Guide Objective Assisted Particle Swarm Optimization and Its Application to History Matching

Alan P. Reynolds[1], Asaad Abdollahzadeh[2], David W. Corne[1], Mike Christie[2], Brian Davies[3], and Glyn Williams[3]

[1] School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, Scotland EH14 4AS
[2] Institute of Petroleum Engineering, Heriot-Watt University
[3] BP

Abstract. As is typical of metaheuristic optimization algorithms, particle swarm optimization is guided solely by the objective function. However, experience with separable and roughly separable problems suggests that, for subsets of the decision variables, the use of alternative 'guide objectives' may result in improved performance. This paper describes how, through the use of such guide objectives, simple problem domain knowledge may be incorporated into particle swarm optimization and illustrates how such an approach can be applied to both academic optimization problems and a real-world optimization problem from the domain of petroleum engineering.

1 Introduction

This paper describes a version of particle swarm optimization (PSO) that uses 'guide objectives' in addition to the overall objective in order to improve performance, in particular when the problem is 'roughly' separable. This introduction briefly describes the real-world problem that motivated this work, in order to give the reader an idea of what is meant by guide objectives and rough separability.

To be able to make effective decisions regarding the exploitation of an oil reservoir, it is necessary to create and update reservoir models. Initial models created using geological knowledge of the reservoir are improved using observations collected over time, in a process called *history matching*. This involves the adjustment of a reservoir model so that, when simulation software is applied, the simulated behaviour is similar to that observed in the real world. This can be posed as an optimization problem, minimizing a measure of misfit.

While we would like to automate the history matching process, incorporating reservoir experts' extensive domain knowledge into metaheuristic optimization algorithms in a generally applicable way has proved difficult. The avenue of research explored in this paper starts with the realization that, given a suitable model parameterization, certain model parameters will affect certain components of the misfit function to a greater degree than others. Indeed, if the reservoir

C.A. Coello Coello et al. (Eds.): PPSN 2012, Part II, LNCS 7492, pp. 195–204, 2012.

consists of distinct regions with little inter-region communication and if the model parameters describe regional features then the history matching problem may be (roughly) separated into a number of smaller, regional subproblems. This suggests that, given a suitable subset of model parameters, it may be possible to select a subset of the misfit components to create a *guide objective* for these parameters. We will show that, when using PSO, these guide objectives may be used in combination with the overall objective in a single optimization run.

In Sect. 2 we describe how metaheuristics perform when applied naively to separable problems and define more precisely what we mean by 'roughly separable' and 'guide objective'. Section 3 describes basic PSO, while Sect. 4 describes how PSO may be modified to exploit guide objectives, producing the guide objective assisted PSO algorithm (GuPSO). Results on simple function optimization problems are provided in Sect. 5.

In Sect. 6, reservoir history matching is described in more detail, including a description of the PUNQ-S3 case study used in this paper. Application of GuPSO to this problem and results obtained are described in Sect. 7. Finally, Sect. 8 presents conclusions and a discussion of potential areas of further research.

2 Optimization and Separable Problems

Consider the minimization of $f(x, y)$, where x and y can take any of a thousand values and nothing is known a priori about f. To guarantee finding the optimal solution one must evaluate all one million solutions. However, if it is known that $f(x, y) = g(x) + h(y)$ then the optimal solution can be found in two thousand evaluations by first optimizing the choice of x and then optimizing the choice of y. Now suppose a metaheuristic is naively applied to the minimization of f. A solution with the optimal value of x may be evaluated early in the search, but if it is coupled with a poor choice for y its significance will be missed. Clearly, knowledge of the problem's separability should be exploited to improve performance, typically by optimizing $g(x)$ and $h(y)$ separately.

Note that x and y may be vectors representing subsets of the decision variables. Also, it may not be obvious when this approach may be used. Suppose we wish to minimize $f(x, y) = x^4 + 2x^2y^2 + y^4$. We may not separate the problem directly, but we note that $f(x, y) = \left(x^2 + y^2\right)^2$ and that, since $x^2 + y^2 \geq 0$, this is equivalent to minimizing $x^2 + y^2$ — a clearly separable problem.

