Progress In Electromagnetics Research B, Vol. 19, 405–425, 2010

IMPROVED CFO ALGORITHM FOR ANTENNA OPTIMIZATION

R. A. Formato

Registered Patent Attorney & Consulting Engineer
P. O. Box 1714, Harwich, MA 02645, USA

Abstract—An improved Central Force Optimization (CFO) algorithm for antenna optimization is presented. CFO locates the global extrema an objective function to be maximized, in this case antenna directivity, by flying "probes" through the decision space (DS). The new implementation includes variable initial probe distribution and decision space adaptation. CFO's performance is assessed against a recognized antenna benchmark problem specifically designed to evaluate optimization evolutionary algorithms for antenna applications. In addition, summary results also are presented for a standard twenty-three function suite of analytic benchmarks. The improved CFO implementation exhibits excellent performance.

1. INTRODUCTION

Central Force Optimization (CFO) [1–4] is a nature-inspired, gradient-like, gravity-based metaheuristic (algorithmic framework) for a multidimensional search and optimization evolutionary algorithm (EA) that locates the extrema of an objective function to be maximized. The objective function is defined on a "decision space" (DS) of unknown topology that is searched by the EA. CFO comprises two simple "equations of motion" drawn from its metaphor of gravitational kinematics. Gravity is deterministic, and so too is CFO because it embraces Newton's mathematically precise laws of gravity and motion.

CFO's deterministic nature is a significant distinction from the many natureinspired EAs that are fundamentally stochastic, Particle Swarm Optimization (PSO) and genetic algorithm (GA) being good examples. PSO and GA, respectively, search DS by analogizing the random swarming behavior of fish and the random processes in genetic

Corresponding author: R. A. Formato (rf2@ieee.org).

selection in natural evolution. As a result of this inherent randomness, each PSO and GA run with the same setup produces a different outcome. By contrast, every CFO run with the same setup produces exactly the same results step-by-step throughout the entire run.

The Central Force Optimization EA creates a metaphorical "CFO space" that is analogous to physical space through which "probes" are flown along trajectories computed from the equations of motion. The value of the objective function to be maximized (its "fitness"), in this case antenna directivity, is computed step-by-step at each probe's location, and then input to a user-defined function that becomes CFO's "mass." Mass in CFO space is analogous to real mass in the Universe moving under Newton's laws.

The first CFO paper applied new the algorithm to two problems in applied electromagnetics (EM): Equalizer design for the canonical Fano load and synthesis of a linear dipole array. It also included a sampling of analytic benchmarks. More recently CFO has been applied to linear and circular array synthesis [5, 6].

This paper presents an improved CFO implementation for antenna optimization and provides detailed results for a recognized numerical antenna benchmark problem and summary results for a suite of analytic EA benchmarks. The improved implementation includes the use of a variable initial probe distribution to better sample the DS topology and an adaptive strategy that periodically shrinks DS around the probe with the best fitness to improve convergence rate.

2. CFO RESULTS FOR VARIABLE LENGTH CENTER-FED DIPOLE

The numerical antenna benchmark suite developed by Pantoja, Bretones, and Martin [7] ("PBM") is designed to objectively evaluate the performance of EAs used to solve complex EM problems, representative examples being: PSO-based array synthesis [8]; swarm intelligence optimization of layered media [9]; GA antenna modeling [10]; array design using GAs, and mimetic and tabu search algorithms [11]; crack detection using a finite-difference frequency domain/PSO methodology [12]; Vee-dipole optimization using a bacteria foraging algorithm [13]; and GA-optimized MRI coil design [14].

EAs solve these problems by "evolving" solutions that are unavailable analytically or numerically. The plethora of nature-inspired EAs makes comparing the algorithms difficult, if not impossible, without a standardized set of benchmarks for testing. The PBM suite, which is based on the Numerical Electromagnetics Code

(NEC), addresses this need for EAs that are applied to optimized antenna design. It therefore is a useful tool for evaluating CFO in the context of antenna optimization.

The PBM suite comprises five antenna problems designed to test an EA's effectiveness (actually locating the global maximum) and efficiency (how many function evaluations are required). Table 1

Table 1. Properties of the PBM benchmark problems.

PBM Benchmark #	Problem Characteristics (in each case objective is to maximize directivity)
1	Variable length center-fed dipole. 2D, unimodal, single global maximum, strong local maxima.
2	Uniform 10-element array of center-fed $\frac{\lambda}{2}$-dipoles. 2D, added Gaussian noise, single global maximum, multiple strong local maxima
3	8-element circular array of center-fed $\frac{\lambda}{2}$-dipoles. 2D, highly multimodal, four global maxima.
4	Vee Dipole. 2D, unimodal, single global maximum, "smooth" landscape.
5	Collinear N_{el}-element array of center-fed $\frac{\lambda}{2}$-dipoles. $(N_{el}-1)$D, unimodal, single global maximum.

Table 2. PBM #1 run parameters and results.

Run #	γ	# Steps	N_{eval}	$D_{\max}$
1	0.000	95	384	1.80301774
2	0.100	92	372	3.24339617
3	0.200	87	352	3.25836701
4	0.300	60	244	3.25836701
5	0.400	72	292	3.25836701
6	0.500	91	368	3.25836701
7	0.600	60	244	3.22849412
8	0.700	67	272	3.01995172
9	0.800	174	700	2.39883292
10	0.900	92	372	3.24339617
11	1.000	193	776	2.34422882

lists their properties. In each case, the objective is to maximize the antenna's directivity ("fitness"). Four of the problems are two-dimensional (2D), while the fifth is $(N_{el} - 1)$D, N_{el} being the number of dipole elements in a collinear array. Of the five problems, the first is by far the most difficult based on the PBM testing data reported in [7]. Problems #1 and #4 are unimodal with a single global maximum. The first problem is "lumpy" with strong local maxima, whereas the fourth is "smooth." Problem #2 is "noisy" in a complex landscape with large amplitude nearby local maxima. The third problem's topology is multimodal with four global maxima. Problem #5 is a unimodal high-dimensionality problem that optimizes the element spacing of an N_{el}-element collinear dipole array.

