Natural Computing 2011: Assignment 2

Particle Swarm Optimisation

This is an exercise worth 20% of the course mark. It requires you to write programs and to use them to carry out a number of investigations. You choose any of the common programming languages, but in choosing you should consider that some parts of this assignment are computationally demanding. You are to write up your investigations in the form of a conference paper with a maximum length of 6 pages (12 pt., one- or two-column layout). The page limit is strict and marking will be done based on these 6 pages. You can submit more detailed descriptions in form of appendices which need to be referred to from the main text. In addition to the report, submit also the code of your program(s). Both should be done electronically using the submit program

'submit nat 2 <filename>'

see the man page of the submit program for more information. Please submit your report as a pdf file and your code separately as a plain text file or an archive of text files.

You should submit your assignment by 4pm of Thursday, 24. November, 2011.

Main idea: Particle Swarm Optimisation is a versatile approach to different types of optimisation problems. It adapts continuous vectors such that a fitness function is minimised. It has found many applications in various fields especially also for dynamic problems, Although dynamic fitness functions would be a very interesting topic as well, you will deal here with a more basic level of investigation.

In the report on your work on this assignment write separate sections answering the following questions and wherever possible include graphical representations of your numerical results.

1. (Introduction) Describe briefly the motivation for particle swarm optimisation, its form and the expected advantages and potential drawbacks. (marks: 5/100)
2. Consider the simple problem to find the minimum in the function $f(\mathrm{x})=\Sigma_i x_i^2$ with $x=(x_1, x_2)^T$. Having fixed a reasonable initialisation scheme and suitable values of ω and of the swarm size, how does the behaviour of the algorithm depend on the parameters α_1 and α_2? Represent the result of this task (for one fixed value of ω) graphically either in a 2D plot (with axes α_1 and α_2 which shows the regions where the swarm is convergent, oscillatory, or divergent) or in a suitable 3D plot. Does the swarm show any other behaviours? Provide the details how you determine the behaviour of the swarm. (20/100)
3. If you inspect the behaviour of the particles in the above task in a parameter setting where the swarm converges rather slowly, you should be able to identify a shortcoming which seems to be intrinsic to the algorithm. Analyse the problem and propose a variant of the algorithm that does not suffer from this problem. (15/100)
4. Construct a fitness function with which you can show that your version of the algorithm is better than the standard algorithm. If you did not solve question 3 you can choose a variant from the literature to compare the standard particle swarm algorithm to. (15/100)
5. The standard particle swarms optimisation algorithms use momentum, personal best and generation best in order to update the particles. Obviously, it is possible to include further information, e.g. from the all-time population best, personal bests or present values of certain neighbours, differences between other vectors in the swarm (this would mean to marry Differential Evolution to PSO), repulsion from neighbours, purely random inputs etc. Each of these terms is supposed to enter the update rule of the velocity with a positive scalar factor α_i. Choose a total of at least five terms (neighbourhood parameters, such as degree of a node count also as a term) and determine the corresponding α-factors by a meta-algorithm. Choose

an example of medium complexity (e.g. the (generalised) Rastrigin function in *D* dimensions) to meta-optimise the PSO algorithm using a metaheuristic algorithm of your choice. Compare the performance of the meta-optimised algorithm with the canonical form (or with you variant from question 3 above). Interesting results would include a considerable speed-up with same quality or an improved success rate or if some of the α_i turn out to be very close to zero (and the algorithm can do well without these terms). (30/100)

6. (Discussion) Discuss here question such as: Did your algorithm(s) perform as expected? If not, why not? (Refer back to earlier sections if you believe that a failure in one of those sections is sufficiently justified to earn you more points.) If yes, why did the algorithm(s) perform as it did? What could have been improved if there was one more week (or one more year) to work on the problem? How general are my results? How (or how not) to interpret the results? (etc.) (10/100)
7. (Conclusion) What are the main results in words? (5/100)

Please state clearly (e.g. simply by following the numbering of the tasks) which part of your report answers what question. Try to write your report such that it is understandable even without repeating the questions (assuming the reader is familiar with PSO in general). You should clearly describe the experimental design, the results, and discuss the results. Use figures where appropriate. It will be helpful if you comment your code, but there will be no marks for good code (you may loose marks if neither the text nor the code is clear).

Feel free to search through the literature, but make sure to cite all the papers that you found useful for this project. The list of references (as a kind of appendix) will not count towards the 6 page limit, but you should place a pointer (cite or endnote) in the main text where you describe in what way the referred literature was useful for your work.

If you need to choose values of parameters of the PSO, you should explain and justify how you came to this choice, given that you (or rather, the individuals in the population) are aiming at good performance. You also need to decide how you are going to present results that such that you present a clear overall picture of the behaviour of the PSO in the present problem.

In the tutorials will be time to clarify most questions related to the assignment. For more critical questions, please contact the lecturer (michael.herrmann@ed.ac.uk) directly.

References

J. Kennedy; R. Eberhart: Particle swarm optimization. *IEEE Int. Conf. on Neural Networks*, vol.4, pp.1942-1948, 1995.

Ioan C. Trelea. The particle swarm optimization algorithm: Convergence analysis and parameter selection. *Information Processing Letters*, **85**(6):317–325, 2003.

NB: The following criteria will be used in evaluating your assignment: Is the design of the algorithm sound? How clearly is your design and implementation described? How sound are the experiments? **Are the results statistically significant?** To what extent did your experiments give the algorithm “a good workout” (i.e. do you understand the effect of the parameters and the design on the results)? Does the discussion of your results shed light on how the algorithm works as you vary the parameters and the difficulty of the problem?