Available online at www.sciencedirect.com
ScienceDirect
Journal of Cleaner Production 15 (2007) 902–906
Journal of Cleaner Production
www.elsevier.com/locate/jclepro

Hybrid evolutionary computation for the development of pollution prevention and control strategies

Raymond R. Tan*

Chemical Engineering Department, De La Salle University-Manila, 2401 Taft Avenue, 1004 Manila, Philippines

Received 7 December 2005; accepted 23 January 2006
Available online 20 March 2006

Abstract

Particle swarm optimization (PSO) is an evolutionary algorithm based on the behavior of social animals. Its key advantage is its computational efficiency compared to related techniques such as genetic algorithm (GA). Use of a modified PSO algorithm in selecting an optimal array of pollution prevention techniques for clay brick production is described. The model is formulated as a multi-constraint knapsack optimization problem. The optimization technique used in the study is a binary PSO augmented with a GA-based mutation operator.

Keywords: Swarm intelligence; Particle swarm optimization; Pollution prevention; Cleaner production

1. Introduction

Pollution prevention or cleaner production is defined as the use of fundamental process or product changes to reduce the emission of direct or indirect pollutants, as opposed to the application of technologies to contain or modify wastes after they have already been formed [1]. Pollution prevention can be achieved through the use of the first two methods in the general hierarchy of waste management approaches:

- Reduction at source;
- Recycle and reuse;
- Treatment;
- Disposal.

In contrast, approaches for the treatment and disposal of wastes are referred to as pollution control techniques. Effective waste management in industry requires the selection of an appropriate, economical and effective set of technologies for a given problem. This results in a combinatorial optimization problem that must consider factors such as performance, cost and technological compatibility. Computational tools and mathematical models can be utilized to facilitate the selection of integrated pollution prevention and control strategies. In its simplest form, this selection process can be formulated as a binary programming model similar to the basic multiple-constraint knapsack problem.

A number of optimization techniques based on biological systems have been developed [2]. These include genetic algorithms (GA), ant colony optimization (ACO), shuffled frog leaping optimization and particle swarm optimization (PSO). PSO is an algorithm developed by Kennedy and Eberhart [3]. The technique is based on the social behavior of animals and was first used for optimizing unconstrained optimization problems with continuous variables.

The basic PSO algorithm consists of velocity update and position update steps. The velocity of each particle is updated using

$$\mathbf{V}(t+1) = a\mathbf{V}(t) + b\mathrm{Rand1}[\mathrm{pbest} - \mathbf{X}(t)] + c\mathrm{Rand2}[\mathrm{gbest} - \mathbf{X}(t)] \quad (1)$$

where $\mathbf{V}(t+1)$ is the updated velocity vector, $\mathbf{V}(t)$, the current velocity vector, $\mathbf{X}(t)$, the current position vector, pbest, the best solution achieved by particle to date, gbest, the best global

* Tel.: +63 2 536 0257; fax: +63 2 524 0563.
E-mail address: tanr_a@dlsu.edu.ph

0959-6526/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jclepro.2006.01.011

solution achieved to date, Rand1, the random number in [0, 1], Rand2, the random number in [0, 1], a, the momentum parameter, b, the self-influence parameter, and c the swarm influence parameter.

The parameters pbest and gbest represent the individual and collective memory of the best location achieved to date. In particular, gbest is defined as the best location found by any particle in the swarm up to the present time. Each particle adjusts its motion on the basis of momentum, self-influence and swarm influence. The latter two influences are modulated by two random numbers, which give the algorithm a "stochastic kick." These random influences prevent the algorithm from being trapped in local optima, and are analogous to probabilistic operators in other evolutionary algorithms (e.g., mutation and cross-over in genetic algorithms and the Metropolis criterion in simulated annealing). In the original PSO [3], the parameters used were $a = 1$, $b = 2$ and $c = 2$. After the velocity update, the particle's position is updated as follows:

$$\mathbf{X}(t+1) = \mathbf{X}(t) + \mathbf{V}(t+1) \tag{2}$$

where $\mathbf{X}(t+1)$ is the updated position vector.

Note that $\mathbf{V}$ and $\mathbf{X}$ are n-dimensional vectors where n is the number of variables in the model to be solved.

