A Peek into the Swarm: Analysis of the Gravitational Search Algorithm and Recommendations for Parameter Selection

Florian Knauf
Hochschule Hannover
Department of Computer Science
Hannover, Germany
f.knauf@mmkf.de

Ralf Bruns
Hochschule Hannover
Department of Computer Science
Hannover, Germany
ralf.bruns@hs-hannover.de

ABSTRACT

The Gravitational Search Algorithm is a swarm-based optimization metaheuristic that has been successfully applied to many problems. However, to date little analytical work has been done on this topic.

This paper performs a mathematical analysis of the formulae underlying the Gravitational Search Algorithm. From this analysis, it derives key properties of the algorithm's expected behavior and recommendations for parameter selection. It then confirms through empirical examination that these recommendations are sound.

CCS CONCEPTS

• **Theory of computation** → **Optimization with randomized search heuristics**; **Theory of randomized search heuristics**; • **Mathematics of computing** → **Optimization with randomized search heuristics**; *Bio-inspired optimization*; • **Computing methodologies** → *Randomized search.*

KEYWORDS

Swarm intelligence, Parameter tuning and algorithm configuration, Metaheuristics, Theory

ACM Reference Format:
Florian Knauf and Ralf Bruns. 2019. A Peek into the Swarm: Analysis of the Gravitational Search Algorithm and Recommendations for Parameter Selection. In *Genetic and Evolutionary Computation Conference (GECCO '19), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3321707.3321774

1 INTRODUCTION

Due to their effectiveness and generality, swarm-based metaheuristics have become an important tool for the solution of global optimization problems. For many of these algorithms, this effectiveness has been demonstrated empirically many times over, but analysis of their inner workings and the circumstances in which they can be expected to be most effective often remains sparse [20]. As such, parameter selection remains a challenge facing every user of these optimization techniques.

One such algorithm is the Gravitational Search Algorithm (GSA). Since its invention by Rashedi et al. in 2009 [17], it has spawned a fair number of variations and hybridizations [18] and was successfully applied to many different problems such as cancer research [1], robotics [4], power-flow study [11], and others [18].

This paper derives key mathematical properties from GSA's definition and uses them to make prediction about its behavior. It uses these predictions to make recommendations for parameter selection and demonstrates by means of measurements on three test functions that the predicted behavior is observable and that the recommendations are sound.

The main contribution is a parameter selection technique for GSA that enables its calibration in accordance with a desired result precision and the extents of the search domain. Secondary contributions are a method to estimate the viability of a selected set of parameters as well as insights about the GSA swarm's inner workings that can be useful in the decision for or against the use of GSA.

The remainder of this paper is structured as follows: Section 2 describes GSA and its mathematical underpinnings. These are analyzed in section 3 to derive the aforementioned properties and recommendations. Section 4 demonstrates the validity of these recommendations by means of measurements on test functions. Section 5 places the paper in its wider scientific context, and section 6 summarizes the results, draws final conclusions, and suggests avenues for future research.

2 THE GRAVITATIONAL SEARCH ALGORITHM

GSA is a swarm-based metaheuristic optimization method that is (loosely [8]) inspired by the physical phenomenon of gravity. As such, it attempts to locate a global minimum (or at least a deep local minimum) in a fitness function $f : \mathbb{R}^n \to \mathbb{R}$ by moving a swarm of N particles in $\mathbb{R}^n$ iteratively over T steps. Better-positioned particles are assigned a higher mass, and their movement is governed by equations similar to those of Newton's law of gravity.

To this end, GSA defines a swarm of particles $P_1, \ldots, P_N$. Each particle $P_i = (\vec{x}_i, \vec{v}_i)$ is associated with a position $\vec{x}_i$ and velocity $\vec{v}_i$. In the reference implementation, $\vec{x}_i$ is initially chosen at random in a search domain $\mathcal{F} \subseteq \mathbb{R}^n$ (usually some interval $[a, b]^n$) while $\vec{v}_i$ is the zero vector [16].

During each step of the optimization process, $\vec{v}_i$ and $\vec{x}_i$ are updated for every P_i, and f is evaluated at $\vec{x}_i$. The result is the position $\vec{x}_*$ such that $f(\vec{x}_*)$ is minimal among all considered positions.

We denote the state of particle P_i at time t as $P_i(t) = (\vec{x}_i(t), \vec{v}_i(t))$ and $f_i(t) = f(\vec{x}_i(t))$, where $0 \leq t \leq T$. In order to calculate the

GECCO '19, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6111-8/19/07…$15.00
https://doi.org/10.1145/3321707.3321774

velocity update (i.e. acceleration) acting on the particles, every particle P_i is assigned a mass $M_i(t)$ according to

$$m_i(t) := \frac{f_{\text{worst}}(t) - f_i(t)}{f_{\text{worst}}(t) - f_{\text{best}}(t)} \tag{1}$$

$$M_i(t) := \frac{m_i(t)}{\sum_{k=1}^{N} m_k(t)} \tag{2}$$

where $f_{\text{worst}}(t) = \max_{1 \leq k \leq N} f_k(t)$, $f_{\text{best}}(t) = \min_{1 \leq k \leq N} f_k(t)$. The force of attraction between particles P_i, P_j is then given by

$$\vec{F}_{ij}(t) = G(t) M_i(t) M_j(t) \frac{\vec{x}_j(t) - \vec{x}_i(t)}{\|\vec{x}_j(t) - \vec{x}_i(t)\| + \varepsilon} \tag{3}$$

where ε ist a very small number[1] and

$$G(t) = G_0 e^{-\alpha \frac{t}{T}} \tag{4}$$

is the gravitational "constant" at time t. G_0 and α are parameters used for calibration.

The combined force $\vec{F}_i(t)$ acting on particle P_i is a randomly weighted sum of the forces exerted on P_i by the heaviest (best-placed) particles. Let $K(t) = \lceil \left(1 - \frac{t}{T}\right) N \rceil$ and $K_{\text{best}}(t)$ be the set of indices of the $K(t)$ heaviest particles at time t, then

$$\vec{F}_i(t) = \sum_{j \in K_{\text{best}}(t), i \neq j} \mathbf{R}_{ij}(t) \vec{F}_{ij}(t) \tag{5}$$

where $\mathbf{R}_{ij}(t)$ is a random diagonal matrix with diagonal elements uniformly distributed in $[0, 1]$ (i.e. $r_{k,k} \sim U(0,1) \forall k$). This means that initially all particles exert force on all other particles while at the end only the single heaviest particle does; the threshold is lowered linearly over time [17].

The acceleration follows from this according to Newton's second Law of Motion:

$$\vec{a}_i(t) = \frac{1}{M_i(t)} \vec{F}_i(t) \tag{6}$$

Finally, the velocity and position of P_i are updated by

$$\vec{v}_i(t+1) := \mathbf{R}_i(t) \vec{v}_i(t) + \vec{a}_i(t) \tag{7}$$

$$\vec{x}_i(t+1) := \vec{x}_i(t) + \vec{v}_i(t+1) \tag{8}$$

where again $\mathbf{R}_i(t)$ is a random diagonal matrix like $\mathbf{R}_{ij}(t)$. Particles that leave $\mathcal{F}$ are randomly reinitialized. The process continues until $t = T$ or a stop criterion is reached.

