Genetic Programming in Europe

Report of the EvoGP Working Group on Genetic Programming of the European Network of Excellence in Evolutionary Computing

W. B. Langdon and R. Poli

30 November 1997

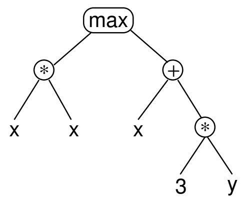
Genetic Programming (GP) is becoming a leading area of computer science in which Europe has a number of very active researchers which are now all gathered in the EvoGP working group. In this report we describe the state of the art in GP technology, the role Europe has in advancing it and the activities of the EvoGP working group. We indicate the likely benefits of this technology now and in the near future and we make recommendations for future European research.

1 Introduction

Genetic Programming (GP) is a powerful set of techniques which allow the automatic production of computer programs. As a field GP was founded by John Koza at the beginning of the 90's, and has grown exponentially since then [Koz92, Koz94, Koz98]. GP is inspired by Darwinian natural evolution, and has been applied successfully to a large number of difficult problems, such as automatic design, pattern recognition, robotic control, optimisation, financial trading and forecasting.

GP is a radically different method of developing software. In GP, the domain experts, instead of trying to transfer their knowledge to computer scientists, can create applications by directly "training" the genetic programs. This is done either by selecting the examples from which the algorithms must learn and generalise, or by grading intermediate solutions. This process of non-mediated automatic knowledge elicitation can dramatically reduce costs and development time while increasing the effectiveness of the programs developed.

GP is based on a special kind of genetic algorithm in which the structures being optimised are variable-size computer programs rather than inflexible fixed-length vectors of parameters. In GP programs are typically expressed as parse trees, rather than as lines of code. For example, the simple expression `max(x * x, x + 3 * y)` would be represented as



The genetic operators used in GP, crossover and mutation, are specialised to act on such structures so as to always produce valid (executable) offspring programs. For example, in its simplest form, crossover works by replacing a subtree in one parent program with another randomly selected subtree taken from another parent program.

The evolution in GP is guided by fitness functions which determine how well each evolved program performs with respect to certain predefined criteria. In many cases the experts can provide a training set of examples which indicate the output of the target program given a certain set of inputs. In those cases the fitness function is simply a measure of the similarity between the outputs produced by the program and those in the training set. In other cases this is impossible and much more complex and slow procedures are necessary to assess the quality of the programs in the population (e.g. in evolutionary robotics, it may be necessary to run each program on a real robot, let the program control the robot for a few minutes and then assess the quality of the resulting behaviour).

2 Genetic Programming

2.1 Taxonomy

In this section we give a classification of genetic programming. Selected research topics are described in Sections 2.2 and an overview of applications of GP is given in Section 2.3.

1. Various changes to the representation of the programs to be evolved
   - Reducing memory requirements and reducing runtime using directed acyclic graphs [Han94a, Kei96]
   - Linear rather than tree representations (Section 2.2.1)
   - Representing programs with graphs (Section 2.2.1)
   - Using trees to specify how to grow another structure (such as an artificial neural network [Gru96], an electrical circuit [KBA+97] or simulations of flowering plants [Jac97])
   - Including semantic information via type systems [Mon95]
   - Incorporating state or database information within a genetic program (Section 2.2.2).
2. Investigation of better ways to assess program fitness (Section 2.2.3)
   - Dynamic fitness such as [Tet96], LEF [GR97], RAT [TA97] and co-evolution [Sie94]
   - Measures aimed at reducing the size of the evolved programs [IdS94, ZM96].
3. Measures to allow programs to be evolved from program modules and to simultaneously evolve the modules
   - Automatically defined functions (ADFs) [Koz94, KA95]
   - Module Acquisition (MA/GliB) [Ang94] and
   - Adaptive Represention through Learning (ARL) [RB96].
4. Search operators and algorithms such as
   - Mutation [BFN96, Lan98]
   - Context preserving crossover (SCPC/WCPC) [D'h94]
   - One-point Crossover [PL97]
   - Non genetic algorithm methods of searching the space of programs [SS97].
5. The adoption of advanced genetic algorithm techniques by genetic programming.
6. An increasing number of implementations, many of them in the public domain.
7. An increasing number of applications (Section 2.3).

2.2 GP Research

2.2.1 Program Representation

In most genetic programming work the programs being evolved are represented using trees [Koz92]. However a number of alternative approaches use linear chromosomes. [Per94] uses a linear postfix language and claims some advantages over the more standard tree approach. [Ban93] also uses a linear approach but various operations convert this to a tree shaped program which is executed. However of the linear chromosome approaches perhaps the most surprising and successful is [Nor94] in which the representation is a small (up to 1000 bytes) machine code program. Enormous (more than thousand fold) speed ups in comparison with LISP are claimed. Later work [NB95b] extended the system so that it was fully Turing complete.

A radically different program representation is used in Parallel Distributed Genetic Programming (PDGP) where program trees are replaced by directed graphs. As nodes within the graph have multiple outputs, the intermediate results calculated by a node can be readily used in many other parts of the program. It is this reuse which gives the technique its power. PDGP is a very flexible framework and has been demonstrated evolving solutions for symbolic regression, parity and artificial ant problems and evolving boolean logic networks, artificial neural networks, finite state automata, etc. [Pol96d, Pol96a, Pol96b, Pol97].

In a system called PADO [Tel96], a new program representation was used to obtain parallel classification programs for signals and images. In PADO programs are made up of several groups of programs each group being used to recognise the instances of images of a given class. The group with the highest output determines the output of PADO. The output of a group is a linear combination of the output of the programs included in the group. Programs are represented as directed graphs in which arcs represent possible flows of control and nodes perform actions and take decisions on which of the possible control flows should be followed. Each node includes an action, a branching condition, and a stack.

2.2.2 Genetic Programming and State Information

Most computer programs make extensive use of storage, yet in almost all genetic programming examples, each genetic program is a function of its inputs and almost all storage requirements are dealt with by the framework used to support the genetic programs. [Lan98] is primarily concerned with extending genetic programming to cover this important topic and provides an overview of GP with evolvable memory.

2.2.3 Parallel Execution and Evolvable Hardware

In common with GAs, the vast proportion of machine resources consumed by GP are used running the fitness function to evaluate the population. Like GAs, GP parallelises easily and can readily take advantage of a wide range of dedicated parallel processing architectures [TOD96, SS96, JP96, Ikr96, AK96, OCPT96]. There is also interest in parallel execution using networks of workstations (e.g. using PVM) and across the Internet using Java. Often GP (and GAs) are parallelised by splitting the population so different parts of it are run on different CPUs. This also has the potential benefits associated with using a demic population. GP may also be parallelised by execution of different fitness cases on different computers.

There is also great interest in evolvable hardware (EHW) which promises considerable speed up in fitness evaluation, either by parallel execution, by execution closer to the hardware or simply by providing faster hardware.

2.3 GP Applications

2.3.1 Prediction and Classification

One application where GP is already making a contribution is that of generating models to automatically fit data. It can be particularly appropriate where the available data are noisy and the underlying processes are non-linear. Applications with both large and small data sets have been reported. Such models may help explain the process, be used for prediction, to optimise the process or control it.