PSO cannot deal explicitly with constrained optimization problems. Penalty functions can be used to convert constrained models into unconstrained ones. For integer variables, additional rules must be used to discretize continuous variable values [4]. In the case of binary variables, the following discretization rule can be used:

$$x_i^* = \begin{cases} 1 & \text{if} \quad xi > \beta \quad \forall i \\ 0 & \text{if} \quad xi \le \beta \end{cases} \tag{3}$$

where x_i* is the discretized value of variable x_i and β, the cut-off for discretization.

In this study, the value $\beta = 0.5$ was used initially. The resulting discretized vector is a string of binary digits similar to a GA chromosome.

Initial tests with the basic binary PSO revealed difficulties in finding the global solutions in binary optimization problems. This effect was due to the limited number of feasible points in the design space. There was a high likelihood that the particles in the swarm would occupy the same position in the design space. Once this happened, the algorithm would cease to make any further progress and thus converge to a false solution. Tuning the PSO algorithm by adjusting the values of a, b and c gave only marginal improvements in the rate of successful convergence. To overcome this shortcoming, the algorithm was modified by incorporating a mutation operator similar to that used in GA. This mutation operator assigns a fixed probability that the value of each binary variable will be reversed [5]. If the calculated updated position according to Eqs. (2) and (3) is 1, the mutation operator converts it to 0, and vice versa. The mutation operator occurs at a predefined, small probability value. The resulting modified algorithm is effectively a PSO–GA hybrid. The principal advantage is that mutation allows the algorithm to progress even when the particles inadvertently converge to a common point.

The pseudocode of the hybrid algorithm is:

```
Begin
 Generate N random particles
 Calculate objective function for each particle
 For each particle:
   Set pbest as the best position of particle to date
   If objective function value is improved then update pbest
 End
 Set gbest as the best position by any particle to date
 For each particle;
   Update particle velocity using Eq. 1
   Update particle position using Eqs. 2 and 3
   Mutate particle position at a specified probability of
   occurrence
 End
Check if the termination criterion is satisfied
End
```

During the development, the robustness of the algorithm was established by testing it on a number of optimization problems. The case study in the next section demonstrates the use of the new computational procedure for a specific application. It is intended to be an illustrative example that represents a general class of combinatorial optimization problems in pollution prevention and control.

2. Case study

Use of PSO in pollution prevention is demonstrated with a case study from Kantardgi et al. [6] on the use of binary programming to select appropriate methods for reducing hydrogen fluoride emissions in the production of clay bricks. The choice of application was based mainly on the availability of well-documented and rigorously reviewed technical and economic data [7]. During brick firing, naturally occurring fluorine compounds in the clay volatilize and are released into the kiln, and eventually into the atmosphere. The process schematic is shown in Fig. 1. Three general schemes for reducing these fluoride emissions have been identified:

- various process modifications that inhibit the release of fluorides from the clay, or that encourage re-absorption of the fluorides into the final product.
- chemical additives such as limestone that react with the fluorides to entrap them in the fired brick.
- end-of-pipe control technologies such as scrubbers to remove fluorides from the flue gas.

Table 1 shows the technical performance and cost data for nine different pollution prevention and control techniques. It is assumed that simultaneous use of two or more technologies results in cumulative reduction of emissions as described below. For example, since the third and fourth technology options each reduce emissions by 25% and 60%, respectively, their

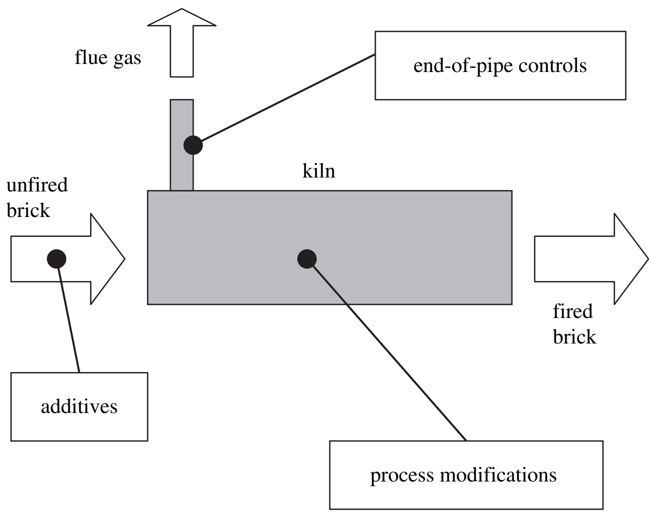

Fig. 1. Brick firing process diagram (adapted from Ref. [2]).

