AIAA 98-4791

Bell-Curve Based Evolutionary Optimization Algorithm

J. Sobieszczanski-Sobieski, K. Laba, and R. Kincaid
NASA Langley
Hampton, VA

7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization
September 2-4, 1998 / St. Louis, MI

BELL-CURVE BASED EVOLUTIONARY OPTIMIZATION ALGORITHM

Jaroslaw Sobieszczanski-Sobieski, Keith Laba†, and Rex Kincaid‡*

Abstract

The paper presents an optimization algorithm that falls in the category of genetic, or evolutionary algorithms. While the bit exchange is the basis of most of the Genetic Algorithms (GA) in research and applications in America, some alternatives, also in the category of evolutionary algorithms, but use a direct, geometrical approach have gained popularity in Europe and Asia. The Bell-Curve Based Evolutionary Algorithm (BCB) is in this alternative category and is distinguished by the use of a combination of n-dimensional geometry and the normal distribution, the bell-curve, in the generation of the offspring. The tool for creating a child is a geometrical construct comprising a line connecting two parents and a weighted point on that line. The point that defines the child deviates from the weighted point in two directions: parallel and orthogonal to the connecting line, the deviation in each direction obeying a probabilistic distribution. Tests showed satisfactory performance of BCB. The principal advantage of BCB is its controllability via the normal distribution parameters and the geometrical construct variables.

0. Introduction

Vigorous research and increasing applications in optimization by Genetic Algorithms (GA) continue because of the GA capability to handle discrete problems, multiple minima, and multiple objectives, e.g., Baeck, 1997. That list of attractors has recently been extended to comprise the GA intrinsic concurrent processing capability. This capability is closely aligned with the present trend in computer technology toward multiprocessor machines and machine clusters. In addition to its utilitarian values, the GA has been seen as an elegant transposition of the biological mechanism of "the survival of the fittest" onto the grounds of engineering.

While the original, biology-inspired GA that employs crossover and random mutations, and operates on 0-1 bits in binary strings as analogs of genes and chromosomes, remains the mainstay of research and applications in America, an alternative approach, also in the category of evolutionary optimization, has been gaining popularity in Europe and Asia. This approach calls for changing design variables by direct applications of the rules of probability without operating on binary strings.

Once the confines of the biology emulation are discarded, the ways one can generate successive generations is limited only by the imagination. This has resulted in a gamut of techniques varying in ingenuity and complexity. An example of a very simple mechanism is found in Grill and Hartmann, 1998. It alters design variables by adding to each an increment whose magnitude is governed by the familiar "bell-curve" probability distribution and generates a child from a single parent (asexual reproduction). A contrasting example of a more complex technique is the algorithm proposed in Ono and Kobayashi, 1997. Their technique connects two parents by a line in the design space and places the child away from that line at a distance controlled by the location of a third parent.

An advantage that motivated the development of techniques such as the two above examples is, primarily, an improved controllability of the entire optimization process. In addition, premature convergence caused by a loss of diversity in the successive generations (population stagnation) that often occurs in the original, biology-inspired GA is remedied. Another motivation is the natural human desire to improve on the devices originally inspired by the natural world.

In this paper, we report on a technique in the same category as that of Ono and Kobayashi, 1997. It also uses two parents connected by a line in the design space as a basis for generation of children. However, instead of using a third parent, it relies throughout on the familiar, "bell-curve" probability distribution. A variant of the technique selects one parent by the objective function criterion, and the other by the degree of satisfaction of the constraints. Testing in structural applications demonstrates the technique's effectiveness in locating constrained minima in a rather complex design

* Manager, Computational AeroSciences and Multidisciplinary Research Coordinator NASA Langley Research Center, MS 139 Hampton VA 23681, AIAA Fellow
† Research Assistant, College of William and Mary
‡ Professor, College of William and Mary

space, and its ease of controllability, achieved without sacrificing the valuable concept of inheritance from two-parents (the concept abandoned in the asexual reproduction) .

1. Notation

B - point where orthogonal subspace intersects L
C - child of P1 & P2
D - distance between P1, and P2 on L
f_1 , f_2 - fitness measures of parents P1 and P2
F(X) - objective function.
F_p - F with a penalty term added for a constraint violation
<g> - vector of NGV violated constraints
g(X) - vector of NG inequality constraint functions
L - line from Parent 1 through Parent 2, from - to + infinity
M - point on L, the origin of coordinate s
N(m, σ) - normal distribution with mean m and variance σ; a code to generate this may be found in Pritsker 1986.
NP - number on individuals in a generation
NX - number of the design variables, length of X.
p - penalty factor
P1, P2 - pair of parent points
r - coefficient used to define σ in the N-distribution of R
R - radius of hypersphere in NX-1 subspace, centered on point B
s - coordinate originating at M, positive toward P2.
W - structural weight (or volume)
X - vector of NX design variables.
X1, X2, X_B, X_C, and X_M - X vectors associated with P1, P2, B, C, and M.
σ - standard deviation

2. The Bell Curve-Based (BCB) Algorithm

The BCB algorithm solves the optimization problem

1)	Find	X
	Minimize	F(X)
	Satisfy	g(X) <= 0

where X, F(X) and g(X) may be continuous or discrete. In simplest terms, the BCB algorithm creates a child from two parents who are selected out of NP individuals in the current generation in the same way as in the original GA. The selection fitness criterion penalizes constraint violations by adding a penalty term to F

2) $$F_p = F + p \Sigma \max(<g>);$$

In the BCB variant (BCB/V), Parent 1 is selected by F and Parent 2 by max(<g>).

The crux of the BCB Algorithm is the way a child is created from parents by the following geometrical construct. The paired parents, P1 and P2 in Figure 1, are connected by line L in the NX-dimensional design space. They are a distance D apart. Point M is placed on L between the parents at a location that deviates from the mid-point toward the parent of better fitness. Coordinate s, originating at M and pointed toward Parent 2, is used to locate point B on L by a random probability distribution, the bell-curve from which the algorithm takes its name:

3) $$s_B = N(0, \sigma)$$

where N(0, σ) is a normally distributed random number with mean 0 and standard deviation σ given in units of D.

