Memetic Particle Swarm for Continuous Unconstrained and Constrained Optimization Problems

Carwyn Pelley[1], Mauro Sebastián Innocente[1,2], Johann Sienz[1,2]

[1] Civil and Computational Engineering Centre,
College of Engineering, Swansea University,
Singleton Park, SA2 8PP, United Kingdom
364915@swansea.ac.uk

[2] ASTUTE project, Digital Technium Building,
College of Engineering, Swansea University,
Singleton Park, SA2 8PP, United Kingdom
m.s.innocente@swansea.ac.uk, j.sienz@swansea.ac.uk

Abstract

Particle Swarm Optimization (PSO) is known for its effective and efficient global search and is one of the most effective Swarm Intelligence (SI) methods. PSO however fails to guarantee convergence to even locally optimal solution and so the method of switching to an effective local search at a safe point in the search is investigated with in-house General-Purpose PSO (GP-PSO). Combining the two algorithms results in guaranteed locally optimal convergence. Relations between various convergence criteria are investigated and methods derived to successfully switch, to the local search. Furthermore, user control is given with the derived method of switching, utilising the choice between accuracy and computational expense. With the added local search, this offers to extend the capabilities of the GP-PSO to competitive results with those in comparison in the literature.

Keywords

Particle swarm optimization, Memetic algorithm

1 Introduction

1.1 Particle Swarm Optimization

Research in the field of Swarm Intelligence (SI) in the past two decades has flourished, where many advances have been made in the direction of effective implementations to a wide range of optimization problem types. SI fits into the field of *modern heuristics* which intends to find a solution to a problem within a suitable computational time without guarantee of optimality. Furthermore, Particle Swarm Optimization (PSO) can be described as a *metaheuristic*, which is where a general framework maybe applied to a wide range of problem types as described by Clerc M.[3] (also called general-purpose heuristics, or polyheuristics). The original definition of the PSO paradigm and its place amongst other paradigms was first described by Kennedy et al.[10]. PSO fits into these categories and can also be considered to have roots in evolutionary algorithms (in particular, genetic algorithms), with its stochastic processes and evolution of its population but also its ability to follow a local and neighbourhood best (being similar to the crossover operator) as described by Kennedy et al[11]. The advantage of modern heuristics over traditional methods (often gradient-based), is that the former is not problem specific. Combining the PSO algorithm and Sequential Quadratic Programming (SQP) gradient-based local search by providing the latter with *good*[1] initial solutions, provides guaranteed local optima convergence which the PSO alone does not have.

A general description of the original PSO by Kennedy[10, 11] can be described as a randomly initialized swarm within feasible space with randomly initialised velocities. The velocity of each of the n-dimensional particles is accelerated towards its own personal best position and that of its local neighbourhood with stochastic weighting between the former and the later.

The basic underlying equations within the GP-PSO are shown below;
The velocity is updated according to:

$$ {v_{ij}}^{(t)} = w \cdot {v_{ij}}^{(t-1)} + iw \cdot U_{(0,1)} \cdot pbest_{ij}^{(t-1)} - x_{ij}^{(t-1)} + sw \cdot U_{(0,1)} \cdot lbest_{ij}^{(t-1)} - x_{ij}^{(t-1)} \tag{1} $$

The position is then updated according to:

$$ x^{(t)} = x^{(t-1)} + v^{(t)} \tag{2} $$

where $U_{(0,1)}$ is a random number from a uniform random distribution in the interval [0,1], sampled anew for each time it is called; w, iw and sw are the inertia, individuality and social weights respectively; pbest and lbest are

[1] Good describes here an initial solution that does not lead to a suboptimal converged solution.

the solution coordinate of the i'th particles best ever position and the solution coordinate of the best of the i'th particles neighbourhood respectively. The number of coefficients in PSO, though small in number, are highly problem dependant and as such, the GP-PSO in [9] can be characterised by the features developed to overcome this inherent problem and produce high quality solutions to trigger a local search.

A brief overview of features describing the GP-PSO are described as follows: the set-up consists of three swarms of different behaviour (different coefficients), complementing each other on the others weaknesses whether exploitative of exploratory; the neighbourhood is a 'forward topology', similar to that of the ring topology only that interconnections are not bidirectional in the former (see [9]). The intention is to slow clustering of the swarm and increase exploration of the search space. The size of these neighbourhoods is dynamic in that they change linearly with time-step so that the search begins highly local (extending the search of the space and delaying clustering), then becoming global toward the end with total cooperation between particles. The original PSO cannot handle constraints, however a number of methods have been adapted for handling them [1]. With the GP-PSO, Innocente[9] utilises a preserving feasibility method with priority rules, where a pseudo-adaptive relaxation of the tolerances for both equality and inequality constraints is used (relaxation is adaptive based on the feasibility of the swarm). For further details, refer to the original work in [9].

