British Journal of Applied Science & Technology
4(6): 912-928, 2014

SCIENCEDOMAIN ***international***
www.sciencedomain.org

SCIENCEDOMAIN

VSO-Optimized Dipole-Loaded Monopole

Richard A. Formato[1*]

[1]Cataldo and Fisher, LLC, P.O. Box 1714, Harwich, MA 02645, USA.

Author's contribution

This work was performed by the sole author RAF, who wrote the first draft of the manuscript, managed literature searches, and read and approved the final manuscript.

Original Research Article

Received 5th September 2013
Accepted 2nd November 2013
Published 27th December 2013

ABSTRACT

Very Simple Optimization (VSO) is introduced as a new a new global search design and optimization algorithm that is applied to a seminal antenna optimization problem: achieving uniform hemispherical coverage in a dipole-loaded monopole. The VSO-optimized monopole's performance is compared to genetic algorithm and hill-climber designs, and VSO is tested against two suites of benchmark functions and several other algorithms.

Keywords: Antenna; wire antenna; monopole; loaded antenna; electromagnetic; Very Simple Optimization; VSO; numerical optimization; optimization algorithm; metaheuristic.

1. INTRODUCTION

This paper introduces Very Simple Optimization (VSO), a new design and optimization (D&O) methodology (met heuristic) with specific application in applied electromagnetic (EM). VSO is used to optimize a dipole-loaded monopole (DLM) antenna as a practical example and also is tested against recognized suites of benchmark functions and other algorithms. VSO thus is generally applicable to any D&O problem regardless of its nature, antenna design or otherwise.

The prototype DLM was introduced by Altshuler [1], and its design subsequently refined using Genetic Algorithm (GA) optimization [2]. GA's have been applied to a variety of wire and other antenna problems including Yagi-Uda arrays, GPS antennas, electrically small

---

**Corresponding author: E-mail: rf2@ieee.org;*

self-resonant structures, dielectric-imbedded antennas, impedance-loaded devices, amorphous antennas, and microstrip patches [3-14].

VSO is introduced as a new deterministic D&O methodology that is applied to the seminal DLM problem and tested against two benchmark suites using several other algorithms. Wire antenna design generally has benefitted substantially from computer-based optimization, and VSO hopefully will become another effective tool to that end.

While GA is fundamentally stochastic, consequently producing a different antenna design every time it is run, VSO is not. Uncertain results can be a significant impediment because there is no good way of knowing why any particular antenna design is different from others. It could be the result of changing the fitness function, or, what is more likely, due to GA's (or another stochastic algorithm's) inherent randomness. The effects of different objective functions are best investigated using deterministic D&O algorithms like VSO because every run with the same setup returns the same results. VSO solves the DLM problem much more efficiently than GA and additionally performs very well against suites of recognized optimization benchmark functions optimized using a variety of other algorithms including Vibrational-PSO, Group Search Optimizer, Genetic Algorithm and Particle Swarm Optimization.

This paper is organized as follows: Section 2.1 describes VSO; Section 2.2 applies it to the DLM problem; and Section 2.3 applies VSO to two suites of benchmark functions and compares its results to those of several other algorithms. In every case VSO's performance is superior, which strongly suggests that this new D&O approach merits further investigation.

2. METHODOLOGY AND RESULTS

2.1 Very Simple Optimization (VSO)

VSO is a novel deterministic iterative D&O algorithm based on a very simple idea, and it appears to work well for a wide range of functions. This section describes the VSO algorithm. Many global search and optimization metaheuristics are drawn from Nature, prime examples being GA, Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Synthetic Annealing (SA), and Biogeography Based Optimization (BBO) as representative metaheuristics. These algorithms are inherently stochastic because they "explore" a decision space in a random manner, just as "survival of the fittest" randomly improves a species, or just as bees or ants randomly search for food, or as other natural processes seek minima, for example, energy level, or maxima, for example, a friendlier habitat. These approaches necessarily return different results on successive runs because some variables must be computed from a probability distribution and cannot be known in advance (true randomness is quite different from pseudo or quasirandomness; see, for example, [15]. VSO by contrast is entirely deterministic because it is purely geometrical in nature (although some measure of randomness can be included if the algorithm designer wishes).

The basic idea underlying VSO is that at every iteration each sample point in the decision space (DS) is moved along a straight line towards the point with the previous iteration's best fitness (which remains fixed). The best fitness can be a maximum or minimum (many algorithms perform minimization), but here VSO is used to maximize an objective function. The VSO algorithm is shown diagrammatically in Fig. 1 (O is the coordinate system origin).

DS is defined by $x_i^{\min} \le x_i \le x_i^{\max}$, $1 \le i \le N_d$, where i is the coordinate number and N_d DS's dimensionality. At VSO's j^{th} iteration the location of each point where the objective function $f(\vec{R})$ is evaluated is specified by a position vector $\vec{R}_j^p = \sum_{k=1}^{N_d} x_k^{p,j} \hat{e}_k$, where p is the sample point number, $x_k^{p,j}$ its coordinates, and $\hat{e}_k$ the unit vector along the x_k axis. The previous best fitness is $F_{j-1}^* = f(\vec{R}_{j-1}^*)$ at the point $\vec{R}_{j-1}^* = \sum_{k=1}^{N_d} x_k^{*,j-1} \hat{e}_k$, where $f(\cdot)$ is the objective function being maximized. Sample point p's position is updated according to $\vec{R}_j^p = \vec{R}_{j-1}^p + \rho(\vec{R}_{j-1}^* - \vec{R}_{j-1}^p)$ in which parameter $0 \le \rho \le 1$ determines where along the line joining $\vec{R}_{j-1}^p$ and $\vec{R}_{j-1}^*$ the sample point is relocated.

When $\rho = 0$ the sample point remains in its original position (no movement), whereas when $\rho = 1$ it is moved to the location of the previous best fitness. When $\rho = 0.5$ the sample point is moved to the midpoint of the line joining $\vec{R}_{j-1}^p$ and $\vec{R}_{j-1}^*$. The value $\rho = 0.5$ was used for all VSO runs reported here, but some other fixed value might work better (other values were not tested). It also may be advantageous to use a variable value for ρ, but this also was not tested. The question of exactly how ρ should be specified has not been investigated further because $\rho = 0.5$ provides good results by bisecting at each step the distance between the sample point's starting location and the location of the best previous fitness. This value serves well for the DLM problem and for the benchmark testing described in Section 2.3.