The BioX system [BGMT95] has been used to model chemical engineering processes and river flows. [WC97] models river flows based on rain fall measurements. On a very seasonal Australian river GP provided a better model than a conventional modelling tool. [WHH+97] shows their GP system can accurately model complex chemical engineering processes using only observed data. While more accurate than a neural network they state "the main advantage ... is that it [GP] has automatically eliminated irrelevant model inputs ... [and] offers a degree of parsimony not attainable using a neural network."

[Han93] uses genetic programming to predict the shape of proteins. He was able to evolve programs which, using the protein's chemical composition, were able to predict whether each part of a protein would have a particular geometric shape (an α-helix) or not. Genetic programming was able to do this as well as the best of other techniques but all suffered from the fact that the structure depends upon more than local composition. [Koz94] reports similar success on other protein geometry problems.

[Lon97] recasts the problem of detecting anomalies in streams of hundreds of thousands of events generated by computer simulations, as the problem of inducing a formal grammar which describes most events. The events it fails to match are then the interesting anomalies. [Lon97, page 452] states "The algorithm itself has proved its worth to us in the areas of data-mining and the analysis of the results from simulation".

Andre has successfully used genetic programming in optical character recognition problems (OCR). In [And94a] he combines genetic programming with a genetic algorithm to produce an OCR program from scratch. In [And94b] he shows genetic programming can be used to maintain existing hand coded programs. He shows genetic programming automatically enhancing an existing manually written OCR program so that it can be used with an additional font. It is perhaps on routine maintenance problems, such as this, that genetic programming will find most immediate commercial application.

2.3.2 Image and Signal Processing

[Tac93] describes the use of genetic programming to extract targets from low contrast noisy pictures. Various (≈20) standard metrics are extracted from the image using standard techniques, which are then processed by a genetic program to yield target details.

[Oak94] describes obtaining blood flow rates within human toes using laser Doppler measurements. These measurements are both noisy and chaotic. He compares the effectiveness (at removing noise but preserving the underlying signal) of special filters evolved by genetic programming and various standard filters. He concludes that a combination of genetic programming and heuristics is the most effective.

[JMD94] was able to evolve 2D image processing programs to extract the location of peoples' hands from still silhouettes. They report these performed better than the best hand coded algorithms. [PC97] evolves programs to improve the usefulness of 2D medical images while [EAS97] evolves digital signal filters which provide channel equalisation to restore the original signal after it has been distorted by noise and dispersion. [NB96] describes applications of GP to image and sound compression.

2.3.3 Design

[KBAK96] shows the automatic design of electrical circuits to meet onerous design requirements.

[NH94] have used genetic programming to evolve 3-D jet aircraft models. The determination of which models are fitter is done manually. This work is similar to that in Section 2.3.6.

2.3.4 Trading

[AP94] used genetic programming to create strategies which have traded in simulated commodity and futures markets (the double auction tournaments held by the Santa-Fe Institute, Arizona, USA). Their automatically evolved strategies have proved superior to many hand-coded strategies. While [CY97] use GP to model volatility in Japanese and New York stock markets and [OCPT96] model currency markets.

Other financial applications include direct marketing, credit card credit worthiness scoring, loan application evaluation, data mining and customer retention modelling [Eib97].

2.3.5 Autonomous Agents and Robotics

One the current active strands in robot research is the development of independent mobile robots which are programmed to react to their environment rather than to follow a global plan [Bro91]. Until recently robot controllers had been hand coded however Spencer has been able to use genetic programming to automatically generate a control program enabling a simulated six legged robot to walk [Spe94]. [GQ96] and [BNO97] have evolved control programs and run them on real robots using GP. While [LHL97] evolve mobile robot controllers using GP and a simulator but demonstrate the controller running on the physical robot.

[Han94b, HWSS95, Iba96, Qur96, RD94, LS96, ZKL96] have applied evolved autonomous agents using genetic programming.

2.3.6 Artistic

There have been a number of uses of genetic programming, perhaps inspired by Dawkins' biomorphs [Daw86] or Karl Sims' panspermia [Sim91], which generate patterns on a computer display. For example [GH97] describes GP being used to make a short single character video. While Reynolds' "boids" technique [Rey92, Rey94b, Rey94a]. has been used as a basis for photo realistic imagery of bat swarms in the films "Batman Returns" and "Cliffhanger" [Rey96].

[DFP+94] uses genetic programming to generate sounds and three dimensional shapes. Virtual reality techniques are used to present these to the user. As in Section 2.3.3, there is an interactive fitness function, with the user indicating a preference between the four objects presented to him. [SA94] have used GP to automatically generate Jazz improvisations. (Examples of genetic music are available from the Internet, e.g. via `http://www.cs.bham.ac.uk/~wbl`).

3 GP in Europe

In this section we review very briefly the main players in genetic programming within Europe. (To save space we do not reiterate work already described in the previous section, more information is available from the EvoGP members, often via the Internet, see Section 7 and Appendix A).

GP is still in its infancy and so it is not surprising that interest is centred on academic institutions, however many of them have commercial partnerships and already some European firms are active in this area, e.g. British Telecom [RM95], Daimler-Benz, Siemens, Voest-Alpine, Cap Volmac, and a number of smaller firms and financial institutions. Such institutions often do not wish to publish their work due to the sensitivity of GP results as this could affect their competitiveness.

3.1 Dortmund

The Computer Science Department at the University of Dortmund has been active in GP since 1993. It has a wide range of interests including linear genomes (Section 2.2.1), theoretical issues [NB95a, NFB96] and applications such as: function regression and model building, data compression [NB96], robotics (Section 2.3.5), interactive evolution (user-in-the-loop approaches), combinatorial optimisation problems and multi-agent systems. It is also investigating issues concerning self-organisation and novel forms of computation such as molecular computation. The principal members of the group have recently published the first GP text book [BNKF97].

3.2 Birmingham

Like the Dortmund group, the EEBIC group within Birmingham University has diverse interests and international collaborative links. At present the major topics include the theoretical underpinnings of GP [PL97, LP97a, Lan98, LP97b], novel representations and operators (Section 2.2.1), as well as applications such as medical imaging [Pol96c] and scheduling [LT97].

3.3 Scotland

Three Scottish universities, Napier, Edinburgh and Glasgow are very active in the GP field. For example at Napier the fundamental scaling problems associated with solving complex problems are under investigation [ABF97] as are issues concerning generating efficient programs from high level specifications, test programs, etc. While Edinburgh has proposed improved methods of testing evolved programs [GR97] and in Glasgow novel means of correcting distortion in radio frequency communications links have been investigated [EAS97]. The University of Essex has also applied GP to Telecommunication Networks [ASS97] and to the evolution of simple software agents.

3.4 Paris

There is an established tradition of evolutionary computation in general and GP in particular in France. For example this is shown by the Evolution Artificielle series of conferences (1994 (Toulouse), 1995 (Brest) and 1997 (Nimes)) and the pioneering work of Gruau at Ecole Normale Superieure de Lyon on neural nets and GP (Sections 2.1 and 2.3.5). Today the Ecole Polytechnique in Paris takes a lead GP role, hosting the first European workshop on GP (Section 4.2) and EvoGP group's WWW pages (Section 4.3). The principal research activities include applying GP to modelling the mechanical properties of novel engineering materials including improved means of fine tuning continuous coefficients [SSJ+96], reducing the cost of fitness evaluation, automatically simplifying evolved programs and extending GP to graphs. The AnimatLab at Ecole Normale Superieure is also active in the GP field.