combined use reduces emissions by 70%. Every kilogram of baseline emissions is reduced (by 25%) to 0.75 kg by applying the third alternative. This quantity is further reduced (by 60%) to 0.3 kg by applying the next technological option. Hence the cumulative reduction is from 1 kg to 0.3 kg, or 70%. Note that the cumulative reduction is multiplicative; the total emissions remaining relative to the baseline are simply the product of the fraction of emissions remaining after the application of each alternative measure:

$$\left(1 - R_{\mathrm{cum}}\right) = \prod_i (1 - R_i) \tag{4}$$

where R_i is the fractional reduction of emissions from alternative (i), and R_{cum}, the total or cumulative fractional reduction of emissions.

Eq. (4) assumes that the individual performance of each option is unaffected by the combined use of multiple technologies. Furthermore, this relationship can be converted to additive form through logarithmic transformation:

Table 1
Performance and cost characteristics of pollution prevention and control options

Technology option	Variable	Emission reduction (%)	Capital cost (10^3 US$)
Modifying the time–temperature profile	x_1	45	37.8
Reducing the airflow through the kiln	x_2	55	151.2
Increasing the turbulence in the preheat zone	x_3	25	189
Increasing the interaction between the product and the flue gas	x_4	60	94.5
Utilizing the flue gas	x_5	45	3.78
Using fluoride reactive additives	x_6	40	189
Dry bed limestone scrubber	x_7	95	945
Dry cloth filter scrubber	x_8	95	1323
Wet scrubber	x_9	95	1323

$$\log(1 - R_{\mathrm{cum}}) = \sum_i \log(1 - R_i) \tag{5}$$

In this case study, it is assumed that the objective is to find the least expensive means of reducing emissions by at least 90% through the use of one or more techniques. This problem can be modeled as a binary programming problem of the form

$$\min \sum_i A_i x_i \tag{6}$$

subject to

$$\sum_i B_{ij} x_i \le \mathrm{C}_j \forall j$$
$$\mathbf{x}_i \,\varepsilon \{1,0\} \forall i$$

where x_i is the binary decision variable for technology option (i), A_i, the economic objective coefficient for technology option (i), B_{ij}, the technology coefficient of option (i) for constraint (j), and C_j, the technological or performance constraint (j).

Each binary variable denotes the choice to use ($x_i = 1$) or not to use ($x_i = 0$) a particular technology option. The objective function is a simple polynomial equation for capital cost; hence the coefficients A_i are in monetary units. It is assumed that the total cost incurred for plant retrofit is the sum of the costs of the individual measures or alternatives selected.

The model constraint coefficients in this example are dimensionless, although in general they may not always be so; the only restriction is that the units of B_{ij} and C_j should be consistent. Constraints include the target fluoride emissions reduction (90%) as well as logical conditions indicating technological incompatibilities among the nine techniques [6,7]. This model is a multi-constraint knapsack problem, which is known to be NP-hard; solution of the model can be found by using evolutionary-based algorithms such as PSO or GA [8]. The model parameters are given in Table 2.

Each coefficient for $j = 1$ is based on the logarithm of the fraction of emissions remaining after implementation of a particular reduction measure. For example, the first technology option reduces emissions by 45%. Thus the coefficient $B_{1,1}$

Table 2
Model coefficients for the case study

	A_i	B_{ij}	
		$j = 1$	2
$i = 1$	37.8	−0.26	0
2	151.2	−0.35	0
3	189	−0.12	0
4	94.5	−0.40	0
5	3.78	−0.26	0
6	189	−0.22	0
7	945	−1.3	1
8	1323	−1.3	1
9	1323	−1.3	1
C_j	–	−1	1

used in the additive model is $\log_{10}(1-0.45)$ or -0.26. This transformation is necessary to allow the cumulative emissions to be expressed additively as a linear polynomial as in Eq. (6). Since the total targeted emissions reduction is 90%, the coefficient $C_1 = \log_{10}(1-0.9)$ or -1. The model coefficients for $j = 2$ indicate incompatibilities among the alternatives—for example, in this case only one type of scrubber can be used at a time. Hence, the coefficients $B_{7,2}$, $B_{8,2}$ and $B_{9,2}$ are all equal to 1. Since only one of these three options will be chosen, C_2 is also 1. Alternatives 1−6 are not considered in this constraint and their corresponding coefficients are all zero.