1.2 Local search

Population-based heuristics inherently improve with the implementation of a local search algorithm, since the heuristic approach of a population of solutions results in rather poor local search properties [6, 13, 15]. The combining of a local search algorithm with the population based heuristic is called a Memetic Algorithm (MA) and was first proposed in 1989 (Moscato 1989) and is quoted by Petalas Y.[16] as being inspired by the notion of Memes as defined by Dawkins (1976) as a unit of cultural evolution. The hybridization of algorithms has gained wide acceptance, due to its ability to solve difficult problems, overcoming the inherent weaknesses of any one particular algorithm, in fact Petalas Y. made a comparative study between the global optimizer and the implementation of the memetic approach and concluded the memetic approaches give superior search capabilities. The use of SQP as a local search to combine with the GP-PSO is made due to the SQPs inherent success in continuous constrained optimization problems (discussed by many authors, amongst them Victoire and Jeyakumar for example, who investigated a hybrid PSO-SQP algorithm for the economic dispatch problem [18]).

Victoire et al.[18] implemented the local search to gbest, triggered each time a new gbest solution was found. It was noticed by Victoire et al[18] that early on in the PSO search, particles were statistically likely to be of proximity to the global optimum but then move away from these areas. For this reason the local search method was chosen for implementation. Various implementation methods are observed in the literature including, what are called tandem search set-ups (where the entire population of particles is subdivided into two subsets, one performing the stochastic search and the other performing the local after which the subsets are merged) and Cascade searches, which perform stochastic searches to all particles and a further improvement is found by the use of a local search method at the end. Methods of investigation and results are varied but overall conclusions indicate that MAs to achieve highly accurate and less computationally expensive results.

Furthermore, stopping criteria are somewhat understudied in the literature, but such indications of stagnation include the reduction of particle velocity components to a certain threshold (Ismael et al.[17]) and the determination of the Euclidean distance of each particle to that of the best position as made by Gimmler et al.[7]. More often than not however, triggering of local search or search stagnation of the swarm is often made by defining a suitably high number of function evaluations (FEs) or time-step limit.

2 Experimental

Three benchmark suites are chosen for implementation, development and testing of the Memetic approach proposed here for the GP-PSO due to certain considerations. One consideration is that the GP-PSO in effect behaves differently with unconstrained problems as to constrained problems due to its constraint handling technique (the conflict and constraint functions handled separately) and so the two popular test suites CEC05[12] (unconstrained) and CEC06[12] (constrained) problems are chosen. Finally, a suitable test for final testing of the approach is through the application of real engineering problems, to which those as tackled in [8, 9] are chosen. With regards to comparison with other authors in the literature, two leading algorithms for the CEC06 are chosen which exhibit very good results and various authors used in comparison with the real-world engineering problems taken from [8]. It should be noted that comparisons are by no means comprehensive, but give indication as to the strength and weaknesses of the GP-PSO-SQP algorithm since the two chosen algorithms with respect to the CEC06 benchmark, in particular, exhibit features that are somewhat similar whilst different in other respects.

Described briefly, one algorithm used in comparison is that of the Dynamic Multi-swarm Optimizer (DMS-PSO) by Liang[13], who also couples a SQP local search to their PSO algorithm. Their method describes sub-populations solving their own objectives, and that their number being assigned adaptively and periodically according to difficulty. Local search implementation occurs with the calling of 5 random pbest particles every 'n' generations (supplying it with initial solutions) and after a number of generations the gbest particles is used to ensure final solution refinement. The DMS-L-PSO[4] is used in comparison with the CEC05 benchmark problems.

Another algorithm considered, is Particle Evolutionary Swarm Optimization Plus (PESO+) by Munoz-Zavala et al.[14]. In this algorithm, no local search is implemented though a perturbation of the solutions is made using operators which are said to be similar to those used in differential evolution plus the implementation of an external file, recording the best tolerant solutions found, though it is arguable that this algorithm in fact remains within the PSO framework.

Another algorithm considered for the comparison with CEC05, is that of the well known TRIBES algorithm by Clerc M.[5]. This algorithm aims to produce a black-box optimisation tool which adapts to the given problem with the use of tribes (multiple swarms) with both intra and inter-tribe communication. The number in each tribe and in fact the number of tribes then changes adaptively based on the findings within the solution space.

Measures derived by Innocente M.[9] for the measurement of swarm dynamics (clustering, diversity and stagnation) are used in this investigation for the purpose of identifying where a local search may be triggered for early switch-over. Such measures include clustering and evolutions measures. Furthermore, evolution of clustering measures are derived, relating the difference between successive time-steps of the clustering measures. It should be noted however that the actual formalization of these measures take a relative approach with the use of a swarm in search of the maximum rather than the minimum and that smoothing with previous iterations occurs (parameter called 'Tref') as defined by Innocente M.[9]. Since there are a number of underlying factors resulting in the successful development of switching criteria, together with the fact that short searches may only allow a small number of search time-steps, Tref is fixed to 10. The number of repeat runs is 20 and a time-step limit of 10,000 for the CEC06 and the real-world engineering problems. The number of particles in the maximizer is 10 and the number in the minimizer is 50. For CEC05, a function evaluation (FE) limits (extraction point) of $1e5$ is defined with 25 repeat runs made.

