An enhanced Gravitational Search Algorithm suited for the optimization of space missions

J. González[♭,1], A. Herrero[♮], S. Moll[a] and J. A. Moraño[♭]

(♭) E.T.S.I. Aeroespacial y Diseño Industrial, Universitat Politècnica de València,
(♮) I.U. de Matemática Multidisciplinar, Universitat Politècnica de València,
(a) Departamento de Matemática Aplicada, Universitat Politècnica de València,
Camí de Vera s/n, València, Spain.

1 Introduction

In the field of engineering and applied mathematics, optimizing an objective function—often referred to as cost or fitness—is a fundamental procedure. For single-variable functions, this process is relatively straightforward, requiring the calculation of the first-order derivative and setting it to zero. However, the complexity increases significantly for multi-variable functions, often necessitating the use of various optimization techniques, such as Lagrange multipliers. These traditional methods generally assume that the function to be optimized can be expressed analytically, a condition that is seldom met in practical engineering problems. Consequently, numerical methods are frequently employed. These methods typically involve computing the numerical gradient of the function to find points where the gradient is zero.

This approach, however, is highly sensitive to the initial point where the gradient is calculated, particularly if the function has multiple local minima. Additionally, gradient-based algorithms face significant challenges when dealing with functions that exhibit discontinuities, such as sudden steps or undefined regions. These limitations make it necessary to develop alternative optimization methods, leading to the emergence of meta-heuristics algorithms [1, 2, 3]. These algorithms are designed to overcome these challenges by employing strategies that do not rely on gradient information. They often mimic natural phenomena, leveraging the inherent optimization capabilities observed in nature.

This research introduces the fundamentals of the Gravitational Search Algorithm (GSA), a meta-heuristic method inspired by Newton's universal law of gravity[4]. The GSA models optimization as a system of masses interacting through gravitational forces, with the objective function acting as a gravitational field influencing the movement of these masses. The gravitational force between two masses m_1 and m_2 separated by a distance $\|\mathbf{r}_{12}\|$ is given by:

$$\mathbf{F}_{12} = G\frac{m_1 m_2}{\|\mathbf{r}_{12}\|^2}\,\mathbf{r_{12}}. \tag{1}$$

In this context, each candidate solution is treated as an object with mass, and the optimization process simulates the gravitational attraction between these objects. The positions of the objects in the search space are updated iteratively based on the gravitational forces, guiding the search

[1] jagonga5@etsid.upv.es

towards optimal solutions. In general cases, the mass of each object is dynamically adjusted based on its fitness value, ensuring that better solutions exert stronger attractive forces and thus have a greater influence on the search process.

2 GSA models

GSA begins with N initial points randomly distributed across the search space, which are treated as point masses. Each mass is assigned based on its fitness function evaluation:

$$m_i(t) = \frac{fitness_i(t) - worst(t)}{best(t) - worst(t)}, \tag{2}$$

where $fitness_i$ is the cost function evaluation for the i-th mass, $worst$ and $best$ are the observed worst and best values, respectively, and t is the current iteration. Once the masses are determined, their acceleration is computed using a modified version of Newton's law:

$$\mathbf{a}_i(t) = \sum_{\substack{j \in K_{best} \\ j \neq i}}^{N} (\mathbf{rand})_j \odot \frac{G(t)\ m_j}{\|\mathbf{r}_{ij}\| + \epsilon}\ \mathbf{r}_{ij}. \tag{3}$$

Here, $(\mathbf{rand})_j$ is a vector of random numbers in the interval [0,1], $\odot$ denotes the Hadamard product, ϵ is a small number to avoid numerical errors when $\mathbf{r}_{ij} \to 0$, and $\mathbf{r}_{ij} = \mathbf{X}_j(t) - \mathbf{X}_i(t)$ with $\mathbf{X}_i(t)$ is the position vector of the i-th mass. The inclusion of a randomized coefficient introduces stochasticity, helping to avoid convergence to local minima. The Euclidean distance in the denominator is not squared, which provides better overall results. Additionally, the acceleration calculation considers only the K_{best} masses, reducing linearly from N to 1 over time.

