

Pseudorandomness in Central Force Optimization

Richard A. Formato^{1*}

¹Consulting Engineer & Registered Patent Attorney
Cataldo & Fisher, LLC
P.O. Box 1714, Harwich, MA 02645 USA

ABSTRACT

Central Force Optimization is a deterministic metaheuristic for an evolutionary algorithm that searches a decision space by flying probes whose trajectories are computed using a gravitational metaphor. CFO benefits substantially from the inclusion of a pseudorandom component (a numerical sequence that is precisely known by specification or calculation but otherwise arbitrary). The essential requirement is that the sequence is uncorrelated with the decision space topology, so that its effect is to pseudorandomly distribute probes throughout the landscape. While this process may appear to be similar to the randomness in an inherently stochastic algorithm, it is in fact fundamentally different because CFO remains deterministic at every step. Three pseudorandom methods are discussed (initial probe distribution, repositioning factor, and decision space adaptation). A sample problem is presented in detail and summary data included for a 23-function benchmark suite. CFO's performance is quite good compared to other highly developed, state-of-the-art algorithms.

Keywords: Central Force Optimization, Optimization, Metaheuristic, Evolutionary Algorithm, Pseudorandomness, Decision Space Exploration and Exploitation

1. INTRODUCTION

This note examines the role of pseudorandomness in Central Force Optimization. CFO is a deterministic Nature-inspired search and optimization metaheuristic for an evolutionary algorithm (EA) based on gravitational kinematics [1-3]. Proofs of convergence for CFO and an extended version have been developed [4,5], and the algorithm has been implemented on a GPU using various topologies [6-8]. The algorithm has been successfully applied to a variety of problems, among them: training neural networks [9]; drinking water distribution networks [10]; solving nonlinear circuits [11]; array synthesis [12,13]; microstrip patch antenna design [14]; multiband slotted bowtie design [15]; rectangular microstrip patch design [16]; microwave broadband absorber design [17]; antenna optimization generally [18]; notched ultra wideband E-shape antenna design [19]; and increasing impedance bandwidth [20,21].

CFO analogizes Newton's mathematically precise laws of motion and gravity, so that its underlying equations are equally precise. The algorithm locates the global maxima of an *objective function* defined on a *decision space* Ω with unknown topology ("landscape"). CFO searches Ω by "flying" a group of "probes" whose trajectories are computed from two deterministic *equations of motion* at a series of discrete "time" steps (iterations). Details of the CFO metaheuristic are in the Appendix. CFO is fundamentally different from Nature-inspired EAs whose underlying equations are inherently stochastic. Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) are examples. Their equations are formulated in terms of true random variables, and removing randomness causes these algorithms to fail completely. By

contrast, CFO's equations are inherently deterministic. Every CFO run with the same setup returns precisely the same values step-by-step throughout the entire run. Nevertheless, effective implementations substantially benefit from a "pseudorandom" component that enters the algorithm indirectly, not through its basic equations. Although pseudorandomness is not required in CFO, numerical experiments show that it is an important feature in effective implementations.

A *pseudorandom* variable is defined here as one whose value is precisely known but arbitrarily assigned. The value can be specified in advance (for example, an arbitrary sequence of numbers) or it can be calculated in a prescribed manner. The variable's "randomness" derives from the fact that its value is arbitrary and uncorrelated with Ω 's topology, not that it is uncertain in the sense of a true random variable. A true *random* variable's value is calculated from a probability distribution with successive calculations yielding different values that cannot be known in advance. This type of randomness is fundamentally different from CFO's pseudorandomness. Even when CFO includes pseudorandomness, it remains deterministic, always yielding the same result for runs with the same setup. Once a pseudorandom variable is specified, either explicitly or by calculation, its value is known with absolute precision, so that CFO's trajectory calculations are deterministic even in the presence of pseudorandomness. There are many ways pseudorandomness can be added to CFO; three simple methods are described here.

2. CFO METHODOLOGY

This paper discusses a CFO implementation with pseudorandomness injected in the following ways: (1) the initial probe distribution (IPD); (2) the "repositioning factor;" and (3) changing the decision space boundaries. The algorithm is referred to as "CFO-PR." Every CFO run starts with a user-specified IPD (total number of probes and their locations in Ω at the beginning of the run, step 0). An arbitrary, variable initial probe distribution is a convenient way to inject pseudorandomness, the effect of which is to provide better sampling of Ω 's topology than a static distribution. Each initial probe distribution in a set of distributions provides different information about Ω 's landscape. As the results below show, certain ones perform much better than others.

The second method of injecting pseudorandomness is the use of a step-by-step variable repositioning factor $\Delta F_{rep} \leq F_{rep} \leq 1$ where ΔF_{rep} is the step increment. "Repositioning" refers to the process of retrieving a probe that has flown outside the decision space (discussed in detail in the Appendix). A variable F_{rep} has the effect of pseudorandomly distributing probes throughout Ω , which provides better sampling of the decision space landscape as a run progresses. The third way pseudorandomness is injected is by shrinking the decision space around the best probe's location. This process coupled with variable F_{rep} redistributes probes in the smaller Ω in an arbitrary but precise, hence pseudorandom, manner. The effect is to speed CFO's convergence, but at the risk of premature convergence [on an empirical basis, this issue does not appear to be significant using the procedures described below].

Pseudocode for CFO-PR appears in Fig. 1. The inner time step loop (" j " loop) is common to all CFO implementations, but the two outer loops inject initial probe pseudorandomness. The γ loop controls where initial probes are deployed, and the N_p/N_d loop determines their number. These two parameters define the initial probe distribution, and the two loops together create a variable, pseudorandom distribution (see Appendix 1 for notation and equations). The variable F_{rep} procedure appears in step (f) of the pseudocode. And the pseudorandom decision space adaptation is in step (g). Ω 's boundaries shrink around the then best probe position vector every 20^{th} step as discussed in detail below. A two-dimensional example is used to illustrate these techniques because it provides a concrete visualization of the different methods. In the actual CFO-PR implementation, of course, these techniques are generalized to the N_d -dimensional case.

2.1 Initial Probes

The manner in which initial probes are deployed using γ is shown schematically in Fig. 2, which provides a 2-dimensional (2D) schematic representation of a variable initial probe distribution comprising an orthogonal array of N_p/N_d probes per axis deployed uniformly on "probe lines" parallel to the coordinate axes that intersect at a point along Ω 's

principal diagonal. N_d is Ω 's dimensionality (here $N_d = 2$), and $x_i^{\min} \leq x_i \leq x_i^{\max}$, $1 \leq i \leq N_d$ define Ω 's domain (decision space).

```

For  $N_p/N_d = 2$  to 8 by 2 :
  For  $\gamma = \gamma_{start}$  to  $\gamma_{stop}$  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,
            $\vec{R}_j^p, 1 \leq p \leq N_p$  [eq.(2)].
      (c) Retrieve errant probes ( $1 \leq p \leq N_p$ ):
           If  $\vec{R}_j^p \cdot \hat{e}_i < x_i^{\min} \therefore$ 
            $\vec{R}_j^p \cdot \hat{e}_i = x_i^{\min} + F_{rep}(\vec{R}_{j-1}^p \cdot \hat{e}_i - x_i^{\min})$ 
           If  $\vec{R}_j^p \cdot \hat{e}_i > x_i^{\max} \therefore$ 
            $\vec{R}_j^p \cdot \hat{e}_i = x_i^{\max} - F_{rep}(x_i^{\max} - \vec{R}_{j-1}^p \cdot \hat{e}_i)$ 

      (d) Compute fitness matrix for current
      probe
      distribution.

      (e) Compute accelerations using current
      probe
      distribution and fitnesses [eq. (1)].

      (f) Increment  $F_{rep}$  by  $\Delta F_{rep}$  :
           If  $F_{rep} > 1 \therefore F_{rep} = \Delta F_{rep}$ .

      (g) If  $j \text{ MOD } 20 = 0 \therefore$ 
           Shrink  $\Omega$  around  $\vec{R}_{best}$ .

    Next  $j$ 
  Next  $\gamma$ 
Next  $N_p/N_d$ 

```

Fig. 1. CFO-PR Pseudocode with Variable Initial Probes and F_{rep} and DS Adaptation

For illustrative purposes, Fig. 2 shows nine probes on each probe line. The lines, which are parallel to the x_1 and x_2 axes, intersect at a point on Ω 's principal diagonal marked by position vector $\vec{D} = \vec{X}_{\min} + \gamma(\vec{X}_{\max} - \vec{X}_{\min})$, where

137 $\vec{X}_{\min} = \sum_{i=1}^{N_d} x_i^{\min} \hat{e}_i$ and $\vec{X}_{\max} = \sum_{i=1}^{N_d} x_i^{\max} \hat{e}_i$ are the diagonal's endpoint vectors. Parameter $0 \leq \gamma \leq 1$ specifies where along
 138 the diagonal the orthogonal probe array is placed by locating the probe lines' intersection point.