Now suppose we wish to minimize $f(x, y) = x^4 + 2x^2y^2 + y^4 + \epsilon x^3 y$, where ϵ is small. The problem may no longer be separated as above. However, the additional term may have limited impact on the quality of solutions: the problem may be thought of as being *roughly* separable. Minimizing $g(x) = x^2$ and $h(y) = y^2$ separately still leads to good values for $f(x, y)$. Therefore it makes sense to start the search by minimizing $g(x)$ and $h(y)$, rapidly finding a near optimal solution, before improving the result by optimizing f directly if desired.

In what follows, we will describe $g(x)$ and $h(y)$ as the *guide objectives* for x and y. These objectives are used to guide our search for good values for x and y and aid in the optimization of $f(x, y)$. So when minimizing $f(x, y) = x^4 + 2x^2y^2 + y^4(+\epsilon x^3 y)$, we use $g(x) = x^2$ and $h(y) = y^2$ as guide objectives.

3 Particle Swarm Optimization

To describe GuPSO we must first describe the basic PSO algorithm. PSO [7] is motivated by the collective behaviour of animals, such as the flocking of birds or swarming of bees. However, instead of a swarm of bees searching for a good source of nectar, PSO uses a swarm of particles moving through a multidimensional search space towards better quality solutions.

Each particle in the swarm has a position and a velocity, initialized at random. In each iteration of PSO, the velocity of each particle is adjusted by applying an acceleration towards the best solution visited by the particle in question and an acceleration towards the best solution visited by the swarm. The position is then adjusted according to the particle's velocity. In detail, if x_{ij} is the jth component of the position of particle i and v_{ij} is the jth component of its velocity, then these are updated as follows:

$$v_{ij} \leftarrow wv_{ij} + \alpha r_1 \left(p_{ij} - x_{ij}\right) + \beta r_2 \left(g_j - x_{ij}\right) \quad , \tag{1}$$

$$v_{ij} \leftarrow \min\left(v_{ij}, V_{\max,j}\right) \quad ,$$

$$v_{ij} \leftarrow \max\left(v_{ij}, -V_{\max,j}\right) \quad ,$$

$$x_{ij} \leftarrow x_{ij} + v_{ij} \quad .$$

Here w is the inertia weight, α and β control the amount of acceleration towards the particle's personal best and the global best solutions, r_1 and r_2 are randomly generated numbers between 0 and 1, p_{ij} is the jth component of the best solution visited by particle i and g_j is the jth component of the best solution visited by the swarm. $V_{\max,j}$ is the maximum velocity permitted in dimension j. Values for w, α and β are supplied by the user.

PSO may also use methods for ensuring that decision variables remain within their permitted bounds. In this paper we apply reflection with random damping. However, since boundary handling is unaffected by the use of guide objectives, we do not provide details but refer the reader to the PSO literature.

4 PSO and Guide Objectives

Now suppose we wish to apply PSO to a separable problem, for example the minimization of $f(x_1, x_2, x_3, x_4) = f_1(x_1) + f_2(x_2, x_3, x_4)$. We have suggested above that two separate optimizations should take place — the minimization of f_1 and the minimization of f_2. However, both these optimizations can be performed concurrently by adjusting the velocity update formula as follows:

$$v_{ij} \leftarrow wv_{ij} + \alpha r_1 \left(p_{ij}^{(j)} - x_{ij}\right) + \beta r_2 \left(g_j^{(j)} - x_{ij}\right) \quad . \tag{2}$$

Here $g^{(j)}$ represents the best solution visited by the swarm according to the guide objective for the jth variable, while $p_i^{(j)}$ is the best solution visited by particle i according to the guide objective for variable j. f_1 acts as the guide objective for x_1 while f_2 acts as the guide objective for x_2, x_3 and x_4. Notice how this separates the optimization so that values taken by x_2, x_3 and x_4 have no affect on the choices for variable x_1 - the evolution of x_1 depends only on the values taken by its guide objective f_1, which is unaffected by the other parameters.