The PBM paper examines four algorithms: GA-FPC, μGA, GA-RC, and PSO (see [7] for details). The first three are GA variants, while the fourth is a particle swarm. Because these algorithms are stochastic, each one was run twenty times to develop statistics. The "mean hit time" in Table 2 in [7] is the average generation number at which the global maximum was located, so that the average number of function evaluations is the product of mean hit time and the population size. Because each of these algorithms is fundamentally stochastic, assessing their performance necessarily requires this type of statistical analysis. CFO, by contrast, is deterministic, so that every run with the same set-up parameters returns precisely the same results. All CFO data consequently are based on a single run.

Because the focus of this paper is the new CFO implementation, not an exhaustive study of its performance, CFOs performance is tested against the most difficult of the five problems, PBM benchmark #1. The four EAs tested in [7] achieved success rates, that is, actually locating the global maximum, of only 10–70% on PBM #1. None of the

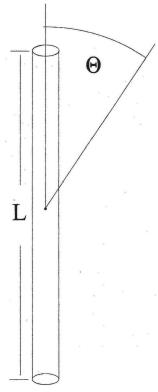


Figure 1. Dipole.

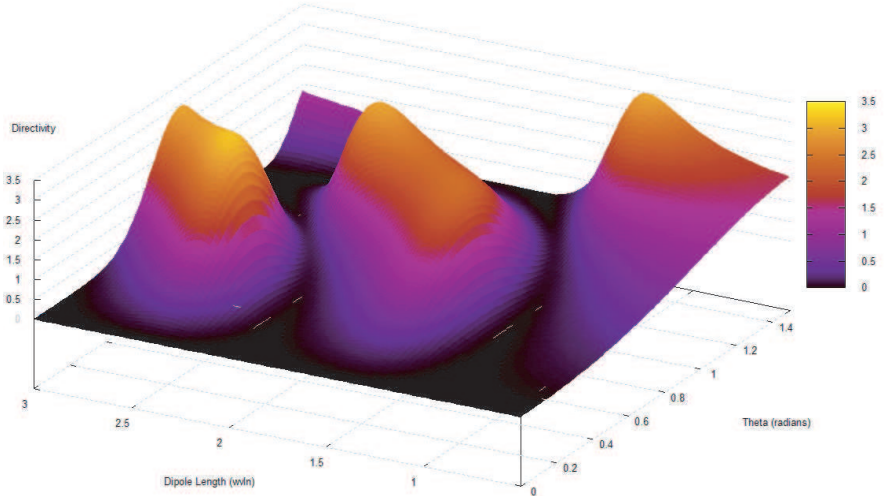


Figure 2. Landscape of 2D PBM benchmark #1.

algorithms was completely successful in locating the global maximum.

The antenna geometry for PBM #1 is shown in Fig. 1. The objective is to maximize a center-fed dipole's directivity, $D(L, \theta)$, as a function of its total length, L, and the polar angle, θ. In other words, determine the "best fitness" of the "objective function" $D(\cdot)$ over the decision space (DS) domain $0.5\lambda \leq L \leq 3\lambda$, $0 \leq \theta \leq \frac{\pi}{2}$. A perspective view of the two-dimensional (2D) DS topology or "landscape" appears in Fig. 2. The topology is smoothly varying with a single global maximum and two local maxima of similar amplitude.

The PBM problems do not have analytic solutions and consequently are solved numerically using NEC. The first step in evaluating CFO is validation of the published PBM results. PBM #1 therefore was modeled using the most recent available version of NEC, Version 4.1 Double Precision ("NEC4") [15] (note that this is not the same version of NEC cited in [7]). The same wire segmentation was used, approximately 100 segments per wavelength (λ), but the wire radius was decreased to 0.00001λ to improve agreement. PBM reports a global maximum directivity of 3.32 at $(L, \theta) = (2.58\lambda, 0.63)$. NEC4's computed directivity at those coordinates is slightly less at 3.2509.

The improved CFO algorithm described in §3 returned a maximum directivity of $D_{\max} = 3.2584$ at $(L, \theta) = (2.5815\lambda, 0.60697)$. CFO appears to have located the global maximum (100% success rate) with a value slightly higher than the value computed by NEC4 at the PBM coordinates. CFO's length coordinate agrees with the value in [7],

but the polar angle is slightly different. For comparison, a conventional CFO run was made with $N_t = 100$, $N_p = 4$ and initial probes located symmetrically in the (L, θ)-plane at $(1.333\lambda, \frac{\pi}{4})$, $((2.167\lambda, \frac{\pi}{4})$, $(1.75\lambda, \frac{\pi}{6})$, $(1.75\lambda, \frac{\pi}{3})$. The best fitness value was 3.02691 at $(L, \theta) = (2.23044\lambda, 1.07992)$. Thus, the conventional CFO implementation did not perform nearly as well as the improved version described below.

Figure 3 plots the evolution of CFO's best fitness, while Fig. 4 the

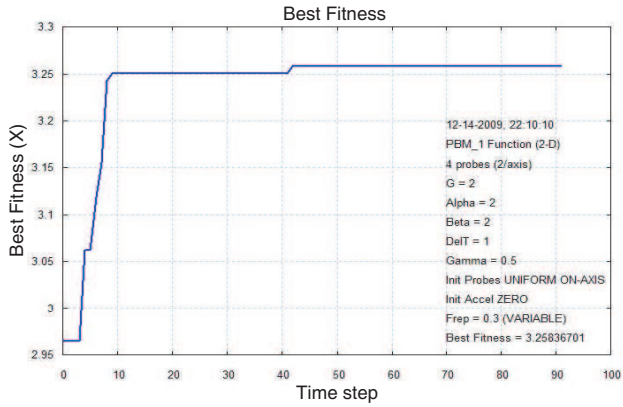


Figure 3. PBM #1 best fitness.