Fig. 2 shows VSO pseudo code. It is evident that VSO is quite simple (hence the name). Initialization begins with the specification of an Initial Sample Point Distribution (ISPD) against which the initial fitnesses are calculated to determine the best starting fitness. The ISPD can be deterministic, as it is here, or stochastic or hybrid in nature. As a general proposition, randomness increases an algorithm's ability to explore DS, which is why Nature-based stochastic metaheuristics are so popular. By comparison, VSO offers the "best of both worlds" because at any iteration full or partial randomness can be added if doing so improves performance. For the runs reported here a variant of the deterministic "probe line" distribution described in [16,17] was employed to create ISPD.

VSO's ISPD is generated by placing sample points uniformly along lines parallel to the coordinate system axes that intersect at a point along the decision space principal diagonal. The intersection point is $\vec{D} = \vec{X}_{\min} + \gamma(\vec{X}_{\max} - \vec{X}_{\min})$ 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 $0 \le \gamma \le 1$ determines where along the diagonal the orthogonal sample point array is placed. Ten γ values were used for the runs reported here uniformly spaced in the intervals $0.05 \le \gamma < 0.49$ and $0.51 \le \gamma < 0.95$. To avoid the possibility of a biased ISPD, $\gamma = 0.5$ is intentionally excluded so that no sample points are placed at the origin in a symmetrical DS, and an even number of sample points, in this case 14 per line, was used for the same reason. A typical three-dimensional ISPD using these parameter values is shown in Fig. 3 (the oblique line is the principal diagonal). VSO then evolves ISPD iteration-by-iteration according to the simple repositioning scheme described above.

A VSO run ends when a user-specified termination criterion is met, often a predetermined number of iterations or saturation of the best returned fitness. The runs reported here terminated on the earlier of 15 iterations or fitness saturation within 0.001 over 4 iterations tested every third step, that is, when $j \text{ MOD } 3 = 0$. In almost every case this early termination criterion was met in fewer than 15 iterations.

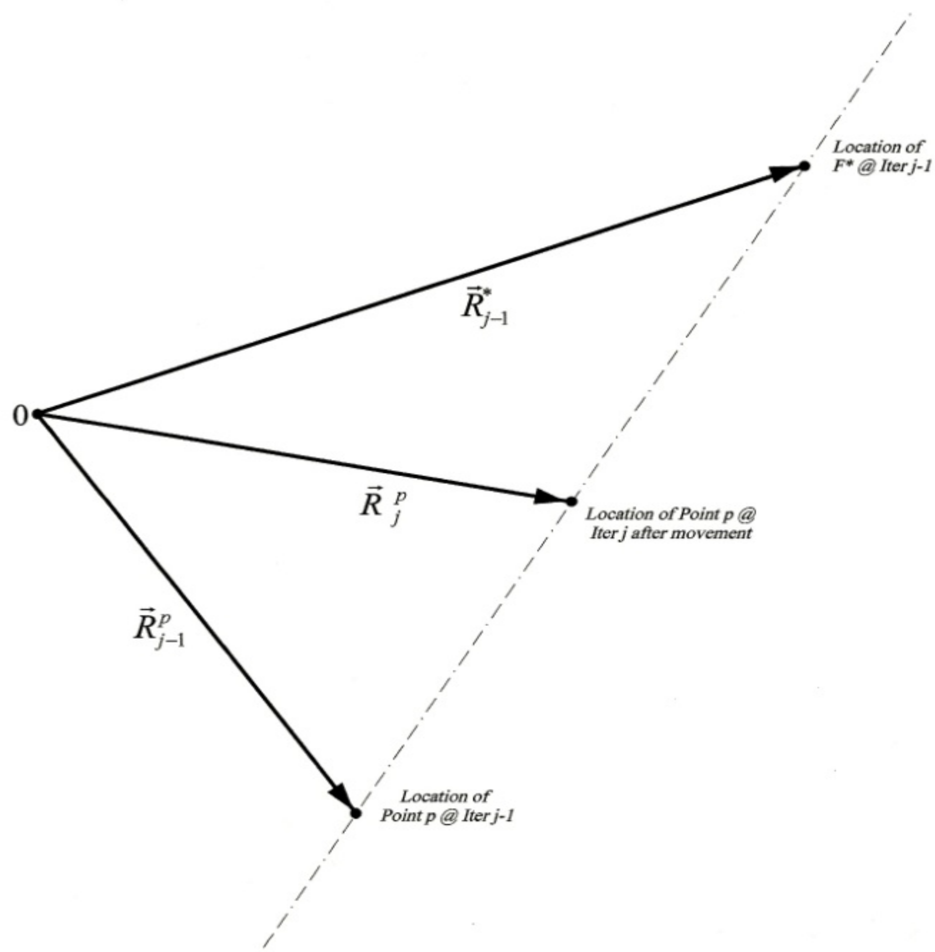


Fig. 1. VSO's Simple Sample Point Relocation Scheme

Algorithm VSO

I - Initialization:
- $j \leftarrow 0$
- (a) ISPD
- (b) Compute initial fitnesses
- (c) Best fitness: $F_0^* = f(\vec{R}_0^*)$

II - Do Until (Termination Criterion)
- $j \leftarrow j+1$
- (a) Reposition sample points: $\vec{R}_j^p = \vec{R}_{j-1}^p + \rho(\vec{R}_{j-1}^* - \vec{R}_{j-1}^p)$
- (b) Compute fitnesses using current sample point distribution
- (c) Best fitness: $F_j^* = f(\vec{R}_j^*)$