Merging two independent optimizations into a single run in this manner does not produce any immediate benefits in the case of separable problems. The advantage of the approach is that it allows for both guide objectives and the true objective to be used when the problem is only roughly separable, via the combination of update formulae (1) and (2) as follows:

$$v_{ij} \leftarrow wv_{ij} + \alpha r_1 (p_{ij} - x_{ij}) + \beta r_2 (g_j - x_{ij}) + \gamma r_3 \left(p_{ij}^{(j)} - x_{ij}\right) + \delta r_4 \left(g_j^{(j)} - x_{ij}\right) \quad . \tag{3}$$

The selection of appropriate values for α, β, γ and δ allows the influence of the guide objectives on the search to be controlled. By changing the values of these parameters during the search, the algorithm may start by using only the guide objectives, but become increasingly influenced by the true objective until finally it behaves like standard PSO. This approach may be effective for roughly separable problems, where guide objectives are used to rapidly finding good solutions but where the final refinements can only be made with reference to the true objective. The resultant algorithm is Guided PSO or GuPSO.

5 Illustrative Results on Academic Problems

The operation of GuPSO on separable or roughly separable problems is best illustrated on academic problems. In this section the objective is always minimized and variables are constrained to lie between -10 and 10. We focus on variations of two functions: the multimodal function of Kvasnika et al. [8] and Rosenbrock's function [13]. The first of these is totally separable and is given by

$$f_1 (x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} g(x_i) \quad ,$$

$$g(x) = 0.993851231 + e^{-0.01x^2} \sin(10x) \cos(8x) \quad .$$

f_1 is used as our first test function, with $n = 20$ and an evaluation limit of 50,000. The guide objective for each variable, x_i, is simply $g(x_i)$.

Rosenbrock's function, given by

$$r(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n-1} \left[(1 - x_i)^2 + 100 \left(x_{i+1} - x_i^2\right)^2 \right]$$

is inseparable. Our second test function is created by splitting fifty decision variables into ten equal sized blocks, summing 10 five variable Rosenbrock functions:

$$f_2 (x_1, x_2, \ldots, x_{50}) = \sum_{j=1}^{10} r(x_{5j-4}, \ldots, x_{5j}) \quad .$$

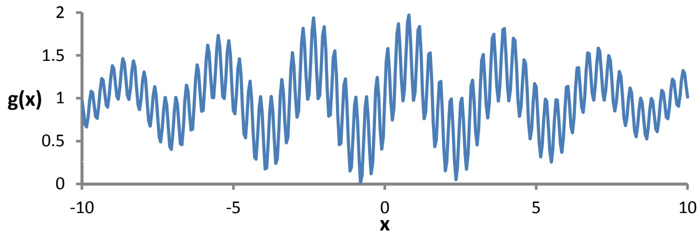


Fig. 1. The highly multimodal function $g(x)$

A limit of 500,000 evaluations is imposed. The guide objective for each variable is simply the Rosenbrock function to which the variable makes a contribution.

A third test function combines f_1 with a twenty variable Rosenbrock function:

$$f_3(x_1, x_2, \ldots, x_{20}) = f_1(x_1, x_2, \ldots, x_{20}) + 0.001h(x_1, x_2, \ldots, x_{20}) .$$

The result can be thought of as being roughly separable. An evaluation limit of 50,000 is imposed. The guide objective for x_i is simply $g(x_i)$, i.e. the contribution of the Rosenbrock function is ignored.

Experiments using the parameter values in table 1 were performed for each problem. Parameters α, β, γ and δ in (3) were set as follows:

$$\alpha = \beta = (1 - \lambda)A, \;\; \gamma = \delta = \lambda A \; . \tag{4}$$

Here λ indicates the degree to which the guide objectives were used in preference to the overall objective. For the two separable functions, λ took the values 0 or 1. For f_3, values of 0, 0.2, 0.5, 0.8 and 1 were tried for λ. Experiments were also performed with λ decreasing linearly from 1.0 to 0.0 over the course of each run.

For each value of λ, thirty runs were performed for every combination of the remaining parameters, in order to find the best values. Thirty runs were then repeated using the best parameter set, allowing for a fair comparison between PSO ($\lambda = 0$) and GuPSO. Results are summarized in Table 2.

It is clear that, on the two separable problems, use of the guide objectives produces significantly better results. Indeed for f_1, using guide objectives resulted in the global optimum being found in all 30 runs, while it was never found using the true objective. However, as has been noted, identical results could be achieved by separating the problem into 20 sub-problems and optimizing each individually using standard PSO.