Now, we define a subspace orthogonal to L and intersecting L at B. The orthogonality reduces the subspace dimensions to NX-1. In that subspace we create a hypersphere of radius R whose length is again governed by the positive side of the normal distribution in units of (r D)

4) $$R = |N(0, \sigma)|;$$

Finally, point C representing the child is placed on the surface of the hypersphere using a uniform, not normal, distribution over that surface. Owing to the subspace orthogonality to L, the chance of the child falling back on either parent is exceedingly small.

Figure 1 shows a three-dimensional design space, in which the hypersphere reduces to a circle of radius R, perpendicular to L at B, and point C may fall anywhere on the perimeter of that circle with equal probability.

In the remainder of BCB one may again use any of the standard GA techniques. In the study reported herein, one repeats the parent selection and the child generation process until the number of children reaches NP. The children are added to the parents resulting in a population that swells to 2 NP. Selection based upon the application of the F_p fitness criterion to all the individuals, young and old alike, reduces the population back to NP, and the process starts over - see Appendix for the BCB & BCB/V step-by-step recipe and further details.

The way BCB algorithm generates children contrasts with that of the original GA (O/GA) as shown in Figure 2 for a case of integer variables in a 2D space. When integer variables are represented as a binary string, there is a limited number of ways these strings may be mixed by the O/GA crossover mechanism. Figure 2a illustrates the consequences, demonstrating the clustering of the O/GA children toward the parents. Intriguingly, this clustering seems to reflect the folk wisdom that among siblings, one child "takes after the mother" and the other "takes after the father", rather than both reflecting features of the parents evenly mixed. Note that the children from random mutations (each mutation changes one 0 to 1 or vice versa) also fall in that clustering pattern. It has also been observed, e.g., Ono and Kobayashi, 1997, that the degree of feature inheritance is affected by the angles formed by L and the coordinate axis. Furthermore, that inheritance also depends on the choice of the numerical system, binary or decimal.

Figs. 2b and c depict the BCB children for a small and a large σ. Comparing to Figure 2a, their distribution is more centered on the line connecting the parents, so that most of the children inherit, mix, and average the features of the parents (within a restriction imposed by integer nature of the variables). In this regard, the BCB improves on both O/GA, and on the asexual reproduction that abandons the two-parent inheritance. Observe, also, that the normal distribution allows some of the children to fall beyond the parents as measured along the coordinate axis and along L. The tails of the bell-curve enable them to fall anywhere within the design space limits so that additional random mutations may not be necessary to maintain the population diversity. Also, it is apparent that owing to the use of the local variable s and the normal distribution, the locations of the children do not depend on the orientation of L and on the choice of the numerical system.

To recapitulate this narrative description of the BCB algorithm, the algorithm may be controlled by:

• lower bound on the inter-parent distance D to keep the parents far enough apart to foster diversity
• standard deviation σ; two different σs are used to control the location of B and the length of R (equations 3 and 4).
• penalty coefficient p
• choosing between BCB and BCB/V

In addition, the usual GA controls apply to the parent selection and formation of the next generation from the parents and their children.

3. Testing and Results

One of the examples is depicted in Figure 3. It is a hub structure which appears also in Balling and Sobieszczanski-Sobieski, 1994. Each member of the hub is an I-beam rigidly attached to the hub and to the wall. The structure is optimized for minimum weight (equivalent to volume for the homogeneous material used), so that the smaller the fitness the better the structure.

The beam cross-sectional dimensions are the design variables, and the constraint functions reflect the material allowable stress and the overall and local buckling. Additional constraints are also imposed on the hub displacements. The top and bottom flanges of the I-beam are not of the same dimensions, hence the cross-section of each I-beam requires 6 design variables. Details are given in the Appendix, including the constraint function formulations; they may also be found in Padula, et. al., 1997.

Utility of the hub structure as an optimization test case stems from its ability to be enlarged by adding as many members as desired without increasing the dimensionality of the load-deflection equations. These remain 3x3 equations for a 2D hub structure regardless of the number of members. While analytically simple, the hub structure design space is complex because the stress, displacement, and buckling constraints are rich in nonlinearities and couplings among the design variables.

The results included herein are for two versions of the hub structure: Version 1 comprising 2 members, and loaded by 2 loading cases; Version 2 is made up of 6 members, and loaded by 3 loading cases. In both versions the members have the same type of I-cross-section defined by 6 design variables.

To validate the GA results and to provide a comparison, a few other methods were exercised:

• Method 2: a crossover method where the child's values are taken directly from one parent or the other. A basic mutation is applied that adds or subtracts a uniform random variable on the range of zero to 1/10 of the range of allowable values for that particular variable.

• Method 3: a crossover method where the child's values are calculated as a direct average of the two corresponding parent values. The mutation is the same as in method 2.

• Method 4: a usable-feasible search algorithm (Vanderplaats 1973)

• Method 5: random generation of 10000 solutions to show that the genetic algorithms produce results better than a random examination of the solution space.

BCB and methods 2-5 were executed for Version 1; Version 2 was optimized using BCB, methods 4 & 5, and by BCB/V.

Table 1 and 2 define the structure for Versions 1 and 2 and show the corresponding results. Note that Version 2 in Table 2 has one member much shorter than others to create a case whose solution is known in advance. That is, all members other than the short one should shrink to minimum gage dimensions. Both Method 4 and BCB confirmed that. Table 2.1 shows results for Version 2 with varied member length. Comparison of the BCB and BCB/V performances is presented in Table 3.

To test the BCB ability to locate the global minimum, a single member hub structure test was devised, shown in Figure 4. In that test the load is limited to F_x only and the member is optimized by adding to the 6 cross-sectional design variables the orientation angle relative to the loading force, as the 7th variable. Zero angle corresponds to pure compression. The global minimum is expected at the angle 180 degrees in pure tension, the second minimum is known to occur at 0 degrees in compression, separated by the maximum weight at about 90 degrees in bending. The convergence of BCB on the optimal angle is demonstrated in Table 4.

The number of individuals in a population was 10 in all the tests reported herein. This relatively small number was found experimentally to be sufficient in these tests.