3.5 Leiden

The GP group at Leiden has concentrated upon financial applications of GP such as, direct marketing, credit card credit worthiness scoring, loan application evaluation, data mining and customer retention modelling as well as theoretical aspects such as improved representations [Kei96, EEK+96, EEKS98].

3.6 Switzerland

The principal centres of GP work in Switzerland are ETH (Zurich), IDSIA (Lugano), Geneva and EPFL (Lausanne). Topics include evolving compact programs [Bli96], improved search techniques and use of GP with parallel computers [MST95].

3.7 Cork

In Ireland there has been work on using GP to automatically parallelise programs [RW97] as well as investigations into the mechanics of GP populations [Rya94].

3.8 Italy

In Parma GP has been applied to making existing medical imaging techniques more effective and to the evolution of cellular-automata-based pattern classifiers. In Naples as an approximate technique to overcome intractable space and time difficulties associated with some mathematical analysis [FCC+97]. Work in Milan has concentrated on fitness functions [Tet96].

3.9 Brussels

The adaptive systems group at the Vrije Universiteit Brussel (VUB) and the IRIDIA group at Université Libre de Bruxelles have long been interested in evolutionary computation and are increasingly interested in Genetic Programming particularly in data mining and autonomous robotics [DC97]. VUB is developing a platform independent distributed implementation of GP written in Java. IRIDIA has been actively involved with the American series of GP conferences [KDD+97].

3.10 Elsewhere

GP research is also active in Vienna [BGS96], Berlin and Bulgaria [SN97].

4 EvoGP Working Group

Genetic Programming has been identified as one of the important recent innovations in Evolutionary Computation. The objective of the EvoGP working group is to foster cooperation among research and development teams in the information-based services, industries and academia throughout Europe.

In a first phase we intended to achieve the following:

- To form a working group representing all EvoNet members with an active interest in the area of Genetic Programming and to establish communication channels (email list, web pages, meetings, etc.) between the members of the group.
- To hold an international workshop on Genetic Programming, with invited speakers from the US and Japan as well as from EvoNet nodes. The aims are to give European and non-European researchers in the area of genetic programming as well as people from industry an opportunity to present their latest research and discuss current developments and applications and also to maximally publicise GP technology in Europe. This event will be the first workshop on Genetic Programming in Europe and should serve as a seed for a possible European conference series on Genetic Programming.
- To write and circulate a report of the state of the art in the area of GP in Europe and the rest of the world, including likely future directions and possible competitive and economic advantages.

In the following subsection the activities of EvoGP are briefly described.

4.1 Meetings Held

EvoGP members had two meetings in the Summer and one electronic meeting in Autumn 1997:

15 July 1997 This was the first (informal) meeting of the working group. It was held in Stanford (USA) during the GP'97 conference. Eight members were present. Several activities were proposed, most of which are now well under way. These included: the organisation of the first European workshop on GP, the setting up of an email list, the preparation of Web pages dedicated to the activities of the group and the preparations of this report. John Koza (chairman of GP'97) was contacted at the conference and agreed to be an invited speaker for the workshop.

22 July 1997 This second informal meeting was held at Michigan State University (East Lansing, USA) during the ICGA'97 conference. Five members were present. In this meeting we mainly discussed the issues related to the organisation of the workshop. The organising committee was set up, the programme committee was sketched, an (electronic) submission format was proposed (this was later modified due to the availability of Springer Verlag to produce the proceedings in time for the event), the site and approximate date of the workshop were also fixed.

5–7 November 1997 This was the first formal meeting of the group. It was held on the Internet using electronic mail via the group's mailing list (`evogp@dcs.napier.ac.uk`). More than 30 messages where exchanged. All members of EvoGP were "present", about two thirds voted and/or participated to the discussion. The main topics for discussion in the meeting were this report, its structure and contents and the EvoGP web pages.

4.2 Workshops Planned

The working group is organising an international workshop, EUROGP'98, the First European Workshop on Genetic Programming, to be held in Paris, on 14–15 April, 1998.

EUROGP'98 is the first event entirely devoted to Genetic Programming to be held in Europe. The aims are to give European and non-European researchers in the area of genetic programming as well as people from industry an opportunity to present their latest research and discuss current developments and applications. The workshop will be held in conjunction with EVOROBOT'98, the first European event on evolutionary robotics, organised by the EvoNet working group on Evolutionary Robotics.

The organising committee includes: Wolfgang Banzhaf, University of Dortmund and Riccardo Poli, The University of Birmingham (program co-chairs), Terry Fogarty, Napier University (publication chair), and Marc Schoenauer, Ecole Polytechnique (local chair). The workshop has an international programme committee including not only the main GP experts in Europe but also several leading GP researchers from around the world: Peter Angeline (USA), Hitoshi Iba (Japan), Una-May O'Reilly (USA), Byoung-Tak Zhang (Korea). John Koza, the founder of GP, will give an invited speech. The workshop proceedings will be published by Springer in the Lecture Notes in Computer Science series.

The workshop Web page is `http://www.cmap.polytechnique.fr/www.EvoGP/EuroGP98.html` . The flyer of the workshop is shown in Appendix B.

4.3 Web Page Development

Web pages dedicated to the activities of the group have been set up at Ecole Polytechnique, France, at URL `http://www.cmap.polytechnique.fr/www.EvoGP/` thanks to the availability and work of Alain Racine and Marc Schoenauer. One of the EvoGP Web pages is shown in Appendix B.

5 Future Role of GP in European Industry and Commerce

GP offers a radically different method of developing software in which the domain experts can play a central role by controlling the process of iterative refinement of the programs being evolved. The process of non-mediated automatic knowledge elicitation used in GP can dramatically reduce costs and development time while increasing the effectiveness of the programs developed. An obvious drawback of GP is the heavy computation load it involves. This is however less and less important as the performance of computers is still doubling every 18 months and low cost parallel computers are now becoming available. What is now considered computationally heavy in 5 years time will be considered perfectly acceptable. Also as GP technology refines, the amount of computation required to solve complex program induction tasks constantly decreases.

Indeed, GP is solving more and more complex problems every year. In many domains (e.g. in electronic design) its performance is now superior to those of human experts.

So, we foresee that in the relatively near future a new generation of software tools based on GP for the automatic design of programs will be available. These will provide new solutions to hard problems in complex domains of high industrial and social relevance where standard methods provide unsatisfactory results. GP will also provide quick and cheap solutions to program design problems in domains where the reduced development times (e.g. in financial applications) and costs (e.g. in small and medium enterprises, SMEs) is vital.

6 Recommendations

In the previous section we have highlighted the potential of GP which in many areas is ripe for exploitation by European industry. To ensure that today's GP technology is fully exploited by European Industry and Commerce, the European Union should support financially GP dissemination and research in Europe.