Fig. 2. VSO Pseudocode

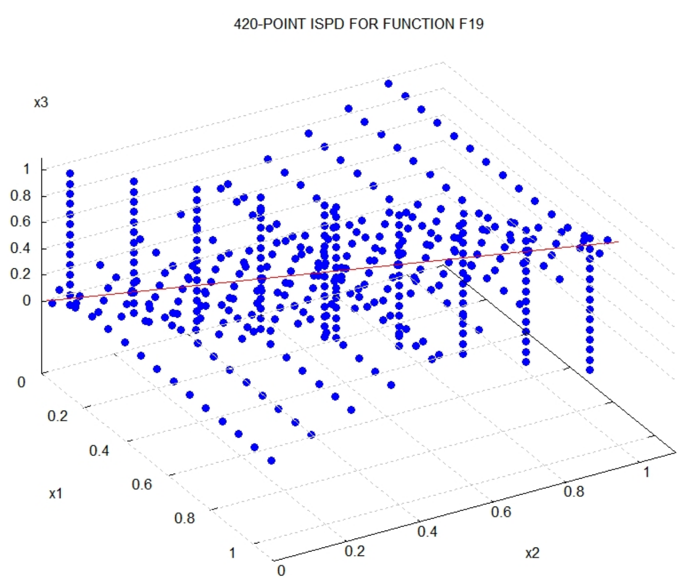


Fig. 3. Typical VSO Three Dimensional ISPD

2.2 VSO Applied to the Dipole-Loaded Monopole

The VSO-optimized DLM is shown schematically in Fig. 4. It comprises eight straight wire segments (the long wire parallel to the X-axis consists of two separate wires along the ±X axes). Additional geometry details may be found in [2, Fig. 1]. All wires are perfectly electrically conducting (PEC), and the antenna is fed at its base against an infinite PEC ground in the plane Z=0 (X-Y plane). This seminal antenna optimization problem was proposed by Altshuler [1,2] and solved using GA. The same problem is solved here using VSO with novel results that are better than GA's.

VSO is reminiscent of the stochastic Hill Climbing algorithm [private communication between Dr. Robert C. Green II, CS Dept., Bowling Green State Univ., and the author] and consequently merits comparison to that approach as well as to GA. The loaded monopole therefore also was optimized using SAHC, a hill climbing variant described in §2.3 below. Perspective views of the VSO, GA, and SAHC-optimized antennas appear in Figs. 5(a-c), respectively (axis length 0.1 meter, frequency 299.8 MHz). While the VSO antenna is quite different from the GA and SAHC designs, whose geometries are similar, the radiation patterns of all three monopoles are quite similar as discussed below.

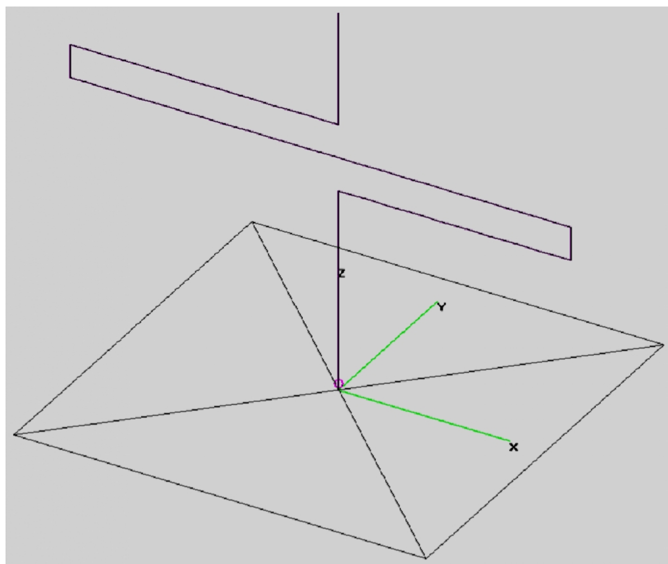


Fig. 4. Geometry of VSO-optimized base-fed DLM

The VSO/SAHC-optimized monopoles' performance was computed using the Numerical Electromagnetics Code version 4.2 double precision (NEC-4.2D) [18] as the optimizer's modeling engine. An earlier version of NEC was used in [2], and details of the GA setup can be found there. NEC is widely regarded as the "gold standard" in wire antenna modeling. Based on the Method of Moments, its development began with the groundbreaking work of Richmond and Mei in 1965 and continues to this day at Lawrence Livermore National Laboratory (USA). Wires comprising the antenna are broken into small segments, and the current amplitude and phase computed segment-by-segment subject to Maxwell's equations with appropriate boundary conditions. NEC's performance has been validated against other analytical and numerical models and against extensive experiments over many years with excellent results.

For the DLM problem considered here, Altshuler defines the design objective as a uniform hemispherical radiation pattern without regard to input impedance. The resulting NEC input files for the three optimized monopoles appear in Figs. 6(a-c). The VSO/SAHC fitness function (to be maximized) was $f(G)=\frac{1}{\sum(G(\theta,\phi)-G_{avg})}$ where $G(\theta,\phi)$ is the power gain as a function of the angles (θ,ϕ) in standard right-handed spherical polar coordinates. G_{avg} is the average gain over all calculation angles. Gain was computed at 5° increments in θ and 45° in ϕ. This fitness function minimizes the difference between the actual and average gains at each calculation point, which has the effect of smoothing the pattern.