3 MATHEMATICAL ANALYSIS

This section analyzes the GSA construction mathematically, complementing the analysis with experimental data where appropriate. In doing so, it derives key properties of GSA's behavior in order to develop suggestions for parameter selection.

3.1 Mass

Equation (2) is reminicient of selection mechanisms in other swarm algorithms (e.g. attractor selection in Glowworm Swarm Optimization [19]) and genetic algorithms (e.g. fitness-proportional roulette wheel selection [10]). Here as there, its purpose is to assign greater but not total influence to well-positioned search particles in order to promote the exploitation of their discoveries without immediate abandonment of worse-positioned particles' exploratory potential.

In GSA, (2) has three obvious (but crucial to sections 3.2, 3.3) effects: For all t,

$$\min\{M_i(t), 1 \leq i \leq N\} = 0 \tag{9}$$

$$\max\{M_i(t), 1 \leq i \leq N\} \leq 1 \tag{10}$$

$$\sum_{i=1}^{N} M_i(t) = 1 \tag{11}$$

Note that (6) can be rewritten without $M_i(t)$ in the denominator so that $\vec{a}_i(t)$ is not undefined in the case described by (9). There is a special case when the swarm is entirely located within a plateau in the fitness function so that $f_i(t) = f_j(t)$ for all $1 \leq i, j \leq N$ and (1) yields undefined results.[2] In this case, all particles are equally well positioned and should be assigned the same mass, so implementations should set $M_i(t) = \frac{1}{N} \; \forall \; i$ to be consistent with (11).

More importantly, (1) is invariant under linear transformation of f: let $f' = af + b$, then

$$\begin{aligned} m'_i &= \frac{f'_{\text{worst}} - f'_i}{f'_{\text{worst}} - f'_{\text{best}}} = \frac{af_{\text{worst}} + b - af_i - b}{af_{\text{worst}} + b - af_{\text{best}} - b} \\ &= \frac{a(f_{\text{worst}} - f_i)}{a(f_{\text{worst}} - f_{\text{best}})} = \frac{f_{\text{worst}} - f_i}{f_{\text{worst}} - f_{\text{best}}} = m_i \end{aligned} \tag{12}$$

Because this is the only place where f is evaluated, the behavior of GSA as a whole is invariant under linear transformation of f.

3.2 Force and Acceleration

One place in which GSA diverges significantly from the physics that inspired it is (3). Gauci et al. note that where the gravitational force exerted in Newtonian physics decreases quadratically with distance, in GSA it does not decrease with distance at all [8].

In addition to this, $\vec{F}_i(t)$ tends to have greater magnitude when the individual force vectors $\vec{F}_{ij}(t)$ have similar directions, which is more the case for particles at the outer limits of the swarm than those at its center. This means that outlier particles are those most strongly drawn towards the center of the swarm, in stark contrast to natural gravity. As a result, in GSA the swarm is unable to split into subswarms and exploit multiple optima simultaneously.

Figure 1 shows this effect on the Rastrigin test function in $\mathbb{R}^2$, with $G_0 = 20, \alpha = 20, N = 50, T = 1000$. Despite an abundance of local minima in the fitness function, the swarm maintains its integrity. It does not exploit multiple minima simultaneously but sprawls as one entity over a quickly contracting area. In doing so, it may wander from optimum to optimum but never disintegrates.

[1] $\varepsilon \approx 10^{-16}$ in the reference implementation [16]

[2] This can, for example, be an issue when GSA is used to handle discrete problems with nearest-integer coding [13].

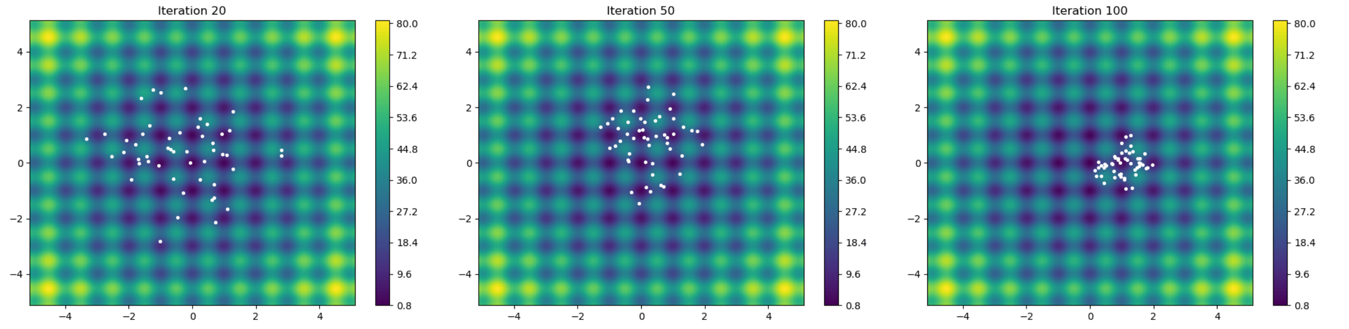

Figure 1: Collapsing swarm in the 2D Rastrigin function

Figure 2 shows the trajectories of 15 particles during the step at time $t = 20$ (center) alongside the swarm configuration at the beginning of that step (left). As predicted, particles tend to cross through the central region of the swarm rather than follow orbital motion, and particles further removed from that central region tend to have longer trajectories. Often particles jump directly from one side of the swarm to the other.

Also shown is the trajectory of a single particle over the next 15 steps (right). The particle rarely moves in the same direction on consecutive steps and almost always undergoes a radical change in direction. This is because once P_i has jumped across the swarm, $\vec{v}_i$ and $\vec{a}_i$ have roughly opposite directions. If the influence of $\vec{v}_i$ is not dampened enough by $\mathbf{R}_i$ in (7), $\vec{a}_i$ and $\vec{v}_i$ nearly cancel each other out, keeping the particle in the same general area for two consecutive steps. However afterwards $\vec{v}_i$ is small, and $\vec{a}_i$ determines the next step alone. Consequently, particle movement is dominated by $\vec{a}_i$, and $\vec{v}_i$ is unable to meaningfully accumulate magnitude.

