International Journal of Mathematics and Computer Science
Volume **20,** Issue no. 1, (2025), 365–371
DOI: https://doi.org/10.69793/ijmcs/01.2025/sabah

A New Hybrid Heuristic Algorithm of Mathematical Numerical Optimization Based on Population Methods

Ahmed Sabah Ahmed Al-jilawi, Huda Amer Hadi

Department of Mathematics
Faculty of Education College for Pure Sciences
Babylon University
Babylon, Iraq

email: aljelawy2000@yahoo.com, hudahdi@uobabylon.edu.iq

(Received October 21, 2024, Accepted November 21, 2024,
Published November 22, 2024)

Abstract

The purpose of this study is to compare population methods. We present the genetic algorithms and particle swarm algorithm of non-linear optimization which include two classes of heuristic algorithms for solving n-dimensional mathematical optimization problems. This work suggests a new hybrid algorithm which is nests particle swarm optimization (PSA) operations in the genetic algorithm (GA). The new hybrid algorithm provides a better convergence between the exploitation compared and exploration of both parent algorithms. However, the existing hybrid algorithms and achieving consistency provide the best accurate results of the optimal solution with relatively small computational cost.

1 Introduction

The purpose of optimization is to select the optimal values of variables that give the maximum or minimum value of the objective function (cost function)

Key words and phrases: Numerical Optimization, Population Methods, Approximate Algorithms, and Duality Convergence Optimization.
AMS (MOS) Subject Classifications: 65K10, 65Z05.
ISSN 1814-0432, 2025, https://future-in-tech.net

within the constraints of inequality and equality [1]. In recent years, optimization has become increasingly popular with the growing need for quick and exact answers to complex issues in several fields including science, design, manufacturing, Applied Mathematics, and heuristic algorithms. There are two main types of heuristic algorithms: The Particle Swarm and Genetic Algorithm. The Particle Swarm Optimization algorithm (PSO), originally introduced in 1995, is based on flocks of birds and schools of fish [3]. The particle swarm consists of randomly initiated candidate solutions, or particles, that move in a multidimensional space. Each particle has a velocity and so information about the best position in a neighborhood is quite important. The first Genetic Algorithm (GA) was used in 1975 by J. H. Holland. It is based on the idea of evolution through random mutation and natural selection [6].

2 The Conic Heuristic algorithms of Optimization

In conic optimization, there are different ways to present the duality of solution of large-scale optimization problems with one strategy being partitioning [5]. We can divide the variables into two subsets: In one subset, minimize first while, in the other subset, fixing the variable [2].

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + g(Bx) \\ \text{subject to} \quad & x \in \mathbb{R}^n \end{aligned} \tag{2.1}$$

where B is an $m \times n$ matrix, $f : \mathbb{R}^n \to (-\infty, \infty]$ and $g : \mathbb{R}^m \to (-\infty, \infty]$ are proper convex functions. Suppose there exists a feasible solution; i.e., an $x \in \mathbb{R}^n$ such that $x \in \text{dom}(f)$ and $Bx \in \text{dom}(g)$.

Then, the problem is equivalent to the constrained optimization problem.

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + g(Bx) \\ \text{subject to} \quad & x_1 \in \text{dom}(f) \text{ and } Bx_2 \in \text{dom}(g), \\ & x_2 = Bx_1. \end{aligned} \tag{2.2}$$

With the convex programming problem of the linear equality constraint $x_2 = Bx_1$, we obtain the dual linear function as

$$\begin{aligned} h(\lambda) &= \inf_{x_1 \in \mathrm{dom}(f)\ ,\ Bx_2 \in \mathrm{dom}(g)} \Big\{ f(x_1) + g(x_2) + \lambda^T(x_2 - Bx_1) \Big\} \\ &= \inf_{x_1 \in \mathbb{R}^n} \Big\{ f(x_1) - \lambda^T B x_1 \Big\} + \inf_{x_2 \in \mathbb{R}^n} \Big\{ f(x_2) + \lambda^T x_2 \Big\} \\ &= \inf_{x_1 \in \mathbb{R}^n} \Big\{ -(B^T\lambda)^T x_1 - f(x_1) \Big\} + \inf_{x_2 \in \mathbb{R}^n} \Big\{ -(-\lambda)^T x_2 - g(x_2) \Big\} \\ &= -\sup_{x_1 \in \mathbb{R}^n} \Big\{ (B^T\lambda)^T x_1 - f(x_1) \Big\} - \sup_{x_2 \in \mathbb{R}^n} \Big\{ (-\lambda)^T x_2 - g(x_2) \Big\} \\ &= -f^*(B^T\lambda) - g^*(-\lambda). \end{aligned} \tag{2.3}$$