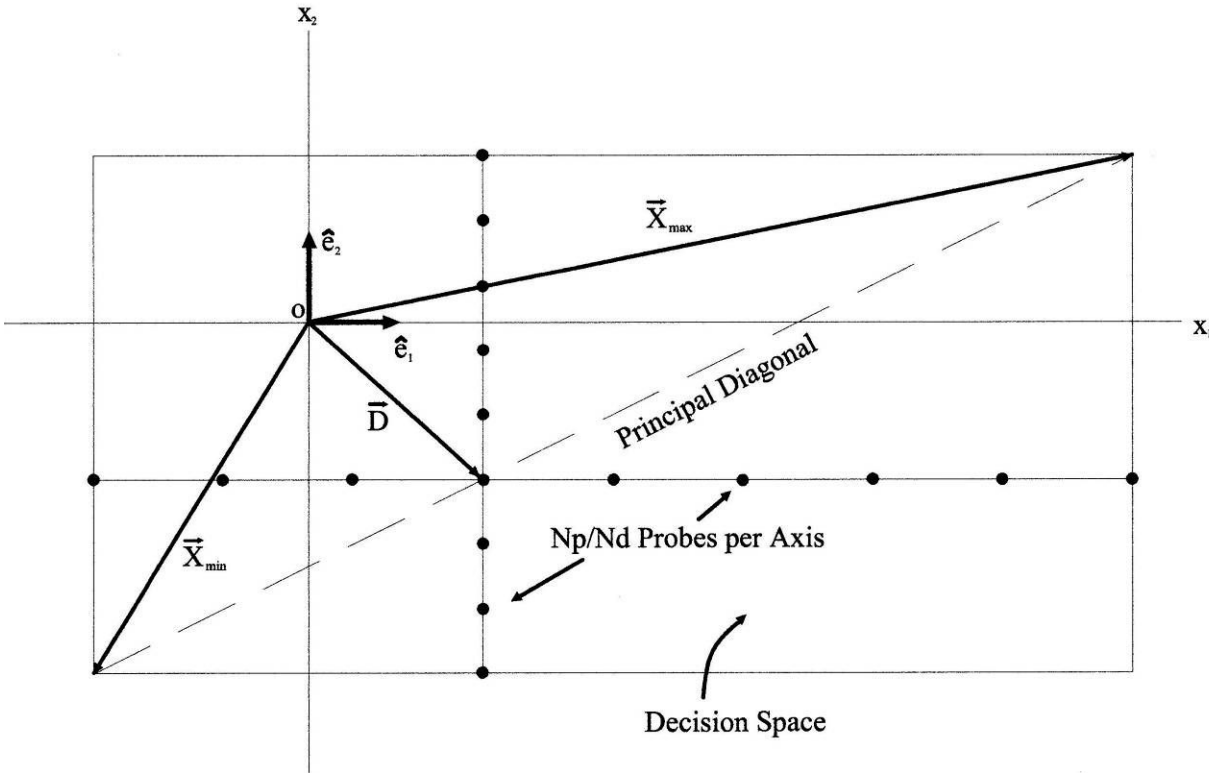
139 While Fig. 2 shows an equal number of probes on each line, a different number of probes per axis can be used
 140 instead. For example, if equal probe spacing were desired in a decision space with unequal boundaries, or if overlapping
 141 probes were to be excluded in a symmetrical space, then unequal numbers would be used. Unequal numbers also might
 142 be appropriate if *a priori* knowledge of Ω 's landscape, however obtained, suggests denser sampling in one area. The
 143 initial probe distribution in Fig. 2 with variable N_p/N_d was used for the CFO-PR runs reported here, but any number of
 144 other variable initial probe distributions could be used instead. The key idea is that the initial probe distribution must be
 145 pseudorandom, that is, arbitrary and therefore uncorrelated with the decision space landscape.

146

147 2.2 Repositioning Factor

148 A variable value for F_{rep} also adds pseudorandomness. F_{rep} starts at some arbitrary initial value that is
 149 incremented at each iteration by an arbitrary amount ΔF_{rep} so that $\Delta F_{rep} \leq F_{rep} \leq 1$. This errant probe retrieval scheme
 150 is an example of an arbitrary sequence of calculated numbers deterministically assigned to a CFO parameter. CFO's
 151 ability to search the decision space depends on where errant probes are reinserted in Ω , and this process is
 152 pseudorandom in nature because F_{rep} is deterministic but arbitrary and uncorrelated with Ω 's topology. By placing errant
 153 probes pseudorandomly throughout the decision space, more information is developed about its topology as the run
 154 progresses. The scheme used here was empirically determined, and appears to work well across a wide range of
 155 objective functions. But, of course, there are many other procedures for setting F_{rep} 's value, some no doubt better than
 156 others.

157



158

159

160

Fig. 2. Variable 2-D Initial Probe Distribution used for CFO Runs Reported Here

2.3 Decision Space Adaptation

CFO-PR also includes adaptive reconfiguration of the decision space in order to improve convergence speed. This feature also is pseudorandom in nature because the way Ω 's boundaries are changed is arbitrary and uncorrelated with the landscape. Fig. 3 illustrates in 2D how Ω 's size is adaptively reduced in this case every 20th step around the probe's location with the then best fitness throughout the run up to the current iteration, $\bar{R}_{best}$. Ω 's boundary coordinates are reduced by one-half the distance from the best probe's position to the each boundary on a coordinate-by-coordinate basis, that is, $x'_i{}^{\min} = x_i{}^{\min} + \frac{\bar{R}_{best} \cdot \hat{e}_i - x_i{}^{\min}}{2}$ and $x'_i{}^{\max} = x_i{}^{\max} - \frac{x_i{}^{\max} - \bar{R}_{best} \cdot \hat{e}_i}{2}$, where the primed coordinate is the new decision space boundary, and the dot denotes vector inner product. For clarity, Fig. 3 shows $\bar{R}_{best}$ as fixed, whereas generally it varies throughout a run. Changing Ω 's boundary every twenty steps instead of some other interval was chosen arbitrarily (another, probably better approach, might be a reactive adaptation based on performance measures such as convergence speed or fitness saturation).

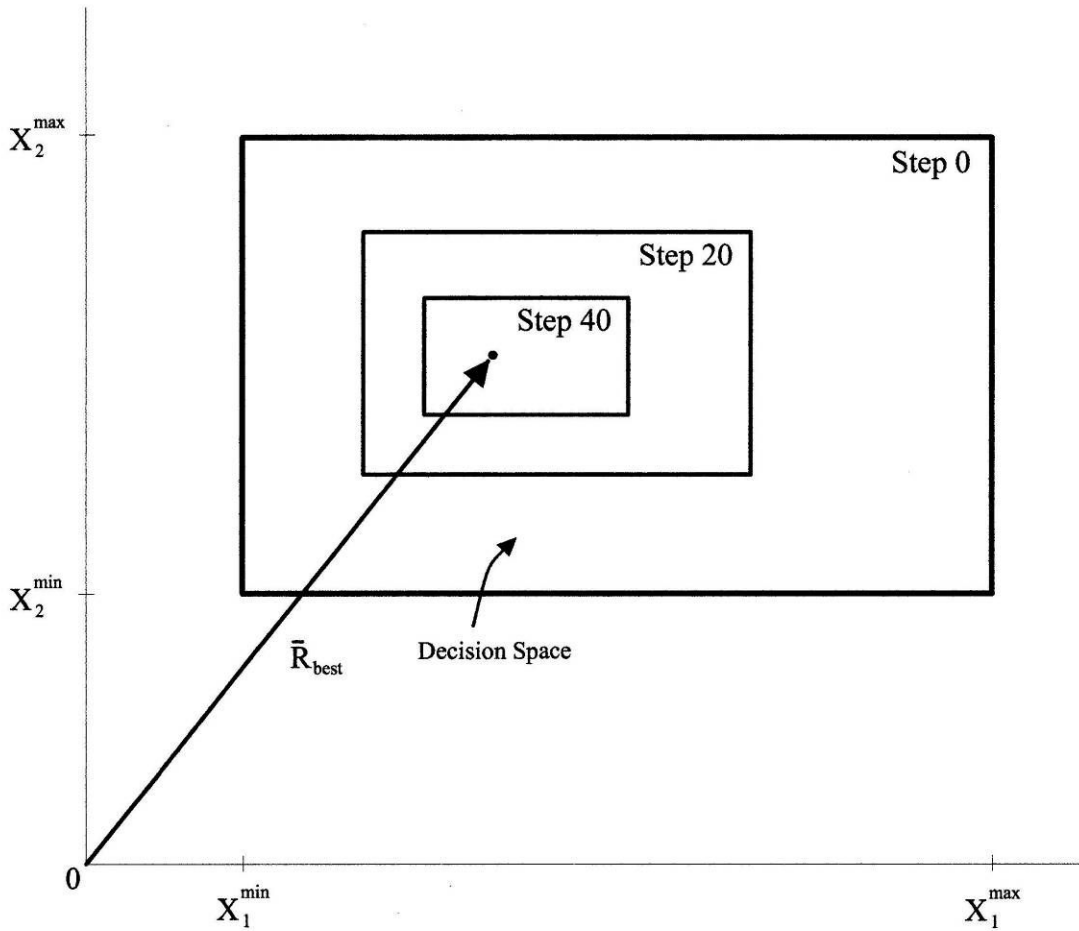


Fig. 3. Schematic 2-D Decision Space Adaptation (with constant $\bar{R}_{best}$)

2.4 A Sample Problem

The effectiveness of injecting pseudorandomness into CFO-PR will be illustrated with the 2D Goldstein-Price function ("GP") plotted in Fig. 4. GP is defined as

$$f(x_1, x_2) = -[1 + (x_1 + x_2 + 1)^2 \cdot (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \\ \times [30 + (2x_1 - 3x_2)^2 \cdot (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$$

where $\Omega: -100 \leq x_1, x_2 \leq 100$ [note that in most published reports Ω is much smaller, viz., $-2 \leq x_1, x_2 \leq 2$]. GP's global maximum is -3 at $(0, -1)$. This function is multimodal with few local maxima, and it varies over nearly nineteen orders of magnitude as shown in Fig. 4.