Being such a new technology, GP has still a number of deficiencies and challenges that need to be overcome. So, the EU should also fund fundamental research on GP to ensure that Europe remains a key player in the development of this increasingly important technology.

In the following sections we make specific recommendations:

6.1 Application Areas

As shown in Section 2.3, today's GP technology can be used to produce innovative solutions in a number of domains. Funding is required for pilot projects which will demonstrate the benefits of this technology in commercial and industrial real-life environments. As with any new technology demonstration projects are needed to convince managers (especially in SMEs) of GP's effectiveness and financial viability before they will adopted it.

Promising application areas include:

- Automatic engineering design (e.g. electronics, mechanical design)
- Financial forecasting
- Signal and image processing (e.g. in industrial inspection)
- Robotics
- Scheduling
- Medicine

6.2 Experimental and Theoretical Studies

Although GP has shown impressive performance on a variety of applications, it is obvious that this technology will move forward rapidly and it is important that Europe remains at the forefront of it or even has an edge over its Japanese and US competitors. EU funding will be necessary to ensure this.

The efficiency and effectiveness of GP depend on the power of its search operators, on the representations chosen, on the quality measures used to direct the search and on the details of the algorithm used. Each of these components need to be studied and their strengths and limitations in terms of speed, flexibility and scalability identified. This is a prerequisite to propose and demonstrate improved forms of GP which overcome such limitations.

The way GP works is still only partially understood. In order to remove its inefficiencies and to understand which problems it is most suitable to solve and how, more research is needed on the theory of GP. The theory can suggest improvements in terms of representations, operators, fitness functions and algorithms. Particularly promising areas include: theories of fitness measures and of genetic operators, schema theories, probabilistic population models, and fitness landscapes.

The following challenges need to be addressed:

- How does GP search the space of possible solutions? What are the best mathematical tools to study program spaces and the evolution of programs?
- What are the relationships between GP and other search algorithms?
- Can dynamic/adaptive fitness functions be fruitfully applied? Can representations and operators be learnt over time?
- Can GP evolution be open-ended? Is there an intrinsic scalability problem in GP? What is its nature?
- For which domains is GP the technology of choice? How can they be characterised? How can domain knowledge be incorporated into GP?
- What are the optimum search operators and representations for any given domain of application?

7 Conclusions

In this report we have summarised the state of the art in genetic programming and emphasised the European contribution. From this it is clear that GP is progressing rapidly internationally and that at present Europe is playing an important role (the members of the EvoGP working group have published more than 70 papers on GP).

EvoGP has gathered together academics and industrial practitioners of GP to maximise the interchange of ideas and the impact of this technology on industry and commerce. However, there is a serious risk that Europe may fall behind in what is obviously a major technology for the future. Therefore, in Section 6 we have made recommendations to ensure Europe remains at the forefront of this increasingly important field.

Our principal recommendation is that the European Union should support financially GP pilot demonstrations to convince SMEs to adopt this technology as well as experimental and theoretical studies to ensure Europe remains at the forefront and has an edge over its competitors.

The EvoGP working group is available to serve as a convenient conduit between The Commission and the European GP community.

More Information

In addition to the references at the end of this paper, there is a great deal of information on genetic programming available on the Internet. The following Internet addresses may be useful; also many of our members (Appendix A) have home pages. These can be found via the EvoGP home page.

Home page	URL
EvoGP	`http://www.cmap.polytechnique.fr/www.EvoGP/`
Email for all EvoGP members	`evogp@dcs.napier.ac.uk`
EvoNet	`http://www.dcs.napier.ac.uk/evonet/Coordinator/evonet_f.htm`
GP bibliography, 1000+ entries	`http://www.cs.bham.ac.uk/~wbl/biblio/gp-bibliography.html`
Search the GP bibliography	`http://liinwww.ira.uka.de/bibliography/Ai/genetic.programming.html`

Acknowledgements

We would like to thank the many members of the EvoGP group who have helped in the preparation of this report. In particular Terry Fogarty, Gusz Eiben, Michele Sebag, Christian Jacob, Wolfgang Banzhaf, Stefano Cagnoni, Marco Tomassini, Marc Schoenauer, Marco Dorigo, Bernard Manderick, and Mark C. Sinclair.

A Members of EvoGP

Member	Electronic Mail / World Wide Web
Wolfgang Banzhaf	banzhaf@cs.uni-dortmund.de http://ls11-www.informatik.uni-dortmund.de/people/banzhaf
Joachim Born	born@dbag.bln.daimlerbenz.com N/A
Markus Brameier	brameier@ls11.informatik.uni-dortmund.de http://ls11-www.informatik.uni-dortmund.de/people/brameier
Stefano Cagnoni	cagnoni@ce.unipr.it http://peggy.mt.unifi.it/cagnoni.html
Bastien Chopard	chopard@cui.unige.ch http://cuiwww.unige.ch/~chopard/
Gianni Degli Antoni	gda@dsi.unimi.it http://www.crema.unimi.it/struttura/docenti/degliantoni.html
Marco Dorigo	mdorigo@ulb.ac.be http://iridia.ulb.ac.be/dorigo/dorigo.html
Gusz Eiben	gusz@wi.leidenuniv.nl http://www.wi.leidenuniv.nl/home/gusz/index.html
Terence Fogarty	t.fogarty@dcs.napier.ac.uk http://www.dcs.napier.ac.uk/~tcf/
Christian Jacob	jacob@informatik.uni-erlangen.de http://www2.informatik.uni-erlangen.de/~jacob
Robert Keller	keller@ls11.informatik.uni-dortmund.de http://ls11-www.informatik.uni-dortmund.de/people/keller
Jerome Kodjabachian	kodjaba@wotan.ens.fr http://www.biologie.ens.fr/AnimatLab
Bill Langdon	W.B.Langdon@cs.bham.ac.uk http://www.cs.bham.ac.uk/~wbl/
Tom Lenaerts	tlenaert@vub.ac.be http://dinf.vub.ac.be/asg/persons/tom.html
Frieder Lohnert	lohnert@DBresearch-berlin.de N/A
Bernard Manderick	bernard@arti.vub.ac.be http://dinf.vub.ac.be/asg/index.html
Jean-Arcady Meyer	meyer@biologie.ens.fr http://www.biologie.ens.fr/AnimatLab
Peter Nordin	nordin@LS11.informatik.uni-dortmund.de http://ls11-www.informatik.uni-dortmund.de/people/nordin/
Antonio Piccolboni	piccolbo@dsi.unimi.it http://eolo.usr.dsi.unimi.it/~piccolbo/
Riccardo Poli	R.Poli@cs.bham.ac.uk http://www,cs.bham.ac.uk/~rmp/
Alain Racine	racineal@cmapx.polytechnique.fr http://blanche.polytechnique.fr/users/www.racineal/racineal.html
Marc Schoenauer	marc@cmapx.polytechnique.fr http://blanche.polytechnique.fr/users/www.marc/marc.html
Michele Sebag	Michele.Sebag@polytechnique.fr http://blanche.polytechnique.fr/www.eeaax/michele/michele.html
Mark C. Sinclair	mcs@essex.ac.uk http://esewww.essex.ac.uk/~mcs
Andrea Tettamanzi	tettaman@dsi.unimi.it http://eolo.usr.dsi.unimi.it/~tettaman/
Blickle Tobias	t.blickle@ids-scheer.de http://www.ch.pgp.net/~blickle/index.html
Marco Tomassini	marco.tomassini@di.epfl.ch N/A
Hans-Michael Voigt	voigt@gfai.FTA-Berlin.de http://zarnow.gfai.de/hmv.htm