Therefore,

$$\max_{\lambda} h(\lambda) = -f^*(B^T\lambda) - g^*(-\lambda).$$

The dual problem of maximizing h over $\lambda \in \mathbb{R}^n$ is converted to a minimization problem by a sign change:

$$\begin{aligned} &\underset{x}{\text{minimize}} && f^*(B^T\lambda) + g^*(-\lambda) \\ &\text{subject to} && \lambda \in \mathbb{R}^m, \end{aligned} \tag{2.4}$$

where f^* and g^* are the conjugate functions of the functions f and g. Moreover, f^* and h^* represent the corresponding optimal dual and prime values.

3 Numerical Result

We implement the numerical result based on Python software. The performance of the new hybrid algorithm shows that the particle swarm algorithm generally converges faster than genetic algorithms in the early iterations. However, the particle swarm algorithm reaches a good optimal solution more quickly, as seen by the steeper initial decline in the figure 1 where both algorithms perform well on this simple sphere function. For more complex problems, the particle swarm algorithm might show better robustness due to its population-based approach and ability to maintain diversity (see Figure (2). Both algorithms have similar computational requirements per iteration. The choice between them may depend more on the specific problem characteristics than on computational cost and Robustness.

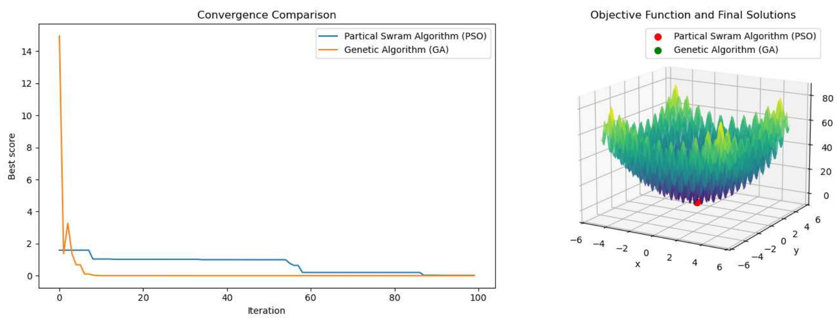

Figure 1: The Cost Function (Objective Function) of Particle Swarm Optimization Via The Genetic Algorithm

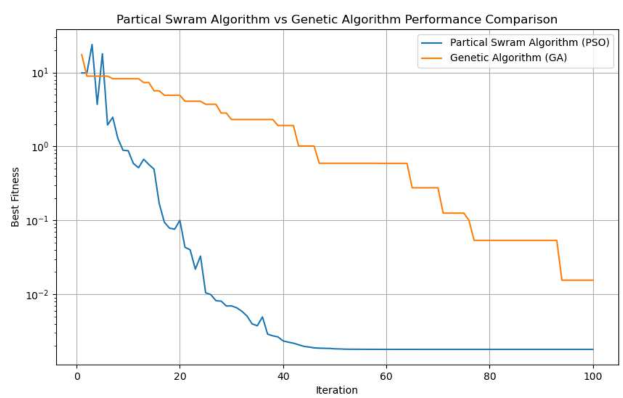

Figure 2: Comparison Convergence of Particle Swarm Optimization Via The Genetic Algorithm

4 Development of Theoratical Convergence

In this section, we present the heuristic algorithm optimizations as a subfield of convex linear programming. Let S^n denote the set of symmetric $n \times n$ matrices: $S^n = \{X \in \mathbb{R}^n \ X = X^T\}$[3].

Theorem 4.1. *Consider the linear programming problem with probability constraints which also can be the objective function as a cone optimization programming*

$$(LP)\begin{cases} (min)_x & \langle c, x_0\rangle \\ s.t. & \mathbb{P}\Big(\langle a_i, x\rangle \le r_i\Big) \ge H \quad i=1,\dots,m, \\ & x \ge 0, \quad x \in \mathbb{R}^n, \end{cases} \tag{4.5}$$

where $\langle x, y\rangle$ *is the inner product of two vectors* x *and* y *in* $\mathbb{R}^n$, $x \ge 0$ *means that* $x_i \ge 0$, *for* $i = 1, \dots, n$, $\mathbb{P}$ *be a probability measure. Then* $H \in [0.5, 1]$.