Table 1. Parameter values

Parameter	Values
Swarm size	10, 20, 50, 100
Inertia weight	0.75, 0.8, 0.85, 0.9, 0.95, 1.0
Acceleration (A)	0.5, 0.8, 1.0, 2.0

Table 2. Comparison of performance using just the true objective against using just the guide objectives. Figures in brackets indicate 95% confidence intervals.

Problem	True objective ($\lambda = 0$)	Guide objectives ($\lambda = 1$)
f_1	3.535 (3.190 – 3.880)	1.138×10^{-9} (1.138×10^{-9} – 1.138×10^{-9})
f_2	14.05 (12.06 – 16.03)	1.703 (0.785 – 2.622)
f_3	5.467 (4.939 – 5.996)	3.696 (3.687 – 3.707)

Results for f_3 also show significant improvements when using the guide objectives. However, the best results were only obtained when both guide objectives and the true objective were used to guide the search, as shown in Fig. 2.

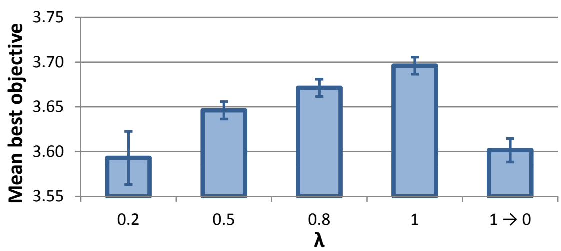


Fig. 2. Results for the third test function. Results obtained using just the true objective are of considerably poorer quality than those shown and are omitted to improve clarity. Error bars show the 95% confidence intervals.

6 The History Matching Problem

History matching, or petroleum reservoir model calibration, is the process of modifying a reservoir model so as to produce simulated outputs that closely match pressure, production and saturation data collected from a real world reservoir. Model parameters that may be modified include rock porosity, vertical and horizontal permeability, pore volume and aquifer volume. The reservoir is divided into regions or layers within which these factors can be assumed to be approximately constant. The location of the boundary between such regions may also be considered a modifiable model parameter. Furthermore, it may be appropriate to adjust various multipliers, rather than the physical characteristics directly.

The objective function is a measure of misfit. Although this paper focuses primarily on simply minimizing misfit, it is useful to obtain a range of different, low misfit models. The resulting ensemble of reservoir models can then be used not only to predict future output, but also to estimate the uncertainty of the prediction — a process known as *uncertainty quantification*.

A number of metaheuristics have been applied to the history matching problem, including simulated annealing [14], tabu search [15], genetic algorithms [12,5], estimation of distribution algorithms [11,3,2] and differential evolution [6]. Recent work has also suggested that PSO may be effectively applied to this

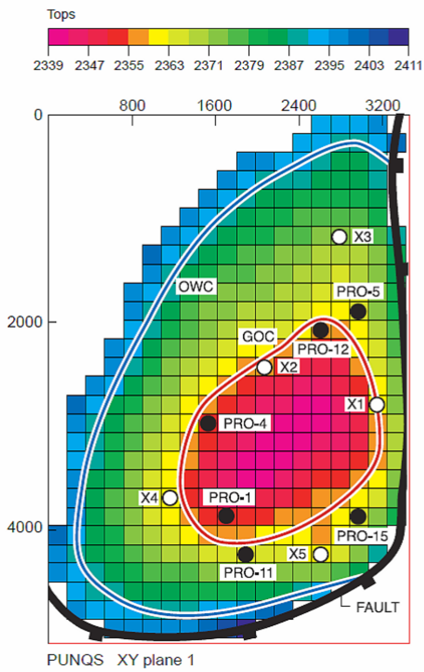


Fig. 3. The depth of reservoir rock and well locations in PUNQ-S3

problem [9,10]. This and the relative ease with which guide objectives may be incorporated provides the motivation for our use of PSO.

6.1 PUNQ-S3

We apply GuPSO to the history matching of the PUNQ-S3 reservoir [1] — a small industrial reservoir engineering model, adapted from a real field example and widely used for performance studies of history matching algorithms.

