps

the greatest benefit of this graph-grammar encoding scheme is its ability to evolve *reusable modules* or *subnetworks*, an aspect of critical importance in the control of the 8-legged robot. This encoding is used to evolve the architecture of the neurocontroller, along with the weights, the number of input and output units, as well as the *type* of units (*temporal* for duration of movement sequence and *spatial* for intensity of movement). The authors propose and use an *interactive approach*, calling for human intervention in identifying and promoting the emergence and adaptation of specific *regularities*. They achieve this by visually monitoring the evolutionary process and rewarding potentially promising features by increasing their fitness evaluations. The authors claim that this *semi-supervised* approach works well, requiring far fewer evolutionary generations to evolve acceptable behaviors than pure evolutionary search. They present a number of interesting walking behaviors that evolved, including *wavewalk* – moving one leg at a time, and the true *quadripod gait* – moving four legs at a time.

MacLennan presents an extremely interesting application of evolution in Chapter 3. He considers the evolution of meaningful communication in agent populations. In the first set of experiments, he arms the agents with *finite state machines* (FSMs) (which are computationally equivalent to a restricted class of neural networks), capable of reading *single symbols* and communicating/writing single symbols in return. The agents are rewarded for recognizing and responding with correct symbols. Using a *direct encoding* scheme to represent the transition tables of the FSMs, he demonstrates interesting results in the evolution of communication in this agent population. He shows that the evolved agents appear to communicate meaningfully using single symbols, and to some extent, pairs of symbols exhibiting rudimentary syntax. Owing to the limitations of FSMs, primarily their inability to *generalize*, MacLennan and his students also explore the use of simple feed-forward neural network structures for communication. In these experiments, they evolve the network architecture and use back-propagation to train the network. Again they demonstrate the evolution of meaningful communication, although the results are characteristically different from those using FSMs.

In Chapter 4, Merelo et al. adopt an evolutionary approach to address an important consideration in unsupervised classification problems. Unsupervised classifiers or clustering algorithms must not only discover compact, well-defined groups of patterns in the data, but also deliver the correct number of groups/clusters. This is a difficult multi-objective optimization problem with few practical algorithmic solutions. The common approach is to fix the num-

ber of desired clusters *a priori* as in the popular k-means algorithm. Vector Quantization (VQ) is a classification technique in which a set of labeled vectors, called the *codebook*, are given, and classification of an input is performed by identifying the codebook vector that matches it best. However, in order to use VQ, one needs the codebook to be specified. Kohonen's Learning Vector Quantization (LVQ) is an approach that can *learn* codebooks from a set of initial vectors. However, even with this approach, the size of the codebook needs to be set in advance. But how does one know what the size should be? Merelo et al. use an evolutionary approach to design the codebook, optimizing accuracy, distortion, and size. This multi-objective optimization leads to codebook designs that are smaller and more accurate than those obtained with LVQ, and contain fewer weights than back-propagation trained neural networks of similar accuracy. The authors demonstrate the power and utility of this approach in separating a set of Gaussian non-overlapping clusters and breast cancer diagnosis.

In Chapter 5, Balakrishnan and Honavar address the evolution of effective box-pushing behaviors. They use a multi-level, variable-length genetic representation that possesses many of the properties described in Chapter 1. Using this mechanism, they evolve both feed-forward and recurrent neuro-controllers for a heavily constrained box-pushing task. They analyze the evolved networks to clearly demonstrate the mechanism by which successful box-pushing behaviors emerge, leading to critical insights into the nature of the task in general, and the ability of evolution to deal effectively with the imposed constraints in particular. Through a variety of experiments the authors highlight the role of *memory* in effective box-pushing behaviors, and show how evolution tailors the designs to match (and surmount) constraints. The authors also discuss the evolution of robot sensory systems, and present empirical results in the co-evolution and co-adaptation of robot sensory and neural systems. They show that evolution often discovers sensory system designs that employ minimal numbers of sensors. Further, they also show that evolution can be trusted to discover robust controllers when the task environment is noisy.