Proof.
First of all, we discuss the constraint $\mathbb{P}\Big(\langle a_i, x\rangle \le r_i\Big) \ge H$, for given $x \in \mathbb{R}^n$. Assume $k_i = \langle a_i, x\rangle$, where $k_i \in \mathbb{R}$ is a scalar random variable which implies the sum of Gaussian random variables with mean $\bar{k}_i = \langle \bar{a}_i, x\rangle$ and variance $\beta_i = x^T \sum_i x_i$. The probability can be computed with cumulative distribution functions $\mathcal{F} : \mathbb{R} \to [0, 1]$. Then, with the random variable $\frac{k_i - \bar{k}_i}{\sqrt{\beta_i}}$,

$$\mathbb{P}\Big(\langle a_i, x\rangle \le r_i\Big) \ge H \iff \Big(\frac{k_i - \bar{k}_i}{\sqrt{\beta_i}} \le \Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big) = \mathcal{F}\Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big).$$

Then, the feasible solution must satisfy the constraint $\mathcal{F}\Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big) \ge H$. Assume $\mathcal{F}^{-1} : [0, 1] \to \mathbb{R}$ is the inverse of $\mathcal{F}$. By the composition rule, we obtain

$$\mathcal{F}\Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big) \ge H \iff \mathcal{F}^{-1} \ \Big(\mathcal{F}\Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big)\Big),$$

the inverse is the identity map based on the composition of the function. Therefore,

$$\mathcal{F}^{-1}(H) \iff \Big(\frac{r_i - \bar{k}_i}{\sqrt{\beta_i}}\Big) \ge \mathcal{F}^{-1}(H).$$

Multiplying both sides by $\sqrt{\beta_i}$, we get

$$r_i - \bar{k}_i \ge \mathcal{F}^{-1}(H)\sqrt{\beta_i} \to r_i - \bar{k}_i = r_i - \langle a_i, x\rangle\,,$$

which is an affine function of x. In fact, $\sqrt{\beta_i} = \sqrt{x^T \sum_i x_i} = \|\sum_i{}^{\frac{1}{2}} x_i\|_2$ by a duality technique. Hence, the dual problem is

$$(DP)\begin{cases} \max_x & \langle c, x\rangle \\ \text{s.t} & r_i - \langle a_i, x\rangle \le r_i) \ge \mathcal{F}^{-1}(H)\|\sum_i^{\frac{1}{2}} x_i\|_2 \quad i=1,\dots,m, \\ & x \ge 0, \quad x \in \mathbb{R}^n, \end{cases} \tag{4.6}$$

On the other hand, ($H \geq 0.5$) such that ($\mathcal{F}^{-1}(H) \geq 0$) that is defined as cone optimization programming

Numerical Computation		
Iteration	Partical Swram Algorithm Best	Genetic Algorithm Best
1	50.836875	22.389485
2	103.450366	22.389485
3	10.346770	22.389485
16	0.028319	2.454720
17	0.044280	2.454720
18	0.002345	0.004567
19	0.002345	0.004567
20	0.002345	0.004567

Table 1: The Best fitness value found by PSA and GA up to that iteration

5 Conclusion

The aim of this study was to develop the hybrid approximate Algorithm that converges faster. The strategy we used was to optimize the duality over Gaussian random variables. This study depended on the population methods: particle swarm algorithm and genetic algorithm. The implemented method showed that the particle swarm algorithm outperforms the genetic algorithm in terms of convergence speed and final solution quality to reach the optimal solution region. The results provided the optimal solution to the initial programming problem (LP) as a conic optimization, which is the upper bound for solving the dual programming problem (DP).

References

[1] M. F. Anjos, J. B. Lasserre, (Eds.), Handbook on semidefinite, conic and polynomial optimization, **166,** Springer Science & Business Media, (2011).

[2] D. Henrion, E. Pauwels, Linear conic optimization for nonlinear optimal control, arXiv preprint:1407.1650.n, **166,** Springer Science & Business Media, (2014).

[3] H. Zhang, X. Sun, G. Li, On second-order conic programming duals for robust convex quadratic optimization problems, Journal of Industrial and Management Optimization, **19,** no. 11, (2023), 8114–8128.

[4] A. Nemirovski, Advances in convex optimization: conic programming. In International Congress of Mathematicians, **1,** (2006), 413–444.

[5] A. D. A. Al-Zamili, A. S. A. Aljilawi, Using Alternating Direction Method of Multipliers to solve optimization problem in Statistics, Computer Science, **19,** no. 1, (2024), 157–162.

[6] A. D. A. Al-Zamili, A. S. A. Aljilawi, Optimizing Logistic Regression with Enhanced Convergence: A Modified ADMM Approach. Computer Science, **20,** no. 1, (2025), 9–15.