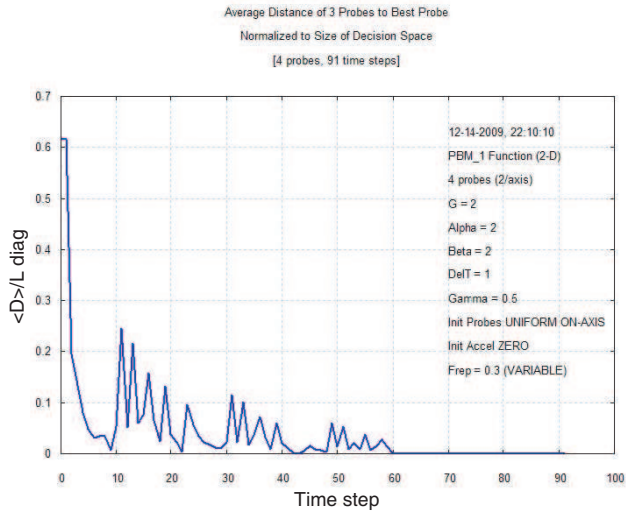


Figure 4. PBM #1 D_{avg} curve.

plots CFO's "D_{avg} curve." D_{avg} is the normalized average distance between the probe with the best fitness and all other probes at each time step, *viz.*, $D_{avg} = \frac{1}{L \cdot (N_p - 1)} \sum_{p=1}^{N_p} \sqrt{\sum_{i=1}^{N_d} (x_i^{p,j} - x_i^{p*,j})^2}$ where $p*$ is the number of the probe with the best fitness, and $L = \sqrt{\sum_{i=1}^{N_d} (x_i^{\max} - x_i^{\min})^2}$ is the length of DS's principal diagonal (see Appendix for definitions).

The directivity initially increases very quickly, plateauing just before step #10 followed by a slight increase at step #41. The rapid initial increase in fitness appears to be characteristic of how CFO converges on maxima. The D_{avg} curve exhibits three distinct quasi-oscillatory regions (steps #10-25, 30-40, and 48-60), after which all four probes have fully converged on the global maximum ($D_{avg} = 0.0008403$ starting at step #60). These results clearly show that CFO effectively locates the PBM #1 global maximum, and it does so deterministically with a total of only 4,376 function evaluations (see Table 2, §3).

3. IMPROVED CFO ALGORITHM

This paper introduces two new elements to how CFO is implemented: (1) Variable initial probe distribution, and (2) DS adaptation that periodically shrinks the DS around the location of the best fitness. The variable initial probe distribution provides CFO with more information about the DS landscape so that trapping is less likely, and DS adaptation takes advantage of CFO's tendency to converge quickly

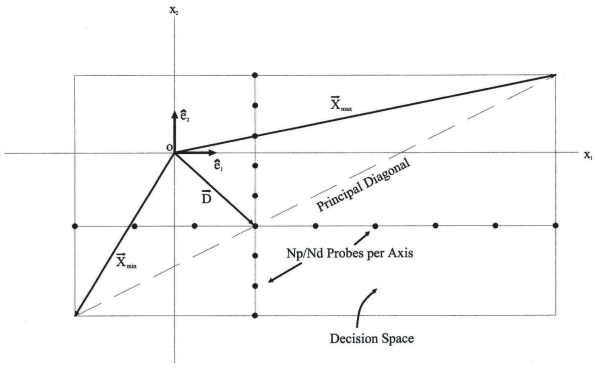


Figure 5. Typical variable 2-D initial probe distribution.

to improve convergence rate.

Like most EAs, CFO contains several user-specified parameters (see Appendix for detailed parameter descriptions). The most important ones (determined empirically) appear to be the initial probe distribution (total number of probes, N_p, and their deployment in the decision space) and the "repositioning factor," $0 \leq F_{rep} \leq 1$. The initial probe distribution determines how well the decision space topology is sampled at the beginning of a run, while F_{rep} is important in avoiding local trapping (a common problem in deterministic algorithms, but there may be an analytical approach to mitigation as discussed below).

Figure 5 provides a 2D schematic representation of the variable initial probe distribution comprising in this case an orthogonal array

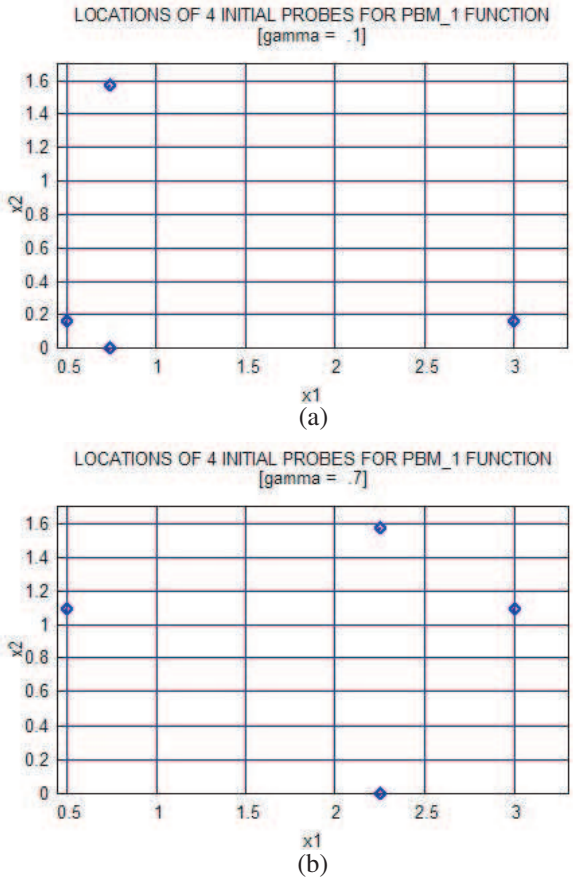


Figure 6. (a) PBM #1 initial probes, $\gamma = 0.1$. (b) PBM #1 initial probes, $\gamma = 0.7$.

of N_p/N_d probes per axis deployed uniformly on lines parallel to the coordinate axes that intersect at a point along DS's principal diagonal, N_d being the DS dimensionality (in this case 2). The DS domain is $x_i^{\min} \leq x_i \leq x_i^{\max}$, $1 \leq i \leq N_d$, where x_i are the "decision variables" and i the coordinate number.

For illustrative purposes, the diagram shows nine probes on each of two lines parallel to the x_1 and x_2 axes whose intersection point is marked by position vector $\vec{D} = \vec{X}_{\min} + \gamma(\vec{X}_{\max} - \vec{X}_{\min})$ where $\vec{X}_{\min} = \sum_{i=1}^{N_d} x_i^{\min} \hat{e}_i$ and $\vec{X}_{\max} = \sum_{i=1}^{N_d} x_i^{\max} \hat{e}_i$ are the diagonal's endpoint vectors. Parameter $0 \leq \gamma \leq 1$ determines where along the diagonal the orthogonal probe array is placed. Different numbers of probes per axis can be used if, for example, equal probe spacing were desired in a space with unequal boundaries or if overlapping probes were to be excluded. Of course, many other variable initial probe distributions could be used as well. As an example of the distribution used here, Fig. 6 shows the initial probes for $\gamma = 0.1$ and $\gamma = 0.7$ used for benchmark PBM #1 (2 probes/axis).