The acceleration vector is integrated twice to update the position for the next iteration $t+1$:

$$\mathbf{v}_i(t+1) = (\mathbf{rand})_i \odot \mathbf{v}_i(t+1) + \mathbf{v}_i(t) \tag{4}$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \tag{5}$$

The time-varying gravitational constant $G(t)$ in Equation (3) is crucial, as a higher value increases the gravitational forces among agents. It is typically expressed as:

$$G(t) = G_0 \cdot e^{-\alpha \cdot t/T} \tag{6}$$

where T is the total number of iterations, and the parameters G_0 and α are usually set to 100 and 20, respectively. Higher $G(t)$ values are used initially for global exploration, decreasing gradually for local exploitation.

The objective of this study is to enhance the GSA for optimizing space missions near Earth by determining the optimal values of G_0 and α, which vary due to the algorithm's stochastic nature. These enhancements aim to improve the performance and reliability of the GSA in practical applications.

3 Bayesian Optimization

Bayesian optimization is a powerful strategy for identifying the extrema of objective functions that are expensive to evaluate [5]. This method is particularly valuable in scenarios where a closed-form expression for the objective function is unavailable, but observations can be obtained at sampled values. It is especially effective for costly evaluations, non-convex problems, and situations where derivatives are inaccessible. This approach is analogous to hyper-parameter tuning in machine learning algorithms, further justifying its application in meta-heuristic optimization [6].

Bayesian optimization is based on Bayes' theorem:

$$P(M|E) \propto P(E|M) \cdot P(M) \tag{7}$$

In this context, M represents an estimate of the unknown function, and E denotes the observations. Bayes' theorem provides a framework for updating the probability of a model given new evidences. Bayesian optimization involves constructing a prior surrogate model of the function and iteratively selecting points in the search space for evaluation. These points are chosen using an acquisition function, which balances exploration and exploitation, making this method well-suited for expensive-to-evaluate functions. As an example, this process can be graphically seen in Figure 1 for the case of a 1D problem.

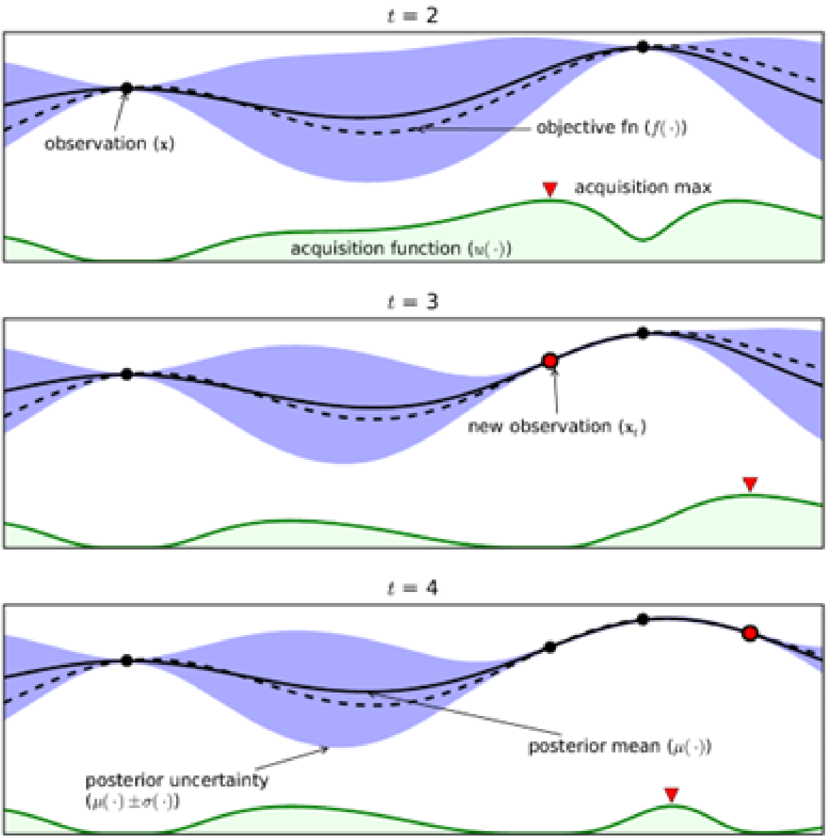

Figure 1: Application of Bayesian optimization to a 1D problem

4 Some Applications

To optimize space trajectories, we define an objective function for the Bayesian optimization process that minimizes the average ΔV impulse error from multiple Hohmann transfers [7, 8, 9]. This involves evaluating different values of T and N, the number of masses, to determine their impact on the parameters G_0 and α. To ensure the algorithm's robustness, the search space is adimensionalized, standardizing all dimensions to the range [0,1].