A typical BCB histogram is in Figure 5. The dotted line represents the best objective (structural weight) fitness in the generation, and the solid line depicts the penalty function F_p. The places where the two lines do not overlap indicate designs where the most fit member is infeasible. In this case, the fitness value is based on the structural weight plus the penalty term, resulting in the discrepancy in the two values. It is apparent that BCB selected feasible designs from the initial population and maintained that feasibility throughout. To do so, BCB had to be supplied with a judiciously chosen value of p.

Controllability of BCB is demonstrated in Figure 6. It shows how the value of σ (eq. 3) that governs the location of B makes the process converge faster. On the other hand, a larger value of σ enables one to "cast a wider net" for more diverse designs at the penalty of converging slower. The coefficient r that governs R (ie. σ in eq.4) has a similar effect as illustrated in Figure 7.

Examination of a large number of designs generated in the BCB process revealed that, typically, a group of feasible and near optimal designs results from the process.

4. Conclusions

An algorithm was developed in the class of evolutionary optimization for creating design populations that improve in a series of successive generations. Distinguishing features of the Bell-Curve Based (BCB) algorithm are the use of a new geometrical construct in the design space, combined with the probability calculus tools of the uniform and normal distributions (bell-curve) for deriving children designs from the parent designs that are paired by a fitness criterion of the objective function penalized for any violation of the constraints. This mechanism replaces the classical, biology-inspired, bit-exchange (crossover) as a means by which to pass the parent features to the children. In a variant of the algorithm (BCB/V), one parent is qualified by the value of the objective, the other by the value of the maximum violated constraint.

BCB and BCB/V have been tested on a number of cases and both performed as intended in all of them. A sample test reported herein was a minimum weight (volume) design of a redundant structure with up to 36 cross-section variables. To create a case of multiple local minima, the member orientation angle was also a design variable in one of the tests. The strength, local buckling, and displacement limitations generated 36 highly nonlinear constraints. In these tests the algorithm was able to do as well or better than a benchmark NLP search algorithm in terms of the objective function value and the ability to eliminate constraint violations. Regarding the number of function evaluations it was on par with other GAs and, as expected, not competitive with gradient-guided NLP techniques. However, superior to the NLP techniques, the algorithm is not hampered by discrete variables, and its capability to find global minima in presence of local minima was demonstrated in the test.

The principal motivation for development of BCB was to have the adaptability and flexibility needed to prevent premature convergence that occurs in many GA processes because of the loss of population diversity. Indeed, owing to the intrinsic controllability of the normal distribution by the mean and variance parameters, combined with the malleability of the geometrical construct, BCB has exhibited the desired adaptability and flexibility in the tests. Another merit of BCB is its ability to identify a number of feasible and near-optimal solutions, as opposed to a point solution typical for an NLP technique.

Future work is aimed at furthering BCB's adaptability to the peculiarities of the design space topology, incorporation of the gradient data wherever available, and on the elapsed time reduction by the use of concurrent processing for simultaneous generation of individual designs.

5. References

Baeck, T., (ed.): Proceedings of the 7th International Conference on Genetic Algorithms; Michigan State University, East Lansing, MI, July 19-23, 1997; Morgan Kaufmann Publ., San Francisco, 1997.

Balling, R.J.; and Sobieszczanski-Sobieski, J.: An Algorithm for Solving the System-Level Problem in Multilevel Optimization; ICASE Report No. 94-96, NASA Contractor Report 195015; NASA Langley Research Center; Institute for Computer Applications in Science and Engineering, Hampton VA 23681; December 1994.

Grill, H.; and Hartmann, D.: Mixed-Discrete Structural Optimization with Distributed Advanced Evolution Strategies. Proceedings of the Australasian Conference on Structural Optimisation, Feb. 11-13, 1998, Sydney, Australia; Oxbridge Press Publ. Highpoint, Victoria, Australia; pp. 79-86.

Ono, I; and Kobayashi, S.: A Real-coded GA for Function Optimization Using Unimodal Normal Distribution Cross-Over. Proceedings of the 7th International Conference on Genetic Algorithms; Michigan State University, East Lansing, MI, July 19-23, 1997; Morgan Kaufmann Publ., San Francisco, 1997. Baeck, T., (ed.). pp. 246-253.

Vanderplaats, G.N.: CONMIN - A FORTRAN Program for Constrained Function Minimization; NASA TM X-62282, August 1973.

Pritsker, A. Alan B.: Introduction to Simulation and SLAM II
Halsted Press, John Wiley & Sons, New York 1986

Knuth Donald E.: The Art of Computer Programming; Addison-Wesley Reading, Massachusetts 1969

Padula, S.L., Alexandrov, N., and Green, L.L..: MDO Test Suite at NASA Langley Research Center. Proceedings of the 6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization; Bellvue, WA, September 4-6, 1996; American Institute of Aeronautics and Astronautics, 1996. pp. 410-420. Accessible as a website:
http://fmad-www.larc.nasa.gov/mdob/

Appendix

1) Step-by-step recipe and mathematical details;
2) Orthogonal hypersphere; and
3) Test Case details

1) The algorithm's step-by-step recipe is as follows (**BCB/V** notes the only steps where **BCB/V** differs):

1) Generate a population of designs by any technique commonly used in a conventional GA.

2) Analyze each design for the value of the objective function and constraints, eq. 2. For each design generate a single "measure of fitness" combining the value of the objective (the smaller the better) and of the constraints (negative = satisfied, zero = active (critical), positive = violated); thus, a smaller fitness reflects a better design.

In BCB/V, place individuals from N in two populations of N each, rank one by F and the other by max(g).

3) Pair-up the designs to form parents for mating, rewarding fitness (as measured by F_p) with more chances to mate. The effect is that individuals that are more fit participate in more parent pairs, and end up contributing their characteristics to more children. BCB uses the electronic roulette to do this, as in the conventional GA.

In BCB/V, one parent is selected from the population ranked by F, the other from the one ranked by max(<g>).