Apart from these observations, we can derive upper bounds for the force and acceleration vectors throughout the optimization process. Let $\tilde{x}_{ij}(t) = \frac{\vec{x}_j(t)-\vec{x}_i(t)}{\|\vec{x}_j(t)-\vec{x}_i(t)\|+\varepsilon}$, then (5) can be rewritten as

$$\vec{F}_i(t) = G(t)M_i(t) \sum_{j\in K_{\text{best}}(t), i\neq j} M_j(t)\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t) \tag{13}$$

Observing (11) and noting that $\|\tilde{x}_{ij}(t)\| \leq 1$, for all i, t

$$\|\vec{F}_i(t)\| \leq G(t)M_i(t) \tag{14}$$

and therefore with (6)

$$\|\vec{a}_i(t)\| \leq G(t) \tag{15}$$

This insight can be extended to the expected value of $\|\vec{a}_i(t)\|$. Note that the distribution of force is not invariant under rotation, since the transformation $\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)$ is not spherically symmetrical. In particular, the expected length of $\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)$ varies depending on the angle between $\tilde{x}_{ij}(t)$ and the axes of the coordinate system:

Let $\hat{x}_{ij}(t) = \frac{\vec{x}_j(t)-\vec{x}_i(t)}{\|\vec{x}_j(t)-\vec{x}_i(t)\|}$ and $X_k \sim U(0,1), 1 \leq k \leq n$ so that $\hat{x}_{ij}(t) \approx \tilde{x}_{ij}(t)$ and

$$\|\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)\| \approx \|\mathbf{R}_{ij}(t)\hat{x}_{ij}(t)\| = \sqrt{\sum_{k=1}^{n} X_k^2 \hat{x}_{ij,k}(t)^2} \tag{16}$$

where $\hat{x}_{ij,k}(t)$ denotes the k-th component of $\hat{x}_{ij}(t)$.

Note here that $\hat{x}_{ij}(t)$ is a unit vector. The expected value of (16) varies between the extremes when $\hat{x}_{ij}(t)$ is one of the axes of the coordinate system and the case when $\hat{x}_{ij}(t) = \frac{1}{\sqrt{n}}(\pm 1, \ldots, \pm 1)^T$, i.e. when the angle between $\hat{x}_{ij}(t)$ and the closest axis is maximized.

In the former case, $\|\mathbf{R}_{ij}(t)\hat{x}_{ij}(t)\| = \sqrt{X_k^2} = X_k$ for some k and $E\left(\|\mathbf{R}_{ij}(t)\hat{x}_{ij}(t)\|\right) = E\left(X_k\right) = \frac{1}{2}$. In the latter case, $\|\mathbf{R}_{ij}(t)\hat{x}_{ij}(t)\| = \sqrt{\frac{1}{n}\sum_{k=1}^{n} X_k^2}$. Let $Y = \frac{1}{n}\sum_{k=1}^{n} X_k^2$. Since $E(X_k^2) = \frac{1}{3}\forall k$ is familiar as the 2nd raw moment of $U(0,1)$, $E(Y) = \frac{1}{n}\sum_{k=1}^{n} E(X_k^2) = \frac{1}{3}$. It follows that

$$E(\sqrt{Y}) = \sqrt{E(Y) - \mathrm{Var}(\sqrt{Y})} \leq \sqrt{E(Y)} = \frac{1}{\sqrt{3}} \tag{17}$$

through the algebraic variance formula. Since X_k are i.i.d., in accordance with the central limit theorem Y approaches a normal random variable with shrinking $\mathrm{Var}(Y) = \frac{1}{n}\mathrm{Var}(X_k^2) = \frac{4}{45n}$ as n grows.[3] $\mathrm{Var}(\sqrt{Y})$ shrinks accordingly from $\mathrm{Var}(\sqrt{Y}) = \frac{1}{12}$ for $n = 1$, when $\sqrt{Y} = X_1$. Therefore

$$\frac{1}{2} \lesssim E\left(\|\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)\|\right) \leq \frac{1}{\sqrt{3}} \tag{18}$$

In combination with (13) and (6), this means that even if all $\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)$ have exactly the same direction,

$$E\left(\|\vec{F}_i(t)\|\right) \leq \frac{G(t)M_i(t)}{\sqrt{3}} \tag{19}$$

$$E\left(\|\vec{a}_i(t)\|\right) \leq \frac{G(t)}{\sqrt{3}} \tag{20}$$

Furthermore, at the end of the optimization process when only the best-positioned particle P_j exerts influence on P_i,

$$\vec{a}_i(T) = G(T)M_j(T)\mathbf{R}_{ij}(T)\tilde{x}_{ij}(T) \tag{21}$$

Since P_j is the heaviest particle, $\frac{1}{N} \leq M_j(t) \leq 1$, and therefore

$$\frac{G_0 e^{-\alpha}}{2N} = \frac{G(T)}{2N} \lesssim E\left(\|\vec{a}_i(T)\|\right) \leq \frac{G(T)}{\sqrt{3}} = \frac{G_0 e^{-\alpha}}{\sqrt{3}} \tag{22}$$

for all but P_j itself, where $\vec{a}_j(T) = \vec{0}$. Given the observation that particle movement is dominated by short-term acceleration rather than a build-up of velocity over several steps, equations (20) and

[3] $\mathrm{Var}(X_k^2) = E(X_k^4) - E(X_k^2)^2 = \frac{4}{45}$. $E(X_k^4) = \frac{1}{5}$ is the 4th raw moment of $U(0,1)$.

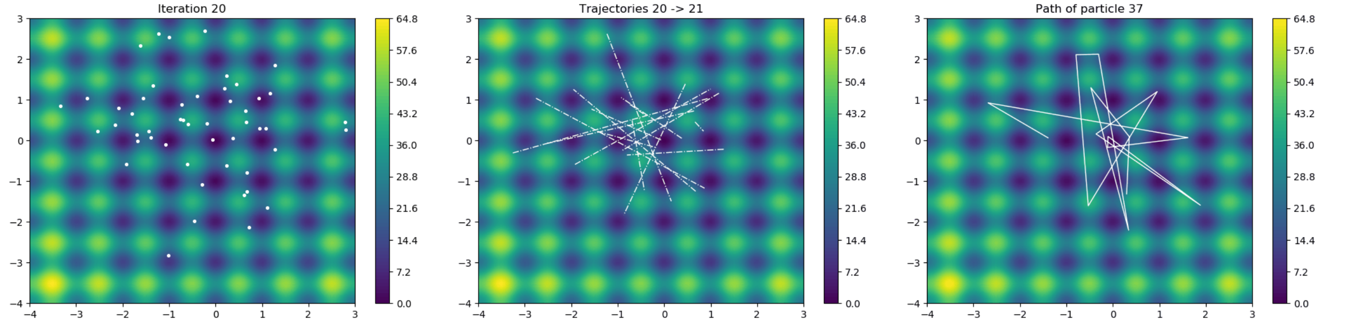

Figure 2: Swarm at $t = 20$ (left), sample trajectories during the next (center) and following 15 iterations (right)

(22) show that the progression of $G(t)$ governs particle mobility. Its effect is akin to that of the mutation rate in evolutionary algorithms.

In the same vein, (22) relates to the step size near the end of the optimization process. This limits the precision to which an optimum can be located, and parameters G_0, α must be chosen accordingly.