B Additional Information

EuroGP'98 flyer

ORGANISING COMMITTEE

- Wolfgang Banzhaf
  University of Dortmund
  (Program co-chair)
  banzhaf@cs.uni-dortmund.de
- Riccardo Poli
  University of Birmingham
  (Program co-chair)
  R.Poli@cs.bham.ac.uk
- Terry Fogarty
  Napier University
  (Publication chair)
  T.Fogarty@dcs.napier.ac.uk
- Marc Schoenauer
  Ecole Polytechnique
  (Local chair)
  marc@cmapx.polytechnique.fr

PROGRAMME COMMITTEE

- Peter Angeline (USA)
- Wolfgang Banzhaf (Germany)
- Tobias Blickle (Germany)
- Marco Dorigo (Belgium)
- Gusz Eiben (The Netherlands)
- Terry Fogarty (UK)
- Frederic Gruau (The Netherlands)
- Hitoshi Iba (Japan)
- W.B. Langdon (UK)
- Jean-Arcady Meyer (France)
- Peter Nordin (Sweden)
- Una-May O'Reilly (USA)
- Riccardo Poli (UK)
- Marc Schoenauer (France)
- Michele Sebag (France)
- Andrea Tettamanzi (Italy)
- Marco Tomassini (Switzerland)
- Hans-Michael Voigt (Germany)
- Byoung-Tak Zhang (Korea)

IMPORTANT DATES

Submission deadline:
1 December 1997
Notification of acceptance:
10 January 1998
Camera ready papers for workshop:
31 January 1998
Workshop:
14-15 April 1998

CALL FOR PAPERS

EUROGP'98

First European Workshop on Genetic Programming
Paris, 14-15 April, 1998

Genetic Programming (GP) is a new branch of Evolutionary Computation in which the structures in the population being evolved are not fixed-length strings of parameters that encode possible solutions to a problem, but *programs* that, when executed, *are* the candidate solutions to the problem.

GP has been applied successfully to a large number of difficult problems like automatic design, pattern recognition, robotic control, synthesis of neural networks, symbolic regression, music and picture generation, etc.

EUROGP'98 is the first event entirely devoted to Genetic Programming to be held in Europe. The aims are to give European and non-European researchers in the area of genetic programming as well as people from industry an opportunity to present their latest research and discuss current developments and applications. The event will be held, Tuesday and Wednesday in the week following Easter, in the center of Paris, Quartier Latin, a 5 minute walking distance from Notre-Dame.

The workshop is sponsored by the EvoNet, the European Network of Excellence in Evolutionary Computation, and is one of the activities of EvoGP, the working group on Genetic Programming of EvoNet. It will be held in conjunction with EVOROBOT'98, the first European event on evolutionary robotics, organised by EvoRob, the EvoNet working group on Evolutionary Robotics. EVOROBOT'98 will be held on April 16-17 at the same site as EUROGP'98.

Topics of interest include, but are not limited to: theoretical developments, experimental results on performance and behaviour of GP runs, new algorithms, representations and operators, novel applications of GP to real-life problems, hybrid architectures including GP components, comparisons with other machine learning or program-induction techniques, new libraries and implementations.

The workshop will consist of: a tutorial on GP for beginners, an invited talk by John Koza, oral and poster sessions with periods for discussion, software demos and industrial stands.

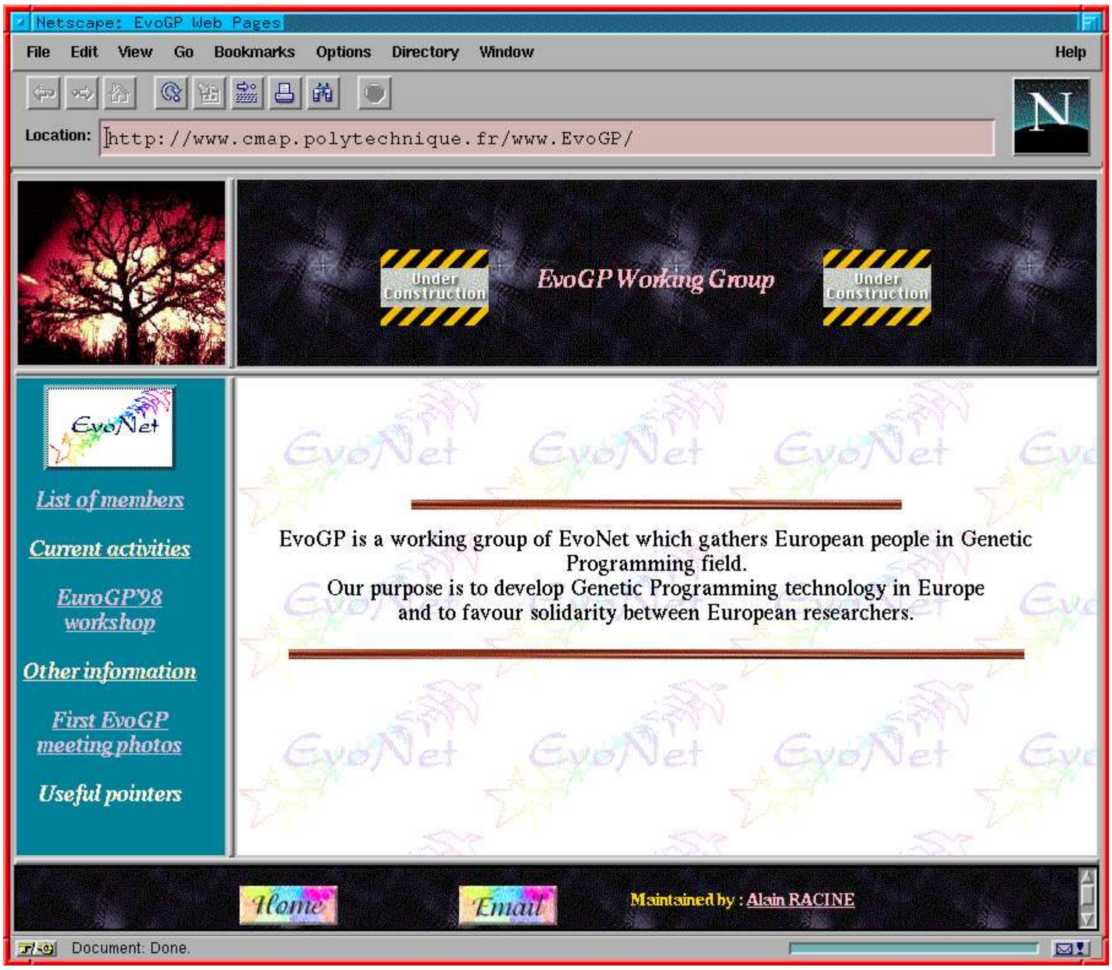
Submissions

To submit, send your manuscript (max length: 10 A4 pages) to one of the co-chairs, Wolfgang Banzhaf or Riccardo Poli, in PostScript (preferably compressed and uuencoded) *by email* not later than December 1, 1997. The papers will be peer reviewed by at least two members of the program committee. Authors will be notified via email on the results of the review by January 10, 1998.