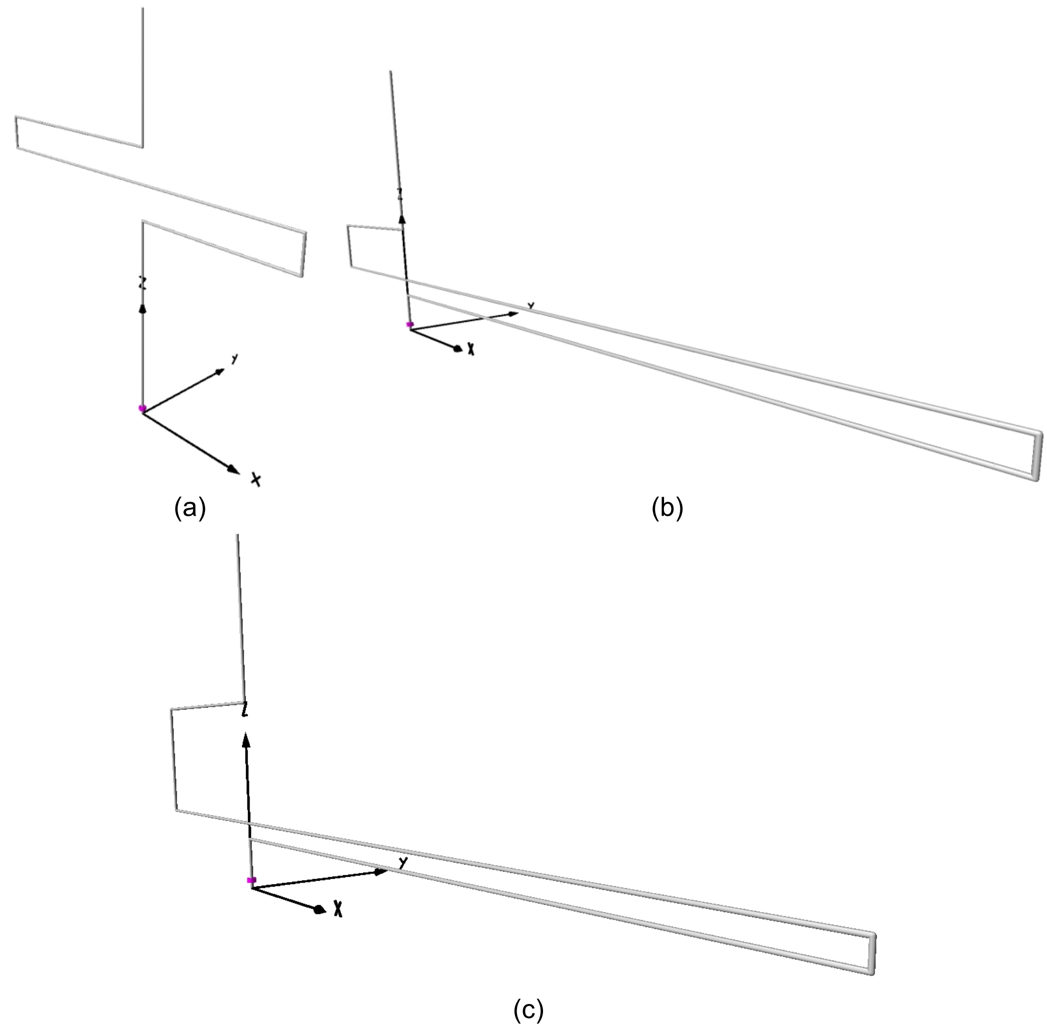


Fig. 5. VSO (a), GA(b), and SAHC(c) Optimized DLM (0.1 m axis length, 299.8 MHz)

The antenna designer must specify a suitable objective (fitness) function, but doing so can be problematic [19]. What appears to be a perfectly reasonable function for achieving a desired balance between various antenna performance measures in fact may be quite poor. Because stochastic D&O algorithms return different results on every run, there is no good way to test the effects of changing the fitness function, for example, by changing coefficient values or by combining parameters differently. These issues are best addressed by using a deterministic D&O methodology such as VSO. In addition, unlike benchmark testing that uses purely analytical formulations, antenna design usually requires a stand-alone numerical modeling engine, in this case NEC, that frequently imposes long computation times making it even more difficult to evaluate the quality of a specific objective function. Thus, in implementing the VSO DLM optimization algorithm, NEC is used as a self-contained stand-alone program shelled by VSO. NEC's data then are returned to VSO which computes each candidate DLM design's fitness. Thus, VSO performs the optimization while NEC computes the antenna's performance.

The VSO/SAHC loaded monopole designs are compared to the GA-optimized monopole by using NEC-4.2D to compute the GA antenna's performance with the geometry data in Fig. 6(b) (taken from Table 1 in [2]. Following standard procedure, these geometry data are in wavelengths (λ) at the operating frequency f_0 because $f_0 = 299.8$ MHz at which $\lambda = 1$ meter precisely, thus permitting the antenna to be scaled to any other operating frequency.

Figs. 7(a-c), respectively, show the VSO, GA, and SAHC-optimized radiation patterns (total power gain) in an azimuth plane containing the maximum gain, while Fig. 8 provides three-dimensional (3-D) perspective views. All three antennas meet the performance objective of providing very uniform coverage of the upper hemisphere. The VSO monopole's gain ranges from a minimum of 2.22 dBi to a maximum of 3.58 dBi (1.36 dB spread), while GA's is between 2.12 and 4.89 dBi (2.77 dB spread), and the corresponding SAHC values are 2.73 to 3.46 dBi (0.73 dB spread). The VSO monopole's gain is 1.41 dB more uniform than the GA's, but less uniform than the SAHC's by 0.68 dB. VSO required far fewer function evaluations than SAHC (8,600 vs. 183,120) because it is deterministic. Even though it was not included as a design objective, NEC outputs antenna input impedance, with the VSO, GA and SAHC values, respectively, being $38.4 + j159.4\Omega$, $298.4 + j755.3\Omega$, and $70.2 + j215\Omega$.

```
CM File: MONO.NEC
CM VSO-OPTIMIZED DIPOLE-LOADED MONOPOLE
CM (see Altshuler, 1997)
CM Frequency, Fc = 299.8 MHz
CM Run ID: 06272013080333
CM Fitness: 1/Sum|Gtot(Th,Ph)-<G>|
CM G=Pwr Gain(dBi); Th,Ph=angles
CM Note: all dimensions are in METERS.
CM File ID 06272013080916
CM Nd= 6, p= 42, j= 9
CE
GW1,16,0,0,0,0,0,.1625,.0005
GW2,14,0,0,.1625,.1351,0,.1625,.0005
GW3,3,.1351,0,.1625,.1351,0,.1895,.0005
GW4,14,.1351,0,.1895,0,0,.1895,.0005
GW5,16,0,0,.1895,-.1565,0,.1895,.0005
GW6,3,-.1565,0,.1895,-.1565,0,.2165,.0005
GW7,16,-.1565,0,.2165,0,0,.2165,.0005
GW8,9,0,0,.2165,0,0,.3082,.0005
GE1
GN1
FR0,1,0,0,299.8,0
EX0,1,1,1,1.,0.
RP0,19,8,1001,0.,0.,5,45,100000.
EN
```

(a)