3 Results and Discussion

Firstly, the implementation of SQP to constrained problems is made. It is clear that major refinement of solutions is apparent on the CEC06 benchmark suite (g01-g13) together with the test engineering problems. The GP-PSO is found to have much difficulty in dealing with problems g02, g05, g07, g09, g10 and g13 where the SQP applied to the best, median and worst gbest solutions, result in refinement of solutions on all these problems with only g02 suffering from a less than 100% global optimum convergence. This is due to g02 being a highly difficult multidimensional multi-modal problem, where the GP-PSO struggles to find the global optimum valley amongst the multitude of suboptimal attractions of the search-space. If the SQP local search is provided with a solution that leads it further down a local sub-optimal region of the search space, then the SQP becomes trapped and merely refines its solution to the minimum of the local valley. Refinement of solution however is apparent on other problems where some other difficulty is apparent by the PSO. In some occasion, the dynamics of the swarm may limit the refinement of the final solution of the PSO and in others, diversity maybe prematurely lost and due to the tight constraints, the swarm maybe artificially confined to a rather small area of the search-space. Sensitivity to scaling and SQP tolerances is apparent in the investigation of these problems, with the two being interlinked. The termination of the SQP search is also linked with the scaling of the problem since it utilises scale sensitive tolerances of absolute magnitudes. Linear scaling of the conflict and constraints however appears to be as much likely to be harmful as they are beneficial, since what it deems an active or inactive constraint then changes. On the one side, a constraint may become inactive, opening a path towards optimality for the SQP but on the other hand this may open up a path to sub-optimality to which the SQP would not have originally have been directed towards. These are just some of the observations made upon application of the SQP algorithm at various points in the GP-PSO search history. The main intent of this investigation however is in the application of the SQP to the PSO at a point in the GP-PSO search where safe switching is likely.

3.1 Unconstrained problems

With the investigation of unconstrained problems of CEC05, the problem-specific nature of clustering measures becomes apparent. Multi-modal problems or highly difficult constrained problems are found to exhibit rather large clustering measures, since the particles become stuck in suboptimal regions of the search-space and require considerable time to converge to the gbest solution. The sheer number of highly attractive areas of the search-space slow the progress of the search toward the 'final' global optimal solution (if found at all). Uni-modal or rather simple functions however cluster well and diversity is lost entirely. On this note, the development of what are called evolution of clustering measures is used in order to utilise information based on the relation between the best and average solutions in the swarm while not relying on absolute measures (i.e. looking at the evolution of these clustering measures between time-steps). The clustering measures however, offer to be the safest method in which to indicate stagnation of the swarm with their highly smooth measures.

The use of pbest particles rather than current particles is used in the calculation of all measures. The current particles set-up consists of the calculations utilising current solutions of the swarm as opposed to the pbest set-up where the 'pbest swarm' receives updates of particle i $(i = 1, m)$, where m is the number of particles, only if a better solution is found (either by a lower maximum constraint violation, lower conflict or both). With the use of pbest particles, the clustering measures become smooth, though at the cost of erratic evolution measures since updates to pbest become further between as the search progresses. This has the undesired effect of producing sudden changes after no changes are apparent over previous time-steps. This is due to the fact that evolution measures relate between time-steps.

In order to tackle this problem, two methods are investigated. One considers the smoothing of all previous time steps up until the current time-step. Another considers the number of iterations the measure remains below a derived threshold. A simple illustration diagram of this method is shown in Fig.1. Regarding the first method, a summing of all previous time-steps appears to be rather problem specific and this owes to the fact that problem specific search histories are apparent on the magnitudes of measures unless given enough time. In the cases where swarm dynamic limitations are apparent, further problem specific magnitudes of measures are likely to be present and so this method is discounted. The later method however offers much in the way of user control but at the

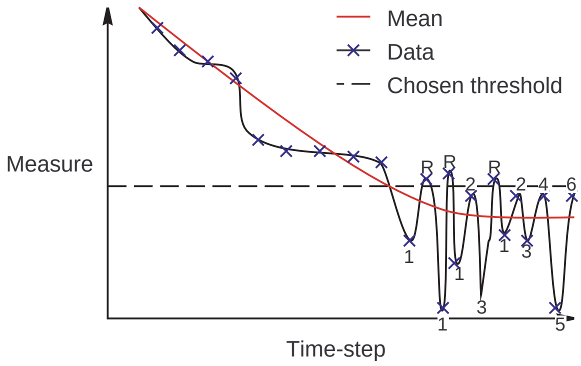

Figure 1: Credibility count method (credibility count is the number of time-steps below the threshold required to trigger early switching).

expense of less than certainty of successful switch-over. Regarding the derivation of the required thresholds, a method in which to define what are safe thresholds is required. In the case of unconstrained problems (investigated first due to it being a simpler case in the behaviour and set-up of the GP-PSO), the points to extract thresholds consist of: the points at which error is attained by the GP-PSO by the gbest particle; where error is attained by the average of the swarm; where error is attained by the SQP when applied at each iteration of the the GP-PSO; and lastly, the extracted values at the final time-step (taken to be the stagnation extracted thresholds since a generous maximum time-step limit is given). From this, the various criteria are chosen with a range of 1 to 500 counts on the *credibility count* (arbitrarily chosen range, where a count of 1 means a disabling of the counter method). The chosen tested set-ups are shown in Table.1, derived on selected problems of CEC05.

