From: AAAI Technical Report FS-00-04. Compilation copyright © 2000, AAAI (www.aaai.org). All rights reserved.

Human and Computer Learning Together In the Exteriorized Particle Swarm

James Kennedy
Bureau of Labor Statistics
2 Massachusetts Ave. NE
Washington, DC 20212
Kennedy_Jim@bls.gov

Abstract

The particle swarm is a computer algorithm for function optimization based on simulation of principles of human social behavior. Individual problem-solution vectors interact in a population until a criterion is met. The present experiment replaces one of the computational agents with a human being. It is shown that the human performs as well as the computational beings on an extremely simple problem, and worse than them on a harder one.

I. Introduction

The particle swarm optimization paradigm emerged from simulations of social influence in humans (e.g., Nowak, Szamrej, and Latané, 1990). Individuals in a computer program are given a problem and some random initial solutions; through interacting with one another in an iterative program, the population members are able to arrive at better and better problem solutions. Particle swarm implementations typically resemble evolutionary computation programs, and the are used to solve difficult mathematical and engineering programs (Yoshida, et. al, 1999, etc.).

Particles in the algorithm are not usually regarded as "agents." Their work is not specialized, and there is no real sense in which they could be said to be "autonomous." If there is an appearance of autonomous search of the problem space, it is because random numbers and a high degree of interaction between individuals cause each particle to follow a unique trajectory. It may turn out to be futile to try to distinguish "real" from apparent autonomy (in humans as well as beings in computer programs).

The word "swarm" is used to describe the movement of the population in the search space. Social influence brings individuals' trajectories into optimal regions of the space, where they oscillate around promising points, constantly updating their targets. The particle swarm differs from some other swarm paradigms, notably those based on ant colonies (Bonabeau, Dorigo, and Theraulaz, 1999), in an important way. In the ant paradigms, intelligence or successful adaptation occurs at the level of the population. No ant obtains the knowledge to solve the problem, rather it is found in the pattern of the entire population. In the particle swarm however each individual represents a complete problem solution. Individuals assess their neighbors' performances and imitate the most successful problem solution in their neighborhood by adjusting their trajectories toward a compromise of their own previous best solution and the best found by any neighbor. Implementations thus far have comprised a permanent sociometry, that is, individuals retain a neighborhood structure for the entire course of a trial.

Where particles in the paradigm are usually arrays in a computer program, there does not seem to be any reason not to substitute a live human for one of them. Given a problem, the solution to which is unknown both to the human and the computational entities, they should be able to interact as members of one population, to solve the problem. Such an algorithm is thus considered to be *exteriorized*, for lack of a better word: It is partly drawn out of the computer, into the physical world.

The introduction of a live actor to the population can raise a number of questions. For instance, is the solution to the problem more likely to be discovered by the person, or by an electronic individual? Will the person's strategies resemble those of the algorithm, or will they differ in some important or noticeable way? Will a population containing a human solve problems faster, or will the wetware entity slow things down? What is it like to be a human in an artificial society? The present paper gives a preliminary report on investigations into these questions.

II. The Particle Swarm

The particle swarm algorithm was introduced in a 1995 paper by Kennedy and Eberhart. The two original authors are an engineer (Eberhart) and a social psychologist (Kennedy), and the paradigm merged simulation as practiced in the social sciences and engineering methods such as evolutionary computation.

The algorithm initializes a population of problem solution vectors and assigns them random positional coordinates and velocities in the search space. Each individual evaluates its current problem solution, and if it is better than previous ones it remembers its current score on the problem and adjusts its velocity vector elements toward the new "best" solution: this is learning from experience. The second and more important effect is social learning, where individuals adjust their velocities towards the best solutions found by any of their neighbors.

The dichotomy of individual experience and social influence has been noted in a number of social-science theories (Ajzen and Fishbein, 1980; Deutsch and Gerard, 1955; Boyd and Richerson. 1985; Festinger, 1954/1999, etc.). In a particle swarm program computer program, algebraic terms representing these two phenomena are added to each element of each individual's velocity, causing the individual "particle" to oscillate around a region between its own previous best solution and the best found by any of its neighbors in a topological neighborhood.