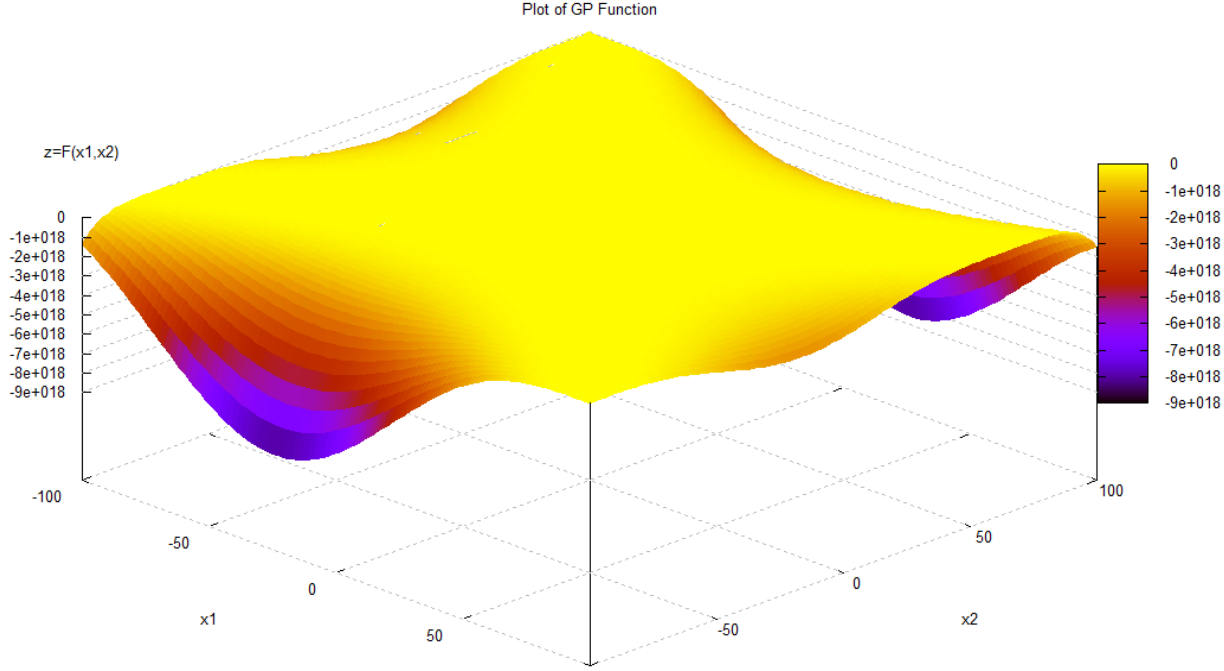


Fig. 4. Goldstein-Price (GP) Function

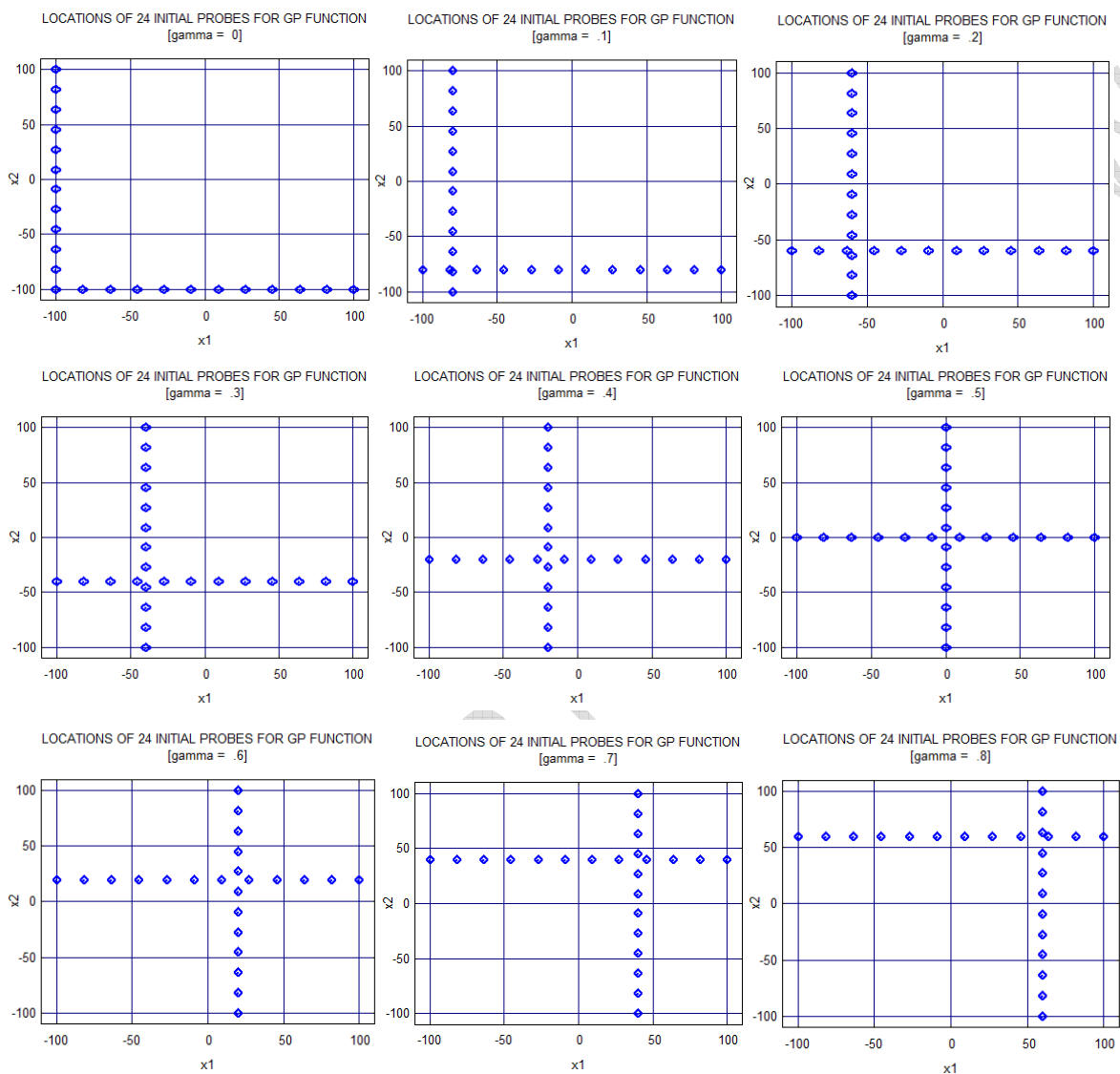
The following parameter values were used for all runs reported in this note: $\alpha = 2$, $\beta = 2$, $G = 2$, $\Delta t = 1$, initial acceleration of zero, initial $F_{rep} = 0.5$, $\Delta F_{rep} = 0.05$, $\gamma_{start} = 0$, $\gamma_{stop} = 1$ with $\Delta\gamma = 0.1$ (eleven runs). For GP, $N_p/N_d = 4$ to 14 by 2, with different ranges of this parameter for the other test functions as described below. In all cases, 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} . The pseudorandom IPD's computed using the procedure illustrated in Fig. 2 for GP are plotted in Fig. 5. A further illustration of the Probe Line IPD concept is shown in 3D in Fig. 6.

Table 1 shows a summary of the results for the GP function. A total of sixty-six optimization runs were made in six groups of eleven runs each [data for "Run #0" are starting values]. The column headings are for the most part self-explanatory. Each run began with $N_t = 500$, but, as the #Steps column shows, in no case were 500 iterations used because every run terminated early. N_{eval} is the number of function evaluations performed for the shortened run, and the total number of evaluations over all runs appears at the bottom of this column. The F_{rep} column lists F_{rep} 's value at the end of the run, and the "V" denotes that F_{rep} was variable as discussed above. Fitness tabulates the best fitness returned during the run. The Initial Probes column shows the type of IPD, in this case probes uniformly spaced along probe lines parallel to Ω 's axes (notated "II-AXIS") as described above and shown in Fig. 5.

The best fitness ranged from a low of $-236.6643...$ in run 12 to the global maximum of -3 at $(0, -1)$, which was returned in run #54 (best results highlighted in blue). Parameters for run #54 were $N_p/N_d = 12$, $N_p = 24$, and

204 $\gamma = 0.9$. The total number of function evaluations over all runs was 180,472 while N_{eval} for the best run was 1,464 (60
 205 iterations). Fig. 7 plots the evolution of GP's best fitness, which in only two steps increases from
 206 $-2.992268247672 \times 10^{-12}$ to GP's actual global maximum of -3 . This seems to be quite remarkable in view of the
 207 initial probe distribution for $\gamma = 0.9$ (Fig. 5) in which all probes are far removed from the maximum's location at $(0, -1)$.
 208 As the data in Table 1 clearly show, some sets of parameters are much better than others. Without pseudorandom initial
 209 probes, a single run would be made with, in this example, only 13.6% probability of locating the global maximum with a
 210 fractional accuracy of 0.03% (9 of 66 runs). This statistic highlights the importance of pseudorandomness in CFO.