Measure	Clustering			Evo-clustering			Evolution			
Criteria label	cb me	pb me	pb cge	evo cb me	evo pb me	evo pb cge	cb av	cb best	pb cge	pb gbest
*SQP1	1e-6	1e-3	1e-3	1e-7	1e-5	1e-4	1e-7	1e-9	1e-4	1e-4
*SQP2	-	-	-	1e-7	1e-5	1e-4	1e-7	1e-9	1e-4	1e-4
ERRB	-	-	-	1e-15	1e-7	1e-7	1e-15	1e-18	1e-7	1e-7
*ERRA	-	-	-	1e-15	1e-8	1e-9	1e-15	0e0	0e-8	0e0
END1	-	-	-	0e0	0e0	0e0	0e0	0e0	0e+0	0e0
END2	-	-	-	-	-	-	-	0e+0	-	0e+0

Table 1: The six chosen criteria for switch-over. *chosen thresholds for further testing.

Of those set-ups shown in Table.1, three set-ups are further tested as indicated by the asterisk. These are chosen on the grounds of extremes. cri_SQP1 criteria offers to be a 100% successful and safe criteria and utilises clustering measures (making it problem specific, however at the advantage of its insensitivity to credibility count), whilst cri_SQP2 and cri_ERRA criteria offer to give user control over computational expense and accuracy though to a varying and unpredictable degree depending on prior knowledge of the user to the applied problem. Further testing of these three criteria on all problems 1-14 in 2,10 and 30dimensions is made and the three chosen measures are shown to be highly suitable. The following results and observations are made:

Three levels of user control are suggested from the derived criteria. One consists of the cri_SQP1 criteria which as already described, utilises information on the clustering measures and so is problem specific, or at least requires am absolute level of clustering. This is identified as the safest switch-over method since it deals with magnitudes that do not relate between time-steps other than the smoothing over ten previous time-steps (tref). This criteria also offers to be suitable for any search length with its insensitivity to credibility count. It is shown that over the problems considered, that 35% more FEs are required to trigger switch-over compared to the number of FEs required for the PSO to attain error. It should be noted however that switch-over also has the added condition that if no switch-over has yet occurred and the final time-step has been reached (user defined), that the local search will be implemented. This criteria is problem specific and in its current set-up, triggers on only uni-modal problems or problems which result in high clustering. This method however which uses clustering measures may have thresholds redefined for problems that do not cluster heavily as indicated by a trial run.

The two remaining criteria however, offer significant user control with adjustment of the credibility count. With the suggested credibility count of 500 however, this results in the triggering on nearly all occasions using both criteria cri_ERRA and cri_SQP2. At a count of 500, the number of FEs required is 78% of the total number required for for error to be attained (using the former criteria) and an increase of 7% over the number of FEs to the point at which error is attained (on the later criteria). This offers two criteria which give control to the user with a fixed credibility count of 500 on each. This fixed count is required since the erratic behaviour on evolution curves is apparent up to a smoothing of 500 time-steps. Some notable results between the two criteria however, are F4(10D) and F5(10), which do not meet success upon triggering by the cri_SQP2 criteria but does with use of cri_ERRA criteria. This suggests the lack of correlation based on the thresholds derived by the success of the local search (i.e. the problems are in fact still threshold sensitive and with use of that derived by success of the local search on a small sample of problems, a general application to the problem range is unsuccessful. Since F4 contains noise on the conflict, it makes sense why this function might deviate with respect to the chosen thresholds for the criteria. Looking in further detail at these two problems, it becomes clear that the two criteria fit their purpose in that, 39% of the total number of FEs were required on average with use of cri_SQP2 and 54% with use of cri_ERRA and that the former has a 2% chance of triggering too early.

It should be noted that since a local search is be triggered at the final time-step if no criteria has not been met, that any search length may be used, however depending on the chosen credibility count, it may be unlikely that a criteria be met before the search terminates if the search is too short. Similarly, in the case of the first criteria cri_SQP1, the high clustering may not occur with very tight restrictions on the search length.