3.3 Velocity

The particle velocity $\|\vec{v}_i(t)\|$ tends to decrease over the course of the optimization, and this would be the case even if all $\mathbf{R}_{ij}(t)\tilde{x}_{ij}(t)$, $\mathbf{R}_i(t)\vec{v}_i(t)$ had the same direction: the damping of $\vec{v}_i(t)$ in (7) uses the same mechanism considered in section 3.2 up to (18), which implies $\frac{1}{2}\|\vec{v}_i(t)\| \le E(\|\mathbf{R}_i(t)\vec{v}_i(t)\|) \le \frac{1}{\sqrt{3}}\|\vec{v}_i(t)\|$. Let X_k, $0 \le k \le t$ denote independent random variables with $E(X_k) = \frac{1}{\sqrt{3}}$, then from (7) and the triangle inequality we know

$$\|\vec{v}_i(t+1)\| \le \sum_{k=0}^{t}\left(\|\vec{a}_k(t-k)\|\prod_{m=1}^{k}X_m\right) \tag{23}$$

Since all X_m are independent, $E(\prod_{m=1}^{k}X_m) = 3^{-\frac{k}{2}}$. Since they are also indepedent from all $\|\vec{a}_k(t-k)\|$, (20) yields

$$E\left(\|\vec{v}_i(t+1)\|\right) \le \sum_{k=0}^{t}\left(\frac{G(t-k)}{\sqrt{3}}3^{-\frac{k}{2}}\right) \tag{24}$$

Let $\eta = e^{\frac{\alpha}{T}}$ so that $G(t-k) = G_0\eta^{k-t}$, then

$$\begin{aligned} E\left(\|\vec{v}_i(t+1)\|\right) &\le \frac{G_0}{\sqrt{3}}\eta^{-t}\sum_{k=0}^{t}\left(\frac{\eta}{\sqrt{3}}\right)^k \\ &= \frac{G_0}{\sqrt{3}}\cdot\frac{\sqrt{3}\eta^{-t} - 3^{-\frac{t}{2}}\eta}{\sqrt{3}-\eta} \\ &\le \frac{G_0\eta^{-t}}{\sqrt{3}-\eta} = \frac{G_0e^{-\alpha\frac{t}{T}}}{\sqrt{3}-e^{\frac{\alpha}{T}}} \end{aligned} \tag{25}$$

This means that if $\alpha \ll T$, for example if $\alpha < T\ln(2/\sqrt{3})$ (which for $T = 1000$ is the case when $\alpha \lesssim 143$), then $\sqrt{3} - e^{\frac{\alpha}{T}} > \frac{1}{\sqrt{3}}$ and

$$E\left(\|\vec{v}_i(T)\|\right) \le \frac{G_0e^{-\alpha\frac{T-1}{T}}}{\sqrt{3}-e^{\frac{\alpha}{T}}} < \sqrt{3}G_0e^{-\alpha\frac{T-1}{T}} \tag{26}$$

For larger values of α, $G(t)$ is even lower, and the accumulated velocity is lower as well. As such, if α is sufficiently large, then the swarm will be approximately stationary towards the end of the optimization.

Note that this does not prove convergence, and indeed if G_0 is very small in relation to the extent of $\mathcal{F}$, the swarm's initial mobility may already be too low for it to collapse. However, in the event that the swarm is initially mobile enough to collapse, its inability to split suggests that it can be expected to converge.[4]

An upper bound to the expected distance W traveled by individual swarm particles over the course of the optimization follows from (25) through

$$\begin{aligned} E(W) &= \sum_{t=0}^{T-1}E\left(\|\vec{v}_i(t+1)\|\right) \le \sum_{t=0}^{T-1}\frac{G_0e^{-\alpha\frac{t}{T}}}{\sqrt{3}-e^{\frac{\alpha}{T}}} \\ &= \frac{G_0}{\sqrt{3}-e^{\frac{\alpha}{T}}}\sum_{t=0}^{T-1}e^{-\alpha\frac{t}{T}} = \frac{G_0}{\sqrt{3}-e^{\frac{\alpha}{T}}}\,\frac{1-e^{-\alpha}}{1-e^{-\frac{\alpha}{T}}} \\ &= \frac{G_0e^{\frac{\alpha}{T}}\left(1-e^{-\alpha}\right)}{\left(\sqrt{3}-e^{\frac{\alpha}{T}}\right)\left(e^{\frac{\alpha}{T}}-1\right)} \end{aligned} \tag{27}$$

If α is large enough that $e^{-\alpha} \approx 0$ and $\alpha \ll T$ so that $e^{\frac{\alpha}{T}} \approx 1$, then

$$E(W) \lesssim \frac{G_0}{\left(\sqrt{3}-1\right)\left(e^{\frac{\alpha}{T}}-1\right)} \tag{28}$$

is a justifiable approximation that can be used to check the plausibility of a selection of G_0 and α.

3.4 Parameter Selection

Given this analysis, parameter selection for GSA has two main objectives:

(1) maximize swarm mobility, and
(2) ensure some level of result precision.

T and N determine the cost of the optimization: f is evaluated $(T+1)N$ times. The derivation of (26) and (28) depends on $\alpha \ll T$, so parameters that depend on the findings presented here should include a large enough T to meet the conditions mentioned there.

[4] We have never observed an optimization process in which the GSA swarm did not converge except with extreme choices of G_0, α (see section 4.1).

Furthermore, (5) implies that GSA $\in \Omega(N^2)$, and as such N should not be extremely large.[5]

G_0 controls the magnitude of the acceleration at the beginning of the optimization process and must be chosen as large as possible to maximize swarm mobility. On the other hand, intuitively G_0 should not be so large that most particles immediately leave $\mathcal{F}$ and are randomly reinitialized. However, a moderate overestimation of G_0 is not a serious problem: in this event GSA behaves like a random search until $G(t)$ has fallen to a sensible level, at which point GSA continues with its normal behavior (see section 4.2).

Let D be the distance from one side of the search domain to the other. We desire that $E\left(\|\vec{a}_i(0)\|\right) \approx D$ for an outlier particle in the initial, random swarm. Then (20) suggests

$$G_0 > \sqrt{3}D \tag{29}$$

Note that D needs to be somewhat meaningful in all directions. If the search domain is highly elongated,[6] there is no choice of G_0 that makes the swarm mobile enough to cross the search domain in a longer dimension without also making it too mobile to stay within the confines of a shorter dimension. If $\mathcal{F}$ is elongated, implementations should distort f in such a way that the distorted search domain is approximately hyperspherical (or, more realistically, hypercubical) and optimize the distorted function or modify the treatment of out-of-bounds particles.

α controls the precision of the result in $\mathcal{F}$. Let δ be the desired precision, then (22) and (26) suggest that α satisfy

$$-\ln\frac{2N\delta}{G_0} \leq \alpha \leq -\ln\frac{\delta}{\sqrt{3}G_0} \tag{30}$$

The observation in section 3.2 that $\vec{v}_i$ does not meaningfully accumulate magnitude suggests further that the upper bound stated in (26) is unlikely to be very tight. It is therefore sensible to choose α towards the lower bound of (30).

A sanity check for the choice of α is provided by (28): It is desirable that the swarm particles be mobile enough to be able to cross the search domain several times, i.e. that $E(W)$ be several times larger than D. The bound stated in (28) is again likely to be loose, therefore the chosen set of parameters should satisfy

$$\frac{G_0}{\left(\sqrt{3}-1\right)\left(e^{\frac{\alpha}{T}}-1\right)} \ggg D \tag{31}$$

If it does not do so, it is possible to relax the precision requirements (decrease δ), increase the computing budget (increase T) or choose a smaller search domain (decrease D, G_0, and α).[7] Section 4.1 refines this condition to (32) using experimental results.