211



214

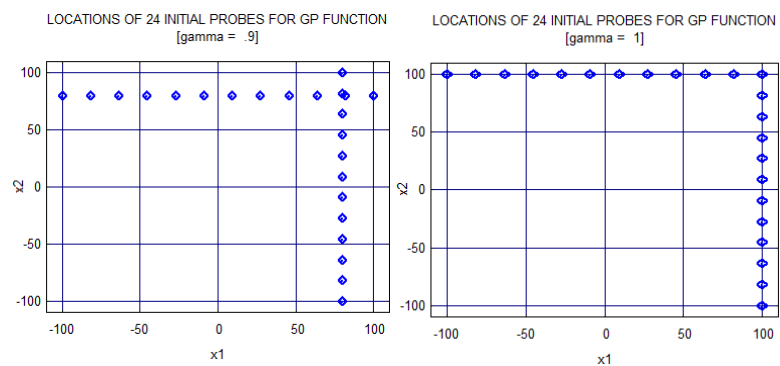


Fig. 5. GP Best Run Initial Probe Distributions

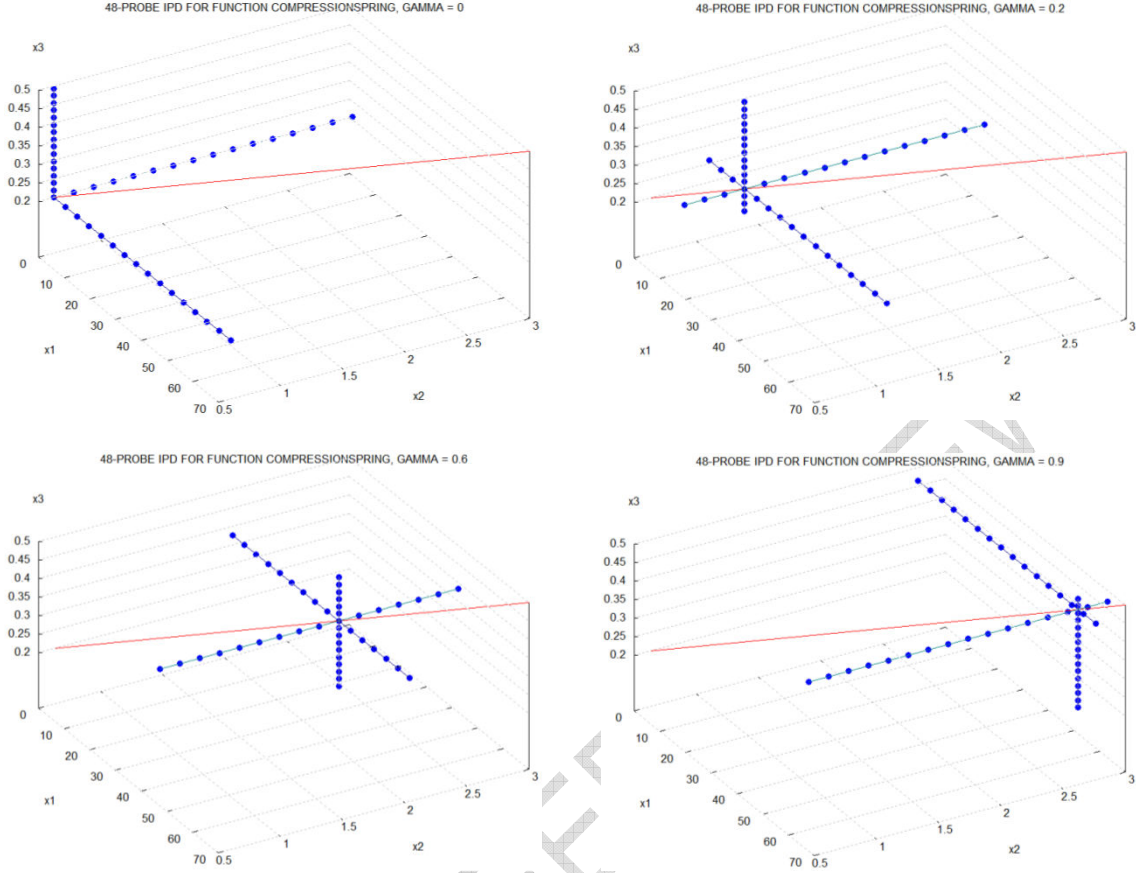


Fig. 6. Example of a 3D Probe Line IPD

Fig. 8 plots CFO's " D_{avg} curve" for GP. 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 Ω 's principal diagonal (see Appendix

for definitions). D_{avg} decreases monotonically through step 10 to 0.0496977, then increases very quickly to a peak of 0.4767301 at step 11, followed by another quasi-monotonic decrease through step 29 to 0.0274355. This cycle repeats through step 48 where D_{avg} is 0.0169657, followed by a jump to 0.1194779 at step 49. After a slight dip through step 53,

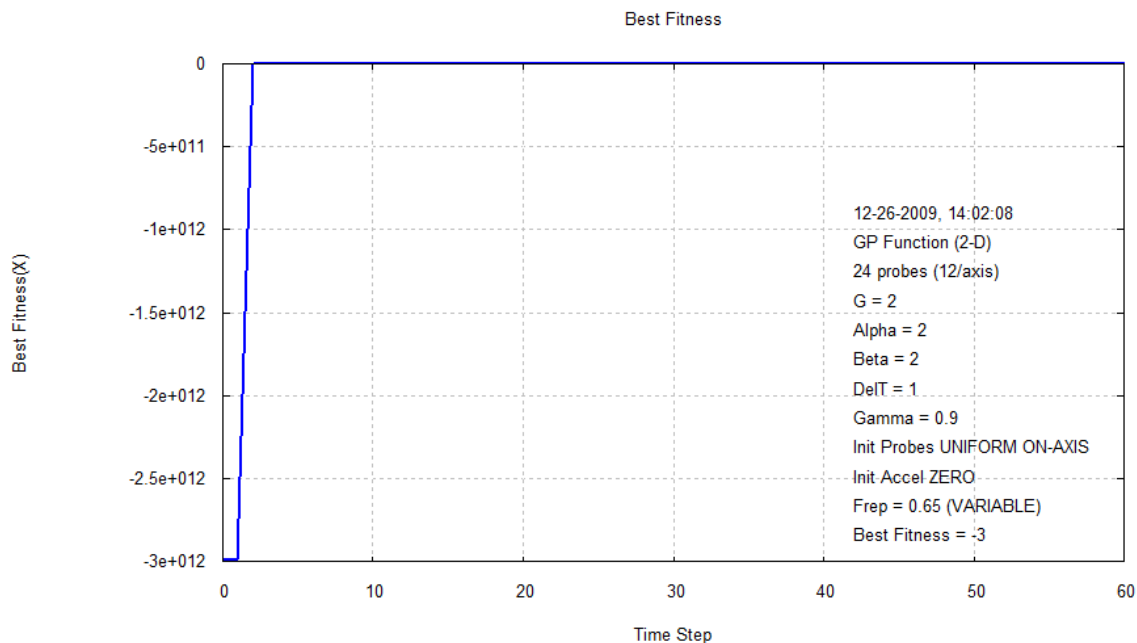
D_{avg} flattens out around a value of 0.11... The quasi-oscillatory behavior in D_{avg} usually correlates with local trapping, which in this case happens to be at the global maximum. Oscillation in D_{avg} may be a similar phenomenon to oscillation seen in " ΔV " curves for gravitationally trapped Near Earth Objects (NEOs), strongly suggesting that NEO theory may hold the key to analytical mitigation or elimination of local trapping at local maxima and possibly another proof of convergence for CFO.

241
242

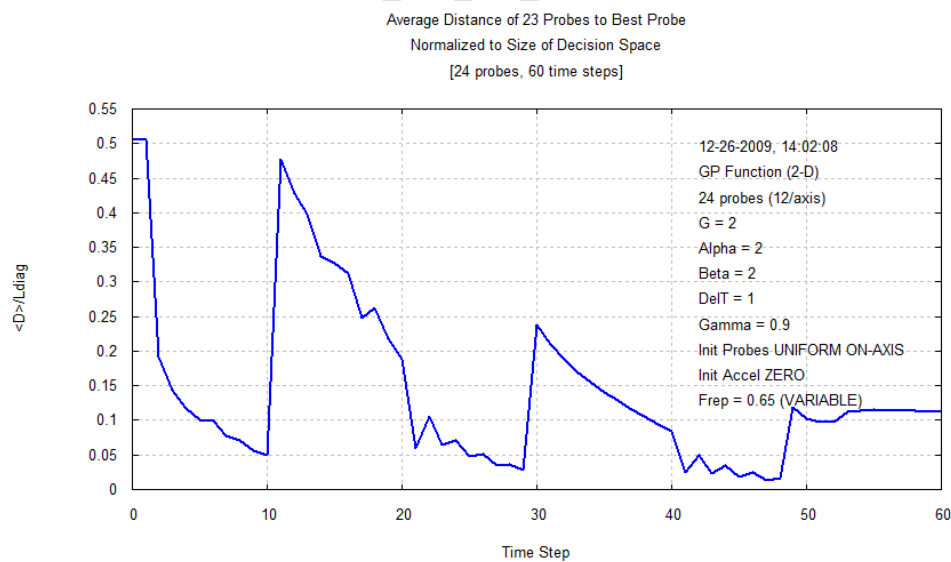
Table 1. Summary Results for GP Function using Pseudorandom CFO.