Problem: The simulation model contains 2660 (19x28x5) grid blocks, of which 1761 are active, and 6 production wells, numbered 1, 4, 5, 11, 12, and 15. The field is structurally bounded to the east and south by a fault, as shown in Fig. 3, while the link to a strong aquifer to the north and west means that no injection wells are required. The field initially has a small gas cap at the center of the structure, and production wells are located around this gas cap.

Porosity and permeability fields for the 'truth case' were generated using a Gaussian Random Fields model in such a way as to be, as much as possible, consistent with the geological model. The reservoir model was completed by using pressure, volume, temperature and aquifer data from the original model. Reservoir simulation was then used to generate production data (bottom-hole pressure

BHP, water cut WCT and gas oil ratio GOR), after which Gaussian noise was added to the well porosities/permeabilities and the synthetic production data to account for measurement error. (For further details see [1].)

Model Parameters and Objective: In this paper we use the parameterization of Hajizadeh et al. [6], with distinct porosity values in each of 9 homogeneous regions (labelled A to I) per layer for 5 layers, resulting in 45 model parameters. Parameter ranges and other details can be found in [6].

The objective function, to be minimized [4] is

$$M = \frac{1}{N_v} \sum_{j=1}^{N_v} \frac{1}{N_p} \sum_{i=1}^{N_p} W_{ij} \left(\frac{O_{ij} - S_{ij}}{\sigma_{ij}} \right)^2$$

where i runs over the time points at which observations are made, j indicates which of the 18 observations (BHP, WCT and GOR at each of the 6 wells) is being referred to, O_{ij} is the truth case value of observation j at time i, S_{ij} is the simulated value, σ_{ij} reflects the measurement error and W_{ij} is a weight factor.

7 Application to PUNQ-S3

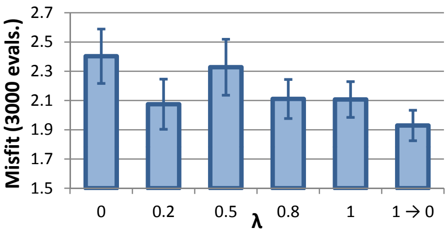
Given a set of porosity parameters for a region, the wells that are primarily affected by these parameters and only marginally affected by the others may be selected. The misfit components for these wells then form the guide objective for these model parameters, as indicated in table 3.

Table 3. Guide objectives for PUNQ-S3 were taken to be the misfit over the set of wells most affected by model parameter in question

Region	A	B	C	D	E	F	G	H	I
Guide wells	5	5, 12	5, 12	5, 12	4, 5, 12	1, 4, 15	1, 4, 11, 15	1, 11, 15	1, 11

Despite the cost of solution evaluation, the basic PSO was tuned by experimenting with a range of swarm sizes (10, 20 and 50) and inertia weights (0.8, 0.85, 0.9, 0.95 and 1). The acceleration parameter A (and hence α and β in basic PSO) was set to one. For each combination of parameters, 30 runs were performed, of 3000 solution evaluations each. The PSO results that are compared with GuPSO in this paper were then obtained by performing additional runs of 3000 evaluations and 1000 evaluations with the best parameter combination.

Results for GuPSO were obtained using the best parameter set found for PSO, with the exception that α, β, γ and δ were set according to (4) using a range of values for λ. It can be seen from the results in Fig. 4 that GuPSO outperforms standard PSO, particularly in shorter runs.



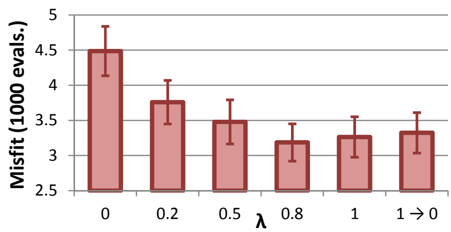


Fig. 4. GuPSO performance for different values of λ after 3000 and 1000 solution evaluations, averaged over 30 runs. Standard PSO is provided by setting λ to 0, while the last bar gives the results obtained by allowing λ to vary from 1 (guide objectives only) to 0 (true objective only) linearly over the course of the search.

8 Conclusions and Further Research

We have presented a modification to PSO whereby guide objectives are utilized in order to improve algorithm performance. The resulting algorithm has been shown to produce improved performance on some simple separable and roughly separable problems. More importantly, GuPSO outperforms standard PSO on a real-world reservoir history matching problem.