Evolution and Development of Agent Programs

Although evolution is a powerful and adaptive search technique, its application is often limited owing to the immense computational effort involved in searching complex solution spaces. Local or environment guided search can often help alleviate this problem by quickly exploring promising areas identified by the evolutionary mechanism. Many authors have explored such combinations

of evolutionary and local search, with much success. One such local mechanism is inspired by developmental and molecular biology. This deals with the notion of *development*, which can be crudely defined as *adaptation that is a result of an interaction between genetically inherited information and environmental influences*. Development is usually associated with determining the *phenotype* from the *genotype*, possibly influenced by environmental factors, as is evidenced in the following three chapters.

In Chapter 6, Boers and Sprinkhuizen-Kuyper present a novel genetic representation for neural network architectures that is inspired by DNA coding of amino acids. This representation or coding mechanism can be thought of as a *recipe* for creating a neural network rather than its direct *blueprint*. This powerful and novel coding is a graph grammar based directly on the parallel string rewriting mechanism of L-systems, where each genotype contains a set of production rules, which when decoded produces the corresponding phenotype (or neural network). The primary benefit of this coding mechanism is its scalability in coding potentially large neural network structures. In addition, such grammar based representations also easily handle and foster the emergence of *modular designs*. The authors present results on the evolution of modular architectures for the famous *what-where* task. They also present an enhancement of the development process that constructively adds modules and units to the decoded network, and address its implications for exploiting the Baldwin effect in the Darwinian evolution framework.

Drawing inspiration from developmental and molecular biology, Michel presents a model of *artificial neurogenesis* in Chapter 7. He uses this procedure to evolve and develop dynamical recurrent neural networks to control mobile robot behaviors such as maze traversal and obstacle avoidance. His model of neurogenesis, based on protein synthesis regulation, includes artificial specifications of processes such as cell differentiation, division, migration, axonal growth, guidance, and target recognition. In addition, the linear threshold function units that are evolved are also adapted using simple Hebbian learning techniques. The preliminary results obtained using this approach offer much promise for the design of large, modular neural networks. The author presents examples of interesting behaviors evolved using this approach for the robot Khepera. Michel also discusses issues involved in the transfer of robot behaviors evolved in simulation, to real robots.

In Chapter 8, Parisi and Nolfi present a comprehensive view of the role of development in the evolutionary synthesis of neural networks. In their view, development is any change in the individual that is genetically *trig-*

gered/influenced. Using this broad definition, they summarize their considerable research in various aspects of development, including self-teaching networks, a model of neurogenesis (axonal growth process), self-directed learning, a model of temporal development with genetic information expressed at different *ages* of the individual, etc. Self-directed learning is demonstrated using an architecture wherein the neural network learns to predict the sensory consequences of the movements of the agent. The authors show that this ability endows the agents with more effective behaviors. Self-teaching networks contain two subnetworks; one produces the behavioral responses of the agent, while the other produces a teaching or supervisory input used to adapt the first subnetwork. The authors also experiment with individuals that are cognizant of their age, allowing adaptation of behaviors with age. The authors demonstrate and discuss results of these different developmental techniques in the context of agent behaviors such as foraging (food finding), wall/obstacle avoidance, goal approaching, etc. In most cases they use *a priori* fixed neural network architectures and a simple direct encoding to evolve the weights of the network.

Enhancing Evolutionary Search Through Local Learning

In addition to the use of recipe-like genetic codings that develop into neural architectures under the influence of environmental factors, other, more powerful local search algorithms can also be used in conjunction with evolutionary search. The following chapters present examples of the use of such learning techniques to further shape the solutions discovered by evolution.

In Chapter 9, Carse et al. address the design of radial basis function (RBF) neural networks using a hybrid approach that involves the use of evolution to discover optimal network architectures (number of hidden units, radial basis function centers and widths, etc.) and a gradient-descent based learning algorithm for tuning the connection weights. The authors introduce two interesting notions in the chapter – a novel crossover operator that identifies *similar units* no matter where they appear on the genotype, and a scaling of training epochs that assigns fewer training cycles to networks in the early stages of evolution, and progressively more as evolution advances. The rationale behind the former idea is to make evolutionary search more efficient by handling the *competing conventions/permutations* problem (where the exact same phenotype has multiple genetic representations), while the latter is inspired by the fact that networks in early stages of evolution are perhaps not architecturally close to ideal, and hence little effort need be wasted in training them. The idea thus is to discover the optimal structure first and then fine tune it. The authors

use variable length genotypes, with a direct encoding of the network architecture. They show that this approach produces relatively small yet accurate RBF networks on a number of problems, including the Mackey-Glass chaotic time series prediction problem.