Run ID: 12-26-2009, 14:02:08														
FUNCTION: GP														
Run #	Gamma	Nt	Nd	Np	G	DelT	Alpha	Beta	#Steps	Neval	Frep	Fitness	Initial Probes	
0	0.000	500	2	8	2.0	1.0	2.00	2.00	0	0	0.50000V	-9999.00000000	UNIFORM	I-AXIS
1	0.000	500	2	8	2.0	1.0	2.00	2.00	78	632	0.60000V	-5.48169471	UNIFORM	I-AXIS
2	0.100	500	2	8	2.0	1.0	2.00	2.00	134	1080	0.55000V	-84.78003234	UNIFORM	I-AXIS
3	0.200	500	2	8	2.0	1.0	2.00	2.00	191	1536	0.55000V	-3.19822531	UNIFORM	I-AXIS
4	0.300	500	2	8	2.0	1.0	2.00	2.00	79	640	0.65000V	-8.62285433	UNIFORM	I-AXIS
5	0.400	500	2	8	2.0	1.0	2.00	2.00	403	3232	0.70000V	-84.47139703	UNIFORM	I-AXIS
6	0.500	500	2	8	2.0	1.0	2.00	2.00	261	2096	0.25000V	-10.56054743	UNIFORM	I-AXIS
7	0.600	500	2	8	2.0	1.0	2.00	2.00	135	1088	0.60000V	-4.98190912	UNIFORM	I-AXIS
8	0.700	500	2	8	2.0	1.0	2.00	2.00	229	1840	0.55000V	-3.00130536	UNIFORM	I-AXIS
9	0.800	500	2	8	2.0	1.0	2.00	2.00	133	1072	0.50000V	-89.42718008	UNIFORM	I-AXIS
10	0.900	500	2	8	2.0	1.0	2.00	2.00	116	936	0.60000V	-8.57642421	UNIFORM	I-AXIS
11	1.000	500	2	8	2.0	1.0	2.00	2.00	193	1552	0.65000V	-3.08573605	UNIFORM	I-AXIS
12	0.000	500	2	12	2.0	1.0	2.00	2.00	78	948	0.60000V	-236.66437724	UNIFORM	I-AXIS
13	0.100	500	2	12	2.0	1.0	2.00	2.00	263	3168	0.35000V	-3.00574349	UNIFORM	I-AXIS
14	0.200	500	2	12	2.0	1.0	2.00	2.00	78	948	0.60000V	-57.61728445	UNIFORM	I-AXIS
15	0.300	500	2	12	2.0	1.0	2.00	2.00	304	3660	0.50000V	-3.00017384	UNIFORM	I-AXIS
16	0.400	500	2	12	2.0	1.0	2.00	2.00	154	1860	0.60000V	-3.55376552	UNIFORM	I-AXIS
17	0.500	500	2	12	2.0	1.0	2.00	2.00	196	2364	0.80000V	-3.00014046	UNIFORM	I-AXIS
18	0.600	500	2	12	2.0	1.0	2.00	2.00	154	1860	0.60000V	-6.35080630	UNIFORM	I-AXIS
19	0.700	500	2	12	2.0	1.0	2.00	2.00	212	2556	0.65000V	-3.03478155	UNIFORM	I-AXIS
20	0.800	500	2	12	2.0	1.0	2.00	2.00	78	948	0.60000V	-16.66444362	UNIFORM	I-AXIS
21	0.900	500	2	12	2.0	1.0	2.00	2.00	99	1200	0.70000V	-6.52063755	UNIFORM	I-AXIS
22	1.000	500	2	12	2.0	1.0	2.00	2.00	250	3012	0.65000V	-3.01375068	UNIFORM	I-AXIS
23	0.000	500	2	16	2.0	1.0	2.00	2.00	78	1264	0.60000V	-22.18779159	UNIFORM	I-AXIS
24	0.100	500	2	16	2.0	1.0	2.00	2.00	60	976	0.65000V	-16.89097168	UNIFORM	I-AXIS
25	0.200	500	2	16	2.0	1.0	2.00	2.00	173	2784	0.60000V	-3.01613943	UNIFORM	I-AXIS
26	0.300	500	2	16	2.0	1.0	2.00	2.00	79	1280	0.65000V	-76.69765275	UNIFORM	I-AXIS
27	0.400	500	2	16	2.0	1.0	2.00	2.00	189	3040	0.45000V	-3.00972155	UNIFORM	I-AXIS
28	0.500	500	2	16	2.0	1.0	2.00	2.00	191	3072	0.55000V	-6.65915188	UNIFORM	I-AXIS
29	0.600	500	2	16	2.0	1.0	2.00	2.00	122	1968	0.90000V	-15.75050525	UNIFORM	I-AXIS
30	0.700	500	2	16	2.0	1.0	2.00	2.00	60	976	0.65000V	-31.95935579	UNIFORM	I-AXIS
31	0.800	500	2	16	2.0	1.0	2.00	2.00	95	1536	0.50000V	-49.88324658	UNIFORM	I-AXIS
32	0.900	500	2	16	2.0	1.0	2.00	2.00	304	4880	0.50000V	-3.00045306	UNIFORM	I-AXIS
33	1.000	500	2	16	2.0	1.0	2.00	2.00	318	5104	0.25000V	-3.00010350	UNIFORM	I-AXIS
34	0.000	500	2	20	2.0	1.0	2.00	2.00	78	1580	0.60000V	-76.94725945	UNIFORM	I-AXIS
35	0.100	500	2	20	2.0	1.0	2.00	2.00	116	2340	0.60000V	-3.12194540	UNIFORM	I-AXIS
36	0.200	500	2	20	2.0	1.0	2.00	2.00	191	3840	0.55000V	-3.03175136	UNIFORM	I-AXIS
37	0.300	500	2	20	2.0	1.0	2.00	2.00	78	1580	0.60000V	-182.65850920	UNIFORM	I-AXIS
38	0.400	500	2	20	2.0	1.0	2.00	2.00	193	3880	0.65000V	-3.14860260	UNIFORM	I-AXIS
39	0.500	500	2	20	2.0	1.0	2.00	2.00	98	1980	0.65000V	-3.62879802	UNIFORM	I-AXIS
40	0.600	500	2	20	2.0	1.0	2.00	2.00	192	3860	0.60000V	-3.04067143	UNIFORM	I-AXIS
41	0.700	500	2	20	2.0	1.0	2.00	2.00	139	2800	0.80000V	-3.54277590	UNIFORM	I-AXIS
42	0.800	500	2	20	2.0	1.0	2.00	2.00	171	3440	0.50000V	-3.00072472	UNIFORM	I-AXIS
43	0.900	500	2	20	2.0	1.0	2.00	2.00	101	2040	0.80000V	-4.37891524	UNIFORM	I-AXIS
44	1.000	500	2	20	2.0	1.0	2.00	2.00	79	1600	0.65000V	-35.88632429	UNIFORM	I-AXIS
45	0.000	500	2	24	2.0	1.0	2.00	2.00	226	5448	0.40000V	-3.00031923	UNIFORM	I-AXIS
46	0.100	500	2	24	2.0	1.0	2.00	2.00	78	1896	0.60000V	-9.06529259	UNIFORM	I-AXIS
47	0.200	500	2	24	2.0	1.0	2.00	2.00	209	5040	0.50000V	-3.03088087	UNIFORM	I-AXIS
48	0.300	500	2	24	2.0	1.0	2.00	2.00	79	1920	0.65000V	-30.56252660	UNIFORM	I-AXIS
49	0.400	500	2	24	2.0	1.0	2.00	2.00	306	7368	0.60000V	-3.00075380	UNIFORM	I-AXIS
50	0.500	500	2	24	2.0	1.0	2.00	2.00	248	5976	0.55000V	-3.00968878	UNIFORM	I-AXIS
51	0.600	500	2	24	2.0	1.0	2.00	2.00	97	2352	0.60000V	-3.34890591	UNIFORM	I-AXIS
52	0.700	500	2	24	2.0	1.0	2.00	2.00	246	5928	0.45000V	-3.01530923	UNIFORM	I-AXIS
53	0.800	500	2	24	2.0	1.0	2.00	2.00	78	1896	0.60000V	-31.34561763	UNIFORM	I-AXIS
54	0.900	500	2	24	2.0	1.0	2.00	2.00	60	1464	0.65000V	-3.00000000	UNIFORM	I-AXIS
55	1.000	500	2	24	2.0	1.0	2.00	2.00	101	2448	0.80000V	-15.41042957	UNIFORM	I-AXIS
56	0.000	500	2	28	2.0	1.0	2.00	2.00	266	7476	0.50000V	-3.00178783	UNIFORM	I-AXIS
57	0.100	500	2	28	2.0	1.0	2.00	2.00	257	7224	0.05000V	-3.02198109	UNIFORM	I-AXIS
58	0.200	500	2	28	2.0	1.0	2.00	2.00	60	1708	0.65000V	-102.95525789	UNIFORM	I-AXIS
59	0.300	500	2	28	2.0	1.0	2.00	2.00	154	4340	0.60000V	-3.40567728	UNIFORM	I-AXIS
60	0.400	500	2	28	2.0	1.0	2.00	2.00	79	2240	0.65000V	-40.28712313	UNIFORM	I-AXIS
61	0.500	500	2	28	2.0	1.0	2.00	2.00	210	5908	0.55000V	-3.00034670	UNIFORM	I-AXIS
62	0.600	500	2	28	2.0	1.0	2.00	2.00	79	2240	0.65000V	-32.25789818	UNIFORM	I-AXIS
63	0.700	500	2	28	2.0	1.0	2.00	2.00	60	1708	0.65000V	-36.68532659	UNIFORM	I-AXIS
64	0.800	500	2	28	2.0	1.0	2.00	2.00	60	1708	0.65000V	-14.65270155	UNIFORM	I-AXIS
65	0.900	500	2	28	2.0	1.0	2.00	2.00	78	2212	0.60000V	-31.47896038	UNIFORM	I-AXIS
66	1.000	500	2	28	2.0	1.0	2.00	2.00	282	7924	0.35000V	-3.00151179	UNIFORM	I-AXIS
Total Function Evaluations:										180472				
54	0.900	500	2	24	2.0	1.0	2.00	2.00	60	1464	0.65000V	-3.00000000	UNIFORM	I-AXIS

243 Because CFO-PR converges so quickly on GP's global maximum, the number of the probe with the best fitness is
 244 constant after step #1 as seen in the best probe plot in Fig. 9. The best probe number (#14) is the same for steps 0 and
 245 1, but it switches to probe #2 at step 2. Neither the number of the best probe nor the fitness change after step 2. Of
 246 course, for most functions the best probe number varies throughout a run, often quite erratically.