For some comparison with suitable algorithms, TRIBES and DMS-L-PSO is chosen as already described. The results are summarised as follows in the following Table.2:

(10D)	FE TRIBES	success	FE DMS-L-PSO	success	FE GP-PSO	success	END FE GP-PSO-SQP		COUNT 100 FE GP-PSO-SQP	
F1	1.36E+003	100%	1.19E+004	100%	4.33E+04	100%	1.0E+05	100%	8.86E+003	100%
F2	6.62E+003	100%	1.23E+004	100%	9.9E+04	20%	1.0E+05	100%	9.88E+003	100%
F3	1.04E+004	100%	1.25E+004	100%	1.0E+05	0%	1.0E+05	100%	1.07E+004	100%
F4	1.74E+004	100%	1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	9.65E+004	0%
F5	5.54E+004	88%	1.00E+005	80%	9.8E+04	24%	1.0E+05	28%	1.53E+004	0%
F6	9.83E+004	4%	5.47E+004	100%	1.0E+05	0%	1.0E+05	88%	1.16E+004	76%
F7	9.66E+004	4%	1.00E+005	16%	1.0E+05	0%	1.0E+05	0%	9.88E+003	0%
F8	1.00E+005	0%	1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	4.72E+004	0%
F9	1.00E+005	0%	3.46E+004	100%	1.0E+05	0%	1.0E+05	0%	1.48E+004	0%
F10	1.00E+005	0%	1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	1.42E+004	0%
F11	1.00E+005	0%	1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	6.69E+004	0%
F12			1.25E+004	76%	1.0E+05	0%	1.0E+05	48%	1.32E+004	12%
F13			1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	1.23E+004	0%
F14			1.00E+005	0%	1.0E+05	0%	1.0E+05	0%	2.09E+004	0%
	6.24E+4		6.70E+4		9.58E+4		1.01E+5		2.52E+4	
	62.00%		66.63%		95.22%		100.00%		25.01%	

Table 2: Comparison with literature.

The results presented in Table.2, indicate that there is significant improvement wih use of the SQP local search with the GP-PSO. With the added restriction of CEC05, that the maximum number of FE be $1e5$, with use of a credibility count of 500, no triggers were made on the problems considered and so not presented. With use of a credibility count of 100 however, not a significant reduction in accuracy is observed over the problems investigated and a significant reduction in FE is observed (on average, significantly less than DMS-L-PSO). This indicates that the current set-up of GP-PSO is in fact perhaps a greedier algorithm. The GP-PSO meets success on F4 if given enough time, together with F9 and F12 which then brings it in line with DMS. An interesting difference is that the DMS attains error on F7, which the GP-PSO-SQP does not, even if not restricted by the CEC05 limits. Comparison with TRIBES, indicates, that again the search appears to be more efficient than with the GP-PSO, although with it achieving rather low success rates on F6, for which the GP-PSO does not share. From [5], it is apparent that this is due to the trapping within a local optimum. It is then speculated that F7 then offers the GP-PSO much difficulty for this very reason.

Comparing the three algorithms, it appears that the GP-PSO has the weaker of the three search algorithms, however this comes to no surprise with its most recent developments targeting constraint handling. It should be noted however, that this serves as an indication only, since more recent versions of the algorithm specifically increasing the efficiency of the search are already under-way. It is conclusive from the results shown that the introduction of the local search enables the GP-PSO to refine its solution if the GP-PSO has successfully searched the solution space.

3.2 Constrained problems

Regarding constrained problems, another problem arises in the derivation of the above thresholds. The pseudo-adaptive relaxation of tolerances based on the feasibility of the swarm means that measures do not relate to the final problem being tackled until either 80% of the search has been reached or the swarm has successfully remained more than or equal to 20% feasible so that final constraint tolerance is reached. Initial investigations into constrained problems reveals this to be a considerable difference with unconstrained problems, since the added requirement for final problem tolerance to be reached, indicates this to be the main drive switching, rather than the chosen thresholds. This is indicated by the lack of sensitivity to credibility count on constrained problems. For this reason, thresholds are defined according to only the points at which the PSO attains error and also the final time-step extraction values. These are then derived based on a sample for the CEC06 suite (problems 1-13), and using these derived criteria, testing is made on the entire 24 problems. The derived thresholds are shown in Table.3.

Measure	Clustering			Evolution-clustering			Evolution			
Criteria label	cb me	pb me	pb cge	evo cb me	evo pb me	evo pb cge	cb av	cb best	pb cge	pb gbest
END1	0e0	1e-10	1e-10	0e0	0e0	0e0	0e0	0e0	0e+0	0e0
END2	0e0	1e-10	1e-10	-	-	-	-	-	-	-
*END3	-	-	-	0e0	0e0	0e0	0e0	0e0	0e+0	0e0
*ERRB	-	-	-	1e-7	0e0	1e-7	0e0	1e-7	1e-7	1e-7
ERRA	-	-	-	1e-11	0e0	1e-12	0e0	1e-11	1e-12	1e-12

Table 3: The chosen criteria for switch-over. *chosen thresholds for further testing.

