The Fifth International Conference on Genetic Algorithms

Michael de la Maza

■ The Fifth International Conference on Genetic Algorithms was held at the University of Illinois at Urbana-Champaign from 17–21 July 1993. Approximately 350 participants attended the multitrack conference, which covered a wide range of topics, including genetic operators, mathematical analysis of genetic algorithms, parallel genetic algorithms, classifier systems, and genetic programming. This article highlights the major themes of the conference by discussing a few papers in detail.

The Fifth International Conference on Genetic Algorithms was hosted by the University of Illinois at Urbana-Champaign from 17–21 July 1993. The conference was organized by Stephanie Forrest (University of New Mexico, conference cochair and editor of the proceedings), David Goldberg (University of Illinois at Urbana-Champaign, conference cochair and local arrangements chair), and J. David Schaffer (Philips Laboratories, New York, conference cochair). Of the 240 papers submitted to the conference, 82 were accepted for oral presentation, and 37 were accepted for poster presentation. Three tutorials were arranged by Lawrence Davis (Tica Associates, Cambridge, Massachusetts, tutorial chair). The conference was supported by the Office of Naval Research, the Naval Research Laboratory, Philips Laboratory (North American Philips Corporation), and the International Society for Genetic Algorithms.

On the first day of the conference, three 4-hour tutorials were presented. The tutorial entitled "Introduction to Genetic Algorithms" featured presentations by Melanie Mitchell (Santa Fe Institute, New Mexico) and Davis. Darrell Whitley (Colorado State University) surveyed genetic algorithm theory in the "Advanced Genetic Algorithms Topics" tutorial, and Rob Smith (University of Alabama) gave an introduction to classifier systems. In the "Spotlighted Technologies" tutorial, David Fogel (Orincon Corp., San Diego, California) discussed evolutionary programming, Thomas Baeck (University of Dortmund) described evolution strategies, and John Koza (Stanford University) spoke on genetic programming. The following three days each began with a plenary session featuring an invited speaker. David Campbell (University of Illinois) spoke on dynamic systems and evolution. Gunter Wagner (Yale University) drew connections between genetic algorithms and the-

> *The most enthusiastic subgroup at the conference was composed of those interested in genetic programming.*

oretical results in evolutionary biology. John Holland (University of Michigan) described his recent work on a new type of genetic algorithm designed to counter hitchhiking. During these three days, the remainder of the time was divided into 1-1/2–hour sessions at which 3 or 4 papers were presented. In the morning of the fifth day, the final papers were presented, and in the afternoon, a business meeting of the International Society for Genetic Algorithms was held. The conference also featured a short poster session.

Because it would be impossible to review all the papers contained in the conference proceedings, this article presents my impressions of some of the major themes of the conference, including summaries of several papers to illustrate these themes.

The previous conference on genetic algorithms, held in 1991, was energized by an influx of German researchers who, for many years, had been studying programs similar to genetic algorithms. This time, the most enthusiastic subgroup at the conference was composed of those interested in genetic programming. *Genetic programming* makes an essential contribution to genetic algorithms: It describes how to evolve individuals with syntactically rich substructures. Simon Handley (Stanford University), who presented one of seven papers in the genetic programming track, described a genetic programming solution to the secondary-structure prediction problem, one of the outstanding problems in computational biology. The primary structure of a protein is the sequential list of amino acids that make up the protein. Groups of amino acids form units of secondary structure, called *alpha helices* and *beta sheets*. Many researchers have developed algorithms that use the primary structure of a protein to predict the secondary structure (alpha helix, beta sheet, or coil).

Handley considered a restricted version of this problem in which the amino acids are grouped into only two categories: alpha helix and coil (any amino acid that is not in an alpha helix). Although this two-state prediction is easier than the typical three-state prediction, Handley's algorithm is worse than three-state prediction algorithms developed half a decade ago. In addition, Handley failed to compare his results to those of Kneller, Cohen, and Langridge (1990), who also describe a two-state prediction algorithm and discuss work that is most directly comparable to Handley's.

These two shortcomings— mediocre results and an underappre-

ciation for the work done by other researchers—appear to permeate the genetic programming field. Peter Angeline (The Ohio State University), for example, in his talk describing work done by himself and Jordan Pollack, criticized traditional AI techniques and then went on to describe a genetic programming system that evolves a procedure that plays suboptimal tick tack toe. Because a simple procedure that uses a traditional AI method—search constrained by heuristics—plays optimal tick tack toe, Angeline's own results do not support his attack on traditional techniques. Handley's lack of familiarity with other relevant research is mirrored by the rest of the papers in the genetic programming section of the proceedings, all of which fail to cite work by N. L. Cramer (1985), who was the first to describe a genetic algorithm with tree-structured individuals.