Figure 7 provides a 2D illustration of how the DS size is adaptively reduced every 20th step around the probe's location with the then best fitness, $\vec{R}_{best}$. DS's boundary coordinates are reduced by one-half the distance from the best probe's position to the boundary on a

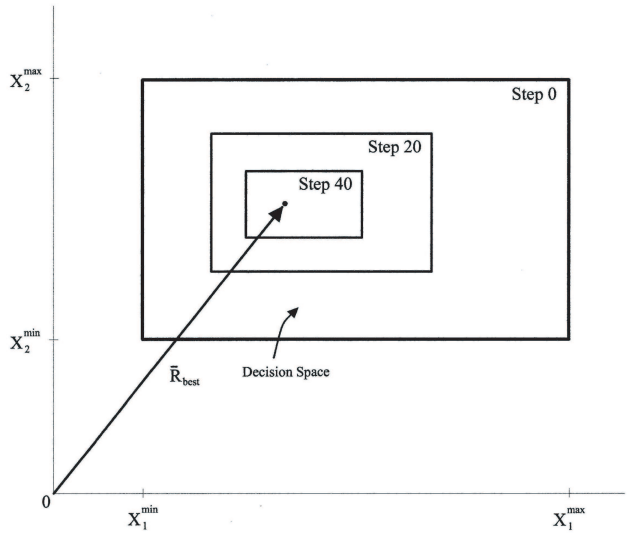


Figure 7. Schematic 2-D decision space adaptation (with constant best $\vec{R}_{best}$).

coordinate-by-coordinate basis. Thus, $x'^{\min}_i = x^{\min}_i + \frac{\vec{R}_{best}\cdot\hat{e}_i - x^{\min}_i}{2}$ and $x'^{\max}_i = x^{\max}_i - \frac{x^{\max}_i - \vec{R}_{best}\cdot\hat{e}_i}{2}$, where the primed coordinate is the new decision space boundary and the dot denotes vector inner product.

For clarity Fig. 7 shows $\vec{R}_{best}$ as fixed, whereas generally it varies throughout a run. Changing DS's boundary every twenty steps as opposed to some other interval was chosen arbitrarily (another, and likely better approach, would be some sort of reactive adaptation based on performance measures such as convergence speed or fitness saturation). Of course, the 2D schemes illustrated in Figs. 5 and 7 are generalized to the N_d-dimensional DS in an actual CFO implementation (the 2D case is used only to provide visualization of the new methodology).

For $\gamma = \gamma_{start}$ to γ_{stop} by $\Delta\gamma$:

(a.1) Compute initial probe distribution.
(a.2) Compute initial fitness matrix.
(a.3) Assign initial probe accelerations.
(a.4) Set initial F_{rep}.

For $j = 0$ to N_t (or earlier termination criterion):

(b) Compute probe position vectors $\vec{R}^p_j, 1 \le p \le N_p$ [Eq.(2)].
(c) Retrieve errant probes ($1 \le p \le N_p$):
If $\vec{R}^p_j \cdot \hat{e}_i \le x^{\min}_i \therefore$
$\vec{R}^p_j \cdot \hat{e}_i = x^{\min}_i + F_{rep}(\vec{R}^p_{j-1} \cdot \hat{e}_i - x^{\min}_i)$.
If $\vec{R}^p_j \cdot \hat{e}_i \ge x^{\max}_i \therefore$
$\vec{R}^p_j \cdot \hat{e}_i = x^{\max}_i + F_{rep}(x^{\max}_i - \vec{R}^p_{j-1} \cdot \hat{e}_i)$.
(d) Compute fitness matrix for current probe distribution.
(e) Compute accelerations using current probe distribution and fitnesses [Eq. (1)].
(f) Increment F_{rep} by ΔF_{rep}:
If $F_{rep} > 1 \therefore F_{rep} = \Delta F_{rep}$.
(g) If $j\ MOD\ 20 = 0\ \therefore$
Shrink DS around $\vec{R}_{best}$.

Next j
Next γ

Figure 8. Pseudocode for CFO with variable initial probes and DS adaptation.

Pseudocode for the improved CFO implementation appears in Fig. 8 (see Appendix for equations). The following parameter values were used for all runs reported in this paper: $\alpha = 2$, $\beta = 2$, $G = 2$, $\Delta t = 1$, initial acceleration of zero, initial $F_{rep} = 0.5$, $\Delta F_{rep} = 0.05$, and $\gamma_{start} = 0$, $\gamma_{stop} = 1$ with $\Delta\gamma = 0.1$ (eleven runs). A run was terminated early if the average best fitness over 50 steps (including the current step) and the current best fitness differed by less than 10^{-6}. For functions PBM #1 and f_1–f_{13} (Table 3 below) two probes per axis were used, while four per axis were used for f_{14}–f_{23}. A probe may "fly" outside DS, and if so it is retrieved using the scheme in Fig. 8 step (c). The probe is repositioned a fraction $\Delta F_{rep} \leq F_{rep} \leq 1$ of the distance between its starting point inside DS and the boundary point that is exceeded on a coordinate-by-coordinate basis.

Table 2 lists the CFO run parameters with variable γ and the returned maximum directivity $D_{\max}$ for benchmark PBM #1. Each run was set up initially with 250 steps, but for every value of γ fewer iterations were needed to meet the early termination criterion. For γ values between 0.2 and 0.5, CFO returned the same $D_{\max}$, which is the global maximum, using between 60 and 91 iterations. The corresponding number of function evaluations per run is between 244 and 368, and the total number of evaluations over all runs is 4,376.

Figure 9 plots trajectories of the best four probes for PBM #1 (note that the probe number of the best probe generally varies throughout a run). This plot is typical of how 2D best probe trajectories appear across a wide range of test functions, visually chaotic with no clear indication of how the probes converge on the objective function's maxima. But this seemingly random motion masks what often is a mathematically precise relationship in the group of probes.