With the additional requirement that the final tolerance be reached, the requirement for such a high credibility count is relaxed. In fact, it is found that the switching technique is somewhat insensitive to the chosen thresholds. As shown from Table.3, two criteria are chosen again from the point of view of two extremes. One considers the safest criteria and the other considers the less computationally expensive criteria. The GP-PSO struggles with problems g05, g07, g09, g10, g14 and g19-g23. Through use of the SQP local search using the gbest end result of the GP-PSO search, success is met on problems g05, g07, g09, g10, g14, g19 and g23. Some notable problems however are g14, which due to the formulation of the problem does not allow the search to leave the feasible intervals, or the conflict has an imaginary part (i.e. a hard interval constraint). To overcome this, the solution coordinates are repaired within the SQP algorithm to the interval border if allowed to cross it. This results from prior knowledge of the function. g21 and g22 however contain logarithms also but the feasible intervals are defined such that solutions should not venture to imaginary space and so problem specific tailoring of the SQP to these problems is required in order to keep it within bounds. Similarly, the GP-PSO does not handle these two problems, since the logarithms are within the constraints and as the relaxation of tolerances of the constraints is used in dealing with the problems of the suite, this makes it unsuitable for tackling them.

The insensitivity of the two chosen criteria are apparent by the similarity of the results with use of the two criteria. The insensitivity then to the credibility count is then indicated by the success with the disabling of the credibility count (count1). Comparison between the accuracy of the two criteria across the suite with the disabling of the count is comparable to that when applied to the final time step (10,000). Significant problems include g10, which is not clear as to the apparent difficulty, though it is suggested that variable scaling may be an issue here. The difficulty of the PSO is somewhat illusive, since the final converged solutions appear to be rather far apart, though with respect to the SQP, the solutions appear to be at the global optimum but with a slight infeasibility at around $1e-11$. Again, the issue of scaling on the problem and or algorithm tolerances (with the later being of fixed magnitude for what is considered a feasible solution ($1e-12$)) appears to be the problem at issue with respect to the SQP. g17 is again a multi-modal problem and as such fails with use of the SQP when provided by a suboptimal solution by the PSO, however the PSO is found to be rather successful with respect to other authors on this function, owing to the constraint handling technique used. The PSO alone finds difficulty with g18, as all solutions are feasible but none are at the global optimum, but with application of the SQP, a 100% success is observed.

To re-iterate, comparisons here are made with two algorithms PESO+ and DMS-PSO as discussed in Section.2. The results of which are shown in Table.4. It should again be noted that where a criteria is not met, the local search is applied at the very final time-step to which is included in the calculation of the mean number of FEs.

Algorithm	Percentage FEs
DMS-PSO	19.2%
GP-PSO	23.3%
GP-PSO-SQP (count1)	23.7%
GP-PSO-SQP (count50)	44.6%
GP-PSO-SQP (count500)	N/A
PESO+	100.0%

Table 4: Percentage FEs compared to the maximum mean algorithm (PESO+) between all 24 problems to achieve the fixed accuracy level (($f(\bar{x}) - f(\bar{x}^*) \leq 1 \times 10^{-4}$)). Problems that are not successful on all four algorithms are removed from statistics including problems g05, g07, g09, g10, g14, g17 and g19-g23.

It is apparent that the application of the local search, bridges the gap between the two optimizers. The PESO+ contains no local search implementation, however the DMS-PSO does, though with its implementation of a local search during the PSO search, it achieves highly optimised solutions at a much reduced computational cost to the GP-PSO. With implementation of the local search to the GP-PSO, it is however possible to achieve similar success to the DMS-PSO and at a rather similar computational cost. A comparison between these algorithms with respect to the problems is then shown in Table.5.

It is clearly observed that the DMS-PSO achieves a 100% success on problem g10 and g21 where the GP-PSO-SQP does not. The GP-PSO however achieves success on problem g17. With the implementation of the SQP local search, the GP-PSO achieves similar success to then the DMS-PSO with the refinement capabilities of the local search.

To conclude on the derived measures for constrained problems, a credibility count of 50 appears to be adequate with the main driving factor being in the final tolerance being reached using the PSO constraint handling technique with the possibility of increasing to a count of 500 or perhaps its disabling in cases where highly difficult multi-modal problems are tackled (in particular g02). In g02, an early switching may result in a refined solution that leads it further down a suboptimal valley and as such, the longer the search is allowed to continue, the more likely that the local search will converge to the correct valley. From this observation, a general case maybe made. Only one criteria is chosen 'cri_ERRB' since insensitivity to derived thresholds are apparent. One implementation is to use a 50 time-step count as default, where this is suitable for the the vast majority of cases, or at least as observed by the suite under investigation. Secondly, a range of 1-500 is another option giving the user control over the likelihood between update of the swarm toward bettering solutions while keeping in mind that a high credibility count may result in a lack of triggering. Thirdly, as with the unconstrained problems, the use of clustering measures maybe