Izzet Sirin described his work with H. Altay Güvenir (both with Bilkent University) on a novel concept-learning algorithm. Their technique fruitfully combines ideas from instance-based learning and genetic algorithms to produce a system that outperforms Aha, Kibler, and Albert's (1991) instance-based algorithms and Quinlan's (1993) c4.5 decision tree algorithm on a selected set of test problems. The first part of their method uses *classification-by-feature partitioning,* a technique that is similar to instance-based learning. However, instead of storing instances and classifying new instances using a distance metric, classification-by-feature partitioning processes the instances and stores partitions of the attributes. A partition simply consists of a lower and an upper bound on an attribute (all attributes must be numeric). A partition is extended when it does not correctly classify an instance. Güvenir and Sirin note that classification-by-feature partitioning overgeneralizes, and they address this problem by using a genetic algorithm to set the limits of each partition and the weight of each attribute. Their results indicate that their hybrid algorithm outperforms classification-by-feature partitioning alone. Unfortu-

nately, like many of the papers that described applications of genetic algorithms, this one failed to include any measure of statistical significance. However, if the results hold up to more rigorous scrutiny, this work will be regarded as a major contribution to the concept-learning literature.

Helen Cobb presented work, conducted with John Grefenstette (both with the Naval Research Laboratory), on the use of genetic algorithms to find the optima of nonstationary environments. They tested three mutation strategies on several types of changing environments. The three mutation strategies were *constant mutation,* as in a traditional genetic algorithm; *random immigrants,* which replaces a constant portion of the population with randomly generated individuals; and *triggered hypermutation,* which increases the mutation rate when it detects a decrease in the fitness of the best individual. On a stationary environment, all three strategies work well. However, in an environment that oscillates between two functions, the random-immigrant genetic algorithm performs best. In an environment in which the optimum is translated at periodic intervals, the hypermutation and standard genetic algorithms outperform the random-immigrant genetic algorithm. Cobb and Grefenstette concluded by suggesting that a hybrid mechanism might be able to capture the advantages of the three mechanisms without suffering from the disadvantages.

In the preface to the proceedings, Forrest expresses satisfaction with the high percentage of papers that describe applications of genetic algorithms. This reaction sharply contrasts with other conferences that cover similar areas, such as simulated annealing and tabu search, because they stress theory over application. Even more surprising, several papers that presented theoretical research, most of which gave more insight into how genetic algorithms work than the many papers that described applications of genetic algorithms, were not accepted for oral presentation. Of particular note is the work of Jenny

Juliany and Michael Vose (both of the University of Tennessee) on the genetic algorithm fractal. Vose has developed a model that facilitates the study of the convergence properties of the genetic algorithm. This model contains a compact description of the composition of a population and a function that explicitly describes how a population changes during a generation. In this paper, Juliany and Vose suggested a way of visualizing the characteristics of repeated iterations of this function. Taking their cue from the way that the Mandelbrot set is often depicted, they color points (each point exactly describes a population) that lead to quick convergence red and points that require many iterations before convergence blue. Intermediate convergence rates are given colors between red and blue. The resulting pictures are complex and suggest that the convergence rates of genetic algorithms should be studied by using the newly developed tools utilized to understand dynamic systems instead of using classical techniques.

The presentation of Juliany and Vose's work during the poster session, rather than the oral presentation sessions, suggests a certain lack of appreciation for genetic algorithm theory. Currently, genetic algorithm theory can be divided into three parts: (1) philosophical musings (De Jong 1993), (2) back-of-the-envelope calculations (the paper by David Goldberg, Kalyanmoy Deb, Hillol Kargupta, and Georges Harik [all of the University of Illinois, Illinois Genetic Algorithms Laboratory] in the proceedings is a good example), and (3) theorems (as in Vose and Liepins [1991]). If genetic algorithm theory is to become more sophisticated, then the third of these three components must be stressed, and just as important, a clear distinction between these three types of work must be maintained. The claim of Goldberg, Deb, Kargupta, and Harik that their results lead to "proofs similar to the probably approximately correct (PAC) results of computational learning theory," when in fact most of their derivations are not even rigorous, does a disservice to the field. Goldberg's

enormous stature—almost all the papers in the proceedings cite him—makes this transgression all the more acute.

Although the field of genetic algorithms is relatively young, the conference demonstrated that a large number of people have discovered the benefits of working with genetic algorithms. The first International Conference on Genetic Algorithms, held in 1985, was attended by 70 people. A simple regression shows that attendance at the conference has increased by approximately 50 percent each time, a fact that was first pointed out in 1989 by Stewart Wilson. The next conference is scheduled to be held in 1995 and promises to be the biggest gathering ever of genetic algorithm enthusiasts.

References

Aha, D.; Kibler, D.; and Albert, M. 1991. Instance-Based Learning Algorithms. *Machine Learning* 6:37–66.

Cramer, N. 1985. A Representation for the Adaptive Generation of Simple Sequential Programs. Presented at the International Conference on Genetic Algorithms and Their Applications, 24–26 July, Pittsburgh, Pennsylvania.