247
 248 Fig. 7. Evolution of GP Function Best Fitness
 249



250
 251 Fig. 8. Evolution of GP Function D_{avg}

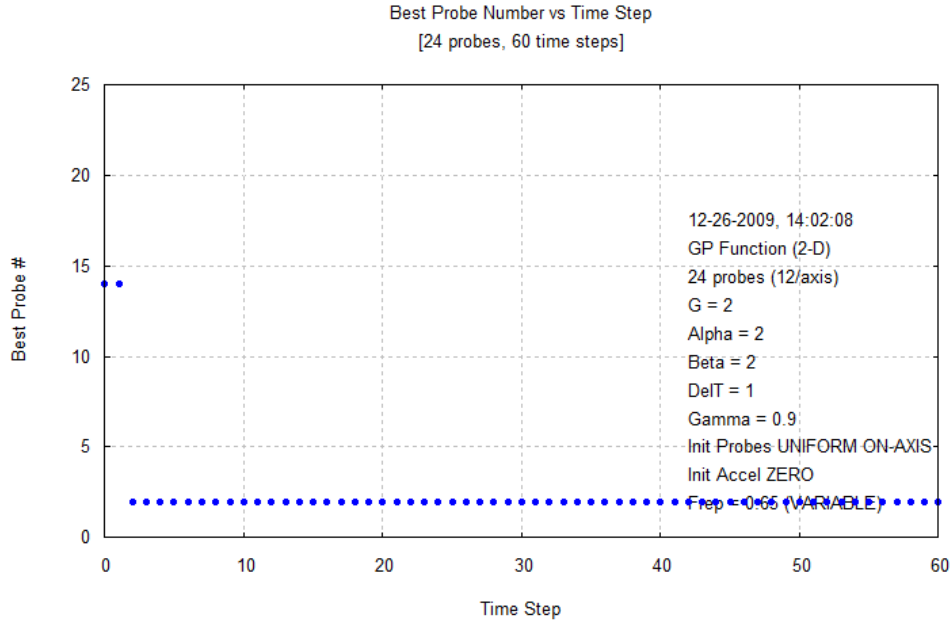


Fig. 9. GP Function Best Probe Number

Figs. 10 and 11, respectively, plot the probe trajectories for the probes with the best ten fitnesses ordered by fitness and for the first sixteen individual probes ordered by probe number [number of trajectories plotted chosen as a matter of convenience]. Both plots are very chaotic with no obvious sign of regularity in how probes gravitate to the global maximum. Nevertheless, there is some measure of regularity as reflected in the D_{avg} curve because its appearance is not nearly as chaotic as the trajectory plots. In fact, in many cases D_{avg} exhibits a mathematically precise oscillation even when the probe trajectories look like Figs. 10 and 11 (see in particular [1]).

3. RESULTS AND DISCUSSION

CFO-PR was tested against the twenty-three function benchmark suite described in [3], and the results compared to the other algorithms discussed in that paper (GSO, PSO, GA). Table 2 summarizes CFO-PR's results. f_{max} is the known global maximum (note that the negative of each benchmark in this standard suite is used because, unlike the other algorithms, CFO locates maxima, not minima). $\langle \cdot \rangle$ denotes average value. Because GSO, PSO and GA are inherently stochastic, evaluating their performance requires a statistical assessment, but CFO's results are repeatable over runs with the same parameters because it is deterministic, and consequently no statistical description is necessary.

The CFO-PR data correspond to the best fitness returned by the single best run in the set of runs with variable N_p/N_d and variable γ . For functions $f_{14} - f_{23}$ eleven runs were made with $0 \leq \gamma \leq 1$ in increments of 0.1 and $4 \leq N_p/N_d \leq 14$ by 2 (66 runs total). For $f_1 - f_{13}$ the same procedure was used, but with $2 \leq N_p/N_d \leq 6$ by 2 (33 runs total) in order to avoid excessive runtimes. Table 2 shows the γ value corresponding to the best fitness, γ_{best} , and the corresponding best value of N_p/N_d . N_{eval} is the number of function evaluations, and it is tabulated for the single best run and for the group of runs used to determine γ_{best} and the best number of probes per axis.

In the first group of high dimensionality unimodal functions, $f_1 - f_7$, CFO-PR returned the best fitness on five of the seven functions ($f_3 - f_7$). PSO performed best on the first two. In the second set of high dimensionality multimodal functions with many local maxima, $f_8 - f_{13}$, CFO-PR performed best on two (f_9, f_{10}) and essentially the same as the best other algorithm (GSO) on f_8 . In last group of ten multimodal functions with few local maxima, $f_{14} - f_{23}$, CFO-PR returned the best fitness on four ($f_{20} - f_{23}$), equal fitnesses on three (f_{14}, f_{17}, f_{18}), and very slightly lower fitnesses on the remaining three.

Even though it is in its infancy, CFO-PR performed very well against three other highly sophisticated algorithms. It returned the best, equal, essentially equal or very slightly lower fitnesses on eighteen of the twenty three test functions. It is reasonable to conclude that, overall, CFO-PR performed as well or better than GSO, which in turn performed better than PSO or GA.

2D GP TRAJECTORIES OF PROBES WITH BEST FITNESSES (ORDERED BY FITNESS)