Finally, ε in (3) is meant to avoid division by zero but not to have a noticeable effect on GSA's behavior. Since the analysis leading to (30) requires that $\hat{x}_{ij}(t) \approx \tilde{x}_{ij}(t)$, ε should be chosen small enough to be negligible compared to $E\left(\vec{a}_i(T)\right)$, i.e. several orders of magnitude smaller than δ. It is also possible to remove ε from (3) and set $\tilde{x}_{ij}(t) = \vec{0}$ whenever $\|\vec{x}_j(t) - \vec{x}_i(t)\| = 0$.[8]

4 EXPERIMENTAL CONFIRMATION

Section 3 made predictions about the behavior of GSA and recommendations for parameter selection. This section tests these predictions and recommendations empirically.

Section 4.1 measures the effect of α on result precision to demonstrate the validity of (22), (30) and augments (31) with experimental results. Section 4.2 shows that the effect of G_0 on particle mobility is in line with the considerations leading into section 3.4 and up to (29). Finally, section 4.3 establishes a link between particle mobility and the quality of detected optima in highly multimodal problems.

In all experiments, we use $T = 1000$, $N = 50$ (these are the values used in the original GSA paper [17]), $\varepsilon = 0$, and $\tilde{x}_{ij}(t)$ is forced to $\vec{0}$ whenever $\|\vec{x}_j(t) - \vec{x}_i(t)\| = 0$.

4.1 Precision

We use the sphere function $f_{\text{sphere}}(\vec{x}) = \|\vec{x}\|^2$ to measure the precision to which GSA is able to locate an optimum once it has found its vicinity. The sphere function is unimodal and has no complicating features [6], so finding the vicinity of the optimum is an easy challenge, and the exploitation of that optimum can be observed with minimal interference from other factors.

Table 1 shows the distances of the median result from the optimum (median error) in $\mathcal{F} = [-100, 100]^n$ for $n \in \{2, 10, 30, 50, 100\}$, $\alpha \in \{2, 5, 10, 20, 30, 50, 80\}$ out of 101 repetitions each. D is the extent of $\mathcal{F}$ along one of the main axes and G_0 set according to suggestion (29), i.e. $D = 200$, $G_0 = 2D = 400$. Alongside the results are shown the upper and lower bounds for $E\left(\|\vec{a}(T)\|\right)$ predicted by (22) and the sanity indicator provided by (31), expressed in relation to D.

For most scenarios, the median error GSA achieves is well below the predicted upper bound for $E\left(\|\vec{a}(T)\|\right)$. Illustrating Bellman's curse of dimensionality [2], GSA achieves higher precision in scenarios of lower dimensionality: for $n = 2$ and $n = 10$, the median error is consistently below even the lower bound for $E\left(\|\vec{a}(T)\|\right)$, albeit barely so in the case of $n = 10$ and not overwhelmingly so in the case of $n = 2$. For $n = 30$, the results are close to but consistently above the lower bound, and for higher dimensionalities they fall comfortably into the middle of the interval for low values of α until they suffer a sharp decline in quality when $\alpha = 80$ for $n = 30$, $\alpha \geq 50$ for $n = 50$ and $\alpha \geq 20$ for $n = 100$. It is around these points that the sanity indicator falls near n.

These failing scenarios are marked red and italic in table 1. GSA's failure in them is due to a combination of the early loss of swarm mobility through a quickly falling $G(t)$ and the problem's higher dimensionality – there are simply more directions for the individual force vectors $\vec{F}_{ij}$ to have, so they are less likely to reinforce each other. In these scenarios, the swarm often fails to collapse,[9] in which case the minimum cannot be effectively exploited. As such the sanity condition (31) should be considered in relation to the

[5] The threshold above which this influence becomes dominant depends on the computational complexity of f. For interesting problems, N need not be very small.

[6] e.g. $[0, 1] \times [0, 1000]$

[7] It is also possible to consider completely different $G(t)$ schedules, but that is beyond the scope of this paper.

[8] This is the authors' preferred solution because it implies $\hat{x}_{ij}(t) = \tilde{x}_{ij}(t)$, in which case the approximate lower bounds in (18) and (22) become simply lower bounds.

[9] For $n = 100$, $\alpha = 20$, it collapsed in 4 out of 101 repetitions, with result errors ranging from $8.999 \cdot 10^{-8}$ to $1.024 \cdot 10^{-7}$. These are within the bounds of (22).

Table 1: Median result precision on f_{sphere} in $\mathbb{R}^n$ for different values of α, 101 repetitions

	Median distance to optimum					$E(\lVert\vec{a}(T)\rVert)$ acc. to (22)		$\frac{G_0}{(\sqrt{3}-1)\left(e^{\frac{\alpha}{T}}-1\right)D}$
α	$n = 2$	$n = 10$	$n = 30$	$n = 50$	$n = 100$	Lower bound	Upper bound	
2	$8.508 \cdot 10^{-3}$	$2.555 \cdot 10^{-1}$	$8.792 \cdot 10^{-1}$	1.406	2.551	$5.413 \cdot 10^{-1}$	31.254	1364.66
5	$5.018 \cdot 10^{-4}$	$1.258 \cdot 10^{-2}$	$4.572 \cdot 10^{-2}$	$7.495 \cdot 10^{-2}$	$1.415 \cdot 10^{-1}$	$2.695 \cdot 10^{-2}$	1.556	545.05
10	$3.346 \cdot 10^{-6}$	$9.193 \cdot 10^{-5}$	$3.323 \cdot 10^{-4}$	$5.503 \cdot 10^{-4}$	$1.220 \cdot 10^{-3}$	$1.816 \cdot 10^{-4}$	$1.048 \cdot 10^{-2}$	271.84
20	$1.667 \cdot 10^{-10}$	$4.269 \cdot 10^{-9}$	$1.683 \cdot 10^{-8}$	$3.375 \cdot 10^{-8}$	*$3.749 \cdot 10^{-1}$*	$8.245 \cdot 10^{-9}$	$4.760 \cdot 10^{-7}$	135.24
30	$9.676 \cdot 10^{-15}$	$2.134 \cdot 10^{-13}$	$9.381 \cdot 10^{-13}$	$2.057 \cdot 10^{-12}$	*5.48*	$3.743 \cdot 10^{-13}$	$2.161 \cdot 10^{-11}$	89.71
50	$2.436 \cdot 10^{-23}$	$5.384 \cdot 10^{-22}$	$2.895 \cdot 10^{-21}$	*$9.933 \cdot 10^{-1}$*	*20.198*	$7.715 \cdot 10^{-22}$	$4.454 \cdot 10^{-20}$	53.29
80	$2.997 \cdot 10^{-36}$	$5.665 \cdot 10^{-35}$	*$3.909 \cdot 10^{-1}$*	*8.13*	*35.648*	$7.219 \cdot 10^{-35}$	$4.168 \cdot 10^{-33}$	32.80

dimensionality of the problem. The measurements suggest that G_0, α should satisfy

$$\frac{G_0}{\left(\sqrt{3}-1\right)\left(e^{\frac{\alpha}{T}}-1\right)D} \gg n \tag{32}$$

in order to have reasonable hope for success even with simple problems.[10]

The conclusion is that the bounds in (22) are a good predictor of result precision, provided that the swarm is mobile enough to find and effectively exploit an optimum. Especially for problems of high dimensionality, high values of α can prevent precisely this, and care should be taken not to demand unreasonable levels of accuracy.