Typically, individuals are programmed to interact with some subset of the population. Within these neighborhoods individuals interact repeatedly throughout the iterations of the program. Neighborhoods can be structured in many ways (Kennedy, 1999), though it is most common simply to let each individual interact with its nearest topological neighbor in the array.

Especially when problems are multimodal, it is typical to note the emergence of cultures within a population. Neighbors become more similar to one another as the program iterates, with subpopulations clustering in optimal regions of the search space. Cultures themselves allow exploitative search of prospectively good regions of the search space, while the diversity of cultures permits simultaneous, parallel explorational search for alternative solutions.

The particle swarm in pseudo code is given in Appendix 1. The algorithm has been shown to perform very well on standard optimization testbed problems and numerous engineering applications.

III. Introducing a Person to the Paradigm

The rule for moving the particles was devised to be an approximation of human cognitive tendencies. To include a live person in the process, it would not seem reasonable to attempt to artificially induce an actual human to simulate the simulation. Instead, in the exteriorized particle swarm the population inside the computer is augmented with a human actor in the outside world who is allowed to interact freely and naturally with the electronic agents. The person is shown the most successful solutions proposed by other members of the population, and is allowed at each iteration to propose one of his or her own. The person in other words receives the same information from a computational particle that particles receive from one another and produces the same kind of output. The human in the system simply produces whatever seems to be a good response, based on their own judgment, and the program iterates until a criterion is met.

A program was written which displayed a graphical field on the screen, with the human's and the human's neighbors' best positions indicated by dots. The first problem implemented in the program was a very simple one; the population attempted to minimize the sphere function $f(\vec{x}) = \sum x_i^2$ in two dimensions. In order to confound the human, a random vector of offsets was added to the optimal vector, keeping the optimum away from the center of the box.

The second problem was more difficult. The Rastrigin function $f(\vec{x}) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2p\ x_i) + 10 \right]$ is a standard optimization testbed problem that features numerous local optima. A hill-climbing algorithm, for instance, tends to get trapped on local optima, typically failing to discover the global optimum. Again the function was offset so the location of the global optimum would not be apparent on the screen. Both functions are graphed in Appendix 2.

In each case, the problem was to find the point on the screen whose coordinates resulted in the smallest function output. Computational individuals searched according to the algorithm given above, while the human simply clicked the mouse with the cursor over the selected points in a graphic box on the screen, using whatever strategy seemed most appropriate. The person made one choice for each iteration of the artifical agents' choices. The program allowed the user to zoom and scroll, in case the search area became too large or small or went off the visible screen.

Populations of six individuals, including the human, were implemented. Search continued until a criterion was met. For the sphere, this was 1.0, and for Rastrigin the criterion was 10.0; both functions were implemented in two dimensions to accommodate display on the computer screen, and particles were initialized in the interval [-50, +50]. Dependent variables included the number of iterations required to reach the criterion, and whether the criterion-beating score was obtained by the human or by one of the artificial particles.

Two experimental conditions were tested for each test function. In the first condition, the human was able to see the best points found by all members of the population. In the second, only two points –the human's best and the best point found by any member of the population – were shown on the screen. If the human's best was the population best, only one point was shown. In either condition, a black circle was drawn around the population best, and a red circle around the human's previous best point. Two human subjects participated in the experiment; each conducted ten trials in each condition for each test function. Ten trials were also run on each function with a comparable non-exteriorized version of the algorithm.

IV. Results

The first table compares the average number of iterations required to meet the criterion for the two functions in the three experimental conditions.

Adding a human to the population did not improve performance. In fact, even though the human performed quite successfully on optimization of the sphere function, a purely computational population solved it faster. The human-in-the-loop implementation performed relatively well on Rastrigin, though as can be seen, most of the time it was a computerized individual who found the best solution.

Table 1. Mean number of iterations to criterion. "All" = human saw all population members' previous bests, "i&g" = human saw only own and population's best. "Computer" = condition with no human. Averaged over 20 trials per cell.