De Jong, K. 1993. Genetic Algorithms Are Not Function Optimizers. In *Foundations of Genetic Algorithms, 2,* ed. L. D. Whitley, 5–17. San Mateo, Calif.: Morgan Kaufmann.

Goldberg, D. E.; Deb, K.; Kargupta, H; and Harik, G. 1993. Rapid, Accurate Optimization of Difficult Problems Using Fast Messy Genetic Algorithms. In *Proceedings of the Fifth International Conference on Genetic Algorithms,* ed. S. Forrest, 56–64. San Mateo, Calif.: Morgan Kaufmann.

Kneller, D.; Cohen, F.; and Langridge, R. 1990. Improvements in Protein Secondary Structure Prediction by an Enhanced Neural Network. *Journal of Molecular Biology* 214:171–182.

Quinlan, J. 1993. *C4.5: Programs for Machine Learning.* San Mateo, Calif.: Morgan Kaufmann.

Vose, M., and Liepins, G. 1991. Punctuated Equilibria in Genetic Search. *Complex Systems* 5:31–44.

Michael de la Maza is a graduate student at the Massachusetts Institute of Technology, where he studies machine learning and optimization.

Books Received

The following books were received by press time:

■ *Artificial Intelligence in Perspective.* Cambridge, Mass.: The MIT Press, 1994. 462 pp. $35.00. ISBN 0-262-52186-5.

■ Anderson, Peter Bøgh. *The Computer as Medium.* New York: Cambridge University Press, 1994. 495 pp. $54.95. ISBN 0-521-41995-6.

■ Brien, Robert, and Eastmond, Nick. *Cognitive Science and Instruction.* Englewood Cliffs, N.J.: Educational Technology Publications, 1994. $34.95.

■ Brown, Robert Alan. *Machines That Learn.* New York: Oxford University Press, 1994. 890 pp. $75.00. ISBN 0-19-506966-8.

■ Charniak, Eugene. *Statistical Language Learning.* Cambridge, Mass.: The MIT Press, 1993. 170 pp. $25.00. ISBN 0-262-03216-3.

■ Durkin, John. *Expert Systems: Design and Development.* New York: Macmillan Publishing Company, 1994. 800 pp. ISBN 0-02-330970-9.

■ Faugeras, Olivier. *Three-Dimensional Computer Vision.* Cambridge, Mass.: The MIT Press, 1994. 663 pp. $65.00. ISBN 0-262-06158-9.

■ Forbus, Kenneth D., and de Kleer, Johan. *Building Problem Solvers.* Cambridge, Mass.: The MIT Press, 1994. 2 volumes. $49.95. ISBN 0-262-06157-0. $19.95. ISBN 0-262-56071-2 (listings).

■ Freedman, David. *Brainmakers.* New York: Simon & Schuster, 1994. 256 pp. $22.00. ISBN 0-671-76079-3.

■ Freuder, Eugene, and Mackworth, Alan K., editors. *Constraint-Based Reasoning.* Cambridge, Mass.: The MIT Press, 1994. 402 pp. $34.95. ISBN 0-262-56075-5.

■ Goldman, Alvin I. *Readings in Philosophy and Science.* Cambridge, Mass.: The MIT Press, 1994. 860 pp. $60.00. ISBN 0-262-57100-5.

■ Harris, Laurence, and Jenkin, Michael, editors. *Spatial Vision in Humans and Robots.* 448 pp. $59.95. ISBN 0-521-43071-2.

■ Krause, Paul, and Clark, Dominic. *Representing Uncertain Knowledge.* Oxford, England: Intellect, 1994. 277 pp. ISBN 1-871516-17-x.

■ KeBler, Christoph W. *Automatic Parallelization.* Wiesbaden, Germany: Vieweg, 1994. 222 pp. ISBN 3-528-05401-8.

■ Lawrence, Jeannette. *Introduction to Neural Networks.* Nevada City, Calif.: Cal Scientific Software, 1993. 324 pp. ISBN 1-883157-00-5.

■ Miller, Dale, ed. *Logic Programming: Proceedings of the 1993 International Symposium.* Cambridge, Mass.: The MIT Press, 1993. 686 pp. $75.00. ISBN 0-262-63152-0.

■ Newquist, Harvey P. *The Brain Makers.* Indianapolis, Ind.: Sams Publishing, 1994. 500 pp. $24.95. ISBN 0-672-30412-0.

■ O'Rourke, Joseph. *Computational Geometry in C.* 346 pp. $59.95. ISBN 0-521-44034-3 (cloth). $24.95. ISBN 0-521-44592-2 (paper).

■ Rips, Lance J. *The Psychology of Proof.* Cambridge, Mass.: The MIT Press, 1994. 449 pp. $45.00. ISBN 0-262-18153-3.

■ Young, Tzay, Y. *Handbook of Pattern Recognition and Image Processing: Computer Vision.* San Diego, Calif.: Academic Press, 1994. 584 pp. ISBN 0-12-774561-0.