

AER1415: Computational Optimization

Assignment 1

University of Toronto Institute for Aerospace Studies

Zouhair Adam Hamaimou
Student Number: 1004981986

Due: February 14, 2025

Contents

1	Introduction	3
2	PSO Algorithm Implementation and Testing	3
2.1	Algorithm Description	3
2.2	Parameter Tuning Studies	4
2.3	Results for Case Studies	4
3	Regression Problem	4
3.1	Problem Description	4
3.2	Solution	5
4	Constraint Relaxation Problem	5
4.1	Part (a): Minimum Relaxation	5
4.2	Part (b): Least Number of Constraints	5
4.3	Part (c): Minimum-Cost Relaxation	6
5	Conclusions	6

1 Introduction

This report presents solutions to Assignment 1 for the AER1415 course, focusing on the implementation and testing of the Particle Swarm Optimization (PSO) algorithm for constrained optimization problems. The following sections describe the PSO algorithm, parameter tuning studies using the bump test function (P3), and the application of the optimized parameters to other case studies. Additionally, solutions to the regression problem and the constraint relaxation problem are included.

2 PSO Algorithm Implementation and Testing

2.1 Algorithm Description

Particle Swarm Optimization (PSO) is a stochastic optimization algorithm inspired by the collective behavior of swarms. The algorithm maintains a population of particles, each representing a potential solution in the search space. The particles adjust their positions based on their own experiences and the experiences of their neighbors.

The main steps of the PSO algorithm are as follows:

1. Initialize a swarm of particles with random positions and velocities within the defined bounds.
2. Evaluate the objective function value for each particle.
3. Update each particle's personal best position and the global best position.
4. Update the velocity and position of each particle using:

$$\begin{aligned}v_i(t+1) &= wv_i(t) + c_1r_1(p_{i,\text{best}} - x_i(t)) + c_2r_2(g_{\text{best}} - x_i(t)), \\x_i(t+1) &= x_i(t) + v_i(t+1),\end{aligned}$$

where w is the inertia weight, c_1 and c_2 are acceleration coefficients, and r_1 and r_2 are random values in $[0, 1]$.

5. Repeat steps 2–4 until a stopping criterion is met.

To handle constraints, the quadratic penalty function approach is employed:

$$\phi(x) = \sum_{i=1}^m \max(0, g_i(x))^2 + \sum_{i=1}^q h_i(x)^2,$$

where $g_i(x)$ are inequality constraints and $h_i(x)$ are equality constraints. The penalty is added to the objective function value to discourage infeasible solutions.

2.2 Parameter Tuning Studies

The bump test function (P3) was used to tune the parameters of the PSO algorithm. The function is defined as:

$$\begin{aligned} \text{Minimize:} \quad & - \left| \sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i) \right| / \sqrt{\sum_{i=1}^n i x_i^2}, \\ \text{Subject to:} \quad & \prod_{i=1}^n x_i > 0.75, \quad \sum_{i=1}^n x_i < 15n/2, \quad 0 \leq x_i \leq 10. \end{aligned}$$

The parameters tuned include the number of particles (n_p), inertia weight (w), cognitive coefficient (c_1), and social coefficient (c_2). Both static and adaptive penalty methods were tested. Each parameter configuration was evaluated over 10 independent runs, and the best-performing configuration was selected based on convergence trends and solution quality.

2.3 Results for Case Studies

The tuned PSO parameters were applied to additional case studies from the "Case Studies" document. Results include the best solution found, the corresponding objective function value, and the feasibility of the solution. Detailed convergence plots and solution statistics are included for each case study.

3 Regression Problem

3.1 Problem Description

The regression model is given by:

$$\hat{f}(x, w) = w_0 + \sum_{i=1}^{M-1} w_i \phi_i(x),$$

where $x \in \mathbb{R}^D$, $w_i \in \mathbb{R}$, and $\phi_i(x)$ are known basis functions. The weights are estimated by minimizing the regularized least-squares error:

$$\sum_{i=1}^N c_i (\hat{f}(x^{(i)}, w) - y^{(i)})^2 + \sum_{i=1}^{M-1} \lambda_i w_i^2,$$

where $c_i > 0$ and $\lambda_i > 0$ are user-defined parameters.

3.2 Solution

The regularized error function can be written in matrix form as:

$$J(w) = (Aw - b)^T C (Aw - b) + w^T \Lambda w,$$

where A , b , C , and Λ are appropriately defined matrices. Taking the derivative with respect to w and setting it to zero gives the system of linear equations:

$$(A^T C A + \Lambda)w = A^T C b.$$

4 Constraint Relaxation Problem

4.1 Part (a): Minimum Relaxation

The goal is to find the smallest $\delta_j \geq 0$ such that the feasible set is non-empty. The optimization problem is:

$$\begin{aligned} \text{Minimize: } & \sum_{j=1}^m \delta_j, \\ \text{Subject to: } & g_j(x) \leq \delta_j, \quad j = 1, \dots, m. \end{aligned}$$

4.2 Part (b): Least Number of Constraints

To relax the least number of constraints, the optimization problem is:

$$\begin{aligned} \text{Minimize: } & \sum_{j=1}^m z_j, \\ \text{Subject to: } & g_j(x) \leq \delta_j z_j, \quad z_j \in \{0, 1\}, \quad j = 1, \dots, m, \\ & \sum_{j=1}^m \delta_j z_j \geq 0. \end{aligned}$$

This problem is challenging due to its combinatorial nature.

4.3 Part (c): Minimum-Cost Relaxation

For minimum-cost relaxation, the problem is:

$$\begin{aligned} \text{Minimize: } & \sum_{j=1}^m c_j \delta_j, \\ \text{Subject to: } & g_j(x) \leq \delta_j, \quad j = 1, \dots, m. \end{aligned}$$

5 Conclusions

This report demonstrates the implementation and application of PSO for constrained optimization, as well as solutions to regression and constraint relaxation problems. The results show the effectiveness of PSO in solving complex optimization problems.