	Sphere	**Rastrigin**
All	13.8	24.0
i&g	13.9	18.7
Computer	10.9	22.4

In the second table it is seen that the human succeeded at finding the optimum a higher proportion of the time than the computational agents on the sphere function, but not on the Rastrigin.

Table 2. Proportion of the times that the optimum was found by the human user, as opposed to a digital population member.

	Sphere	**Rastrigin**
All	0.55	0.40
i&g	0.7	0.35

As discussed below, some strategies were available for the sphere function that did not work on the Rastrigin, especially ones that capitalized on the monotonicity of the function. By exploiting these, the human user was able to find the criterion most of the time on the simpler function. Such strategies were not available though for the Rastrigin function, because the user is never sure that the global optimum, or one that meets the criterion, has been found.

V. Phenomenology: What It's Like to Be a Particle

The particle swarm algorithm was intended to be a simulation of human sociocognitive behavior, and the present embellishment is meant to underscore that analogy by inserting an actual human into the population. The argument has been that particles are like minds. But to a human, the experience of having a mind involves subjectivity, thinking is more than information processing, it is conscious experience as well. What is it like to find oneself participating in a society of computational simpletons? The present section employs first-person narrative to describe the phenomenology of a human participant in the particle swarm. This departure from the norm of scientific literature may say more about the author than about the experimental paradigm itself, but this report would seem incomplete without an introspective statement. I describe here my own experience with the exteriorized particle swarm.

I found that the attempt to find the optimum evoked in me an immediate feeling of competition with the electronic individuals. I could have felt relief or elation when one of the them found a good point. Instead, I tended to be disappointed that one of them, and not I, had found it.

On the other hand, when I found a better point than theirs, it was quietly satisfying to watch them attempt to imitate me. It was sometimes disconcerting, in fact, to see just how fast they emulated my discovery and bettered it.

In the company of humans, there is a similar glow in knowing that others have found your wit worth copying, your comments worth repeating; it is pleasurable to lead and be followed (though a good follower tactfully allows some separation between the initial event and its replication). The universal stratification of human societies tends to evoke theoretical assumptions about drives for power; seeing the effect of this "drive for power" in the present paradigm though makes one question whether this might be better thought to be a *social drive for improvement*. Individuals may experience a desire to influence others, to have power over them, but the real result is a constant push upward in performance of the group, what Tomasello (1999) calls the "ratchet effect." This appears to be one of many cases (Leary's analysis of self-esteem as a sociometer is another prominent one) (Leary, Tambor, Terdal, and Downs, 1995) where the experience of being human blinds one to a functional view of human behavior. Scientific study of human behavior requires us to consider the effects of a behavioral tendency; what it feels like to perform the behavior is only a biological distraction.

Though both the population best and the human's best were marked with circles, it sometimes happened that the best performer in the population was in fact the human, and there was only one circled point. This produced a very discomforting feeling, as I realized that the only clues available to me were ones that I created myself.

VI. Rational Strategies

There were several decent strategies for optimizing these functions. The choice of them is influenced by knowledge about the function. For instance, to know that you are optimizing the sphere function is to know that wherever a good point has been found, a better point is near it on one side or another. Thus the "clicking nearby" strategy eventually works. With the Rastrigin function though the participant does not know if the best optimum found so far is a very good one, or if a better one exists somewhere else. Thus the "clicking far away" strategy sometimes discovered better points than those found so far.

Knowledge about the evaluations of points that had been found also influenced the strategic choice. For instance, if the criterion for Rastrigin was 10.0, and the best point found so far was evaluated at 10.05, then clicking nearby would likely find a point on the same hill that met the criterion. When the best evaluation was, for instance, 450.0, then random clicking worked almost as well.

On the Rastrigin function, is was often useful to investigate regions about points that were not the population best. That is, if a particle had discovered a good point, but not the best, then it might be that it was resting on the side of an optimum, and not at the peak. Therefore better points could be found in its immediate vicinity.