4) Generate a child:

- Consider a design space in n-dimensions. Figure 1 illustrates a 3D example. Design points P1 and P2 are the parents, assignment of the P1 and P2 labels is arbitrary. The hyperline L connects the parents. The prefix "hyper" reminds us that we are dealing with n-dimensional space of which 3D simplification is depicted in Figure 1. Line L extends beyond P1 and P2 to infinity, emphatically, it is not merely a (P1,P2) segment. P1 and P2 define vectors X1 and X2. The distance D between P1 and P2 is

$$D = \|(X1^2 + X2^2)^{1/2}\|.$$

- Coordinates X_M of point M on L between P1 and P2 are

$$k = f_2/(f_1 + f_2)$$

$$X_M = X2 + (X1 - X2)\,k$$

(recall, a smaller fitness is better, thus as f_2 improves relative to f_1, k will decrease, moving point M closer to X2 and vice versa)

- The variate, N_B, is generated relative to X_M

$$N_B = N(0, \sigma);$$

Or, equivalently, relative to X2

$$N_B = N(X_M, \sigma)$$

- Assume σ is in the units of D and place point B at a distance from M measured by s_B from eq.3

$$X_B = X_M + (X2 - X1)\, N(0, \sigma)/D;$$

or use the equivalent formula

$$X_B = X2 + (X1 - X2)\, N(X_M, \sigma)/D;$$

- Generate an N-dimensional hypersphere with radius R that is orthogonal to L at B, and place child C on the surface of the above hypersphere, using a uniform probability distribution over that surface, i.e., each point on the surface has the same chance of selection (details in Appendix, Section 2)

- X_C defines the child design point. If any element of X_C exceeds its bounds it is reset to that bound

5) Repeat #3 to 5 to produce the entire offspring generation

6) Group together the newly generated children population and the previous population from which their parents were drawn (the grouped population comprises 2 NP individuals) and select the most fit individuals. These individuals represent the next generation. The size of the next generation is kept at NP by discarding the least fit half in each successive grouped population. Then, repeat from #2 until termination criteria are met.

2) Child C Placement on Hypersphere that is Orthogonal to L and has N-distributed R

1) Shift the coordinate system X origin to B.

2) Rotate X so that X_1 coincides with L pointing from P1 to P2.

3) Apply algorithm by Knuth 1969:

- Set and maintain $X_1 = 0$
- Let $X_2, X_3, \ldots X_{n-1}$ be normal variates distributed as N(0,1).
- Let $b = (X_2^2 + X_3^2 + \ldots + X_{n-1}^2)^{1/2}$
- Generate radius R distributed as the positive half of N(0, (r D)) for assumed C
- Generate point C whose coordinates are

$$X_C = \{0, (R^*X_2)/b, (R * X_2)/b, \ldots, (R * X_{n-1})/b\}$$

3) Test Case Details

The solution method is the standard, linear, displacement-based finite element method, using the beam element in a slender beam formulation that neglect the transverse shear stress deformations and preserves the Kirchoff's assumption of the beam cross-sections remaining planar under the load. The structure is two-dimensional for analysis purposes, with the exception of out-of-plane member buckling. The following defines the constraints.

Displacement Constraints at the hub:

$$d/d_a - 1 \leq 0 \qquad q/q_a - 1 \leq 0$$

d = resultant translational displacement
q = rotational displacement

	q_a	d_a
two-member hub:	0.2 cm	1.0 rad
six-member hub:	0.2 cm	1.0 rad

Stress Constraints:

Normal and shear stresses (s and t) were evaluated within the cross section at the top and bottom extreme fibers, at the centroid, and at the top and bottom of the web. This was done at both ends of the member except for the centroidal stresses which are constant along the length of the member. The following stress constraint was imposed at each location:

$$s_{eq}/s_a - 1 \leq 0$$

s_{eq} = von Mises-Huber equivalent stress = $(s^2 + 3t^2)^{1/2}$

s_a = allowable stress = 25 kN/cm^2

In-Plane Buckling Constraint for each beam:

$N/N_{cr} - 1 \leq 0$

N = axial force (compression positive)
$N_{cr} = 2.05\Pi^2 EI_{xx}/L^2$
E = modulus of elasticity = 20,000 kN/cm^2
I_{xx} = strong axis moment of inertia
L = member length

Out-of-Plane / Lateral-Torsional Buckling Constraint (at each end):

$(N/N_{cr}) + (M/M_{cr})^{1.75} - 1 \leq 0$

N = axial force (compression positive)
M = magnitude of bending moment
$N_{cr} = 2.05\Pi^2 \ EI_{yy}/L2$
$M_{cr} = \Pi(EIyy/GI_{zz})1/2/L$
E = modulus of elasticity = 20,000 kN/cm^2
G = shear modulus of elasticity = E/2(1+v)
v = Poisson's ratio = 0.3
I_{yy} = weak axis moment of inertia
I_{zz} = torsional moment of inertia = $b_1t_13+b_2t_2^3$
$+(h- t_1-t_2)b_3^3$

L = member length

Local Flange and Web Buckling Constraints (at each end):
$s/s_{cr} + (t/t_{cr})^2 - 1 \leq 0$; where
s = normal stress (compression positive)
t = shear stress

The above stresses were evaluated at the following points:

	t	s
flanges:	mid-flange	extreme fiber
web:	centroid	mid-web

and constrained by the following critical stress values:

	t_{cr}	s_{cr}
Top Range:	$0.55E(2t_1/b_1)^2$	$0.41E(2t_1/b_1)^2$
Bottom Range:	$0.55E(2t_2/b_2)^2$	$0.41E(2t_2/b_2)^2$
Web:	$4.80E(b_3/(h-t_1-t_2))^2$	$3.60E(b_3/(h-t_1-t_2))^2$

E = modulus of elasticity = 20,000 kN/cm^2

Bounds on Section Variables:

	lower	**uppper**
b_1:	2.0 cm	6.0 cm
t_1:	0.1 cm	1.0 cm
b_2:	2.0 cm	6.0 cm
t_2:	0.1 cm	1.0 cm

	lower	**upper**
b_3:	0.1 cm	1.0 cm
h:	3.0 cm	8.0 cm
A:	0.68 cm^2	10.00 cm^2
I:	1.00 cm^2	100.0 cm^2