The authors of accepted papers will have three weeks to improve their paper on the basis of the reviewers' comments and will be asked to send a camera ready version of their manuscripts in LNCS format (12 pages recommended, 15 pages max) by January 31st, 1998. The papers accepted will appear in the workshop proceedings, published by Springer in the Lecture Notes in Computer Science series, which will be available at the workshop.

Workshop Web page:
http://www.cmap.polytechnique.fr/www.EvoGP/EuroGP98.html

Sample EvoGP *Web page*



References

[ABF97] Manu Ahluwalia, Larry Bell, and Terence C. Fogarty. Co-evolving functions in genetic programming: A comparison in ADF selection strategies. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 3–8, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[AK96] David Andre and John R. Koza. Parallel genetic programming: A scalable implementation using the transputer network architecture. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 16, pages 317–338. MIT Press, Cambridge, MA, USA, 1996.

[And94a] David Andre. Automatically defined features: The simultaneous evolution of 2-dimensional feature detectors and an algorithm for using them. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 23, pages 477–494. MIT Press, 1994.

[And94b] David Andre. Learning and upgrading rules for an OCR system using genetic programming. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[Ang94] Peter John Angeline. Genetic programming and emergent intelligence. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 4, pages 75–98. MIT Press, 1994.

[AP94] Martin Andrews and Richard Prager. Genetic programming for the acquisition of double auction market strategies. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 16, pages 355–368. MIT Press, 1994.

[ASS97] P. Aiyarak, A. S. Saket, and M. C. Sinclair. Genetic programming approaches for minimum cost topology optimisation of optical telecommunication networks. In *Second International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, GALESIA*, University of Strathclyde, Glasgow, UK, 1-4 September 1997. IEE.

[Ban93] Wolfgang Banzhaf. Genetic programming for pedestrians. In Stephanie Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93*, page 628, University of Illinois at Urbana-Champaign, 17-21 July 1993. Morgan Kaufmann.

[BFN96] Wolfgang Banzhaf, Frank D. Frankone, and Peter Nordin. The effect of extensive use of the mutation operator on generalization in genetic programming using sparse data sets. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature IV, Proceedings of the International Conference on Evolutionary Computation*, volume 1141 of *LNCS*, pages 300–309, Berlin, Germany, 22-26 September 1996. Springer Verlag.

[BGMT95] K. D. Bettenhausen, S. Gehlen, P. Marenbach, and H. Tolle. BioX++ – New results and conceptions concerning the intelligent control of biotechnological processes. In A. Munack and K. Schügerl, editors, *6th International Conference on Computer Applications in Biotechnology*, pages 324–327. Elsevier Science, 1995.

[BGS96] Walter Bohm and Andreas Geyer-Schulz. Exact uniform initialization for genetic programming. In Richard K. Belew and Michael Vose, editors, *Foundations of Genetic Algorithms IV*, University of San Diego, CA, USA, 3–5 August 1996. Morgan Kaufmann.

[Bli96] Tobias Blickle. Evolving compact solutions in genetic programming: A case study. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving From Nature IV. Proceedings of the International Conference on Evolutionary Computation*, volume 1141 of *LNCS*, pages 564–573, Berlin, Germany, 22-26 September 1996. Springer-Verlag.

[BNKF97] Wolfgang Banzhaf, Peter Nordin, Robert E. Keller, and Frank D. Francone. *Genetic Programming – An Introduction On the Automatic Evolution of Computer Programs and its Applications.* Morgan Kaufmann, dpunkt.verlag, November 1997.

[Bro91] Rodney A. Brooks. Intelligence without reason. Technical report, MIT AI Lab, Apr 1991. AI Memo no 1293, *Prepared for Computers and Thought.*

[BNO97] Wolfgang Banzhaf, Peter Nordin, and Markus Olmer. Generating adaptive behavior for a real robot using function regression within genetic programming. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 35–43, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[CY97] Shu-Heng Chen and Chia-Hsuan Yeh. Using genetic programming to model volatility in financial time series. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 58–63, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Daw86] Richard Dawkins. *The blind Watchmaker.* Harlow : Longman Scientific and Technical, 1986.

[DC97] Marco Dorigo and Marco Colombetti. *Robot Shaping: An Experiment in Behavior Engineering.* MIT Press/Bradford Books, 1997.

[DFP+94] Sumit Das, Terry Franguidakis, Michael Papka, Thomas A. DeFanti, and Daniel J. Sandin. A genetic programming application in virtual reality. In *Proceedings of the first IEEE Conference on Evolutionary Computation*, volume 1, pages 480–484, Orlando, Florida, USA, 27-29 June 1994. IEEE Press. Part of 1994 IEEE World Congress on Computational Intelligence, Orlando, Florida.

[D'h94] Patrik D'haeseleer. Context preserving crossover in genetic programming. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, volume 1, pages 256–261, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[EAS97] Anna I. Esparcia-Alcazar and Ken Sharman. Evolving recurrent neural network architectures by genetic programming. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 89–94, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[EEK+96] A.E. Eiben, T.J. Euverman, W. Kowalczyk, E. Peelen, F. Slisser and J.A.M. Wesseling. Comparing Adaptive and Traditional Techniques for Direct Marketing. In H.-J. Zimmermann, editor, *Proceedings of the 4th European Congress on Intelligent Techniques and Soft Computing*, pages 434–437, Verlag Mainz, Aachen, 1996.

[EEKS98] A.E. Eiben, T.J. Euverman, W. Kowalczyk and F. Slisser. Modelling Customer Retention with Statistical Techniques, Rough Data Models and Genetic Programming. In A. Skowron and S.K. Pal, editors, *Fuzzy Sets, Rough Sets and Decision Making Processes*, Springer-Verlag, Berlin, 1998 (in press).

[Eib97] Gusz Eiben. GP in leiden. electronic communication, 10 November 1997.

[FCC+97] I. De Falco, M. Conte, A. Della Cioppa, E. Tarantino, and G. Tautteur. Genetic programming estimates of kolmogorov complexity. In Eric Goodman, editor, *Genetic Algorithms: Proceedings of the Seventh International Conference*, Michigan State University, East Lansing, MI, USA, 19-23 July 1997. Morgan Kaufmann.

[GH97] Larry Gritz and James K. Hahn. Genetic programming evolution of controllers for 3-D character animation. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 139–146, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[GQ96] Frederic Gruau and Kameel Quatramaran. Cellular encoding for interactive evolutionary robotics. Cognitive Science Research Paper 425, School of Cognitive and Computing Sciences, University of Sussex, Falmer, Brighton, Sussex, UK, July 1996.