The model can be converted to unconstrained form using penalty functions to allow the use of PSO

$$\min \sum_i \mathrm{A}_i x_i + \mu \sum_j \alpha_j \left[\sum_i B_{ij}\, x_i - C_j \right]^2 \tag{7}$$

subject to

$$x_i \varepsilon \{1, 0\}\ \forall i$$

where

$$\alpha_j = \begin{cases} 0 & \text{if } B_{ij}x_i - C_j \le 0 \\ 1 & \text{if } B_{ij}x_i - C_j > 0 \end{cases}$$

μ = penalty constant

The constant μ is a large positive number and α_j is a unit step function so that the penalty applies to only one side of an inequality constraint j. In this study, the value $\mu = 10^{10}$ was used. The penalty term artificially inflates the value of the objective function whenever a constraint is violated. The PSO algorithm was implemented using a Visual Basic program using a swarm of 10 particles. Details of the parameters used in the case study are shown in Table 3. The lowest capital cost for achieving the target 90% reduction is US$249,480, which requires the combined use of the following techniques:

- modifying the time−temperature profile (x_2);
- increasing the interaction between the product and the flue gas (x_4); and
- utilizing the flue gas (x_5).

Table 3
Case study parameters

Parameter	Value
Number of particles	10
Momentum coefficient	1
Self-influence coefficient	2
Swarm influence coefficient	2
Maximum number of iterations	250
Penalty coefficient	10^{10}
Number of variables	9
Probability of mutation	0.1

Note that these results indicate that the target 90% reduction in fluoride emissions can be achieved at low costs without resorting to the use of end-of-pipe treatment. The optimization algorithm converged to the correct solution in an average of 24 ± 18 iterations. Note that, with a swarm of 10 particles, this means that the solution was reached after evaluating the function an average of 240 (or 24×10) times. In comparison, the same solution can be reached only by evaluating all 512 (or 2^9) possible combinations of x_i. Hence PSO required about half as many function evaluations as compared to listing down all possible combinations of alternatives. This result agrees with prior work that showed the computational efficiency of PSO [3,7]. Fig. 2 shows the evolution of the objective function value for one particular test run, with the correct solution being reached on the ninth iteration.

Further tests were run at different emission reduction targets ranging from 10 to 99% to determine the sensitivity of the solution to the emission reduction goal. The optimal solutions for the target levels are shown in Table 4. Each row indicates which alternatives should be utilized to achieve the lowest capital cost of retrofits capable of meeting the emission targets. Note that the actual reductions achieved, shown in the last column, may exceed the target values. Alternative x_5 (utilizing the flue gas) appears in all the optimal solutions because of its relatively low cost. It is also notable that no scrubbers are called for with target reductions of up to 95%. At this level of emissions reduction, alternatives x_3 (increasing turbulence in the preheat zone) and x_6 (using fluoride reactive additives) may substitute for each other as they have similar performance and identical cost characteristics. Use of the latter gives slightly higher reduction of emissions although both options meet the 95% target. The dry bed limestone scrubber is required to reduce emissions by 98% or more.

3. Conclusions

Binary PSO has been shown to be effective in developing optimal pollution prevention strategies. Significant improvement in the rate of successful convergence was achieved by incorporating a mutation operator after the position update step.

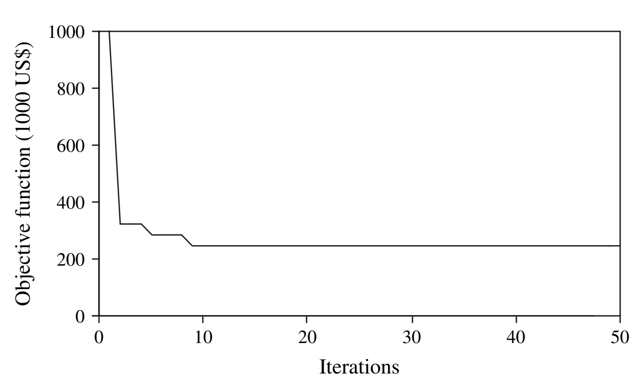

Fig. 2. Example of objective function evolution.