The results, exemplified in Figure 2, illustrate the average error as a function of G_0 and α for specific values of N and T. By evaluating these parameters across different Hohmann transfers, we can assess their effectiveness and reliability.

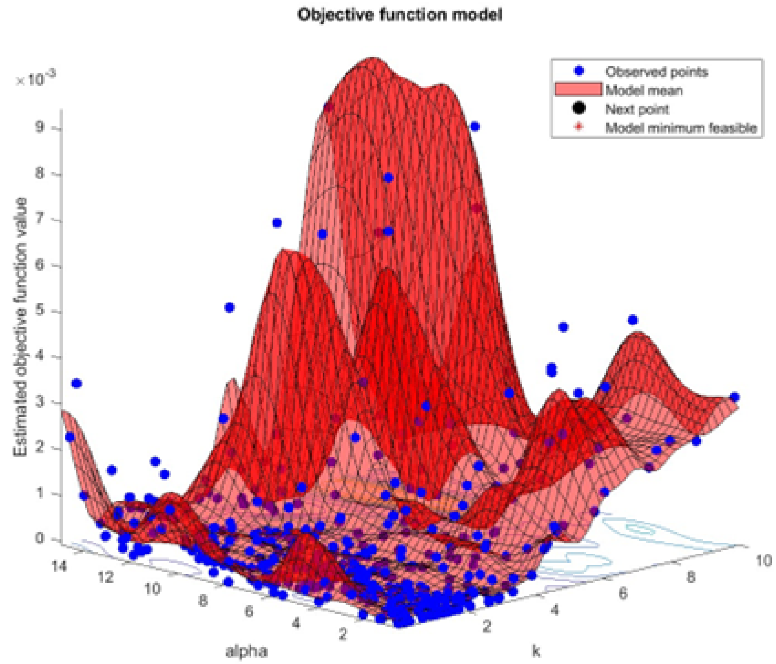

Figure 2: Results example of a Bayesian optimization process.

A practical application, such as a water-propelled satellite designed to collect space debris, showcases the method's ability to significantly improve fuel efficiency. Additionally, we discuss the implications of the No Free Lunch Theorems [1] and the potential benefits of hybridizing this approach with other optimization techniques.

References

[1] Wolpert, D.H., Macready, W.G., "No free lunch theorems for optimization". *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[2] Brochu, E., Cora, V., Freitas, N., "A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning". *CoRR*, abs/1012.2599, 2010. `https://doi.org/10.48550/arXiv.1012.2599`.

[3] Wang, J., "An intuitive tutorial to gaussian process regression". *Computing in Science and Engineering*, 25(4):4–11, 2023.

[4] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., "GSA: A gravitational search algorithm". *Information Sciences*, 179(13):2232–2248, 2009.

[5] Wang, W., Bayesian Optimization Concept Explained in Layman Terms.

[6] Snoek, J., Larochelle, H., Adams, R.P., "Practical Bayesian Optimization of Machine Learning Algorithms". In: Advances in Neural Information Processing Systems. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. `https://doi.org/10.48550/arXiv.1206.2944`.

[7] Curtis, H.D., Orbital Mechanics for Engineering Students: Revised fourth edition. Elsevier, 2021.

[8] Moraño Fernández, J.A., Maniobras orbitales impulsivas. Universitat Politècnica de València, 2021. http://hdl.handle.net/10251/164022.

[9] Tatay Sangüesa, J., "Multiobjective Optimization of Impulsive Orbital Trajectories". Bachelor's thesis. Universitat Politècnica de València. http://hdl.handle.net/10251/124982.