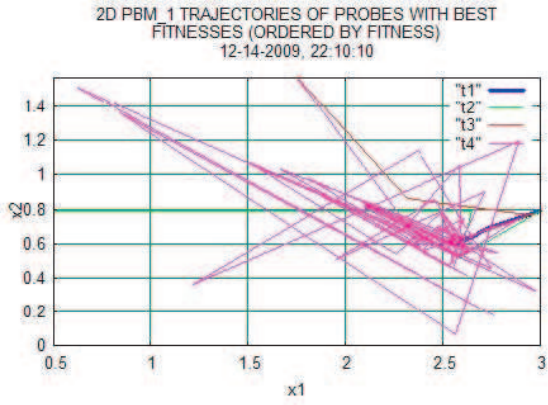
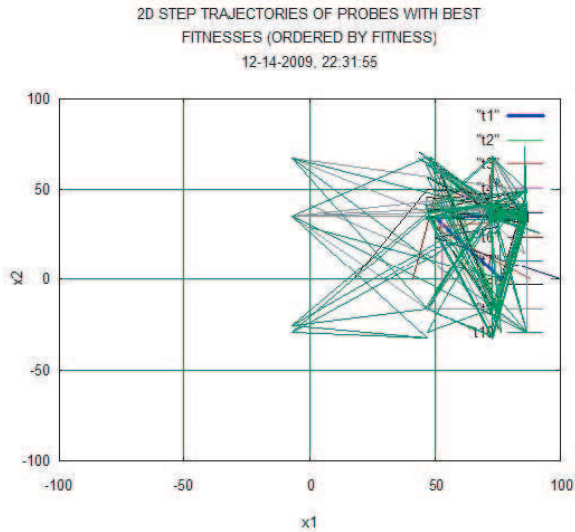
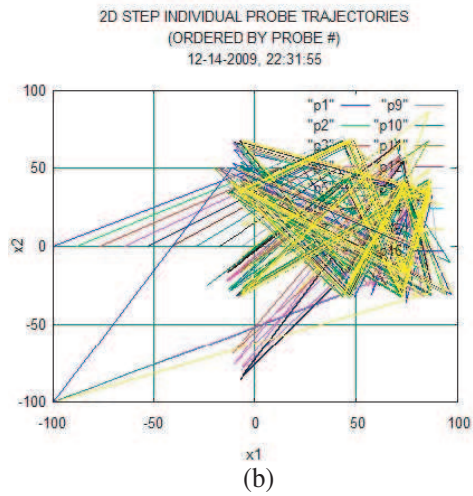


Figure 9. Best probe trajectories for PBM #1.

An example is provided by the 2D Step function $f(x) = -\sum_{i=1}^{N_d} (\lfloor x_i - x_o^i + 0.5 \rfloor)^2$, $N_d = 2$, $x_o^1 = 75$, $x_o^2 = 35$, $-100 \leq x_i \leq 100$, which, like PBM #1, is unimodal with a global maximum of zero at the point (75, 35). CFO's best 10 probe trajectories for the Step are plotted in Fig. 10(a), while Fig. 10(b) plots the individual



(a)



(b)

Figure 10. (a) 2D step best probe trajectories. (b) Step individual probe trajectories.

probe trajectories for probes #1 through #16 [note that this run was made without DS adaptation using 18 probes/axis and fixed instead of variable F_{rep}, γ, both 0.5]. As with PBM #1, they are visually chaotic. But, even though there is no obvious sign of regularity, they actually are characterized by a mathematically precise oscillation in the D_{avg} curve that is plotted in Fig. 11. The oscillation starting at step 257 continues thereafter with the repeating sequence: 0.6000747, 0.5884655, 0.6085178, 0.5926851, 0.5988160, 0.5965264, 0.6002386, 0.5995855, 0.5967055, 0.5981010, 0.6016253, 0.5947017. This observation is quite significant because it establishes what appears to be a strong connection between CFO and the theory of gravitationally trapped Near Earth Objects (NEOs).

Oscillation (or quasi-oscillation) in CFO's D_{avg} curve appears to be a reliable harbinger of trapping. In many cases, the trapping is local, that is, at a maximum that is not a global maximum. But in some cases the oscillation sets in at a global maximum, as is the case with the 2D step with its maximum offset to (75, 35). The NEO-CFO connection is based on the NEO behavior shown in Fig. 12, which reproduces with permission Fig. 2 in [16]. In many cases CFO's oscillatory D_{avg} curves bear a striking resemblance to the typical NEO ΔV plot in Fig. 12, which is based on the theory of gravitationally trapped NEOs developed in [17] (curve computed by Professors Andrea Carusi and Andrea Milani, private communication, Astronaut "Rusty" Schweickart).

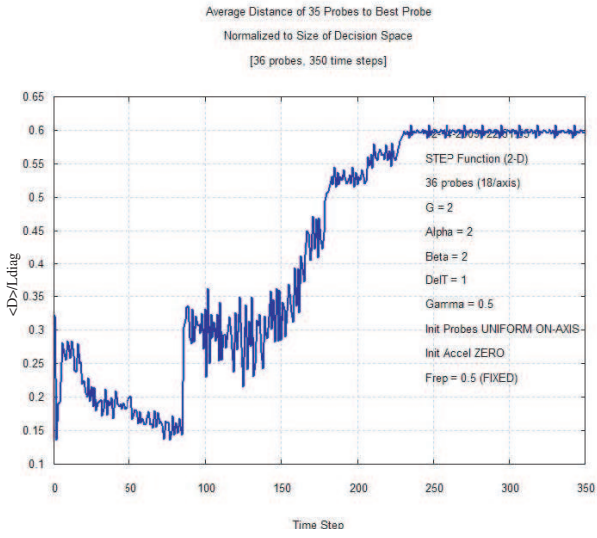


Figure 11. CFO's D_{avg} for 2D step function.

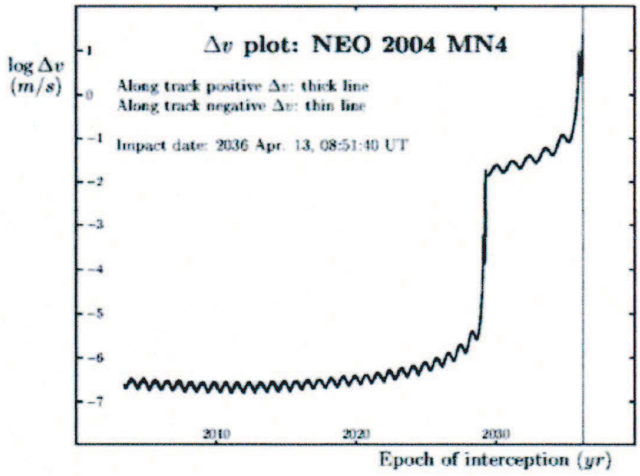


Figure 12. Reproduction of Schweickart's Fig. 2 for asteroid apophis.