Table 4
Optimal solutions for different emission reduction targets

Target reduction	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	Capital cost (10^3 US$)	Actual reduction
10%	0	0	0	0	1	0	0	0	0	3.78	45.0%
20%	0	0	0	0	1	0	0	0	0	3.78	45.0%
30%	0	0	0	0	1	0	0	0	0	3.78	45.0%
40%	0	0	0	0	1	0	0	0	0	3.78	45.0%
50%	1	0	0	0	1	0	0	0	0	41.58	69.8%
60%	1	0	0	0	1	0	0	0	0	41.58	69.8%
70%	0	0	0	1	1	0	0	0	0	98.28	78.1%
80%	1	0	0	1	1	0	0	0	0	136.08	88.0%
90%	0	1	0	1	1	0	0	0	0	249.48	90.2%
95%	1	1	1/0	1	1	0/1	0	0	0	476.28	95.9–96.8%
98%	1	0	0	0	1	0	1	0	0	986.58	98.5%
99%	1	0	0	1	1	0	1	0	0	1081.08	99.4%

This modification allows the algorithm to continue to make progress even after the particles converge to the same coordinates in the design space. Compared to the basic PSO algorithm, the modified procedure had a significantly reduced propensity to converge prematurely to suboptimal solutions. The technique seems particularly suited to medium-sized and large problems, where it was found to be more computationally efficient than simple enumeration of all potential combinations of alternatives.

Further studies on the use of the computational technique shown here will have to focus on models incorporating more variables and constraints. Uncertainties in cost or performance data are also bound to be encountered in real-world problems, which will require the use of probabilistic or fuzzy formulation of the model [8]. A potential variant combining PSO with Monte Carlo may be developed in the future to handle problems wherein the model coefficients are defined statistically through probability distributions, instead of being assumed to be fixed values. These results also suggest that PSO has considerable potential for solving MINLP problems encountered in life cycle analysis, process integration and industrial ecology. Parameter tuning studies for specific types of models will be necessary to ensure reliable convergence. Dynamic or adaptive PSO variants should be tested; furthermore, multiple particle classes as originally suggested by Kennedy and Eberhart [7] can be used to balance the algorithm's exploration and exploitation tradeoff. In many cases, it will also be necessary to make use of multiple objective PSO (MOPSO) based on Pareto dominance rough sets or fuzzy sets to balance conflicting technological, economic and environmental targets.

Acknowledgements

The author thanks Dr. Alvin Culaba, Professor Mike Purvis, Dr. Elmer Dadios, Dominic Foo, Dennis Benghui and Dennis Cruz for their inputs.

References

[1] Asian Development Bank. Guidelines for policy integration and strategic action planning for the achievement of cleaner production; 2002.
[2] Elbeltagi E, Hegazy T, Grierson D. Comparison among five evolutionary-based optimization algorithms. Applied Engineering Informatics 2005;19: 43–53.
[3] Kennedy J, Eberhart RH. Particle swarm optimization. Proceedings of the IEEE conference on neural networks IV, Piscataway, NJ, USA; 1995.
[4] Eberhart R, Shi Y, Kennedy J. Swarm intelligence. San Diego: Academic Press; 2001.
[5] Goldberg DE. Genetic algorithms in search, optimization and machine learning. Reading: Addison-Wesley; 1989.
[6] Kantardgi I, Purvis MRI, Cherviakov L, Khudoshina M. Approaches to the modeling of energy utilization in product life cycles. Proceedings of the second conference on humanoid, nanotechnology, information technology, communications and control, environment and management, Manila, Philippines; 2005.
[7] Department of Environment, Transportation and the Regions. Reducing fluoride emissions in brick, tile and pipe manufacture. Environmental Technology Best Practice Programme GG166, United Kingdom; 1999.
[8] Lin FT, Yao JS. Using fuzzy numbers in kanpsack problems. European Journal of Operational Research 2001;135:158–76.