12-26-2009, 14:02:08

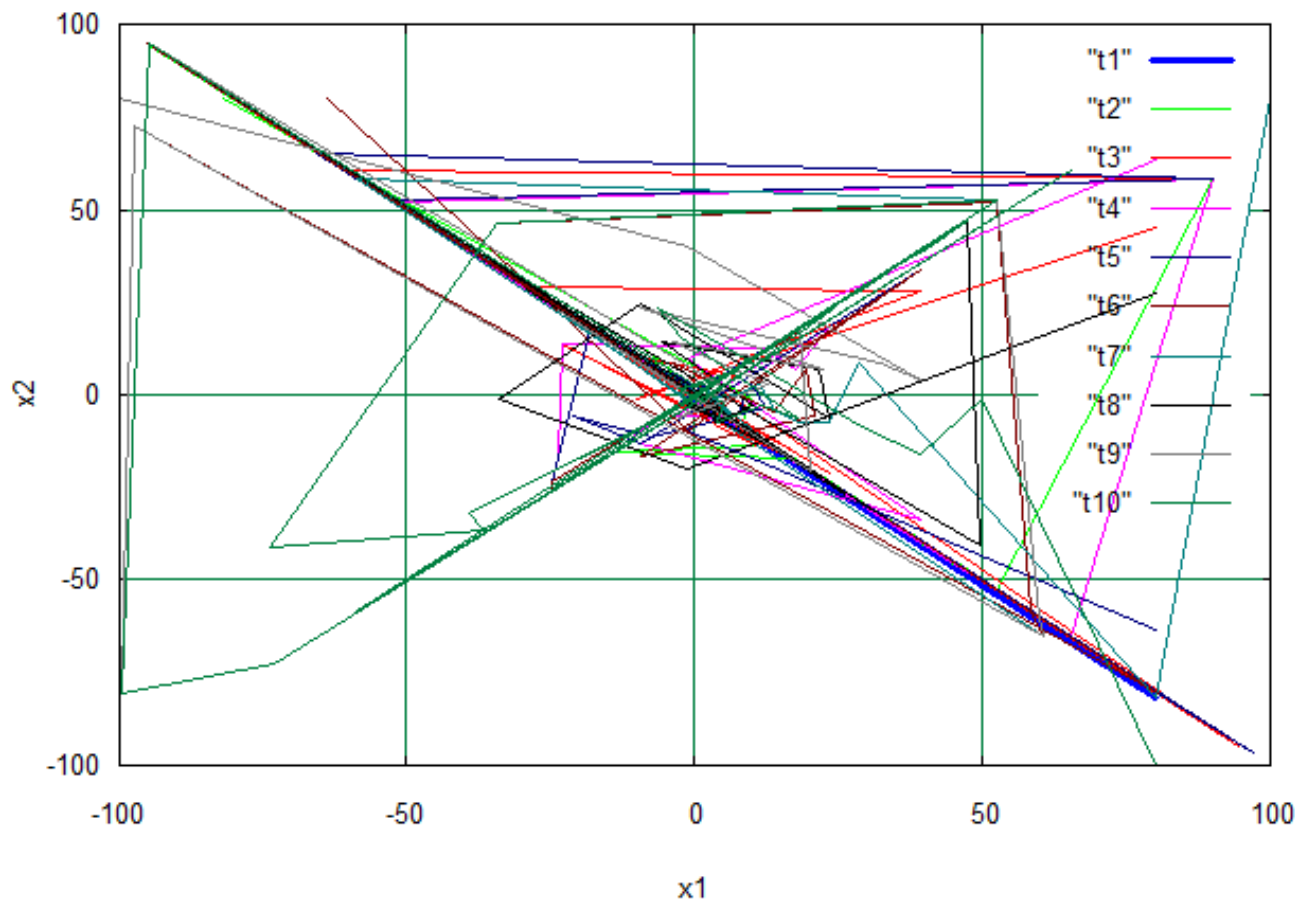


Fig. 10. GP Function Trajectories of Probes with Best Fitness

4. CONCLUSION

This note suggests that pseudorandomness is an important, indeed perhaps essential, aspect of effective CFO implementations. A pseudorandom variable has an arbitrary but precisely known value that may be assigned or calculated. Its essential characteristic is that the value is uncorrelated with the decision space's topology, so that it has the effect of distributing probes pseudorandomly throughout the landscape. While in a general sense this process may appear to be similar to the randomness in an inherently stochastic algorithm, it is in fact fundamentally different. The equations underlying stochastic algorithms are formulated in terms of true random variables whose values are computed from probability distributions and consequently unknowable until the calculation is made. Therefore successive calculations yield different values, and every optimization run has a different outcome. By contrast, a pseudorandom variable in the context of CFO is known with absolute precision because of how its value is determined (assignment or deterministic calculation). This property allows CFO to compute probe trajectories precisely because it is inherently

deterministic. Every CFO with the same setup, with or without a pseudorandom component, yields exactly the same results step-by-step throughout an entire run. Importantly, CFO's reproducibility lends itself well to "reactive" implementations in which run parameters are "tuned" in response to performance metrics such as rate of convergence or fitness saturation, for example. Reactive stochastic algorithms, on the other hand, are not easily implemented.

This note provides examples of how pseudorandomness can improve CFO's performance. Three different approaches are used (initial probe distribution, repositioning factor, and decision space adaptation), and each was discussed in detail. A sample CFO problem was presented in detail, and summary data included for a standard 23-function benchmark suite. CFO's performance is quite good compared to other highly developed, state-of-the-art algorithms. Hopefully these results will encourage further work on improved methodologies for injecting pseudorandomness into CFO. Of course, any or all of CFO's run parameters can be pseudorandomized, not only the three considered here. But even with respect to those parameters, different approaches to how they are pseudorandomized may yield better results or faster runtimes. There are many fruitful areas of research on CFO, and it is the author's hope that this and the other CFO papers will provide the foundation and catalyst for that work.

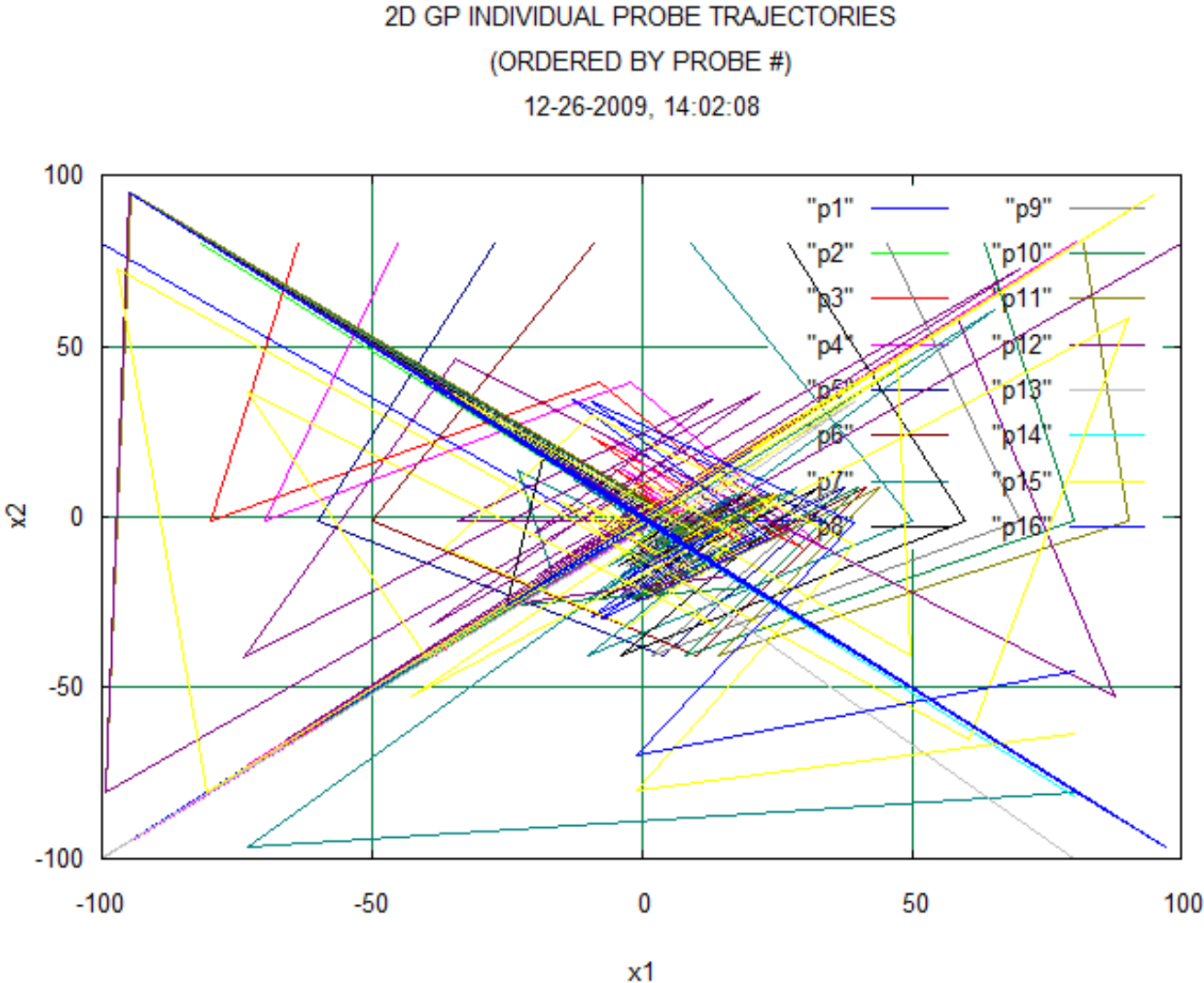


Fig. 11. GP Function Probe Trajectories by Probe Number

319

320

321

322

Table 2. CFP-PR 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 N_p/N_d	N_{eval} Best Run	N_{eval} Total
Unimodal Functions (other algorithms: average of 1000 runs)								
f_1	30	0	-3.6927x10 ⁻³⁷ / PSO	-4.8438x10 ⁻⁴	0.1	4	20,640	507,060
f_2	30	0	-2.9168x10 ⁻²⁴ / PSO	-4x10 ⁻⁸	0.5	2	5,040	716,400
f_3	30	0	-1.1979x10 ⁻³ / PSO	-6x10 ⁻⁸	0.5	2	10,260	1,534,260
f_4	30	0	-0.1078 / GSO	-4.2x10 ⁻⁷	0.5	2	5,160	332,340
f_5	30	0	-37.3582 / PSO	-1.09289x10 ⁻³	0.9	6	34,560	845,640
f_6	30	0	-1.6000x10 ⁻² / GSO	0	1.0	6	10,980	350,280
f_7	30	0	-9.9024x10 ⁻³ / PSO	-4.249x10 ⁻⁵	0.1	4	60,120	1,983,960
Multimodal Functions, Many Local Maxima (other algorithms: avg 1000 runs)								
f_8	30	12,569.5	12,569.4882 / GSO	12,569.4866	0.5	4	12,720	448,800
f_9	30	0	-0.6509 / GA	-2.05x10 ⁻⁶	0.7	4	16,440	680,640
f_{10}	30	0	-2.6548x10 ⁻⁵ / GSO	-1.5x10 ⁻⁷	0.5	2	5,100	904,980
f_{11}	30	0	-3.0792x10 ⁻² / GSO	-9.97293x10 ⁻²	0.1	6	42,660	489,060
f_{12}	30	0	-2.7648x10 ⁻¹¹ / GSO	-2.067x10 ⁻⁵	0.5	2	3,660	341,400
f_{13}	30	0	-4.6948x10 ⁻⁵ / GSO	-3.2853x10 ⁻³	0.6	6	16,920	679,620
Multimodal Functions, Few Local Maxima (other algorithms: avg 50 runs)								
f_{14}	2	-1	-0.9980 / GSO	-0.9980	0.2	12	5,952	141,076
f_{15}	4	-0.0003075	-3.7713x10 ⁻⁴ / GSO	-4.889x10 ⁻⁴	0	12	3,360	304,664
f_{16}	2	1.0316285	1.031628 / GSO	1.031626	0.4	12	6,288	124,340
f_{17}	2	-0.398	-0.3979 / GSO	-0.3979	0	8	1,872	108,340
f_{18}	2	-3	-3 / GSO	-3	0.9	12	1,464	180,472
f_{19}	3	3.86	3.8628 / GSO	3.8627	0.2	14	3,150	200,268
f_{20}	6	3.32	3.2697 / GSO	3.32173	0.3	12	18,072	730,212
f_{21}	4	10	7.5439 / PSO	10.1532	0.4	6	1,896	336,712
f_{22}	4	10	8.3553 / PSO	10.4029	0.8	6	2,208	386,176
f_{23}	4	10	8.9439 / PSO	10.5363	0.8	6	2,256	394,320