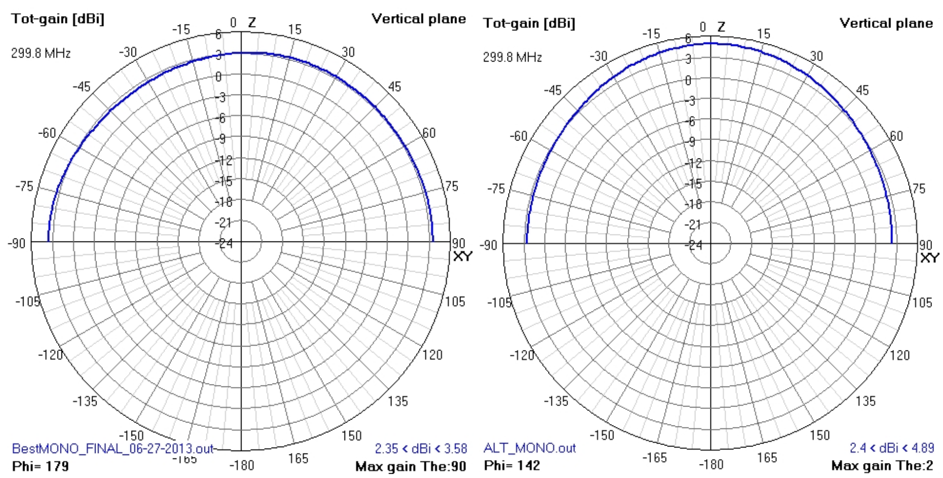
```
CM ALTSHULER'S DIPOLE-LOADED MONOPOLE
CM Ref: Altshuler, 1997
CM Frequency, Fc = 299.8 MHz
CM Note: all dimensions are in METERS.
CE
GW1,3,0,0,0,0,0,.0299,.0005
GW2,46,0,0,.0299,.4565,0,.0299,.0005
GW3,1,.4565,0,.0299,.4565,0,.0427,.0005
GW4,46,.4565,0,.0427,0,0,.0427,.0005
GW5,15,0,0,.0427,-.1515,0,.0427,.0005
GW6,4,-.1515,0,.0427,-.1515,0,.0848,.0005
GW7,15,-.1515,0,.0848,0,0,.0848,.0005
GW8,13,0,0,.0848,0,0,.2107,.0005
GE1
GN1
FR0,1,0,0,299.8,0
EX0,1,1,1,1.,0.
RP0,19,8,1001,0.,0.,5,45,100000.
EN
```

(b)

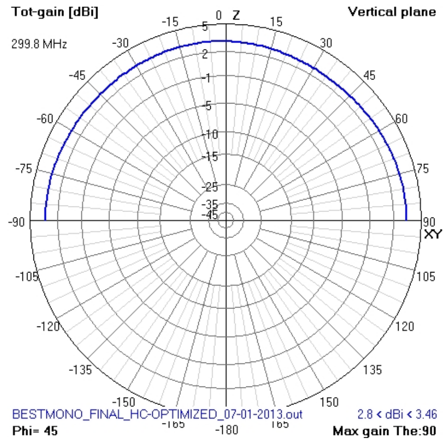
```
CM File: MONO.NEC
CM SAHC-OPTIMIZED DIPOLE-LOADED MONOPOLE
CM (see Altshuler, 1997)
CM Frequency, Fc = 299.8 MHz
CM Run ID: 07012013170847
CM Fitness: 1/Sum|Gtot(Th,Ph)-<G>|
CM G=Pwr Gain(dBi); Th,Ph=angles
CM Note: All dimensions are in METERS.
CM File ID 07012013173527
CM Nd= 6, p= 577, j= 1
CE
GW1,3,0,0,0,0,0,.0318,.0005
GW2,40,0,0,.0318,.4048,0,.0318,.0005
GW3,1,.4048,0,.0318,.4048,0,.0418,.0005
GW4,40,.4048,0,.0418,0,0,.0418,.0005
GW5,14,0,0,.0418,-.1368,0,.0418,.0005
GW6,8,-.1368,0,.0418,-.1368,0,.1179,.0005
GW7,14,-.1368,0,.1179,0,0,.1179,.0005
GW8,10,0,0,.1179,0,0,.2198,.0005
GE1
GN1
FR 0,1,0,0,299.8,0
EX 0,1,1,1,1.,0.
RP0,19,8,1001,0.,0.,5,45,100000.
EN
```

(c)

Fig. 6. Optimized DLM NEC Input Files: VSO(a), GA(b), SAHC(c)

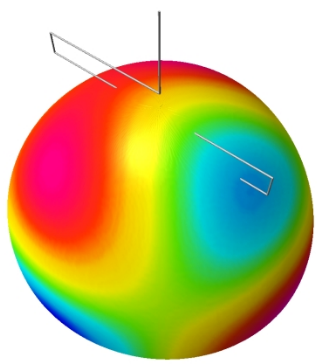


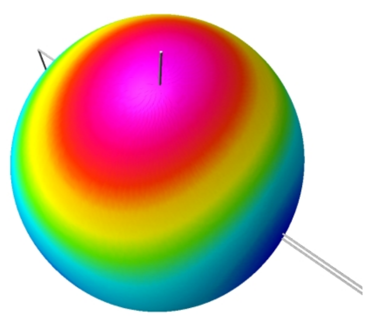
(a) (b)



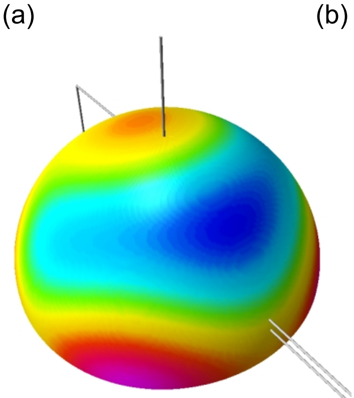
(c)

Fig. 7. Optimized DLM Radiation Patterns: VSO(a), GA(b), SAHC(c)





(a) (b)



(c)

Fig. 8. Optimized DLM 3-D Patterns: VSO(a), GA(b), SAHC(c)

2.3 VSO Applied to Benchmark Functions

Global search and optimization algorithms typically are tested against standard suites of benchmark functions. Two sets of benchmarks were used to test VSO. The six-function vibrational-PSO (v-PSO) suite [20] was selected because it provides a direct comparison to v-PSO, which is a new state-of-the-art D&O methodology tested against both benchmark functions and practical engineering problems in the same way VSO has been tested. Table 1 shows the form of the test functions, DS, and the location and value of the known maximum (same notation as [20]). Note that the signs have been changed because VSO performs maximization whereas v-PSO performs minimization, and that the form of f_1 in [20] is missing the last two terms and DS is slightly smaller (see f_{10} in [21]).

