Theoretical Background for Aerodynamic Shape Optimization

John C. Vassberg

Boeing Technical Fellow
Advanced Concepts Design Center
Boeing Commercial Airplanes
Long Beach, CA 90846, USA

Antony Jameson

T. V. Jones Professor of Engineering
Dept. Aeronautics & Astronautics
Stanford University
Stanford, CA 94305-3030, USA

Von Karman Institute
Brussels, Belgium
7 April, 2014

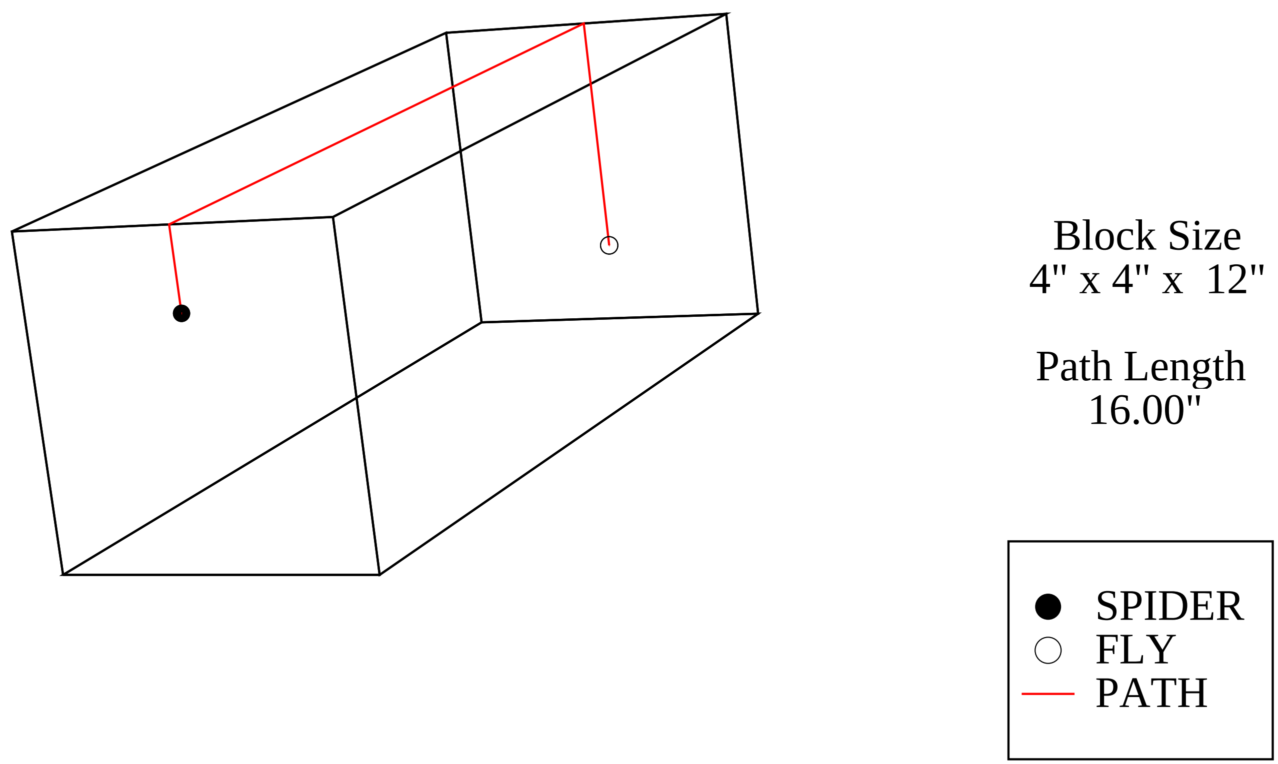
LECTURE OUTLINE

- **INTRODUCTION**
- **THEORETICAL BACKGROUND**
  - SPIDER & FLY
  - BRACHISTOCHRONE
- **SAMPLE APPLICATIONS**
  - MARS AIRCRAFT
  - RENO RACER
  - GENERIC 747 WING/BODY
- **DESIGN-SPACE INFLUENCE**