On the sphere function, it was possible to use the distance and direction between the population best and the human best points to pick a next point. Depending on their relative evaluations, the user could extrapolate, clicking on a point beyond the better of the two known points, or interpolate, clicking between them.

Knowledge of the direction of recent improvement could also be used on the sphere function. That is, it is possible to remember, over time, the direction from early good points to later ones. This information, which is not available to the computational particles, gives some clues about which direction one might look to find the global optimum.

Despite this arsenal of sensible strategies, finely tuned human intelligence, brilliant ideas, profound rationality, in most cases the best bet was to imitate the others' good performance, which is just what the particles did.

An important aspect of participating in the particle swarm involves the planning and plotting of rational strategies. The sphere function is the simplest test function imaginable; it would seem that a reasonably intelligent person should be able to figure out which direction the successful searches pointed, and make an educated guess about where the optimum would be found. As can be seen in the experimental results, this strategy was successful more than half the time, as the human participants were usually the ones to beat the criterion. But the victory was modest: Even on this simplest of functions the computational particles were first to exceed the criterion nearly half the time when all previous best locations were displayed, and nearly a third of the time when only the human's and the population's bests were shown.

The Rastrigin function was even worse. In that function it is possible to move with continual improvement, and yet be "barking up the wrong tree," so to speak; that is, a subpopulation may begin to converge on a local optimum while a better site exists in some other corner of the space.

This is interestingly similar to other kinds of blind search, for instance the search for scientific truths. Some facts are found, verified, and publicly announced to the research community; members of that community then test facts related to the verified ones, and adjust their theories to be more similar to the ones that produce successful results, and so on. All of them are proceeding without knowledge of what their next experiment will find, and without knowledge of what the ultimate scientific truth will be. And of course, generally only successes are reported in scientific journals and meetings. The participants in the process of science are simply, blindly building on their own and one another's successes.

VII. Implications

Among human society, there seems to be a universal tendency, sometimes interpreted as hypocrisy, to present oneself positively, concealing one's embarrassing errors and awkwardnesses while displaying and even exaggerating one's accomplishments (Goffman, 1959; Schlenker, 1982; Baumeister, 1982 . This is just what the particles do in the particle swarm algorithm. Members of the population announce their own and attend to others' previous successes; none are influenced by poor or mediocre choices. This suggests a functional benefit from biased self-presentations: Positive self-presentation gives others information about best positions in the large search space of human behaviors. In the particle swarm algorithm, *most* problem-solving attempts are ignored by the rest of the population; only the best ones influence others, and their memory is retained until they are outdone.

VIII. Conclusions

The particle swarm algorithm was developed to mimic social behaviors of live organisms, including humans. Simulated individuals represented as vectors in a search space interact "socially" with one another to find answers to hard mathematical problems. The present study replaced one of the simulated individuals with a real person, situated outside the computer environment.

It is not uncommon in other paradigms for a human user to participate in computational optimization processes by giving suggested starting points, guiding search away from known local optima, etc. *Shared-initiative processing* is usually considered to be a way to supplement the search algorithm with human knowledge and information processing strategies. In the present case though the human did not always turn out to be the smart one. As in many real-world situations, the optimum was not known ahead of time. The human in the loop did have information about the nature of the test function, while the electronic participants only knew about previous best solutions; this did not turn out to be an advantage, however.

It has been theorized that humans optimize their cognitive structures through a process involving learning from experience and social influence. Neither of these suggest any great amount of rational reasoning or the application of well-designed strategies for searching through the problem space, though the phenomenological experience for the human comprises mainly these kinds of processes. We feel we are constantly trying to "figure out" and "understand" what needs to be done to find a good solution – but in fact computationally intelligent individuals who do not try to figure out anything actually perform as well as an intelligent human agent.

We humans experience cognition as a process largely composed of logical and clear thinking, sequences of narrative and imagery wherein we analyze information and make decisions based on rational decisions. Psychologists though have had very little luck in discovering evidence of consistent patterns of reasoning in human research subjects. One result has been the "anti-introspectionist" tradition in social psychology heralded by Nisbett and Wilson (1977). That tradition of research accepts that people have little if any access to their cognitive processes, and argues that we are much more likely to give plausible-sounding explanations for events than correct ones.