[GR97] Chris Gathercole and Peter Ross. Tackling the boolean even N parity problem with genetic programming and limited-error fitness. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 119–127, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Gru96] F. Gruau. Modular genetic neural networks for six-legged locomotion. In J.-M. Alliot, E. Lutton, E. Ronald, M. Schoenauer, and D. Snyers, editors, *Artificial Evolution*, volume 1063 of *LNCS*, pages 201–219. Springer Verlag, 1996.

[Han93] Simon Handley. Automatic learning of a detector for alpha-helices in protein sequences via genetic programming. In Stephanie Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93*, pages 271–278, University of Illinois at Urbana-Champaign, 17-21 July 1993. Morgan Kaufmann.

[Han94a] S. Handley. On the use of a directed acyclic graph to represent a population of computer programs. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, pages 154–159, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[Han94b] Simon G. Handley. The automatic generations of plans for a mobile robot via genetic programming with automatically defined functions. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 18, pages 391–407. MIT Press, 1994.

[HWSS95] Thomas Haynes, Roger Wainwright, Sandip Sen, and Dale Schoenefeld. Strongly typed genetic programming in evolving cooperation strategies. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pages 271–278, Pittsburgh, PA, USA, 15-19 July 1995. Morgan Kaufmann.

[Iba96] Hitoshi Iba. Emergent cooperation for multiple agents using genetic programming. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1996 Conference Stanford University July 28-31, 1996*, pages 66–74, Stanford University, CA, USA, 28–31 July 1996. Stanford Bookstore.

[IdS94] Hitoshi Iba, Hugo de Garis, and Taisuke Sato. Genetic programming using a minimum description length principle. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 12, pages 265–284. MIT Press, 1994.

[Ikr96] I. M. Ikram. An occam library for genetic programming on transputer networks. In Hamid R. Arabnia, editor, *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, pages 1186–1189, Sunnyvale, California, 9-11 August 1996. CSREA.

[Jac97] Christian Jacob. *Principia Evolvica - Simulierte Evolution mit Mathematica*. dpunkt.verlag, Heidelberg, Germany, August 1997.

[JMD94] Michael Patrick Johnson, Pattie Maes, and Trevor Darrell. Evolving visual routines. In Rodney A. Brooks and Pattie Maes, editors, *ARTIFICIAL LIFE IV, Proceedings of the fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 198–209, MIT, Cambridge, MA, USA, 6-8 July 1994. MIT Press.

[JP96] Hugues Juille and Jordan B. Pollack. Massively parallel genetic programming. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 17, pages 339–358. MIT Press, Cambridge, MA, USA, 1996.

[KA95] John R. Koza and David Andre. Evolution of both the architecture and the sequence of work-performing steps of a computer program using genetic programming with architecture-altering operations. In E. V. Siegel and J. R. Koza, editors, *Working Notes for the AAAI Symposium on Genetic Programming*, pages 50–60, MIT, Cambridge, MA, USA, 10–12 November 1995. AAAI.

[KBA+97] John R. Koza, Forrest H Bennett III, David Andre, Martin A. Keane, and Frank Dunlap. Automated synthesis of analog electrical circuits by means of genetic programming. *IEEE Transactions on Evolutionary Computation*, 1(2):109–128, July 1997.

[KBAK96] John R. Koza, Forrest H. Bennett III, David Andre, and Martin A. Keane. Automated WYWIWYG design of both the topology and component values of electrical circuits using genetic programming. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 123–131, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[KDD+97] John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors. *Genetic Programming 1997: Proceedings of the Second Annual Conference*, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Kei96] Maarten Keijzer. Efficiently representing populations in genetic programming. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 13, pages 259–278. MIT Press, Cambridge, MA, USA, 1996.

[Koz92] John R. Koza. *Genetic Programming: On the Programming of Computers by Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.

[Koz94] John R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge Massachusetts, May 1994.

[Koz98] John R. Koza. *Genetic Programming III*. MIT Press, Cambridge Massachusetts, 1998 (forthcoming).

[Lan98] W. B. Langdon. *Data Structures and Genetic Programming*. Kulwer, 1998. forth coming.

[LHL97] Wei-Po Lee, John Hallam, and Henrik Hautop Lund. Applying genetic programming to evolve behavior primitives and arbitrators for mobile robots. In *Proceedings of IEEE 4th International Conference on Evolutionary Computation*, volume 1. IEEE Press, 1997. to appear.

[Lon97] Tom Longshaw. Evolutionary learning of large grammars. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, page 445, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[LP97a] W. B. Langdon and R. Poli. An analysis of the MAX problem in genetic programming. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 222–230, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[LP97b] W. B. Langdon and R. Poli. Fitness causes bloat: Mutation. In John Koza, editor, *Late Breaking Papers at the GP-97 Conference*, pages 132–140, Stanford, CA, USA, 13-16 July 1997. Stanford Bookstore.

[LS96] Sean Luke and Lee Spector. Evolving teamwork and coordination with genetic programming. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 150–156, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[LT97] W. B. Langdon and P. C. Treleaven. Scheduling maintenance of electrical power transmission networks using genetic programming. In Kevin Warwick, Arthur Ekwue, and Raj Aggarwal, editors, *Artificial Intelligence Techniques in Power Systems*, chapter 10, pages 220–237. IEE, 1997.

[Mon95] David J. Montana. Strongly typed genetic programming. *Evolutionary Computation*, 3(2):199–230, 1995.

[MST95] Christian Miccio, Eduardo Sanchez, and Marco Tomassini. Parallel genetic programming induction of binary decision diagrams. Technical Report 7, Ecole Polytechnique Federal de Lausanne, EPFL, Switzerland, November 1995.

[NB95a] Peter Nordin and Wolfgang Banzhaf. Complexity compression and evolution. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pages 310–317, Pittsburgh, PA, USA, 15-19 July 1995. Morgan Kaufmann.

[NB95b] Peter Nordin and Wolfgang Banzhaf. Evolving turing-complete programs for a register machine with self-modifying code. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pages 318–325, Pittsburgh, PA, USA, 15-19 July 1995. Morgan Kaufmann.

[NB96] Peter Nordin and Wolfgang Banzhaf. Programmatic compression of images and sound. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 345–350, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[NFB96] Peter Nordin, Frank Francone, and Wolfgang Banzhaf. Explicitly defined introns and destructive crossover in genetic programming. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 6, pages 111–134. MIT Press, Cambridge, MA, USA, 1996.

[NH94] Thang Nguyen and Thomas Huang. Evolvable 3D modeling for model-based object recognition systems. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 22, pages 459–475. MIT Press, 1994.

[Nor94] Peter Nordin. A compiling genetic programming system that directly manipulates the machine code. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 14, pages 311–331. MIT Press, 1994.

[Oak94] Howard Oakley. Two scientific applications of genetic programming: Stack filters and non-linear equation fitting to chaotic data. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 17, pages 369–389. MIT Press, 1994.

[OCPT96] Mouloud Oussaidene, Bastien Chopard, Olivier V. Pictet, and Marco Tomassini. Parallel genetic programming: An application to trading models evolution. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 357–380, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[PC97] Riccardo Poli and Stefano Cagnoni. Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 269–277, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Per94] Tim Perkis. Stack-based genetic programming. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, volume 1, pages 148–153, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[PL97] Riccardo Poli and W. B. Langdon. A new schema theory for genetic programming with one-point crossover and point mutation. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 278–285, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Pol96a] R. Poli. Some steps towards a form of parallel distributed genetic programming. In *The 1st Online Workshop on Soft Computing (WSC1)*, http://www.bioele.nuee.nagoya-u.ac.jp/wsc1/, 19–30 August 1996. Nagoya University, Japan.