THE SPIDER & THE FLY

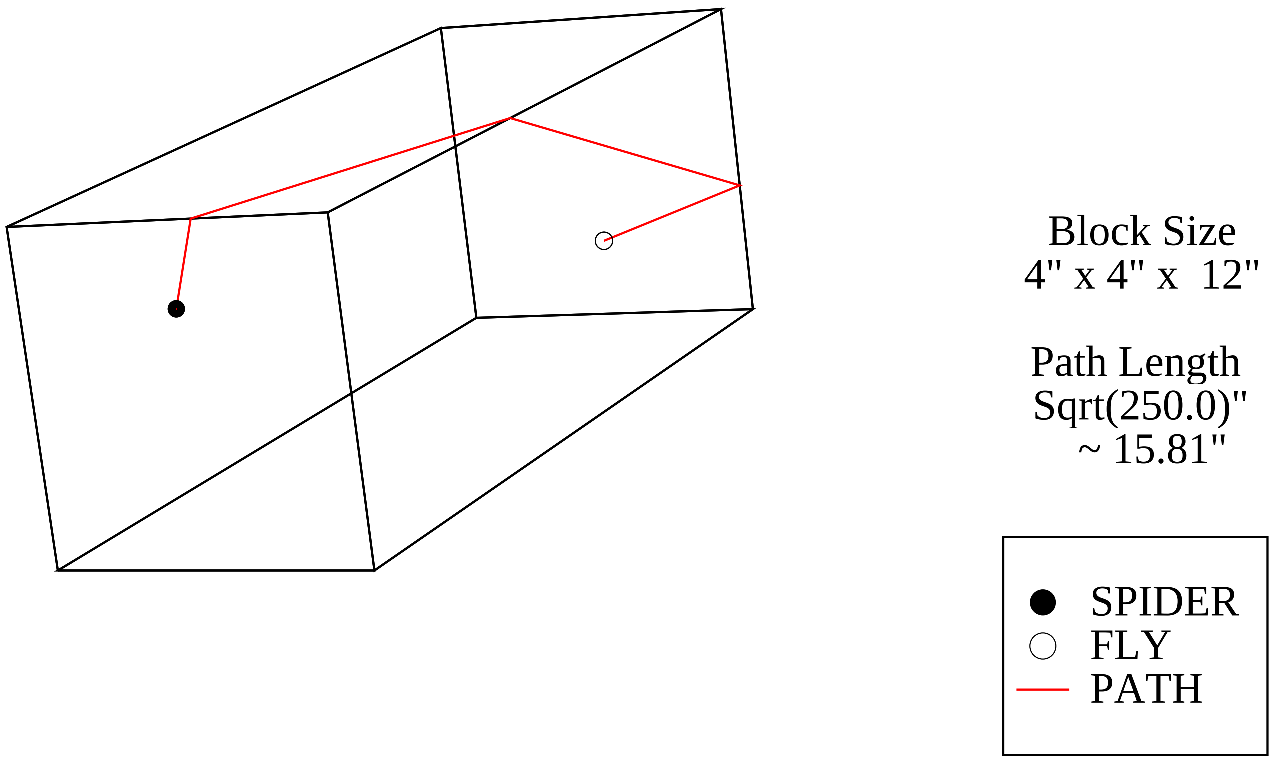
- **PROBLEM STATEMENT**
- **PROBLEM SET-UP**
  - COST FUNCTION
  - DESIGN SPACE
  - GRADIENT & HESSIAN
- **SEARCH METHODS**
  - STEEPEST DESCENT
  - NEWTON ITERATION
  - NASH EQUILIBRIUM
- **EXACT SOLUTION**

THE SPIDER & THE FLY



Obvious Local-Minimum Path between Spider and Fly.

THE SPIDER & THE FLY



Non-Obvious Global-Minimum Path between Spider and Fly.

SPIDER-FLY DESIGN SPACE

Path type to optimize is partitioned into four segments. Path described as the piecewise linear curve that connects:

$$(2,0,3),\ (X,0,4),\ (4,Y,4),\ (4,12,Z),\ (2,12,1).$$

Three design variables (X,Y,Z), constrained by:

$$\begin{array}{rcccr} 0 & \leq & X & \leq & 4, \\ 0 & \leq & Y & \leq & 12, \\ 0 & \leq & Z & \leq & 4. \end{array}$$

SPIDER-FLY COST FUNCTION

Segment Lengths:

$$
\begin{aligned}
S_1 &= \left[1 + (X-2)^2\right]^{\frac{1}{2}}, \\
S_2 &= \left[(X-4)^2 + Y^2\right]^{\frac{1}{2}}, \\
S_3 &= \left[(Y-12)^2 + (Z-4)^2\right]^{\frac{1}{2}}, \\
S_4 &= \left[(Z-1)^2 + 4\right]^{\frac{1}{2}}.
\end{aligned}
$$

Total Path Length:

$$I \equiv S = S_1 + S_2 + S_3 + S_4.$$

Minimize I Subject to Constraints.