4.2 Swarm mobility

The extended Rosenbrock "banana" function [12]

$$f_{\text{rosenbrock}}(\vec{x}) = \sum_{i=1}^{n-1}\left[100\left(x_{i+1}-x_i^2\right)^2 + (1-x_i)^2\right] \tag{33}$$

is useful to test the effects of G_0 on swarm mobility. $f_{\text{rosenbrock}}$ in $\mathbb{R}^2$ is unimodal with a minimum at $(1,1)^T$ and features a deep parabolic valley [6] that is easy to find but difficult to traverse. For dimensionalities $3 \le n \le 7$, (33) has two local minima and contains many saddle points with narrow ranges of descent directions [12]. Once located, these saddle points are difficult to escape for many optimization algorithms [12].

Figure 3 shows a typical progression of the GSA swarm on $f_{\text{rosenbrock}}$ in two dimensions. The swarm collapses at some point the valley, usually near the origin,[11] then crawls as a unit towards the optimum at $(1,1)^T$ (marked with a black cross). As it does so, $G(t)$ falls, the swarm contracts and loses mobility. Towards the end of the optimization, progress slows to negligible levels, and the swarm is never able to reach the optimum. This highlights the importance of swarm mobility for success on $f_{\text{rosenbrock}}$.

Table 2 shows the median fitness achieved by 1001 repetitions[12] of GSA on $f_{\text{rosenbrock}}$ in $\mathbb{R}^2$ and $\mathbb{R}^7$ for various G_0 alongside the average percentage of particles that were flung out of $\mathcal{F}$ and randomly reinitialized. $\delta = 10^{-10}$ is fixed and α set to $-\ln\frac{2N\delta}{G_0}$ so that $G(T)$ is equal in all scenarios. The optimization takes place in $\mathcal{F} = [-2.048, 2.048]^n$. The values considered for G_0 are the powers of 2 from 2 to 4096 and the lower bound suggested by (29) for $D = 4.096$ ($G_0 \approx 7.09$).

Table 2: Median fitness, average percentage of displaced particles on $f_{\text{rosenbrock}}$ for various G_0, 1001 repetitions

		$n = 2$		$n = 7$	
G_0	α	Med. Fit.	% disp.	Med. Fit.	% disp.
2	19.114	$5.396 \cdot 10^{-2}$	$1.92 \cdot 10^{-5}$	3.298	0
4	19.807	$5.054 \cdot 10^{-2}$	0.0181	3.187	$6 \cdot 10^{-5}$
7.09	20.379	$3.847 \cdot 10^{-2}$	0.0316	3.115	0.0714
8	20.5	$2.783 \cdot 10^{-2}$	0.0778	3.098	0.103
16	21.193	$1.195 \cdot 10^{-2}$	1.326	3.01	0.602
32	21.886	$7.176 \cdot 10^{-3}$	4.188	2.913	3.3
64	22.58	$6.104 \cdot 10^{-3}$	7.084	2.826	6.248
128	23.274	$5.305 \cdot 10^{-3}$	9.83	2.769	9.02
256	23.966	$4.258 \cdot 10^{-3}$	12.418	2.685	11.616
512	24.659	$3.985 \cdot 10^{-3}$	14.854	2.632	14.068
1024	25.352	$3.518 \cdot 10^{-3}$	17.152	2.574	16.376
2048	26.045	$3.324 \cdot 10^{-3}$	19.33	2.539	18.564
4096	26.738	$2.863 \cdot 10^{-3}$	21.388	2.501	20.634

For low values of G_0, displacements are rare; with $G_0 \approx 7.09$, fewer than 0.1% of particles are displaced on average. Soon beyond this point, the percentage of displaced particles increases sharply. For $n = 7$, this increase happens noticeably later than for $n = 2$.

This is in line with the considerations in section 3.4 leading up to (29): the suggested bound is a good ballpark estimate of the point beyond which increases of G_0 cause significant amounts of particle displacements, which in turn cause fitness function evaluations at random points and are not generally desirable. This increase happens later for higher dimensionalities because $\mathcal{F}$'s inscribed hypersphere[13] (whose diameter is D) covers less of its volume as n grows and particles that are moved beyond the hypersphere but remain in $\mathcal{F}$ are not displaced.

However, on $f_{\text{rosenbrock}}$ the result GSA achieves continues to improve as G_0 increases well beyond this point. The best median fitness achieved in these measurements is for $G_0 = 4096$, a ludicrously high value in relation to the search domain that causes more

[10] Note that this condition should be considered necessary but not necessarily sufficient.

[11] This may be related to center-seeking bias in GSA, see Davarynejad et al. [5].

[12] The sample is larger here because the results were less stable than in section 4.1.

[13] For simplicity, the effects of (18) are neglected here.

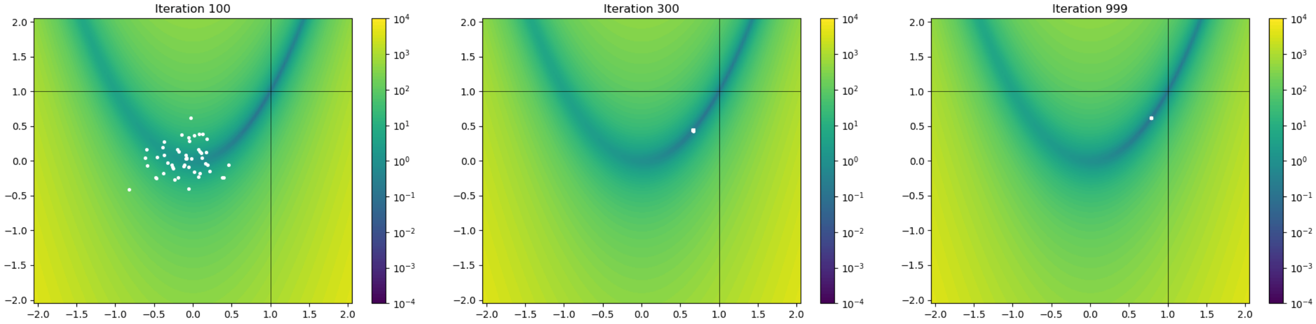

Figure 3: Swarm movement on $f_{\text{rosenbrock}}$ in $\mathbb{R}^2$, $G_0 = 16, \alpha \approx 21.193$

than a full fifth of the fitness function evaluations to take place at random positions. The measurements in section 4.3 show that the effect is not universal, and the exact reasons for it are unclear. We suspect that it is related to the specific challenges of the Rosenbrock function, where it is difficult for the swarm to move towards the optimum and mobility is paramount.