Because v-PSO is stochastic, it was tested statistically in [20] by making 100 runs and averaging the results. The algorithm was applied to each benchmark in 10, 20 and 30 dimensions using a total of 200,000 function evaluations per run (10,000 generations, 20 particle swarm), thus resulting in a total of 20,000,000 evaluations for each of the six benchmarks. Results for v-PSO are reported in Table IV in [20] and reproduced here in Table 2, along with VSO's results for the same cases.

VSO out-performed v-PSO in every case, both in terms of the quality of its solution and by far in the number of function evaluations, N_{eval}. Compared to 20,000,000 evaluations per function for v-PSO, VSO required a worst case maximum of only 67,200 for the 30-dimensional Schwefel and returned much better results. VSO's determinism greatly reduces the number of function evaluations, but at the same time VSO may not explore DS as well as a stochastic algorithm. It is evident that how VSO's ISPD is configured is an important consideration, perhaps the most important single factor in setting up effective VSO runs, an issue that requires further investigation.

In addition to testing against the v-PSO benchmarks, VSO also was tested against a recognized suite of twenty three benchmarks. Those results are summarized in Table 3. This suite's description appears in detail in [16,21,22] and consequently is not repeated here (note that there is some overlap with the v-PSO suite). In Table 3, VSO is compared directly to Central Force Optimization (CFO) and to Group Search Optimizer (GSO), which is based on a metaphor of animal foraging [22], and indirectly to two other GA and PSO variants. CFO was selected because, like VSO, it is deterministic. GSO was selected because it has been extensively tested and was itself compared to two other algorithms, GA and PSO, as described in [22].

The test functions in Table 3 are numbered as they are in the GSO paper. $f_{\max}$ is the known global maximum (note that the negative of each function is used because, unlike the other algorithms, CFO, like VSO, searches for maxima instead of minima). The $<\cdot>$ brackets denote mean value because all the algorithms discussed in [22] are stochastic, thus requiring a statistical assessment. The data in Table 3 for the other algorithms are reproduced from [22]. As noted, the high dimensionality results are averages over 1,000 runs, whereas the lower dimensionality data are 50 run averages. The GSO experimental setup is described in detail in [22]. By contrast, because both CFO and VSO are deterministic, their results are repeatable over runs with the same parameters so that only a single run is required. The same VSO setup described above for the v-PSO suite was used for the 23-benchmark suite.

The results in Table 3 speak for themselves. VSO returned the best or equal fitness on 12 of the benchmarks, which is a very robust performance in view of the relatively small number of function evaluations (maximum of 67,200 in only two cases). In every other case, VSO's best fitness was close to the actual maximum, except for benchmark f_{14} where its returned fitness was poor compared to the known maximum. It is likely that VSO would perform considerably better against f_{14} using a different ISPD, which again highlights ISPD's importance.

A version of Steepest Ascent Hill Climbing with Replacement (SAHC) [23] was implemented using essentially the same setup parameters as VSO. The number of sample points was equalized by increasing the number per dimension from 14 to 140 because VSO utilized 10 gamma values in its ISPD, whereas gamma is not a parameter in SAHC. Its ISPD was computed as $\vec{R}_0^p = \sum_{k=1}^{N_d} [x_k^{\min} + r_0 (x_k^{\max} - x_k^{\min})]\hat{e}_k$, $p = 1,...,N_p$, where $0.05 \le r_0 < 0.95$ is a uniformly distributed random variable (RV). At each step the sample points' locations were tweaked according to $\vec{R}_j^p = \sum_{k=1}^{N_d} (x_k^{p,j-1} + r_j L_{diag})\hat{e}_k$ where $-0.1 \le r_j < 0.1$ is a uniformly distributed RV and $L_{diag} = \sqrt{\sum_{k=1}^{N_d} (x_j^{\max} - x_j^{\min})^2}$ the length of DS's principal diagonal. Each tweaked coordinate is constrained to remain inside DS by setting it to $x_i^{\max}$ or $x_i^{\min}$ if it is greater or less than the boundary value, respectively. The same VSO early termination criterion was applied. A total of 1,000 independent SAHC runs was made, each with a new random ISPD, and the best fitness over all runs returned.

Results comparing VSO and SAHC using the GSO benchmark suite appear in Table 4. VSO returned the best fitness against all but five benchmarks, and those were all low-dimensionality functions. Perhaps the most important data in Table 4 are the number of function evaluations. VSO never required more than 67,200 calculations, whereas SAHC required no fewer than 1,120,840, which is a staggering difference. On the 30-D functions the difference is even greater, with SAHC requiring at least 16,850,400 function evaluations and yielding solutions that generally were quite poor compared to VSO. It is evident that VSO markedly outperforms SAHC for real-world problems like antenna design where the number of modeling engine runs can be a significant impediment in formulating an effective objective function.

Table 1. v-PSO benchmark functions

f	Function	f(x)	DS	x*	f(x*)
f_1	Ackley	$20\exp\left(-0.2\sqrt{\frac{1}{N_d}\sum_{i=1}^{N_d}x_i^2}\right)+\exp\left(\frac{1}{N_d}\sum_{i=1}^{N_d}\cos(2\pi x_i)\right)-20-e$	$[-30,30]^{N_d}$	$[0]^{N_d}$	0
f_2	Cosine Mixture	$-\sum_{i=1}^{N_d}x_i^2+0.1\sum_{i=1}^{N_d}\cos(5\pi x_i)$	$[-1,1]^{N_d}$	$[0]^{N_d}$	$0.1N_d$
f_3	Exponential	$\exp(-0.5\sum_{i=1}^{N_d}x_i^2)$	$[-1,1]^{N_d}$	$[0]^{N_d}$	1
f_4	Griewank	$-\frac{1}{4000}\sum_{i=1}^{N_d}(x_i-100)^2+\prod_{i=1}^{N_d}\cos\left(\frac{x_i-100}{\sqrt{i}}\right)-1$	$[-600,600]^{N_d}$	$[0]^{N_d}$	0
f_5	Rastrigin	$-\sum_{i=1}^{N_d}[x_i^2-10\cos(2\pi x_i)+10]$	$[-5.12,5.12]^{N_d}$	$[0]^{N_d}$	0
f_6	Schwefel	$-418.9829\,N_d+\sum_{i=1}^{N_d}[x_i\sin(\sqrt{\lvert x_i\rvert})]$	$[-500,500]^{N_d}$	$[420.9687]^{N_d}$	0