Table 1: Hub Structure Version 1

Member Data

	Length	**Angle**
Member 1	200.0	293.867
Member 2	200.0	337.050

Load Case Data

	Fx	**Fy**	**Mz**
Load 1	10.6727	17.2130	12.6227
Load 2	44.8890	18.8343	12.5973

Solution Comparison

	BCB	**Method 2**	**Method 3**	**Method 4**	**Method 5**
Best Obj. Value	604.265	586.777	592.245	670.9	755.008
Feasible?	Yes	Yes	Yes	Yes	Yes
Time (Seconds)	115	190	187	1	67
# Solutions Before Best Solution Found	7000	30000	88000	186	39195
Total Solutions Examined	50000	100000	100000	386	100000

Table 2: Hub Structure Version 2

Member Data

	Length	**Angle**
Member 1	120.00	320.771
Member 2	120.00	92.717
Member 3	120.00	335.389
Member 4	12.00	99.809
Member 5	120.00	257.694
Member 6	120.00	174.027

Load Case Data

	Fx	**Fy**	**Mz**
Load 1	26.219	17.516	14.269
Load 2	20.611	25.009	15.613
Load 3	13.667	19.360	16.835

Solution Comparison

	BCB	CONMIN	Random
Best Obj. Value	461.5	474.2	2276.7
Feasible?	YES	YES	YES
Time (Seconds)	71	1	21
# Solutions Examined	5000	280	10000

Table 2.1 Hub Structure Version 2.1

Solution Comparison

	BCB	CONMIN	Random
Best Obj. Value	532.8	535.9	2843.3
Feasible?	YES	YES	YES
Time (Seconds)	194	1	22
Total Solutions Examined	10000	90	10000

Table 3: Performance comparison of BCB and BCB/V

BCB:

Number of Individuals	Best Solution Found	Generation Best Found
10	461.3	367
50	468.7	446
100	465.2	476
500	462.0	380

BCB/V

Number of Individuals	Best Solution Found	Generation Best Found
10	463.8	500
50	464.0	498
100	463.1	309
500	462.4	466

Table 4: Convergence of α on Π (3.14159)

Generation Number	Mean α	Maximum α	Minimum α
0	2.73911	4.93799	.007982
50	3.10717	3.11264	3.10122
100	3.11468	3.13170	3.10776
500	3.14148	3.14218	3.14057
1000	3.14159	3.14235	3.14057