SPIDER-FLY GRADIENT

First Variation of Cost Function:

$$\delta I = I_X \delta X + I_Y \delta Y + I_Z \delta Z \equiv G\ \delta \mathcal{X}$$

$$\begin{aligned} I_X &= \frac{(X-2)}{S_1} + \frac{(X-4)}{S_2} \\ I_Y &= \frac{Y}{S_2} + \frac{(Y-12)}{S_3} \\ I_Z &= \frac{(Z-4)}{S_3} + \frac{(Z-1)}{S_4} \end{aligned}$$

$$G \equiv Gradient\ Vector$$

$$\mathcal{X} \equiv Design\ Space\ Vector$$

SPIDER-FLY HESSIAN MATRIX

$$A = \begin{bmatrix} I_{XX} & I_{YX} & I_{ZX} \\ I_{XY} & I_{YY} & I_{ZY} \\ I_{XZ} & I_{YZ} & I_{ZZ} \end{bmatrix},$$

$$\begin{aligned}
I_{XX} &= \frac{1}{S_1^3} + \frac{Y^2}{S_2^3} \\
I_{XY} &= I_{YX} = \frac{(4-X)Y}{S_2^3} \\
I_{XZ} &= I_{ZX} = 0 \\
I_{YY} &= \frac{(X-4)^2}{S_2^3} + \frac{(Z-4)^2}{S_3^3} \\
I_{YZ} &= I_{ZY} = \frac{(Y-12)(4-Z)}{S_3^3} \\
I_{ZZ} &= \frac{(Y-12)^2}{S_3^3} + \frac{4}{S_4^3}
\end{aligned}$$

FINITE-DIFFERENCE APPROXIMATION

Consider the Taylor series expansion of a function f.

$$f(x+\Delta x) = f(x) + \Delta x\ f_x(x) + \frac{\Delta x^2}{2} f_{xx}(x) + \ldots + \frac{\Delta x^n}{n!} f_n(x) + \ldots$$

A first-order accurate approximation of $f_x(x)$ can be determined with the forward differencing formula

$$f_x(x) \simeq \frac{f(x+\Delta x) - f(x)}{\Delta x}.$$

Here Δx is a small perturbation of the X coordinate.

FINITE-DIFFERENCE APPROXIMATION

In the case of the spider-fly, let's approximate I_X.

$$I_X \simeq \frac{I(X+h,Y,Z) - I(X,Y,Z)}{h}$$

For example, using $h = 10^{-3}$ at $(X,Y,Z) = (2,6,2)$ gives:

$\boxed{I_X \simeq -0.31565661\text{, an error of about 0.1\%.}}$

The exact value of I_X at this location is $-\frac{2}{\sqrt{40}} \simeq -0.31622777$.

COMPLEX-VARIABLE APPROXIMATION

Consider the Taylor series expansion of a complex function f.

$$f(x+\Delta x) = f(x) + \Delta x\ f_x(x) + \frac{\Delta x^2}{2} f_{xx}(x) + \ldots + \frac{\Delta x^n}{n!} f_n(x) + \ldots$$

A second-order accurate approximation of $f_x(x)$ can be found with the complex-variable formula

$$f_x(x) \simeq \frac{Im[f(x+ih)]}{h}.$$

Here $\Delta x = ih$ is an imaginary perturbation of X.

COMPLEX-VARIABLE APPROXIMATION

In the case of the spider-fly, let's approximate I_X.

$$I_X \simeq \frac{Im[I(X+ih,Y,Z)]}{h}$$

For all $h \leq 10^{-3}$ at $(X,Y,Z) = (2,6,2)$, we get:

$$\boxed{I_X \simeq -0.31622777.}$$

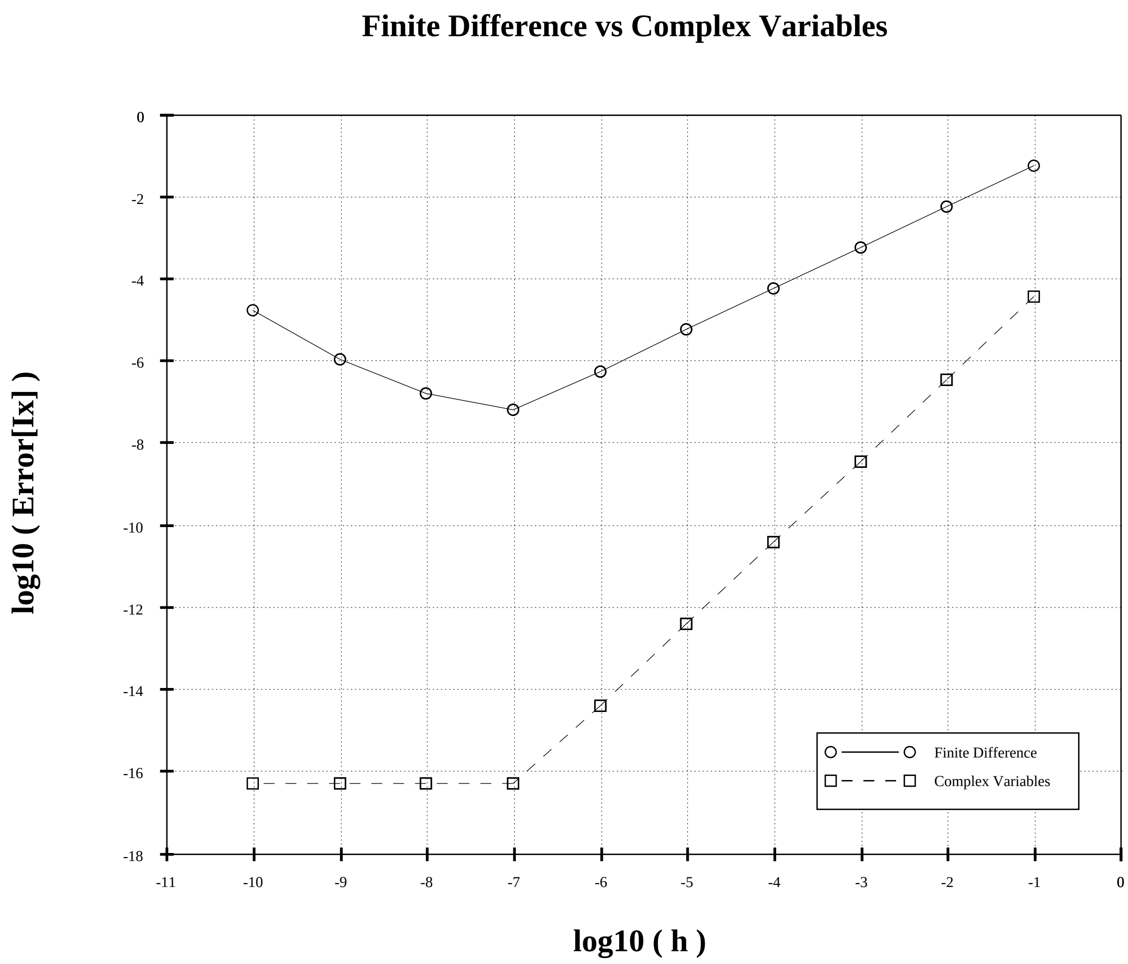
This is identical to the exact value to 8 significant digits.

GRADIENT APPROXIMATION

	$log_{10}(Error\ I_X)$	
$log_{10}(h)$	Finite Difference	Complex Variable
-1	-1.244	-4.449
-2	-2.243	-6.449
-3	-3.243	-8.449
-4	-4.243	-10.449
-5	-5.243	-12.449
-6	-6.244	-14.449
-7	-7.192	-16.256
-8	-6.778	-16.256
-9	-5.977	-16.256
-10	-4.768	-16.256

Stability of Finite-Difference and Complex-Variable Methods

GRADIENT APPROXIMATION



SPIDER-FLY SEARCH METHODS

Trajectory:

$$\mathcal{X}^{n+1} = \mathcal{X}^n + \delta\mathcal{X}^n$$

Steepest Descent:

$$\delta\mathcal{X}^n = -\lambda G, \quad \lambda > 0$$

$$\delta I^n = G\,\delta\mathcal{X}^n = -\lambda G^2 \leq 0$$

Newton Iteration:

$$\delta\mathcal{X}^n = -A^{-1}G = -HG$$

SPIDER-FLY SEARCH METHODS

Rank-1 quasi-Newton:

$$H^{n+1} = H^n + \frac{(\mathcal{P}^n)(\mathcal{P}^n)^T}{(\mathcal{P}^n)^T \delta G^n},$$

where

$$\delta G^n = G^{n+1} - G^n$$

and

$$\mathcal{P}^n = \delta \mathcal{X}^n - H^n \delta G^n.$$

SPIDER-FLY SEARCH METHODS

Nash Equilibrium:

$$\begin{array}{lllll} minimize\ I(X^\star, Y^n, Z^n) & => & I_x(X^\star, Y^n, Z^n) = 0 & => & X^\star, \\ minimize\ I(X^n, Y^\star, Z^n) & => & I_x(X^n, Y^\star, Z^n) = 0 & => & Y^\star, \\ minimize\ I(X^n, Y^n, Z^\star) & => & I_x(X^n, Y^n, Z^\star) = 0 & => & Z^\star. \end{array}$$