In the exteriorized particle swarm, it was seen that rational search strategies could produce correct answers when the problem to solve was very simple, for instance, a two-dimensional sphere function. Hill-climbing and gradient-following, using information from peers' progress, were about as successful as simple social influence by itself. When a more complex function was attempted however, e.g., the Rastrigin function, the social conformity algorithm by itself was superior to "rational" search; the human participant did no better than a simple computational being.

We have to think that most problems in the world are complex. Monotonic smooth unimodal functions such as the sphere are the exception. Social, cognitive, and environmental problems tend to have more than one right answer. Besides that, it is not uncommon for "real" problems to be deceptive, that is, performance degrades as the optimum is approached, leading a "rational" incremental searcher away from the best answers. Cliffs and contradictions in the search space make cognitive problems more similar to Rastrigin than to the sphere.

Individuals in the particle swarm exhibit an extremely simple type of behavior which has been distilled from observations of human sociality. They explore around their previous successes and around the successes of their neighbors. They make no pretense of rationality or reasonableness; they do not "think" in the ordinary sense of the word. Their thinking rather comprises simple *evaluation* of their own mental states and the states of their peers; *comparison* of their own states and of their neighbors' states with previous successes, and *imitation* of previous successes (own and others'). These three kinds of phenomena – evaluation, comparison, and imitation – are easily demonstrated in human beings, in contrast to loftier phenomena such as rational cogitation, which are rarely seen.

References

Ajzen, I, and Fishbein, M. (1980). *Understanding Attitudes and Predicting Social Behavior*. Englewood Cliffs, NJ: Prentice-Hall.

Baumeister, R. F. (1982). A self-presentational view of social phenomena. *Psychological Bulletin, 91*, 3-26.

Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford: Oxford University Press.

Boyd, R. and P. J. Richerson (1985). *Culture and the Evolutionary Process*. Chicago: University of Chicago Press.

De Jong, K. A. (1975). An analysis of the behavior of a class of genetic adaptive systems. Doctoral dissertation, University of Michigan.

De Jong, K., Potter, M., and Spears, W. (1997). Using problem generators to explore the effects of epistasis. In T. Bäck, (Ed.), *Proceedings of the Seventh International Conference on Genetic Algorithms*, 338-345. Morgan Kaufmann.

Deutsch, M., and Gerard, H. B. (1955). A study of normative and informational social influences upon individual judgment. *Journal of Abnormal and Social Psychology, 51*, 629-636.

Eberhart, R. C., P. K. Simpson, and R. W. Dobbins (1996). *Computational Intelligence PC Tools*. Boston, MA: Academic Press Professional.

Festinger, L. (1954/1999). Social communication and cognition: A very preliminary and highly tentative draft. In Harmon-Jones, E., and J. Mills (Eds.), *Cognitive Dissonance: Progress on a Pivotal Theory in Social Psychology*. Washington DC: AP Publishing.

Goffman, E. (1959). *The Presentation of Self in Everyday Life*. Garden City NJ: Doubleday.

Kennedy, J. (1999). Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance. *Proceedings of the 1999 Congress on Evolutionary Computation*, 1931-1938. Piscataway, NJ: IEEE Publishing.

Kennedy, J. and Eberhart, R.C. (1995). Particle swarm optimization. *Proceedings of the 1995 International Conference on Evolutionary Computation*, Perth, Western Australia.

Kennedy, J., and Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. *Pro-*

ceedings of the 1997 Conference on Systems, Man, and Cybernetics, 4104-4109. IEEE Service Center, Piscataway, NJ.

Kennedy, J., and Spears, W. M. (1998). Matching algorithms to problems: An experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. *Proceedings of the 1998 International Conference on Evolutionary Computation*, 78-83. IEEE Service Center, Piscataway, NJ.