Angle α stated in radians

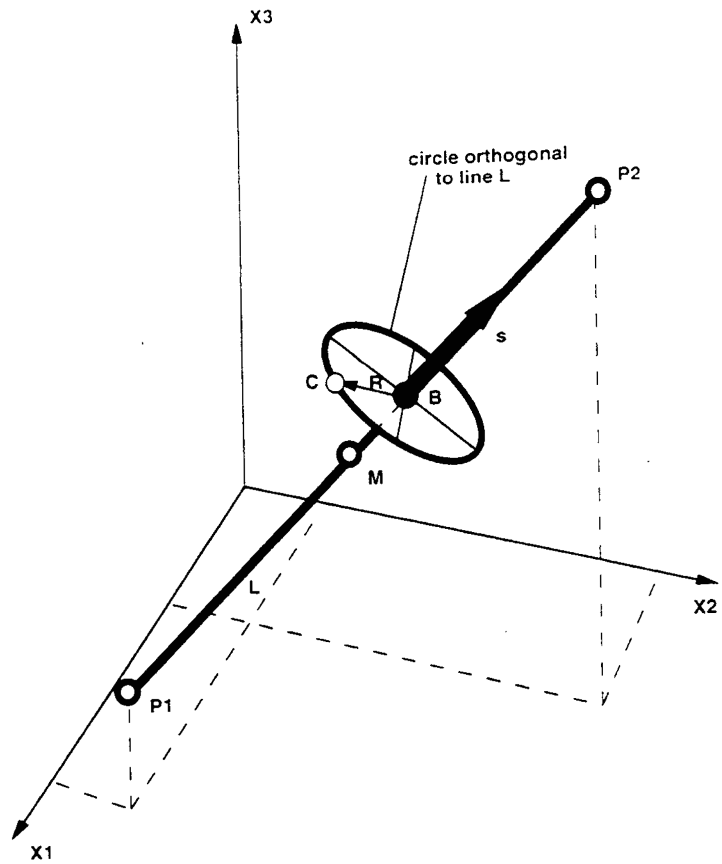

Figure 1. BCB Geometrical Construct in 3D Space: P1,P2 - parents; C - child

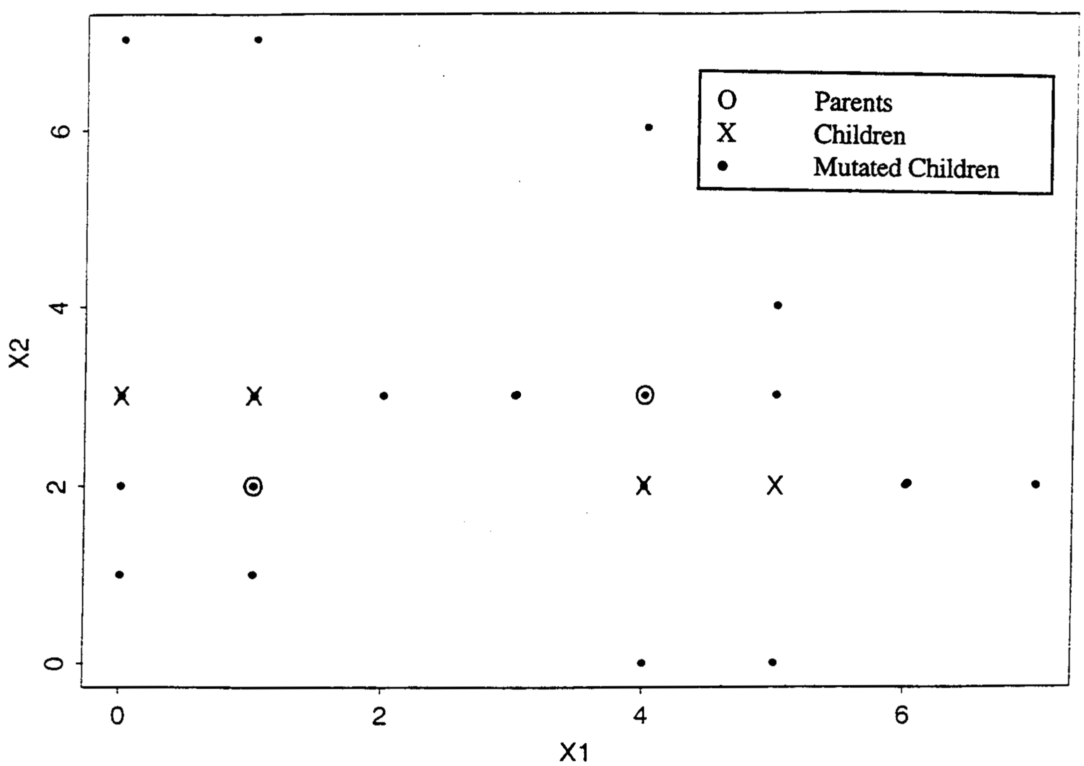

Figure 2a. Potential Children From Standard Genetic Algorithm

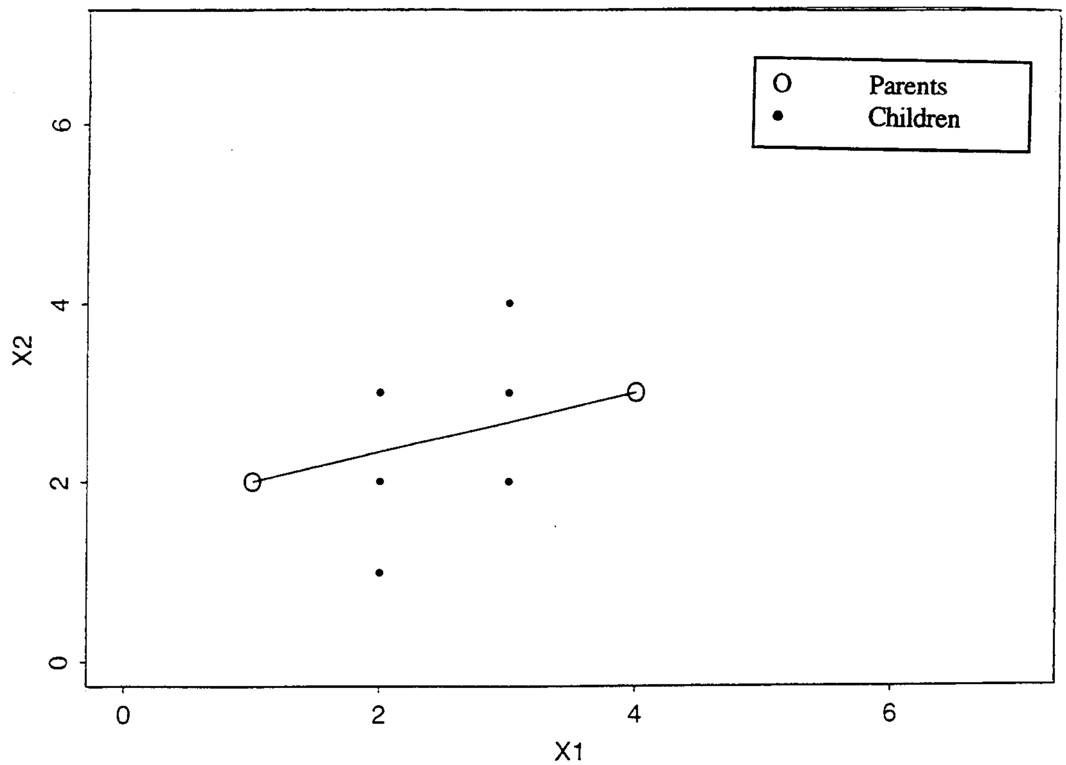

Figure 2b. Potential Children From BCB Algorithm (Sm. Variation)

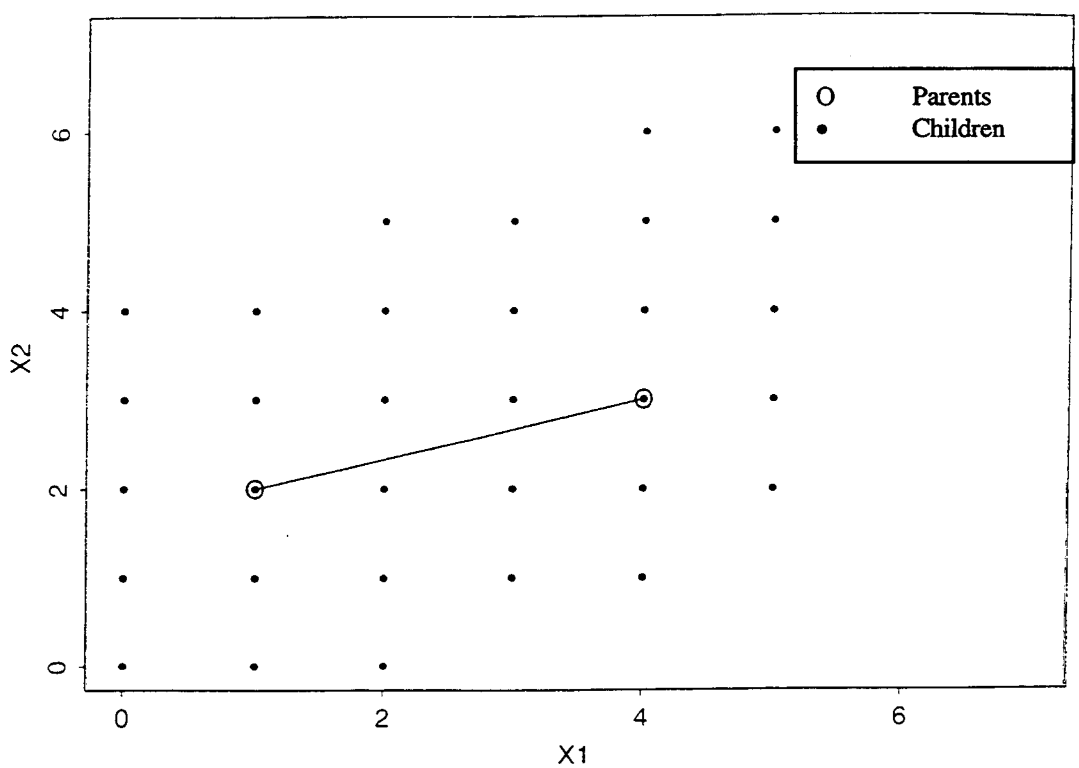

Figure 2c. Potential Children From BCB Algorithm (Lg. Variation)