Table 2. VSO results for v-PSO benchmarks

f	N_d	v-PSO*	VSO	Neval
f_1	10	–1.84e-15±2.9e-16	4.77e-18	9,800
	20	–2.84e-15±1.5e-16	4.77e-18	19,600
	30	–4.93e-15±3.4e-16	4.77e-18	29,400
f_2	10	1±0	1	9,800
	20	2±0	2	19,600
	30	3±0	3	29,400
f_3	10	1±0	1	9,800
	20	1±3e-18	1	19,600
	30	1±1e-17	1	29,400
f_4	10	–0.020±0.006	0	9,800
	20	–0.0026±0.002	0	19,600
	30	–8.8568e-4±0.001	0	29,400
f_5	10	0±0	0	9,800
	20	0±0	0	19,600
	30	–5.6843e-16±1e-15	0	29,400
f_6	10	–620.8131±50.4	–1.305e-4	18,200
	20	–1.3384e+3±68.5	–2.551e-4	44,800
	30	–2.1395e+3±103.3	–3.827e-4	67,200

** average best fitness; data reproduced from Table IV in [20]*

Table 3. Comparative results for GSO 23-function benchmark suite

f	N_d	$f_{max}^{(1)}$	<Best Fitness>/ Other Alg.	CFO		VSO	
				Best Fitness	N_{eval}	**Best Fitness**	N_{eval}
Unimodal Functions (*other algorithms:* average of 1000 runs)							
f_1	30	0	$-3.6927x10^{-37}$ / PSO	$-2.6592x10^{-2}$	108,660	$-1.2037x10^{-35}$	29,400
f_2	30	0	$-2.9168x10^{-24}$ / PSO	$-4x10^{-8}$	161,640	$-4.3368x10^{-19}$	29,400
f_3	30	0	$-1.1979x10^{-3}$ / PSO	$-6x10^{-8}$	239,340	0	29,400
f_4	30	0	-0.1078 / GSO	$-4.2x10^{-7}$	59,160	$-3.4694x10^{-18}$	29,400
f_5	30	0	-37.3582 / PSO	$-2.1719x10^{-2}$	164,160	$-1.3154x10^{-5}$	54,600
f_6	30	0	$-1.6000x10^{-2}$ / GSO	0	73,620	0	29,600
f_7	30	0	$-9.9024x10^{-3}$ / PSO	$-3.5599x10^{-3}$	66,660	$-7.7487x10^{-4}$	29,400
Multimodal Functions, Many Local Maxima (*other algorithms:* average of 1000 runs)							
f_8	30	12,569.5	12,569.4882 / GSO	12,569.4852	69,720	12,569.4866	67,200
f_9	30	0	-0.6509 / GA	$-3.52x10^{-6}$	117,120	0	29,400
f_{10}	30	0	$-2.6548x10^{-5}$ / GSO	$-1.5x10^{-7}$	111,660	$4.7705x10^{-18}$	29,400
f_{11}	30	0	$-3.0792x10^{-2}$ / GSO	-2.00124	160,680	$-8.2269x10^{-2}$	67,200
f_{12}	30	0	$-2.7648x10^{-11}$ / GSO	-0.105859	68,220	$-5.0111x10^{-6}$	29,400
f_{13}	30	0	$-4.6948x10^{-5}$ / GSO	$-6.5966x10^{-2}$	103,320	$-3.2007x10^{-6}$	42,000
Multimodal Functions, Few Local Maxima (*other algorithms:* average of 50 runs)							
f_{14}	2	-1	-0.9980 / GSO	-1.005284	12,824	-6.9034	2,800
f_{15}	4	$-3.075x10^{-3}$	$-3.7713x10^{-4}$ / GSO	$-2.4163x10^{-3}$	19,920	$-1.6333x10^{-3}$	3,920
f_{16}	2	1.0316285	1.031628 / GSO	1.031607	9,256	1.0316239	2,800
f_{17}	2	-0.398	-0.3979 / GSO	-0.398	8,824	-0.3979	2,800
f_{18}	2	-3	-3 / GSO	-3	15,784	-3	2,800
f_{19}	3	3.86	3.8628 / GSO	3.86157	12,612	3.774	4,200
f_{20}	6	3.32	3.2697 / GSO	3.31976	52,128	3.0333	10,920
f_{21}	4	10	7.5439 / PSO	10.1466	25,376	10.1532	7,280
f_{22}	4	10	8.3553 / PSO	10.4028	29,168	10.4029	7,280
f_{23}	4	10	8.9439 / PSO	10.5362	24,784	10.5364	7,280

(1) Negative of the functions in [22] computed by CFO/VSO because they search for maxima instead of minima.

Table 4. Comparative results SAHC & VSO

f	N_d	$f_{\max}^{(1)}$	SAHC		VSO	
			Best Fitness (1,000 runs)	N_{eval}	***Best Fitness***	N_{eval}
Unimodal Functions						
f_1	30	0	-3.1296×10^{4}	16,850,400	-1.2037×10^{-35}	29,400
f_2	30	0	-118.1927	16,863,000	-4.3368×10^{-19}	29,400
f_3	30	0	-2.9379×10^{4}	16,900,800	0	29,400
f_4	30	0	-56.9567	16,888,200	-3.4694×10^{-18}	29,400
f_5	30	0	-1.9033×10^{14}	16,938,600	-1.3154×10^{-5}	54,600
f_6	30	0	-2.6567×10^{4}	16,888,200	0	29,600
f_7	30	0	-23.7855	16,900,800	-7.7487×10^{-4}	29,400
Multimodal Functions, Many Local Maxima						
f_8	30	12,569.5	5,462.47	16,913,400	12,569.4866	67,200
f_9	30	0	-4,340.43	16,850,400	0	29,400
f_{10}	30	0	-18.8522	16,850,400	4.7705×10^{-18}	29,400
f_{11}	30	0	-263.789	16,913,400	-8.2269×10^{-2}	67,200
f_{12}	30	0	-5.0287×10^{7}	16,913,400	-5.0111×10^{-6}	29,400
f_{13}	30	0	-1.2672×10^{8}	16,888,200	-3.2007×10^{-6}	42,000
Multimodal Functions, Few Local Maxima						
f_{14}	2	-1	-0.998	1,126,720	-6.9034	2,800
f_{15}	4	-0.0003075	-9.3807×10^{-4}	2,246,720	-1.6333×10^{-3}	3,920
f_{16}	2	1.0316285	1.0316071	1,125,880	1.0316239	2,800
f_{17}	2	-0.398	-0.397986	1,120,840	-0.3979	2,800
f_{18}	2	-3	-3.00144	1,125,880	-3	2,800
f_{19}	3	3.86	3.8625	1,690,080	3.774	4,200
f_{20}	6	3.32	3.2756	3,372,600	3.0333	10,920
f_{21}	4	10	8.9088	2,248,400	10.1532	7,280
f_{22}	4	10	9.2653	2,250,080	10.4029	7,280
f_{23}	4	10	9.5548	2,245,040	10.5364	7,280