Figure 11 exhibits several oscillatory plateaus connected by jumps in D_{avg}, much in the same way that asteroid Apophis' ΔV curve does. The visual similarity is self-evident and compelling. It seems implausible that this structural similarity is entirely coincidental. NEO trajectories are based on real gravity, while CFO's D_{avg} curves are based on an analogous "gravity" in metaphorical CFO space. That different curves based on analogous phenomena have very similar appearances not only seems reasonable, but highly probable. The striking similarity between NEO and CFO plots appears to be a convincing validation of the CFO metaphor. More importantly, it arguably points to an analytical framework that may provide a deterministic solution to the local trapping problem [18]. Perhaps the theory of gravitational resonant returns [17] can be applied to CFO, directly or with modification or extension, so that a completely deterministic solution to local trapping can be formulated. It is the author's hope that this possibility will be investigated by researchers motivated by these observations, hopefully leading to still more robust CFO implementations.

4. ANALYTICAL BENCHMARKS

The PBM benchmark suite was developed specifically because the usual analytical benchmark suites rely on problems that may not be representative of the types of problems encountered in antenna

Table 3. Comparative results for 23 benchmark functions (N_d = Function Dimension, $f_{\max}$ = Known Global Maximum).

*Test Function**	N_d	$f_{\max}$*	*<Best Fitness>/ Other Algorithm*	*---------- CFO ----------*			
				Best Fitness	γ_{best}	*Best Run* N_{eval}	*Total* N_{eval}
Unimodal Functions (other algorithms: average of 1000 runs)							
f_1	30	0	**-3.6927×10^{-37}** / PSO	-2.6592×10^{-2}	0.5	6,960	108,660
f_2	30	0	**-2.9168×10^{-24}** / PSO	-4×10^{-8}	0.5	5,040	161,640
f_3	30	0	-1.1979×10^{-3} / PSO	**-6×10^{-8}**	0.5	10,260	239,340
f_4	30	0	−0.1078 / GSO	**-4.2×10^{-7}**	0.5	5,160	59,160
f_5	30	0	−37.3582 / PSO	**-2.17187×10^{-2}**	0.2	21,600	164,160
f_6	30	0	-1.6000×10^{-2} / GSO	**0**	0	4,980	73,620
f_7	30	0	-9.9024×10^{-3} / PSO	**-3.55996×10^{-3}**	0.3	6,060	66,660
Multimodal Functions, Many Local Maxima (other algorithms: avg 1000 runs)							
f_8	30	12,569.5	**12,569.4882** / GSO	12,569.4852	0.8	7,980	69,720
f_9	30	0	−0.6509 / GA	**-3.52×10^{-6}**	1.0	6,840	117,120
f_{10}	30	0	-2.6548×10^{-5} / GSO	**-1.5×10^{-7}**	0.5	5,100	111,660
f_{11}	30	0	**-3.0792×10^{-2}**/ GSO	−2.00124	1.0	13,980	160,680
f_{12}	30	0	**-2.7648×10^{-11}** / GSO	−0.105859	1.0	11,760	68,220
f_{13}	30	0	**-4.6948×10^{-5}**/ GSO	-6.5966×10^{-2}	0.5	11,100	103,320
Multimodal Functions, Few Local Maxima (other algorithms: avg 50 runs)							
f_{14}	2	−1	**−0.9980** / GSO	−1.005284	0.3	920	12,824
f_{15}	4	−0.0003075	**$-3.7713\text{x}10^{-4}$** / GSO	-2.41631×10^{-3}	0.5	1,584	19,920
f_{16}	2	1.0316285	**1.031628 /** GSO	1.031607	0.1	896	9,256
f_{17}	2	−0.398	−0.3979 / GSO	**−0.398**	0.6	616	8,824
f_{18}	2	−3	**−3 /** GSO	**−3**	0	2,776	15,784
f_{19}	3	3.86	**3.8628 /** GSO	3.86157	0.4	924	12,612
f_{20}	6	3.32	3.2697 / GSO	**3.31976**	0.2	3,000	52,128
f_{21}	4	10	7.5439 / PSO	**10.1466**	0.6	4,368	25,376
f_{22}	4	10	8.3553 / PSO	**10.4028**	0.4	4,672	29,168
f_{23}	4	10	8.9439 / PSO	**10.5362**	0.4	2,256	24,784

* Note: Negative of the functions in [19] are computed by CFO because it searches for maxima instead of minima.

optimization or other problems in applied EM. Nevertheless, analytical benchmarks serve the purpose of comparing EAs across a wide range of functions of varying dimensionality with known maxima, thereby providing a another measure of EA effectiveness. The improved CFO algorithm was tested against the twenty-three function benchmark suite in [19] and the results compared to the other algorithms' results in that paper. The benchmarks, their DSes, and the other algorithms ("GA", "PSO", "GSO") are discussed in detail in [19].

GSO (Group Search Optimizer) is a novel Nature-inspired metaheuristic that mimics animal searching behavior based on a "producer-scrounger" model. The version of PSO implemented in [19] was "PSOt," a MATLAB-based toolbox that includes standard and variant PSO algorithms. Similarly, the genetic algorithm in [19] was implemented using GAOT (genetic algorithm optimization toolbox). Recommended default parameter values were used for the PSOt and GA algorithms as described in [19]. This section provides a detailed comparison of CFOs results to these other widely used and highly developed algorithms by displaying the improved CFO's results side-by-side those reported in [19].

Tables 3 and 4 summarize CFO's results using the same numbering as [19]. $f_{\max}$ is the known global *maximum* (note that the negative of each benchmark in [19] is used because, unlike the other algorithms, CFO locates maxima, not minima). $< \cdot >$ denotes average value. Because GA, PSO, and GSO are inherently stochastic, evaluating their performance requires a statistical assessment. Data in the tables for those algorithms are reproduced from [19].

CFO's results are repeatable over runs with the same parameters because it is deterministic but see [20] for a discussion of "pseudorandomness" in CFO. The CFO data correspond to the best fitness returned by the single best run in the set of runs with variable γ. As described above, a total of eleven runs were made with $0 \leq \gamma \leq 1$ in increments of 0.1. The γ value corresponding to the best fitness is γ_{best}. N_{eval} is the total number of function evaluations for the single best run and for the group of eleven runs used to determine γ_{best}.