PROBLEM	GP-PSO SUCCESS	GP-PSO FEASIBLE	GP-PSO-SQP SUCCESS	GP-PSO-SQP FEASIBLE	PESO+ SUCCESS	PESO+ FEASIBLE	DMS-PSO SUCCESS	DMS-PSO FEASIBLE
g01	100%	100%	100%	100%	100%	100%	100%	100%
g02	30%	100%	30%	100%	56%	100%	84%	100%
g03	90%	100%	100%	100%	100%	100%	100%	100%
g04	100%	100%	100%	100%	100%	100%	100%	100%
g05	0%	100%	100%	100%	100%	100%	100%	100%
g06	100%	100%	100%	100%	100%	100%	100%	100%
g07	0%	100%	100%	100%	96%	100%	100%	100%
g08	100%	100%	100%	100%	100%	100%	100%	100%
g09	0%	100%	100%	100%	100%	100%	100%	100%
g10	0%	100%	70%	85%	16%	100%	100%	100%
g11	100%	100%	100%	100%	100%	100%	100%	100%
g12	100%	100%	100%	100%	100%	100%	100%	100%
g13	70%	100%	100%	100%	100%	100%	100%	100%
g14	0%	100%	100%	100%	0%	100%	100%	100%
g15	100%	100%	100%	100%	100%	100%	100%	100%
g16	100%	100%	100%	100%	100%	100%	100%	100%
g17	95%	100%	95%	100%	0%	100%	0%	100%
g18	5%	100%	100%	100%	92%	100%	100%	100%
g19	0%	100%	100%	100%	0%	100%	100%	100%
g20	NaN	0%	NaN	0%	NaN	0%	NaN	0%
g21	0%	0%	NaN%	NaN%	0%	100%	100%	100%
g22	0%	0%	NaN%	NaN%	0%	0%	0%	0
g23	0%	0%	100%	100%	0%	96%	100%	100%
g24	100%	100%	100%	100%	100%	100%	100%	100%

Table 5: Final success rates and feasibility rates using the end values (corresponding to 10,000 iterations in the case of the GP-PSO and GP-PSO-SQP, roughly corresponding to 500,000 FEs).

used if thresholds are derived specifically for the chosen problem (trial run or a priori knowledge of the function to be optimized). Similarly to the unconstrained problems, the clustering measures are shown to be problem specific, though to an even greater extent in constrained problems since the constraints confine the swarm in problem specific ways, to which the user may not be able to predict a priori. The most significant result from this investigation is that the implementation of the local search to the in-house GP-PSO, results in the ability to guarantee local optimum convergence which the PSO does not alone meet. It also results in the successful switching of the algorithm to the local search at the point at which the PSO on average finds the global optimum.

3.3 Test Engineering problems

To conclude on the derived switching method described. A selection of typical test engineering problems are to be made as taken from [8]. Such problems consist of the design of a pressure vessel (PVD), welded beam design (WBD), minimization of the weight of a tension/compression spring design (TCSD) and finally Himmelblau's non-linear optimization problem (HBNLP). It should be noted that the formulation of the PVD problem is discrete, however a continuous version of this problem is tackled for application of the SQP local search called C-PVD. With regards to the application of the SQP to the GP-PSO. The SQP achieves 100% accuracy when applied from the very first time-step on C-PVD, WBD and TCSD, however this is not the case for HBNLP. The solutions found are 100% feasible and the objective function is within 1×10^{-02} of the literature optimum. With respect to the switching criteria however, only a credibility count of 10 was found to be necessary for these problems with no degradation in the converged solution (with this required for the HBNLP alone). Furthermore, with a credibility count of 10, all sample runs were triggered on all problems. With a count of 1, a comparison is made with a selection of authors in the literature in order to emphasise the effectiveness of the local search, together with the minimum required knowledge of the user in its application.

Prob.	Optimum	SQP Conflict (count1)				GP-PSO			
		Best	Mean	Fes	Runs	Best	Mean	Fes	Runs
C-PVD	5885.332774	5885.332774	5885.332774	2.8E+04	20	5885.431448	5894.288539	1.6E+05	20
WBD	1.724852	1.724852	1.724852	7.7E+04	20	1.724927	1.724942	2.2E+04	20
TCSD	0.012665	0.012665	0.012665	1.9E+04	20	0.012673	0.012735	1.3E+04	20
HBNLP	-31025.561420	-31025.561420	-31025.551307	6.2E+04	20	-31025.561368	-31025.561328	5.9E+04	20
		Leticia C. et al.[2]				De Freitas Vaz et al.[17]			
		Best	Mean	Fes	Runs	Best	Mean	Fes	Runs
C-PVD	5885.332774	-	-	-	-	5885.33	-	8.8E+05	-
WBD	1.724852	1.724852	2.057400	2.40E+04	30	1.81429	-	9.6E+05	-
TCSD	0.012665	0.012665	0.013100	2.40E+04	30	0.0131926	-	7.6E+05	-
HBNLP	-31025.561420	-	-	-	-	-31012.1	-	7.8E+05	-
		Hu X. et al.[8]				Worasucheep C.[19]			
		Best	Mean	Fes	Runs	Best	Mean	Fes	Runs
C-PVD	5885.332774	-	-	-	-	-	-	-	-
WBD	1.724852	$*^1$ 1.72485084	$*^2$ 1.72485084	2.00E+05	11	1.724852	1.724852	2.0E+05	30
TCSD	0.012665	0.012666	0.012719	2.00E+05	11	-	-	-	30
HBNLP	-31025.561420	-31025.56142	-31025.56142	2.00E+05	11	*-31026.647264	-31002.170814	2.0E+05	30

Table 6: Comparison of results for the four tested problems with that in the literature. *Corrected conflict according to given design variables (in brackets), $*^1$1.72485084 (1.724855), $*^2$1.72485084 (1.724855).