Leary, M. R., Tambor, E. S., Terdal, S. K., & Downs, D. L. (1995). Self-esteem as an interpersonal monitor: The sociometer hypothesis. *Journal of Personality and Social Psychology, 68*, 518-530.

Nisbett, R. E., and Wilson, D. W. (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review, 84*, 231-259.

Nowak, A., Szamrej, J., and Latané, B. (1990). From private attitude to public opinion: A dynamic theory of social impact. *Psychological Review, 97*, 362-376.

Schlenker, B. R. (1982). Translating actions with attitudes: an identity analytic approach to the explanation of social conduct. In L. Berkowitz (Ed.), *Advances in Experimental Social Psychology, Vol. 15*. New York: Academic Press.

Yoshida, H., K. Kawata, Y. Fukuyama, and Y. Nakanishi, A particle swarm optimization for reactive power and voltage control considering voltage stability. *Proc. Intelligent System Application to Power Systems*, G. L. Torres and A. P. Alves da Silva, Eds., Rio de Janeiro, Brazil, 1999, pp. 117-121.

Appendix 1: The Particle Swarm in Pseudocode

Loop
For i = 1 to number of individuals
If $G(\vec{x}_i) > G(\vec{p}_i)$ then do //G() evaluates goodness
For d = 1 to dimensions
$p_{id} = x_{id}$ //p_{id} is best so far
Next d
End do
$g = i$ //arbitrary
For j = indexes of neighbors
If $G(\vec{p}_j) > G(\vec{p}_g)$ then g=j //g is index of best performer in the neighborhood
Next j

For d = 1 to number of dimensions
$v_i(t) = c\,(v_{id}(t-1) +$
$\quad j_1(p_{id} - x_{id}(t-1))$
$\quad + j_2(p_{gd} - x_{id}(t-1)))$
$x_{id} = x_{id} + v_{id}$
Next d
Next i
Until criterion

where x_{id} is the position of individual i in the d-dimensional parameter space, v_{id} is i's velocity, and p_{id} is the best point found by i so far. The coefficient χ controls the convergence of the population; in the current studies it is set to approximately 0.729, and φ_1=φ_2=2.05.

Appendix 2: The Test Functions

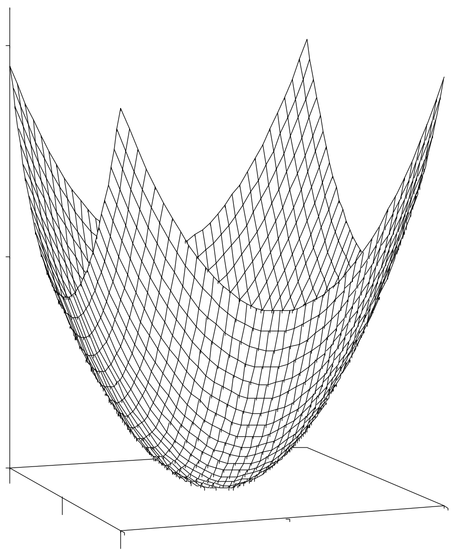

Figure 1. The Sphere Function

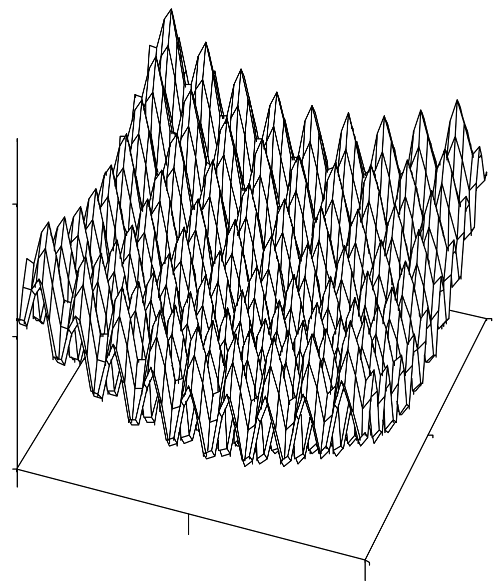

Figure 2. The Rastrigin Function