Cooperative PSO with Spatially Meaningful Neighbors

Peter Wilson

Supervised by Beatrice M. Ombuki-Berman

Submitted in partial fulfillment
of the requirements for COSC 4F90

Department of Computer Science

Brock University

St. Catharines, Ontario

Abstract

Contents

List of Tables

List of Figures

1 Introduction

2 Background

2.1 Particle Swarm Optimization

2.1.1 Overview

Particle Swarm Optimization (PSO) is a population-based computational intelligence algorithm that uses multiple potential solutions (Particles) and a measure of how good those solutions are (Fitness) to find an optimal solution for continuous problems. Like a number of other Computational Intelligence algorithms, PSO is inspired by patterns that are observed in nature. Particle Swarm Optimization specifically draws from the flight patterns of birds and insects in their search of food, who tend to flock together and hone in on an ideal convergence point from various directions.

PSO builds off this metaphor by representing the birds as Particles. In this case, the particles have positions (represented as x and y in 2D space) that directly correlate to the values used in the solution. This allows the potential solution to be easily visualized in a graph,

1.

2.1.2 Position and Velocity Update

2.1.3 Parameters

Table 1: GP Algorithm Pseudo Code

$pop \leftarrow randomPopulation(popSize)$;
for $gen \leftarrow 0, maxGen$ **do**
  $evaluateFitness(pop)$;
  $parents \leftarrow selectBest(pop)$;
  $children \leftarrow applyCrossover(parents)$;
  $applyMutation(children)$;
  $gen \leftarrow gen + 1$;
end for
return $bestIndividual$;

Talk about Adjusted and Standardized fitness (also hits, and about human interpretation)

2.2 Cooperative Particle Swarm Optimization

et al. [1]

2.2.1 Variant: CPSO-S

¡Include algorithms¿

2.2.2 Variant: CPSO-Sk

¡Include algorithms¿

2.2.3 Variant: CPSO-Hk

¡Include algorithms¿

2.2.4 Variant: CPSO-Rk

¡Include algorithms¿

2.3 Delaunay Triangulation

¡discuss the creation as well as the usages¿ ¡include the pseudocode for n-dimensions¿

3 Literature Review

¡discuss the cpso papers and the delaunay triangulation papers¿

$$addgeneralGPsymbolicregressionhere(nofinance)$$

4 Experiments

4.1 Experiment 1: CPSO-S

4.1.1 Setup

4.1.2 Results

4.1.3 Discussion

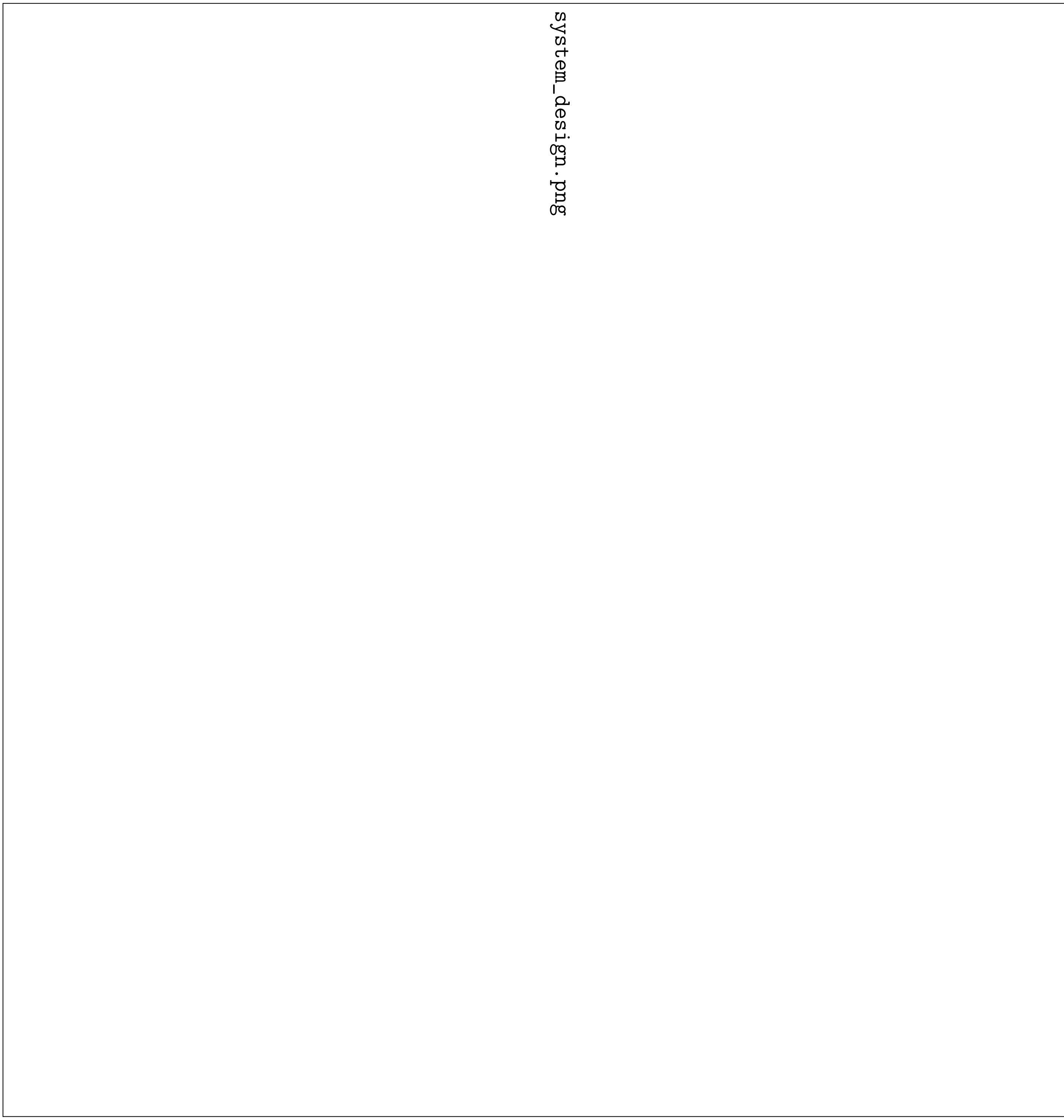

Figure 1: Design of the GPP System.

4.2 Experiment 2: CPSO-Sk

4.2.1 Setup

4.2.2 Results

4.2.3 Discussion

4.3 Experiment 3: CPSO-Hk

4.3.1 Setup

4.3.2 Results

4.3.3 Discussion

4.4 Experiment 4: CPSO-Rk

4.4.1 Setup

4.4.2 Results

4.4.3 Discussion

5 Conclusion

A Preliminary Trials

References

[1] Kalyan Veeramachaneni, Owen Derby, Una-May O'Reilly, and Dylan Sherry. Learning regression ensembles with genetic programming at scale. *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, pages 1117–1124, 2013.