3. CONCLUSION

This paper applies VSO to the optimization of a dipole-loaded monopole antenna with very good results. It describes VSO as a new easily implemented deterministic, iterative D&O algorithm, and reports test data using two benchmark suites that compare it to other stochastic algorithms, again with very good results. VSO holds promise as an effective D&O methodology, especially for problems requiring the formulation of a suitable objective function like antenna D&O, and consequently merits further study, especially with respect to how ISPD should be specified. Wire antenna design has improved dramatically using optimization tools such as GA, and VSO holds promise to further extend this capability.

COMPETING INTERESTS

The author has declared that no competing interests exist.

REFERENCES

1. Altshuler EE. A Monopole Antenna Loaded with a Modified Folded Dipole. IEEE Trans. Ant. Prop. 1993;41(7):871-76.

2. Altshuler EE, Linden DS. Wire-Antenna Designs Using Genetic Algorithms. IEEE Ant. Prop. Mag. 1997;39(2):33-43.
3. Altshuler EE, Linden DS, Process for the Design of Antennas Using Genetic Algorithms. United States Patent No. 5,719,794, Feb. 17, 1998.
4. Herscovici NM, Osorio F, Peixeiro C. Miniaturization of Rectangular Microstrip Patches Using Genetic Algorithms. IEEE Ant. Wireless Prop. Lett. 2002;1:94-97.
5. Altshuler EE, Linden DS. Wire-Antenna Designs Using Genetic Algorithms. IEEE Ant. Prop. Mag. 1997;39(2):33-43.
6. Altshuler EE, Linden DS. Design of a Vehicular Antenna for GPS/Iridium Using a Genetic Algorithm. IEEE Ant. Prop. Soc. Intl. Sym. 1997. IEEE Digest 1997;3:1680-1683.
7. Griffiths LA, Furse C, Chung YV. Broadband and Multiband Antenna Design Using the Genetic Algorithm to Create Amorphous Shapes Using Ellipses. IEEE Trans. Ant. Prop. 2006:54(10):2776-2782.
8. Jayasinghe, JW, Anguera J, Uduwawala DN. A High-directivity Microstrip Patch Antenna Design by Using Genetic Algorithm Optimization. Prog. Elec. Res. 2013;37:131-144.
9. Altshuler EE. Design of a Vehicular Antenna for GPS/Iridium Using a Genetic Algorithm. IEEE Trans. Ant. Prop. 2000;48(6):968-972.
10. Altshuler EE. Electrically Small Self-resonant Wire Antennas Optimized Using a Genetic Algorithm. IEEE Trans. Ant. Prop. 2002;50(3):297-300.
11. Jayasinghe JW, Anguera J, Uduwawala DN. Genetic Algorithm Optimization of a High-Directivity Microstrip Patch Antenna Having a Rectangular Profile. Radio engineering. 2013;22(3):700-707.
12. Altshuler EE. Electrically Small Genetic Antennas Immersed in a Dielectric. IEEE Ant. Prop. Soc. Intl. Sym. 2004;3:2317-2320.
13. Altshuler EE, Linden DS. An Ultrawide-band Impedance-Loaded Genetic Antenna. IEEE Trans. Ant. Prop. 2004;52(11):3147-3150.
14. Altshuler EE, Best SR, O'Donnell TH, Herscovici N. An Electrically-small Multi-frequency Genetic Antenna Immersed in a Dielectric Powder," IEEE International Workshop on Antenna Technology 2009, iWAT. 2009;1-3.
15. Hayes B. Quasirandom Ramblings. Amer. Scientist. 2011;99:282-87.
16. Formato RA. Central Force Optimization with Variable Initial Probes and Adaptive Decision Space. App. Math. Comp. 2011;217:8866-72.
17. Formato RA. Improved CFO Algorithm for Antenna Optimization. Prog. Electro. Res. B. 2010;19:405-25.
18. (i) Burke GJ. Numerical Electromagnetics Code - NEC-4.2 Method of Moments Part I: User's Manual. U.S. Dept. of Energy Lawrence Livermore National Laboratory, LLNL-SM-490875, July 15, 2011. (ii) Burke GJ, Miller EK, Poggio AJ. The Numerical Electromagnetics Code (NEC)–A Brief History. 2004 IEEE AP-S Intl. Sym. USNC/URSI Natl. Radio Sci. Monterey, California, June 20-25, 2004.
19. Formato RA. Issues in Antenna Optimization - A Monopole Case Study. App. Comp. Electro. Soc. J. (in press).
20. Pehlivanoglu YV. A New Particle Swarm Optimization Method Enhanced with a Periodic Mutation Strategy and Neural Networks. IEEE Trans. Evol. Comp. 2013:17(3);436-52.
21. Yao X, Liu Y, Lin G. Evolutionary Programming Made Faster. IEEE Trans. Evol. Comp. 1999;3(2):82-102.

22. He S, Wu QH, Saunders JR. Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior. IEEE Trans. Evol. Comp. 2009;13(5):973-90.
23. Luke S. Essentials of Metaheuristics, 2nd ed. (Algorithm 6). Lulu, June 2013, (free online) available: http://cs.gmu.edu/~sean/book/metaheuristics/.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sciencedomain.org/review-history.php?iid=366&id=5&aid=2844