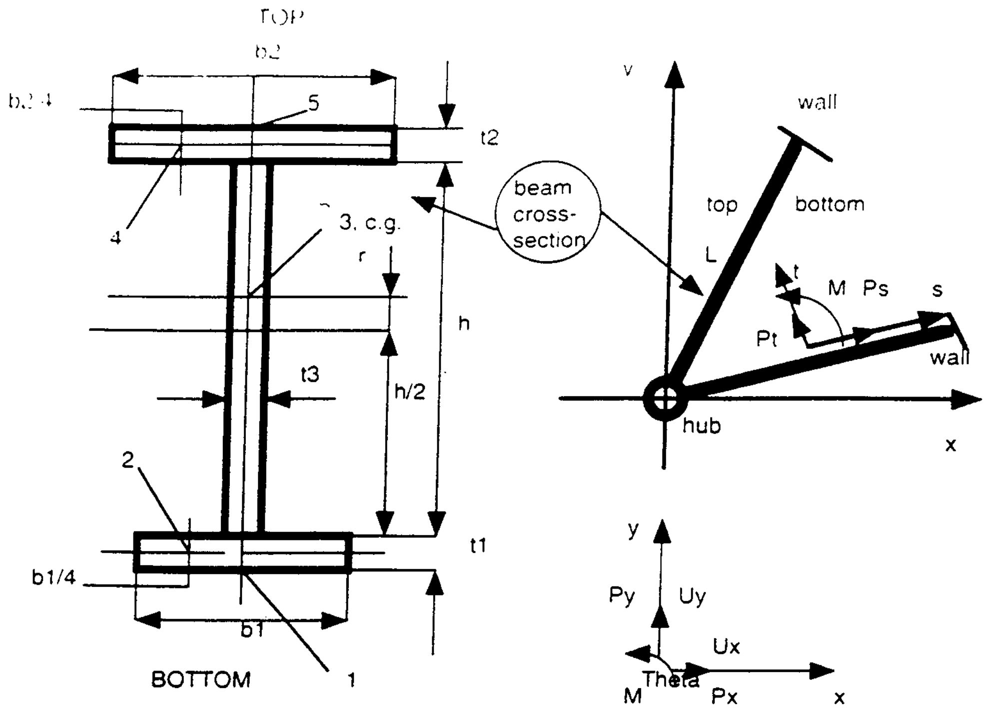

Figure 3. Hub Structure

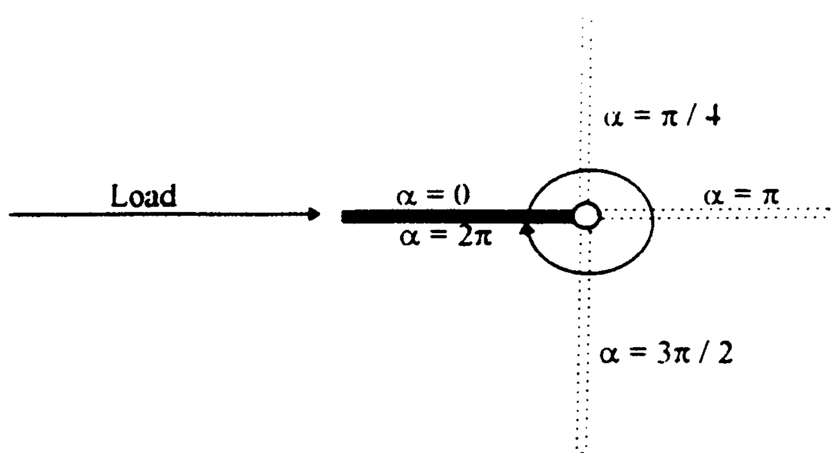

Figure 4. Schematic of Variable α Problem

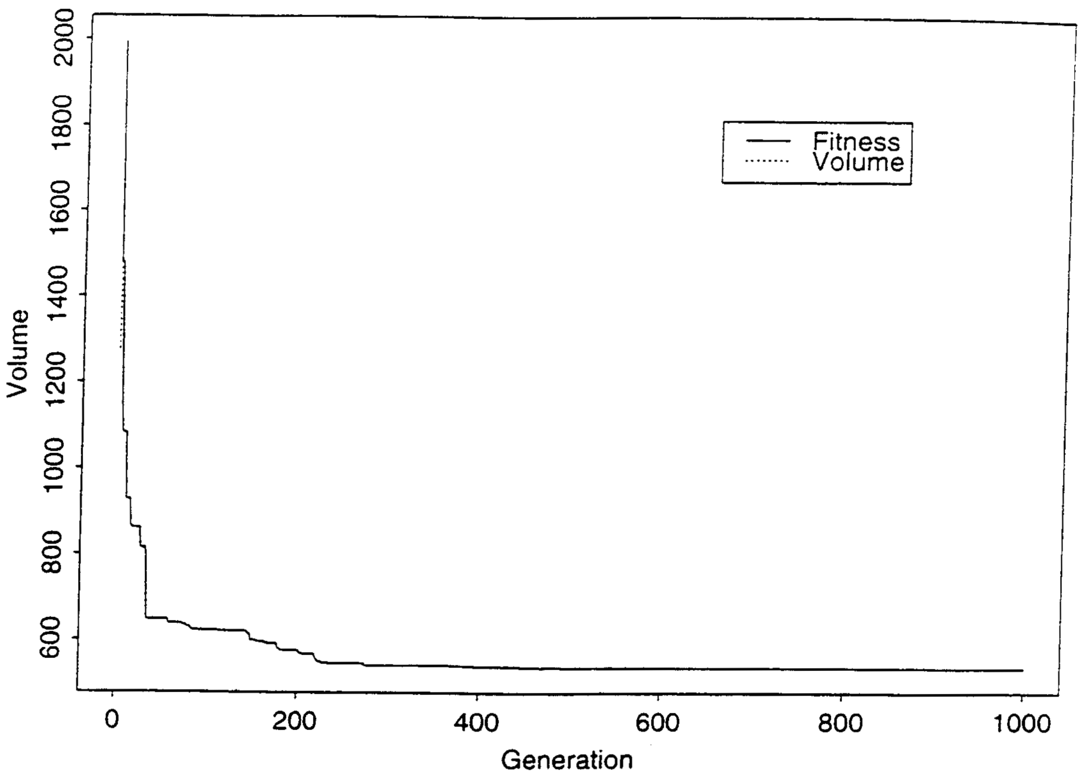

Figure 5. Improving Fitness Using BCB Algorithm (General Case)