Figure 4 shows the median best-so-far fitness over the course of the optimizations with $G_0 \in \{1, 7.09, 32, 4096\}$ and $n = 7$. Also shown is the same median over 1001 random searches that evaluated $N = 50$ points in $\mathcal{F}$ per iteration. Swarms with lower G_0 achieve better early results because they collapse quickly, but they are overtaken at a later time because their mobility is exhausted earlier.

With $G_0 = 4096$, GSA behaves like a random search for the first ≈ 200 iterations, at which point $G(t)$ has dropped to around 20 and the usual behavior takes over. This pattern is common to all cases with very high values of G_0, and higher values of G_0 correspond with a longer random-searching phase. In these optimizations, all or nearly all particles are moved past the boundaries of $\mathcal{F}$ and randomly repositioned in every iteration until $G(t)$ is low enough.

4.3 Mobility and multimodality

Since the GSA swarm moves from optimum to optimum rather than split into subswarms, mobility has a profound effect on its ability to minimize multimodal functions. We consider the Rastrigin function

$$f_{\text{rastrigin}}(\vec{x}) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right] \tag{34}$$

in $\mathcal{F} = [-5.12, 2.56]^n$ (thus $D = 7.68$) as a highly multimodal problem whose optimum is significantly removed from $\mathcal{F}$’s geometric center to minimize the possible influence of center-seeking bias [5]. The challenge is to identify the global (or at least a good local) minimum among the multitude of local minima.

Table 3 shows the median and average fitness achieved by 1001 repetitions of GSA on $f_{\text{rastrigin}}$ in $\mathbb{R}^{30}$ for various G_0 and the average percentage of displaced particles. δ is fixed to 10^{-5}, α set to $-\ln \frac{2N\delta}{G_0}$ as before. The considered values for G_0 are the powers of 2 from 8 to 4096, $\sqrt{3}D \approx 13.302$ and two points of interest: $\sqrt{3}D' \approx 72.859$ where $D' = \sqrt{30}D$ is the diameter of $\mathcal{F}$’s *circumscribed* hypersphere, and 100 for closer spacing near the optimal G_0.

Table 3: Average and median fitness, average percentage of displaced particles on $f_{\text{rastrigin}}$ in $\mathbb{R}^{30}$, various G_0, 1001 rep.

G0	α	Avg. Fitness	Med. Fitness	% displaced
8	8.987	41.274	40.793301	$6.194 \cdot 10^{-5}$
13.302	9.496	30.955	30.843710	0.03
16	9.68	25.98	25.868925	0.118
32	10.373	19.326	18.904217	1.516
64	11.067	19.034	18.904217	5.567
72.859	11.196	18.956	18.904217	6.704
100	11.513	18.49	18.904207	9.258
128	11.76	18.918	18.904212	11.145
256	12.453	19.694	18.904217	16.028
512	13.146	19.999	18.904217	20.383
1024	13.839	20.229	19.899171	24.293
2048	14.532	21.007	19.899176	27.82
4096	15.226	21.273	20.894135	31.023

Of the values considered, the best result is achieved for $G_0 = 100$, larger than $\sqrt{3}D'$ but not overwhelmingly so. The median fitness is, however, virtually identical for $G_0 = 32$ through 512; $G_0 = 100$ has the best result by a margin smaller than the target precision δ, and as such this difference is unlikely to be meaningful. This may suggest that D' is more relevant to a good selection of G_0 than D, but further research is necessary to substantiate this claim.

Unlike $f_{\text{rosenbrock}}$, results on $f_{\text{rastrigin}}$ deteriorate for larger values of G_0, although the impact of an underestimated G_0 is more severe than that of an overestimation. This result is in line with the analysis in section 3.4.

5 RELATED WORK

To date only a handful of publications have analyzed GSA in a manner comparable to this paper, and to the authors’ knowledge none have proposed a method of parameter selection other than meta-optimization.

One of the first was by Ghorbani et al. in 2012 [9], who show that the GSA swarm would become stationary if t increased past T towards infinity. The idea to derive upper limits for relevant quantities (such as (14), (15)) has a precedent there.

In 2014, Davarynejad et al. investigated center-seeking and initialization-region biases in GSA and suggested a mitigating modification [5]. Their analysis is largely empirical.

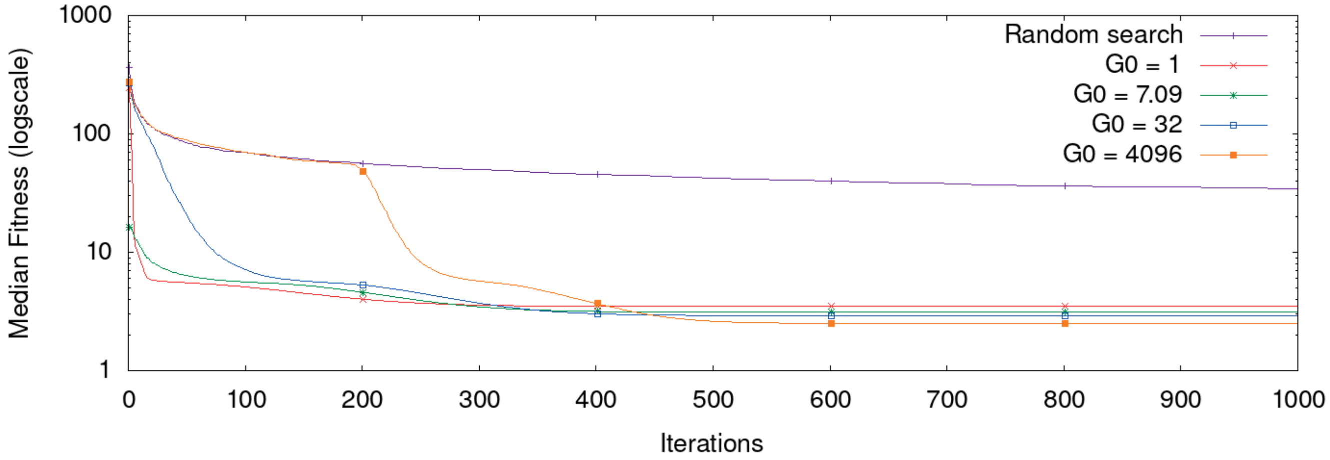

Figure 4: Median fitness for various G_0 over time on $f_{\mathbf{rosenbrock}}$ in $\mathbb{R}^7$

In 2016, Farivar and Shoorehdeli proved the existence of Lyapunov-stable equilibrium points in GSA and suggested a modification that makes these equilibrium points asymptotically stable (i.e., guarantees local convergence to the equilibrium point) [7].

In 2017, Pelusi et al. presented a method of parameter control for GSA in which α is tuned over time by a fuzzy controller whose membership functions are in turn optimized by a separate metaheuristic [15].