There are a number of areas of possible future research.

Other sources of guide objectives: Much of this paper assumes that guide objectives are found via the rough separability of the problem. However, *any* alternative objective that provides a better guide for the improvement of decision variables than the true objective could be used in this approach.

Multiple guide objectives: The approach need not be limited to a single guide objective for each decision variable. Multiple guide objectives may be used, either at different points in the search or through further modification of the velocity update formula.

Forgetfulness: History matching problems may be roughly separable, with the exception of one or two model parameters that affect the entire reservoir. GuPSO may remember a best solution for one guide objective that depends upon old, long discarded values for the 'global' parameters. It may be useful to allow GuPSO to 'forget' such solutions.

Other applications: In particular, GuPSO may be an appropriate approach to reservoir development optimization. When locating new wells, predicted oil recovery from the well should make a suitable guide objective for the well location. However, since placement of each well affects the output of both old wells and the other new wells, the problem is only roughly separable.

References

1. PUNQ-S3 test case. Department of Earth Science and Engineering. Imperial College, London (2010), `http://www3.imperial.ac.uk/earthscienceandengineering/research/perm/punq-s3model` (retrieved January 7, 2010)

2. Abdollahzadeh, A., Reynolds, A., Christie, M., Corne, D., Davies, B., Williams, G.: Bayesian optimization algorithm applied to uncertainty quantification. In: SPE EUROPEC/EAGE Annual Conf. and Exhibition, Vienna, Austria (2011)
3. Abdollahzadeh, A., Reynolds, A., Christie, M., Corne, D., Williams, G., Davies, B.: Estimation of distribution algorithms applied to history matching. In: SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A. (2011)
4. Barker, J.W., Cuypers, M., Holden, L.: Quantifying uncertainty in production forecasts: Another look at the PUNQ-S3 problem. SPE Journal 6(4), 433–441 (2001)
5. Erbaş, D., Christie, M.A.: Effect of sampling strategies on prediction uncertainty estimation. In: SPE Reservoir Simulation Symposium, Houston, U.S.A. (2007)
6. Hajizadeh, Y., Christie, M., Demyanov, V.: Comparative study of novel population-based optimization algorithms for history matching and uncertainty quantification: PUNQ-S3 revisited. In: Abu Dhabi Int. Petroleum Exhibition and Conf. (2010)
7. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. of the IEEE Int. Conf. on Neural Networks, vol. 4, pp. 1942–1948 (1995)
8. Kvasnicka, V., Pelikan, M., Pospichal, J.: Hill climbing with learning (an abstraction of genetic algorithm). Neural Network World 6(5), 773–796 (1995)
9. Mohamed, L., Christie, M.A., Demyanov, V.: Comparison of stochastic sampling algorithms for uncertainty quantification. SPE Journal 15(1), 31–38 (2010)
10. Mohammed, L., Christie, M., Demyanov, V.: Reservoir model history matching with particle swarms. In: SPE Oil and Gas India Conf. and Exhibition, Mumbai, India (2010)
11. Petrovska, I., Carter, J.N.: Estimation of distribution algorithms for history-matching. In: Proc. of the 10th European Conf. on the Mathematics of Oil Recovery. EAGE, Amsterdam (2006)
12. Romero, C.E., Carter, J.N., Gringarten, A.C., Zimmerman, R.W.: A modified genetic algorithm for reservoir characterisation. In: Int. Oil and Gas Conf. and Exhibition, Beijing, China (2000)
13. Rosenbrock, H.H.: An automatic method for finding the greatest or least value of a function. The Computer Journal 3(3), 175–184 (1960)
14. Sultan, A.J., Ouenes, A., Weiss, W.W.: Automatic history matching for an integrated reservoir description and improving oil recovery. In: Proc. of the Permian Basin Oil and Gas Recovery Conf., Midland, Texas, pp. 741–748 (1994)
15. Yang, C., Nghiem, L., Card, C., Bremeier, M.: Reservoir model uncertainty quantification through computer-assisted history matching. In: SPE Annual Technical Conf. and Exhibition, Anaheim, California, U.S.A. (2007)