In Chapter 10, Floreano et al. investigate chase-escape behaviors co-evolving in a framework of two miniature mobile robots, a predator with a vision system and a faster prey with only proximity sensors. Both robots are controlled using recurrent neural networks evolved in a competitive co-evolutionary setting. Inspired by the thesis that unpredictable and dynamic environments require the *evolution of adaptivity*, the authors explore mechanisms that produce solutions capable of coping with changing environments. This becomes particularly critical in co-evolving populations owing to the phenomenon known as the *Red Queen Effect*. In such situations, adaptive traits evolved by one species are often annulled by the evolution of counteractive traits in a competing species, thereby producing *dynamic fitness landscapes*. In order to evolve agent designs that can cope with such dynamic and unpredictable changes, the authors explore three broad approaches – (1) maintaining diversity in evolving populations, (2) preferentially selecting genotypes whose mutant neighbors are different but equally viable, and (3) incorporating mechanisms of ontogenetic adaptation. The authors use a direct encoding representation to evolve weights within *a priori* fixed network architectures, and in later experiments, noise levels and parameters for Hebbian learning. They present results of interesting evolved chase-escape behaviors.

Crabbe and Dyer, in Chapter 11, present a flexible neural network architecture that can learn to approach and/or avoid multiple goals in dynamic environments. Artificial agents endowed with this architecture, called MAXSON (max-based second order network), learn spatial navigation tasks faster than other reinforcement-based learning approaches such as Q-learning. The learning architecture employs two subnetworks – a second-order policy network that generates agent actions in response to sensory inputs, and a first-order value network that produces reinforcement signals to adapt the policy networks. Of particular importance are two thresholds, θ_1 and θ_2, that control the response of the agent to sudden sensory changes or fluctuations. The authors demonstrate the emergence of effective behaviors for agents seeking food and drink and avoiding poisons. The final set of experiments demonstrates the ability of this approach in producing agents driven by *internal goals*, i.e., agents seek food or water only when they are hungry or thirsty, respectively. The authors use an evolutionary approach to determine optimal settings for the

thresholds, which strongly influence the learning rules of the agents. They also present results in evolving portions of the learning rules themselves.

Novel Extensions and Interpretations of Evolutionary Search

Although evolutionary algorithms have proven themselves to be efficient and effective heuristic search techniques, their applicability is often limited owing to a few drawbacks such as the phenomenon of *premature convergence*, lack of concrete theoretical underpinnings, etc. This has prompted researchers to extend or interpret evolutionary search mechanism in novel and useful ways. The final set of chapters in this compilation extend basic evolutionary search along interesting and useful dimensions.

In Chapter 12, Menczer et al. present an interesting variation of evolutionary algorithms, namely, *local selection*. Their algorithm ELSA (Evolutionary Local Selection Algorithm) creates and maintains diversity in the population by placing *geographic* constraints on evolutionary search. In their approach, individuals/agents in the population compete for finite environmental resources, with similar agents obtaining less energy than dissimilar ones. Agents accumulating more than a certain fixed threshold of energy reproduce via mutation, while agents with no energy die. This environment-mediated crowding leads to diverse populations containing heterogeneous individuals. Such heterogeneity is a must for certain kinds of cover or Pareto optimization problems. For example, on problems with multi-criterion or expensive fitness functions, ELSA can quickly find a set of solutions using a simplified fitness function. Some other technique can then be used to compare these solutions to pick the right one. Using a direct encoding of feed-forward neural networks, the authors demonstrate the power of ELSA on a number of application domains, including evolution of sensors and adaptive behaviors, design of realistic ecological models, web-information browsing agents, and feature selection for classification tasks.

Moriarty and Miikkulainen present the SANE (Symbiotic, Adaptive Neuro-Evolution) approach in Chapter 13. The SANE approach differs from conventional evolution of neural architectures in that each individual in the population is a neuron rather than a neural network. In the fitness evaluation phase, pools of neurons are randomly drawn from the population to construct neural networks, which are then evaluated on the target task. The fitness of the resulting neural network is then distributed among the component neurons. Evolution thus favors neurons that can cooperate with each other. This approach supports heterogeneity by maintaining a diverse set of neurons, since