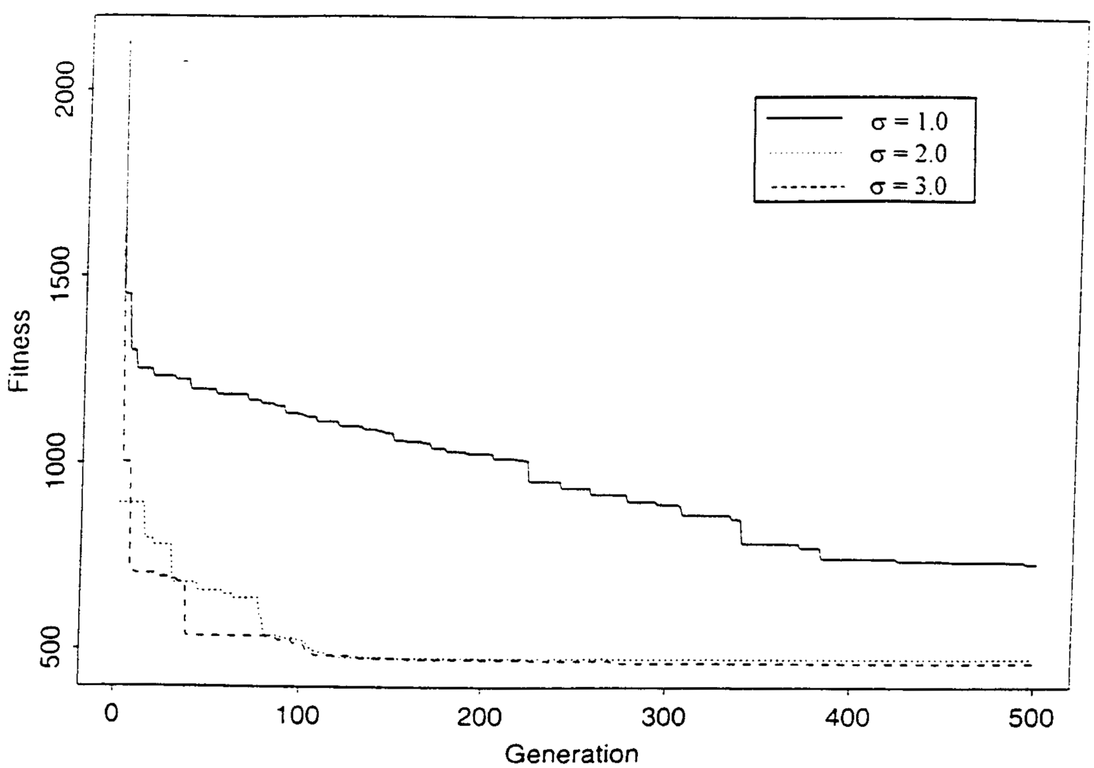

Figure 6. Varying σ Values for Initial Child Generation

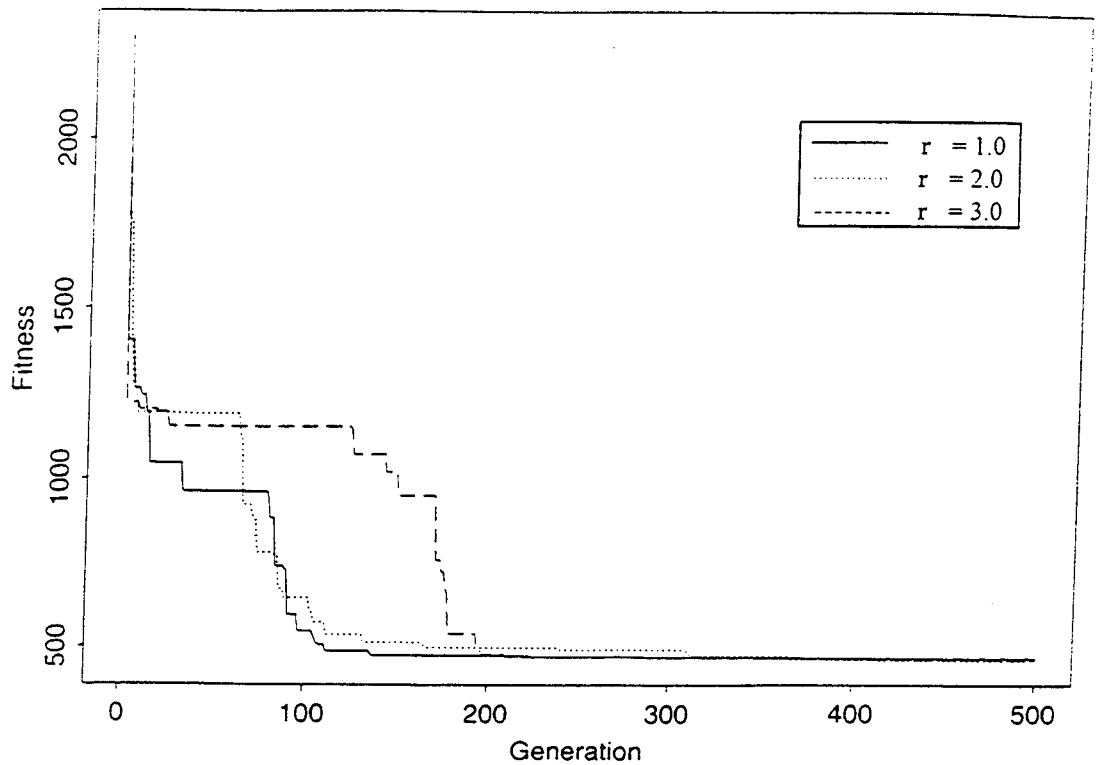

Figure 7. Varying r Values for Radius Generation