CFO's results are quite good. It returned the best fitness (bold font) on twelve of the twenty three functions, equal fitness on one (f_{18}), and essentially the same fitness on another (f_8). CFO returned very similar best fitnesses on three benchmarks (f_{14}, f_{16}, and f_{19}). On seventeen benchmarks CFO performed better than or essentially as well as the best of the algorithms GSO, GA and PSO. On the remaining six, CFO's performance was mixed. But CFO required far fewer function evaluations than the other algorithms in all cases because it is deterministic (see [19] for details).

Table 4. Results for six 300-dimensional multimodal benchmark functions.

Test Function	$f_{\max}$	<Best Fitness>/ Other Algorithm	- - - - - CFO - - - - -	
			Best Fitness	γ
f_8	125,694.7	125,351.2 / GSO	**125,691.261**	0.8
f_9	0	−98.9 / GSO	$\mathbf{-3.491 \times 10^{-5}}$	1.0
f_{10}	0	-3.9540×10^{-6} / PSO	$\mathbf{-5 \times 10^{-8}}$	0.5
f_{11}	0	$\mathbf{-1.8239 \times 10^{-7}}$ / GSO	−37.20625	1.0
f_{12}	0	$\mathbf{-8.2582 \times 10^{-8}}$/ GSO	−0.018772	0.6
f_{13}	0	$\mathbf{-2.0175 \times 10^{-7}}$ / GSO	−12.23783	0.5

Scalability is another issue in optimization, that is, how well an EA performs as the DS dimensionality changes. In [19] the tested algorithms were evaluated for scalability using benchmarks f_8–f_{13} in 300D. CFO was tested in the same way using $\gamma = \gamma_{best}$ from the $N_d = 30$ runs, except for f_{12} where the second best 30D γ value was used (of course, there is no way of knowing what value of γ is the best choice without making 300D variable-γ runs which are prohibitively long). In order to avoid excessive runtime, each run was manually (and therefore subjectively) terminated when fitness saturation seemed to set in. The scalability results appear in Table 4. In three cases (f_8, f_9, f_{10}) CFO returned the best fitness. CFO also performed well against f_{12}, but compared to GSO it did not perform well on f_{11} and f_{13}. Nevertheless, the data presented in [19, Table XI] arguably support the conclusion that CFO scales as well as the other algorithms described in that report.

5. CONCLUSION

This paper presents an improved implementation of Central Force Optimization for use in antenna optimization. The improved algorithm implements a variable initial probe distribution to provide better sampling of the DS landscape and shrinks the decision space around the location of the best fitness in order to speed convergence. These enhancements are suggested by experience with CFO and consequently are inherently empirical. This note also suggests what appears to be a compelling argument that the theory of gravitationally trapped NEOs may be applicable to CFO for the purpose of developing a deterministic methodology for mitigating or eliminating local trapping. Testing CFO against antenna benchmark problem PBM #1 and a suite of

twenty-three analytic benchmarks shows that the improved algorithm performs quite well. Future efforts might explore alternative variable initial probe distributions, including looping over the number of probes in addition to their placement, and reactive DS adaptation to achieve still better performance.

APPENDIX A. CFO THEORY

CFO locates the global *maxima* of an objective function $f(x_1, x_2, \ldots, x_{N_d})$ defined on decision space DS: $x_i^{\min} \leq x_i \leq x_i^{\max}$, $1 \leq i \leq N_d$, where x_i are the *decision variables*, i the coordinate number, and N_d DS's dimensionality. The value of $f(\vec{x})$ at $\vec{x}$ is its *fitness*. $f(\vec{x})$'s topology is unknown, and there is no *a priori* information about its maxima. CFO searches by flying "probes" through DS at discrete "time" steps (iterations). Each probe's location is specified by its position vector computed from *equations of motion* that are analogous to their real-world counterparts for material objects moving through physical space under the influence of gravity without energy dissipation.

$\vec{R}_j^p = \sum_{k=1}^{N_d} x_k^{p,j} \hat{e}_k$ is probe p's position vector at step j. $x_k^{p,j}$ are its coordinates and $\hat{e}_k$ the unit vector along the x_k-axis. Indices p, $1 \leq p \leq N_p$, and j, $0 \leq j \leq N_t$, respectively, are the probe number and iteration number. N_p and N_t are the corresponding *total* number of probes and *total* number of time steps.

Each probe experiences an acceleration created by the "gravitational pull" of "masses" in the decision space. Probe p's acceleration at step $j-1$ is given by

$$\vec{a}_{j-1}^p = G \sum_{\substack{k=1 \\ k \neq p}}^{N_p} U\left(M_{j-1}^k - M_{j-1}^p\right) \cdot \left(M_{j-1}^k - M_{j-1}^p\right)^{\alpha} \times \frac{\left(\vec{R}_{j-1}^k - \vec{R}_{j-1}^p\right)}{\left\|\vec{R}_{j-1}^k - \vec{R}_{j-1}^p\right\|^{\beta}}. \tag{A1}$$

$M_{j-1}^p = f(x_1^{p,j-1}, x_2^{p,j-1}, \ldots, x_{N_d}^{p,j-1})$ is the objective function's fitness at probe$p$'s location at time step $j-1$. Each of the other probes at that iteration has associated with it fitness M_{j-1}^k, $k = 1, \ldots, p-1, p+1, \ldots, N_p$. G is CFO's "gravitational constant." $U(\cdot)$ is the Unit Step function, $U(z) = \left\{ \begin{array}{ll} 1, & z \geq 0 \\ 0, & \text{otherwise} \end{array} \right\}$.

The acceleration $\vec{a}_{j-1}^p$ causes probe p to move from position $\vec{R}_{j-1}^p$

at step $j-1$ to $\vec{R}_j^p$ at step j according to the trajectory equation

$$\vec{R}_j^p = \vec{R}_{j-1}^p + \frac{1}{2}\vec{a}_{j-1}^p\,\Delta t^2, \quad j \geq 1. \tag{A2}$$

Equations (A1) and (A2) collectively are the *equations of motion* in CFO space. They combine to compute a new probe distribution at each time step using the CFO "masses" discovered by the probe distribution at the previous step. Δt is the "time" interval between steps during which the acceleration is constant. This terminology has no significance beyond reflecting CFO's kinematic roots, as does the factor one-half in Eq. (A2). G and Δt are analogous to parameters in the equations of motion for real masses moving under real gravity. By contrast, the exponents α and β have no analogs in Nature. They are included to provide added flexibility to the user, because in metaphorical "CFO space" the algorithm designer is free to change how CFO "gravity" varies with distance or mass or both if doing so results in better performance.