[Pol96b] Riccardo Poli. Evolution of recursive transistion networks for natural language recognition with parallel distributed genetic programming. Technical Report CSRP-96-19, School of Computer Science, University of Birmingham, B15 2TT, UK, December 1996. Presented at AISB-97 workshop on Evolutionary Computation.

[Pol96c] Riccardo Poli. Genetic programming for feature detection and image segmentation. In T. C. Fogarty, editor, *Evolutionary Computing*, number 1143 in Lecture Notes in Computer Science, pages 110–125. Springer-Verlag, University of Sussex, UK, 1-2 April 1996.

[Pol96d] Riccardo Poli. Parallel distributed genetic programming. Technical Report CSRP-96-15, School of Computer Science, University of Birmingham, B15 2TT, UK, September 1996. Submitted to Evolutionary Computation.

[Pol97] Riccardo Poli. Discovery of symbolic, neuro-symbolic and neural networks with parallel distributed genetic programming. In *3rd International Conference on Artificial Neural Networks and Genetic Algorithms, ICANNGA'97*, University of East Anglia, Norwich, UK, 1997.

[Qur96] Adil Qureshi. Evolving agents. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 369–374, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[RB96] Justinian P. Rosca and Dana H. Ballard. Discovery of subroutines in genetic programming. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 9, pages 177–202. MIT Press, Cambridge, MA, USA, 1996.

[RD94] Simon Raik and Bohdan Durnota. The evolution of sporting strategies. In Russel J. Stonier and Xing Huo Yu, editors, *Complex Systems: Mechanisms of Adaption*, pages 85–92. IOS Press, 1994.

[Rey92] Craig W. Reynolds. An evolved, vision-based behavioral model of coordinated group motion. In Meyer and Wilson, editors, *From Animals to Animats (Proceedings of Simulation of Adaptive Behaviour)*. MIT Press, 1992.

[Rey94a] Craig W. Reynolds. Evolution of obstacle avoidance behaviour:using noise to promote robust solutions. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 10, pages 221–241. MIT Press, 1994.

[Rey94b] Craig W. Reynolds. An evolved, vision-based behavioral model of obstacle avoidance behaviour. In Christopher G. Langton, editor, *Artificial Life III*, volume XVII of *SFI Studies in the Sciences of Complexity*, pages 327–346. Addison-Wesley, Santa Fe Institute, New Mexico, USA, 15-19 June 1992 1994.

[Rey96] Craig Reynolds. Boids. WWW home page, 12 August 1996. http://reality.sgi.com/craig/boids.html.

[RM95] Gerald Robinson and Paul McIlroy. Exploring some commercial applications of genetic programming. Project 4487, British Telecom, Systems Research Division, Martelsham, Ipswitch, UK, 9/3/95 1995.

[RW97] Conor Ryan and Paul Walsh. The evolution of provable parallel programs. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 295–302, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Rya94] Conor Ryan. Pygmies and civil servants. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 11, pages 243–263. MIT Press, 1994.

[SA94] Lee Spector and Adam Alpern. Criticism, culture, and the automatic generation of artworks. In *Proceedings of Twelfth National Conference on Artificial Intelligence*, pages 3–8, Seattle, Washington, USA, 1994. AAAI Press/MIT Press.

[Sie94] Eric V. Siegel. Competitively evolving decision trees against fixed training cases for natural language processing. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 19, pages 409–423. MIT Press, 1994.

[Sim91] K. Sims. panspermia. In Christopher G. Langton, editor, *Artificial Life II Video Proceedings*. Addison-Wesley, Sante Fe Institute, New Mexico, USA, February 1990 1991.

[SN97] Vanio Slavov and Nikolay I. Nikolaev. Inductive genetic programming and superposition of fitness landscapes. In Eric Goodman, editor, *Genetic Algorithms: Proceedings of the Seventh International Conference*, Michigan State University, East Lansing, MI, USA, 19-23 July 1997. Morgan Kaufmann.

[Spe94] Graham F. Spencer. Automatic generation of programs for crawling and walking. In Kenneth E. Kinnear, Jr., editor, *Advances in Genetic Programming*, chapter 15, pages 335–353. MIT Press, 1994.

[SS96] Kilian Stoffel and Lee Spector. High-performance, parallel, stack-based genetic programming. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 224–229, Stanford University, CA, USA, 28–31 July 1996. MIT Press.

[SS97] R. P. Salustowicz and J. Schmidhuber. Probabilistic incremental program evolution. *Evolutionary Computation*, 5(2):123–141, 1997.

[SSJ+96] Marc Schoenauer, Michele Sebag, Francois Jouve, Bertrand Lamy, and Habibou Maitournam. Evolutionary identification of macro-mechanical models. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 23, pages 467–488. MIT Press, Cambridge, MA, USA, 1996.

[TA97] Astro Teller and David Andre. Automatically choosing the number of fitness cases: The rational allocation of trials. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 321–328, Stanford University, CA, USA, 13-16 July 1997. Morgan Kaufmann.

[Tac93] Walter Alden Tackett. Genetic programming for feature discovery and image discrimination. In Stephanie Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93*, pages 303–309, University of Illinois at Urbana-Champaign, 17-21 July 1993. Morgan Kaufmann.

[Tel96] Astro Teller. Evolving programmers: The co-evolution of intelligent recombination operators. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 3, pages 45–68. MIT Press, Cambridge, MA, USA, 1996.

[Tet96] Andrea G. B. Tettamanzi. Genetic programming without fitness. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1996 Conference Stanford University July 28-31, 1996*, pages 193–195, Stanford University, CA, USA, 28–31 July 1996. Stanford Bookstore.

[TOD96] Turton, Openshaw, and Diplock. Some geographic applications of genetic programming on the cray T3D supercomputer. In Jesshope and Shafarenko, editors, *UK Parallel'96*. Springer, 1996.

[WC97] P. A. Whigham and P. F. Crapper. Applying genetic programming to model rainfall-runoff. CSIRO Land and Water, Canberra, Australia, October 1997.

[WHH+97] Mark Willis, Hugo Hiden, Mark Hinchliffe, Ben McKay, and Geoffrey W. Barton. Systems modelling using genetic programming. *Computers in Chemical Engineering*, 21:S1161–1166, 1997. Supplemental.

[ZKL96] Byoung-Tak Zhang, Ju-Hyun Kwak, and Chang-Hoon Lee. Building software agents for information filtering on the internet: A genetic programming approach. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1996 Conference Stanford University July 28-31, 1996*, page 196, Stanford University, CA, USA, 28–31 July 1996. Stanford Bookstore.

[ZM96] Byoung-Tak Zhang and Heinz Mühlenbein. Adaptive fitness functions for dynamic growing/pruning of program trees. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 12, pages 241–256. MIT Press, Cambridge, MA, USA, 1996.