From this it becomes apparent that the GP-PSO achieves competitive results (according to not only the solutions to which it finds but also the number of FEs required to achieve such level of accuracies). With the SQP applied with a credibility count of 1, a similar number of FEs is required to the point at which the GP-PSO attains error, however with the increased accuracy of the mean solution, apart from problem HBNLP which as already discussed, requires a credibility count of 10 (enabled). This then results in a mean FE of 9.5×10^4 which still compares well with other authors.

4 Conclusion

The investigation into the implementation of a local search with in-house particle swarm optimizer is made. Methods in which to successfully switch between the global and local search have been derived and tested, resulting in a method that achieves a mean computational expense (indicated by FEs) comparable to the point at which the GP-PSO attains error. The accuracies are then shown to be refined, limited by only the computational accuracy. This then identifies the GP-PSO as being competitive amongst other algorithms, at least on the tested problems, whilst giving total control to the user if a priori knowledge is known on the problem being tackled.

References

[1] Alec Banks, Jonathan Vincent, and Chukwudi Anyakoha. A review of particle swarm optimization. part ii: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications. *Natural Computing*, 7:109–124, 2008.

[2] Leticia C. Cagnina, Susana C. Esquivel, and Carlos A. Coello Coello. Solving engineering optimization problems with the simple constrained particle swarm optimizer, 2008.

[3] Maurice Clerc. *Particle Swarm Optimization*. ISTE Publishing Company, 2006.

[4] Yann Cooren, Maurice Clerc, and Patrick Siarry. Performance evaluation of tribes, an adaptive particle swarm optimization algorithm. *Swarm Intelligence*, 3:149–178, 2009. 10.1007/s11721-009-0026-8.

[5] Yann Cooren, Maurice Clerc, and Patrick Siarry. Performance evaluation of tribes, an adaptive particle swarm optimization algorithm. *Swarm Intelligence*, 3:149–178, 2009. 10.1007/s11721-009-0026-8.

[6] Marco Dorigo and Thomas Stutzle. *Ant Colony Optimization*. The MIT Press, 2004.

[7] Jens Gimmler, Thomas Sttzle, and Thomas E. Exner. Hybrid particle swarm optimization: An examination of the influence of iterative improvement algorithms on performance. volume 4150, pages 436–443, Brussels, 2006.

[8] X. H. Hu, R. C. Eberhart, and Y. H. Shi. Engineering optimization with particle swarm. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium (sis 03)*, pages 53–57, Indianapolis, 2003.

[9] Mauro Sebastin Innocente. *Development and testing of a Particle Swarm Optimizer to handle hard unconstrained and constrained problems*. PhD thesis, Civil and Computational Engineering Centre, College of Engineering, Swansea University, Swansea, United Kingdom, 2010.

[10] J. Kennedy and R. Eberhart. A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, pages 39–43, Nagoya, 1995.

[11] J. Kennedy and R. Eberhart. Particle swarm optimization. In *IEEE International Conference on Neural Networks*, volume 1-6, pages 1942–1948, Perth, 1995.

[12] J. J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. N. Suganthan1, C. A. Coello Coello, and K. Deb. Problem definitions and evaluation criteria for the cec 2006", special session on constrained real-parameter optimization. In *Technical Report*, Singapore, 2006.

[13] J. J. Liang and P. N. Suganthan. Dynamic multi-swarm particle swarm optimizer with a novel constraint-handling mechanism. In *IEEE Congress on Evolutionary Computation*, volume 1-6, pages 9–16, Vancouver, 2006.

[14] Angel E. Munoz-Zavala, Arturo Hernandez-Aguirre, Enrique R. Villa-Diharce, and Salvador Botello-Rionda. Peso+ for constrained optimization. In *IEEE Congress on Evolutionary Computation*, volume 1-6, page 231, 2006. Vancouver, Canada.

[15] Paola Pellegrini and Elena Moretti. A computational analysis on a hybrid approach: Quick-and-dirty ant colony optimization. *Applied Mathematical Sciences*, 3:1127–1140, 2009.

[16] Y. G. Petalas, K. E. Parsopoulos, and M. N. Vrahatis. Memetic particle swarm optimization. *Annals of operations research*, 156(1):99–127, Dec 2007.

[17] De Freitas Vaz, Antnio Ismael, Pinto Fernandez, and Edite Manuela Da Graa. Optimization of nonlinear constrained particle swarm. *Technological & Economic Development of Economy*, 12(1):30–36, 2006.

[18] T. Aruldoss Albert Victoire and A. Ebenezer Jeyakumar. Hybrid pso-sqp for economic dispatch with valve-point effect. *Electric Power Systems Research*, 71(1):51–59, 2004.

[19] C. Worasucheep. Solving constrained engineering optimization problems by the constrained pso-dd. In *Proceedings of the ECTI-CON*, volume 1-2, pages 5–8, Krabi, 2008.