The concept of "mass" in CFO space is very important because, unlike real mass, it is a positive-definite *user-defined function* of the objective function's fitness, not (necessarily) the fitness itself. In the CFO implementation described here, for example, mass is defined as $MASS_{CFO} = U(M_{j-1}^k - M_{j-1}^p) \cdot (M_{j-1}^k - M_{j-1}^p)^\alpha$ [difference in fitness values raised to the α power multiplied by the Unit Step]. Some other function can be used instead if it provides better results. In this particular case, the Unit Step is critical to the definition, because without it CFO mass could be negative depending on which fitness is greater. Mass always is positive in the real world, and as a consequence gravity always attractive. In metaphorical CFO space, where mass can be positive or negative depending on how it is defined, undesirable effects can result from the wrong definition. For example, negative mass creates a repulsive gravitational force that flies probes away from maxima instead of toward them, which defeats the very purpose of the algorithm.

Because probes may fly outside space DS into regions of unallowable solutions, a question arises as what to do. While many schemes are possible, a simple, empirically determined one is used here. On a coordinate-by-coordinate basis, probes flying out of the decision space are placed a fraction $\Delta F_{rep} \leq F_{rep} \leq 1$ of the distance between the probe's starting coordinate and the corresponding boundary coordinate. F_{rep} is the variable "repositioning factor" introduced in [2]. Its value, as well as those of all the CFO parameters, were determined empirically.

REFERENCES

1. Formato, R. A., "Central force optimization: A new metaheuristic with applications in applied electromagnetics," *Progress In Electromagnetics Research*, PIER 77, 425–491, 2007.
2. Formato, R. A., "Central force optimization: A new computational framework for multidimensional search and optimization," *Nature Inspired Cooperative Strategies for Optimization (NICSO 2007)*, Studies in Computational Intelligence 129, N. Krasnogor, G. Nicosia, M. Pavone, and D. Pelta (eds.), Vol. 129, Springer-Verlag, Heidelberg, 2008.
3. Formato, R. A., "Central force optimisation: A new gradient-like metaheuristic for multidimensional search and optimisation," *Int. J. Bio-inspired Computation*, Vol. 1, No. 4, 217–238, 2009.
4. Formato, R. A., "Central force optimization: A new deterministic gradient-like optimization metaheuristic," *OPSEARCH*, *Jour. of the Operations Research Society of India*, Vol 46, No. 1, 25–51, 2009.
5. Mohammad, G. and N. Dib, "Synthesis of antenna arrays using central force optimization," *Mosharaka International Conference on Communications, Computers and Applications*, 6–8, Amman, Jordan, Feb. 2009.
6. Qubati, G. M., R. A. Formato, and N. I. Dib, "Antenna benchmark performance and array synthesis using central force optimization," *IET (U.K.) Microwaves, Antennas & Propagation*, 2000 (in press).
7. Pantoja, M. F., A. R. Bretones, and R. G. Martin, "Benchmark antenna problems for evolutionary optimization algorithms," *IEEE Trans. Antennas and Propagation*, Vol. 55, No. 4, 1111–1121, Apr. 2007.
8. Li, W. T., X. W. Shi, and Y. Q. Hei, "An improved particle swarm optimization algorithm for pattern synthesis of phased arrays," *Progress In Electromagnetics Research*, PIER 82, 319–332, 2008.
9. Ghaffari-Miab, M., A. Farmahini-Farahani, R. Faraji-Dana, and C. Lucas, "An efficient hybrid swarm intelligence-gradient optimization method for complex time Green's functions of multilayered media," *Progress In Electromagnetics Research*, PIER 77, 181–192, 2007.
10. Sijher, T. S. and A. A. Kishk, "Antenna modeling by infinitesimal dipoles using genetic algorithms," *Progress In Electromagnetics Research*, PIER 52, 225–254, 2005.
11. Cengiz, Y. and H. Tokat, "Linear antenna array design with use

of genetic, memetic and tabu search optimization algorithms," *Progress In Electromagnetics Research C*, Vol. 1, 63–72, 2008.

12. Zainud-Deen, H. H., W. M. Hassen, and K. H. Awadalla, "Crack detection using a hybrid finite difference frequency domain and particle swarm optimization techniques," *Progress In Electromagnetics Research M*, Vol. 6, 47–58, 2009.
13. Mangaraj, B. B., I. S. Misra, and A. K. Barisal, "Optimizing included angle of symmetrical V-dipoles for higher directivity using bacteria foraging optimization algorithm," *Progress In Electromagnetics Research B*, Vol. 3, 295–314, 2008.
14. Yau, D. and S. Crozier, "A genetic algorithm/method of moments approach to the optimization of an RF coil for MRI applications — Theoretical considerations," *Progress In Electromagnetics Research*, PIER 39, 177–192, 2003.
15. Burke, G. J., "Numerical electromagnetics code — NEC-4, method of moments, Part I: User's manual and Part II: Program description — Theory," UCRL-MA-109338, Lawrence Livermore National Laboratory, Livermore, California, USA, Jan. 1992. https://ipo.llnl.gov/technology/software/softwaretitles/nec.php.
16. Schweickart, R., C. Chapman, D. Durda, P. Hut, B. Bottke, and D. Nesvorny, "Threat characterization: Trajectory dynamics (white paper 039)," 2006. http://arxiv.org/abs/physics/0608155.
17. Valsecchi, G. B., A. Milani, G. F. Gronchi, and S. R. Chesley, "Resonant returns to close approaches: Analytical theory," *Astronomy & Astrophysics*, Vol. 408, No. 3, 1179–1196, 2003. http://www.aanda.org.
18. Formato, R. A., "Are near earth objects the key to optimization theory?" arXiv:0912.1394v1 [astro-ph.EP].
19. He, S., Q. H. Wu, and J. R. Saunders, "Group search optimizer: An optimization algorithm inspired by animal searching behavior," *IEEE Trans. Evol. Computation*, Vol. 13, No. 5, 973–990, Oct. 2009.
20. Formato, R. A., "Pseudorandomness in central force optimization," arXiv:1001.0317v1[cs.NE], www.arXiv.org, 2010.