323

324

325

REFERENCES

1. Formato, RA. Central Force Optimization: A New Metaheuristic with Applications in Applied Electromagnetics. *Progress In Electromagnetics Research*. 2007;77:425–491. doi:10.2528/PIER07082403.
2. Formato, RA. Parameter-Free Deterministic Global Search with Simplified Central Force Optimization. In: Huang D.-S., Zhao Z, Bevilacqua V, and Figueroa JC, editors. *Advanced Intelligent Computing Theories and Applications (ICIC2010)*, Lecture Notes in Computer Science (LNCS 6215). Springer-Verlag Berlin Heidelberg; 2010.
4. Formato RA. Central Force Optimization with variable initial probes and adaptive decision space. *Applied Mathematics and Computation*. 2011;217(21):8866–8872. doi:10.1016/j.amc.2011.03.151.
4. Ding D, Luo X, Chen J, Wang X, Du P, Guo Y. A Convergence Proof and Parameter Analysis of Central Force Optimization Algorithm. *Journal of Convergence Information Technology (JCIT)*. 2011;6(10):16-23.
5. Ding D, Qi D, Luo X, Chen J, Wang X, Du P. Convergence analysis and performance of an extended central force optimization algorithm. *Applied Mathematics and Computation*. 2012;219(4):2246–2259
<http://dx.doi.org/10.1016/j.amc.2012.08.071>.
6. Green RC, Wang L, Alam M, Formato RA. Central Force Optimization on a GPU: A Case Study in High Performance Metaheuristics using Multiple Topologies. *The Journal of Supercomputing*. 2012;62(1):378-398.
doi:10.1007/s11227-011-0725-y.
7. Green RC, Wang L, Alam M. Evaluating the Impact of Multiple Neighborhood Topologies on Central Force Optimization.
8. Green R, Wang L, Alam M, Formato R. Central Force Optimization on a GPU: A case study in high performance metaheuristics using multiple topologies. *IEEE Congress on Evolutionary Computation*, New Orleans, Los Angeles, June 2011;550–557. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5949667.
9. Green R, Wang L, Alam M. Training neural networks using Central Force Optimization and Particle Swarm Optimization: Insights and comparisons,” *Expert Systems with Applications*, 2012;39:555–563.
<http://www.sciencedirect.com/science/article/pii/S0957417411010086>.
10. Haghighi A, Ramos HM. Detection of Leakage Freshwater and Friction Factor Calibration in Drinking Networks Using Central Force Optimization. *Water Resource Management*. 2012;26:2347–2363. doi:10.1007/s11269-012-0020-6.
11. Roa O, Amaya I, Ramírez F, Correa R. Solution of Nonlinear Circuits with the Central Force Optimization Algorithm. 978-1-4614-6/12/\$31.00 ©2012 IEEE. doi:10.1109/CWCAS.2012.6404079. 2012 IEEE 4th Colombian Workshop on Circuits and Systems (CWCAS), 1-2 Nov. 2012; pp.1-6.
12. Mohammad G, Dib N. Synthesis of Antenna Arrays using Central Force Optimization. *Mosharaka International Conference on Communications, Computers and Applications, MIC-CPE 2009*.
13. Qubati GM, Formato RA, Dib NI. Antenna Benchmark Performance and Array Synthesis using Central Force Optimisation. *IET Microwaves, Antennas & Propagation*. 2010;4(5):583–592. doi:10.1049/iet-map.2009.0147.
14. Qubati GM, Dib NI. Microstrip Patch Antenna Optimization using Modified Central Force Optimization. *Progress in Electromagnetics Research B*. 2010;21:281-298.
<http://www.jpier.org/pierb/pier.php?paper=10050511>.
15. Montaser AM, Mahmoud KR, Elmikati HA. Tri-band Slotted Bow-Tie Antenna Design for RFID Reader using Hybrid CFO-NM Algorithm. *29th National Radio Science Conference (NRSC 2012)*. April 10-12, 2012. 978-1-4673-1887-7/12/\$31.00 ©2012 IEEE.
16. Mahmoud KR. Central Force Optimization: Nelder-Mead Hybrid Algorithm for Rectangular Microstrip Antenna Design. *Electromagnetics*. 2011;31(8): 578-592. doi:10.1080/02726343.2011.621110.

17. Asi MJ, Dib NI. Design of Multilayer Microwave Broadband Absorbers using Central Force Optimization. Progress in Electromagnetics Research B. 2010;26:101-113. doi:10.2528/PIERB10090103.
18. Formato RA. Improved CFO Algorithm for Antenna Optimization. Progress In Electromagnetics Research B. 2010; 19: 405-425. doi:10.2528/PIERB09112309.
19. Montaser AM, Mahmoud KR, Abdel-Rahman AB, Elmikati HA. Design Bluetooth and Notched-UWB E-Shape Antenna using Optimization Techniques. Progress In Electromagnetics Research B. 2013;47:279-295. doi:10.2528/PIERB12101407.
20. Formato RA. New Techniques for Increasing Antenna Bandwidth with Impedance Loading. Progress In Electromagnetics Research B. 2011;29:269-288. doi:10.2528/PIERB11021904.
21. Formato RA. Improving Bandwidth of Yagi-Uda Arrays. Wireless Engineering and Technology. 2012;3(1):18-24. doi:10.4236/wet.2012.31003.

APPENDIX

CFO searches an N_d -dimensional decision space Ω for the global *maxima* of an *objective function* $f(x_1, x_2, \dots, x_{N_d})$ defined on $\Omega: x_i^{\min} \leq x_i \leq x_i^{\max}, 1 \leq i \leq N_d$. The x_i are the *decision variables*, and i the coordinate number. The term *fitness* refers to the value of $f(\vec{x})$ at point $\vec{x}$ in Ω . There is no *a priori* information about the objective function's maxima, that is, $f(\vec{x})$'s topology or "landscape" is unknown.

CFO searches Ω by flying "probes" through the space at discrete "time" steps (iterations). Each probe's location is specified by its position vector computed from two *equations of motion* that analogize their real-world counterparts for material objects moving through physical space under the influence of gravity without energy dissipation.

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

Equations of Motion. In metaphorical "CFO space" each of the N_p probes experiences an acceleration created by the "gravitational pull" of "masses" in Ω . Probe p 's acceleration at step $j-1$ is given by

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

which is the first of CFO's two equations of motion. In equation (1), $M_{j-1}^p = f(x_1^{p,j-1}, x_2^{p,j-1}, \dots, x_{N_d}^{p,j-1})$ is the objective function's fitness at probe p 's location at time step $j-1$. Each of the other probes at that step (iteration) has associated with it fitness $M_{j-1}^k, k = 1, \dots, p-1, p+1, \dots, N_p$. G is CFO's *gravitational constant*, and $U(\cdot)$ is the Unit Step function, $U(z) = \begin{cases} 1, & z \geq 0 \\ 0, & \text{otherwise} \end{cases}$.

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

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

which is CFO's second equation of motion.

The CFO equations of motion, (1) and (2), combine to compute a new probe distribution at each time step using “masses” discovered by the probe distribution at the previous step. Δt is the “time” interval between steps during which the acceleration is constant. Note that CFO’s terminology has no significance beyond reflecting CFO’s kinematic roots, as does the factor $\frac{1}{2}$ in eq. (2). The gravitational constant, G , and time increment, Δt , have direct analogues in Newton’s equations of motion for real masses moving under real gravity through three-dimensional physical space. The CFO exponents α and β , by contrast, have no analogues in Nature. They provide added flexibility to the algorithm designer who, in metaphorical “CFO space,” is free to change how “gravity” varies with distance, or mass, or both, if doing so creates a more effective algorithm.

“Mass.” The concept of “mass” in CFO space is very important and quite different than it is in real space. Mass in the physical Universe is an inherent, immutable property of matter, whereas in CFO space it is a positive-definite *user-defined function* of the objective function’s fitness [not (necessarily) the fitness itself]. For example, in equation (1) 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]. A different function can be used if it results in a better performing CFO algorithm. In this specific implementation the Unit Step is a critical element because it prevents negative mass. Without the Unit Step CFO mass could be negative depending on which fitness is greater. But mass in the real Universe always is positive, and as a consequence the force of gravity always attractive. By contrast, mass can be positive or negative in metaphorical CFO space, depending on how it is defined, and undesirable effects may result from the wrong definition. Negative mass creates a *repulsive* gravitational force that flies probes away from maxima instead of toward them, thus defeating the very purpose of the algorithm.

Errant Probes. At any iteration in a CFO run, it is possible that a probe’s acceleration computed from eq. (2) may be too great to keep it inside Ω . If any coordinate $x_i < x_i^{\min}$ or $x_i > x_i^{\max}$, the probe enters a region of *unfeasible* solutions that are not valid for the problem at hand. The question is what to do with an errant probe, and it arises in many algorithms. There are many approaches. 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” (see, for example, [2] for a more detailed discussion). Its value, as well as those of all the CFO parameters, was determined empirically.