All these analyses fit in a wider context of similar research done for other swarm-based metaheuristics, starting with Ozcan and Mohan's 1999 analysis of the Particle Swarm Optimization [14] that Clerc and Kennedy refined in 2002 to introduce a popular method of parameter selection [3]. Similar work exists for other swarm algorithms and aspects of optimization metaheuristics [20]; this work is far from exhaustive but too numerous to list here.

6 CONCLUSION

This paper set out to discover key properties of GSA and provide its users with a method of parameter selection. Through mathematical analysis it uncovered the following properties:

- GSA is invariant to linear transformations of f,
- its swarm is unable to split into subswarms and exploit several optima simultaneously,
- particles do not usually accumulate velocity to any meaningful degree, and their movement is dominated by short-term influences; particle mobility is controlled by $G(t)$,
- unless α is very small, the swarm will be approximately stationary towards the end of the optimization,
- the result precision of a successful application of GSA is predicted by (22),
- a loose upper bound for the expected travel distance of swarm particles is given by (27),
- an overestimation of G_0 causes less severe problems than an underestimation, and
- the search domain should have similar extents in all directions or the treatment of out-of-bounds particles be modified.

It used these to derive recommendations for parameter selection along with a sanity check for a given set of parameters and demonstrated them to be sensible. The recommendations are detailed in section 3.4, and the sanity check is refined in section 4.1 to (32).

Research, as ever, does not stop with this paper. Here the focus was on the main variant of GSA, but the methods of analysis are, in principle, applicable to many variations as well and could be used to derive methods of parameter selection in these contexts.

Nor do we claim to have spoken the last word on mainline GSA: (27) gives a loose bound that could be improved in the future, which in turn would improve the parameter sanity check (32), and (29) only states a lower bound for G_0 when an upper bound would be desirable – although the measurements in section 4.2 suggest that such a bound may be difficult to determine for some problems.

Nevertheless, this paper gives users of GSA the tools to tune the algorithm for good performance, and those contemplating the use of GSA may find the list of derived properties helpful when trying to determine whether or not GSA is a good fit for their problem.

REFERENCES

[1] Behnam Mohseni Bababdani and Mehdi Mousavi. 2013. Gravitational search algorithm: A new feature selection method for QSAR study of anticancer potency of imidazo[4,5-b]pyridine derivatives. *Chemometrics and Intelligent Laboratory Systems* 122 (2013), 1–11. https://doi.org/10.1016/j.chemolab.2012.12.002

[2] Richard Bellman. 1957. *Dynamic Programming.* Princeton University Press, Princeton, NJ.

[3] Maurice Clerc and James Kennedy. 2002. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation* 6, 1 (2002), 58–73. https://doi.org/10.1109/4235.985692

[4] Pradipta Kumar Das, Himansu Sekhar Behera, and Bijaya Panigrahi. 2016. A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. *Swarm and Evolutionary Computation* 28 (2016), 14–28. https://doi.org/10.1016/j.swevo.2015.10.011

[5] Mohsen Davarynejad, Jan van den Berg, and Jafar Rezaei. 2014. Evaluating center-seeking and initialization bias: The case of particle swarm and gravitational search algorithms. *Information Sciences* 278 (2014), 802–821. https://doi.org/10.1016/j.ins.2014.03.094

[6] Kenneth Alan De Jong. 1975. Analysis of the behavior of a class of genetic adaptive systems. (1975).

[7] Faezeh Farivar and Mahdi Aliyari Shoorehdeli. 2016. Stability analysis of particle dynamics in gravitational search optimization algorithm. *Information Sciences* 337-338 (2016), 25–43. https://doi.org/10.1016/j.ins.2015.12.017

[8] Melvin Gauci, Tony J. Dodd, and Roderich Groß. 2012. Why 'GSA: a gravitational search algorithm' is not genuinely based on the law of gravity. *Natural Computing*

11, 4 (01 Dec 2012), 719–720. https://doi.org/10.1007/s11047-012-9322-0

[9] Farzaneh Ghorbani and Hossein Nezamabadi-pour. 2012. On the convergence analysis of gravitational search algorithm. (2012).

[10] David E. Goldberg and Kalyanmoy Deb. 1991. A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. Foundations of Genetic Algorithms, Vol. 1. Elsevier, 69 – 93. https://doi.org/10.1016/B978-0-08-050684-5.50008-2

[11] Mohammad Sadegh Jahan and Nima Amjady. 2013. Solution of large-scale security constrained optimal power flow by a new bi-level optimisation approach based on enhanced gravitational search algorithm. *IET Generation, Transmission & Distribution* 7, 12 (2013), 1481–1491. https://doi.org/10.1049/iet-gtd.2012.0697

[12] Schalk Kok and Carl Sandrock. 2009. Locating and Characterizing the Stationary Points of the Extended Rosenbrock Function. *Evolutionary Computation* 17, 3 (2009), 437–453. https://doi.org/10.1162/evco.2009.17.3.437

[13] Jonas Krause, Jelson Cordeiro, Rafael Stubs Parpinelli, and Heitor Silvério Lopes. 2013. A Survey of Swarm Algorithms Applied to Discrete Optimization Problems. In *Swarm Intelligence and Bio-Inspired Computation*. Elsevier, Oxford, 169–191. https://doi.org/10.1016/B978-0-12-405163-8.00007-7

[14] E. Ozcan and C. K. Mohan. 1999. Particle swarm optimization: surfing the waves. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, Vol. 3. 1939–1944. https://doi.org/10.1109/CEC.1999.785510

[15] Danilo Pelusi, Raffaele Mascella, and Luca Tallini. 2017. Revised Gravitational Search Algorithms Based on Evolutionary-Fuzzy Systems. *Algorithms* 10, 2 (2017). https://doi.org/10.3390/a10020044

[16] Esmat Rashedi. 2011. Gravitational Search Algorithm (GSA), Reference implementation. https://www.mathworks.com/matlabcentral/fileexchange/27756-gravitational-search-algorithm-gsa. [Online; accessed 12-Dec-2018].

[17] Esmat Rashedi, Hossein Nezamabadi-pour, and Saeid Saryazdi. 2009. GSA: A Gravitational Search Algorithm. *Information Sciences* 179, 13 (2009), 2232 – 2248. https://doi.org/10.1016/j.ins.2009.03.004

[18] Esmat Rashedi, Elaheh Rashedi, and Hossein Nezamabadi-pour. 2018. A comprehensive survey on gravitational search algorithm. *Swarm and Evolutionary Computation* 41 (2018), 141 – 158. https://doi.org/10.1016/j.swevo.2018.02.018

[19] Mehmet Polat Saka, Erkan Doğan, and Ibrahim Aydogdu. 2013. Analysis of Swarm Intelligence–Based Algorithms for Constrained Optimization. In *Swarm Intelligence and Bio-Inspired Computation*. Elsevier, Oxford, 25–48. https://doi.org/10.1016/B978-0-12-405163-8.00002-8

[20] Xin-She Yang. 2012. Efficiency Analysis of Swarm Intelligence and Randomization Techniques. *Journal of Computational and Theoretical Nanoscience* 9, 2 (2012), 189–198. https://doi.org/doi:10.1166/jctn.2012.2012