These reduce to:

$$X^\star = \frac{2(2+Y^n)}{(1+Y^n)}, \qquad Y^\star = \frac{12(4-X^n)}{(8-X^n-Z^n)}, \qquad Z^\star = 4 - \frac{3(12-Y^n)}{(14-Y^n)}.$$

Update design vector:

$$[X^{n+1}, Y^{n+1}, Z^{n+1}]^T = [X^\star, Y^\star, Z^\star]^T$$

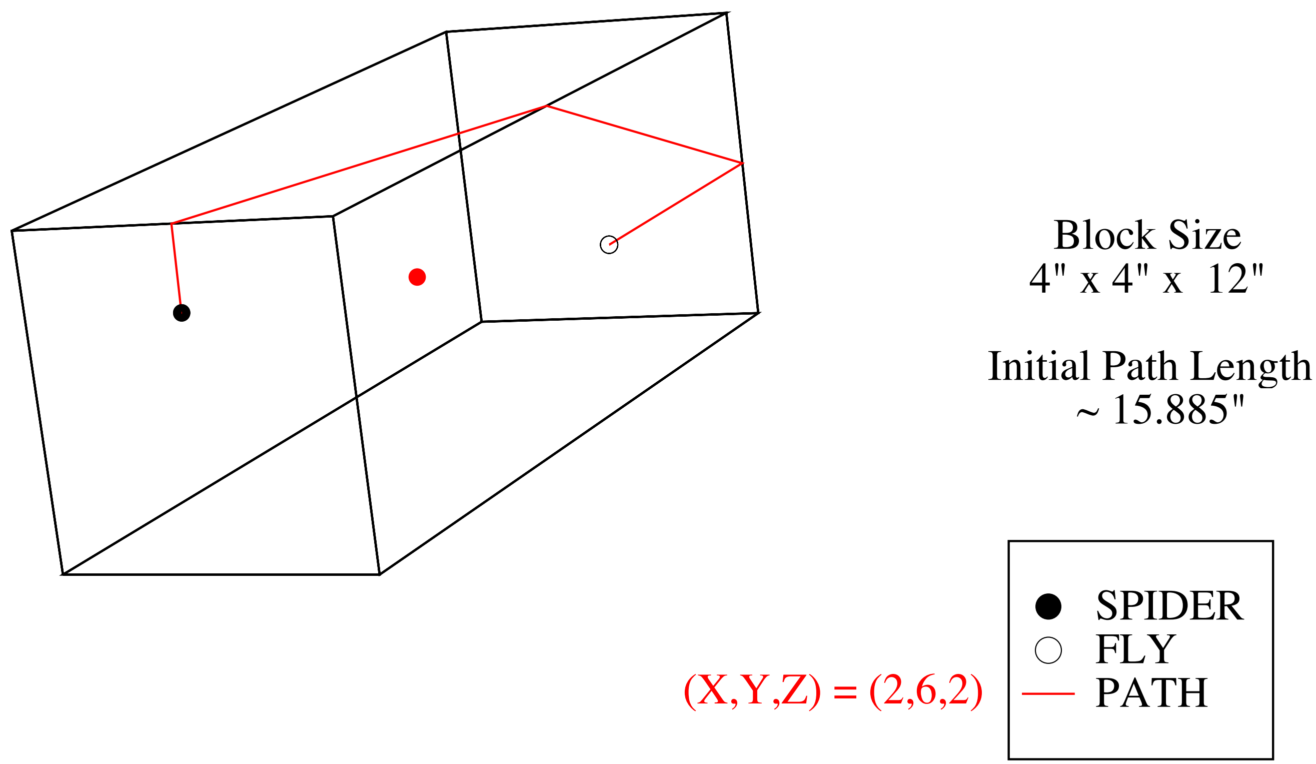
SPIDER-FLY INITIAL PATH

$$\mathcal{X}^0 = \begin{bmatrix} 2 \\ 6 \\ 2 \end{bmatrix}, \qquad G^0 = \begin{bmatrix} \frac{-2}{\sqrt{40}} \\ 0.0 \\ (\frac{-3}{\sqrt{40}} + \frac{1}{\sqrt{5}}) \end{bmatrix} \approx \begin{bmatrix} -0.31623 \\ 0.0 \\ -0.02713 \end{bmatrix},$$

$$A^0 \approx \begin{bmatrix} 1.14230 & 0.04743 & 0.0 \\ 0.04743 & 0.03162 & -0.04743 \\ 0.0 & -0.04743 & 0.50007 \end{bmatrix},$$

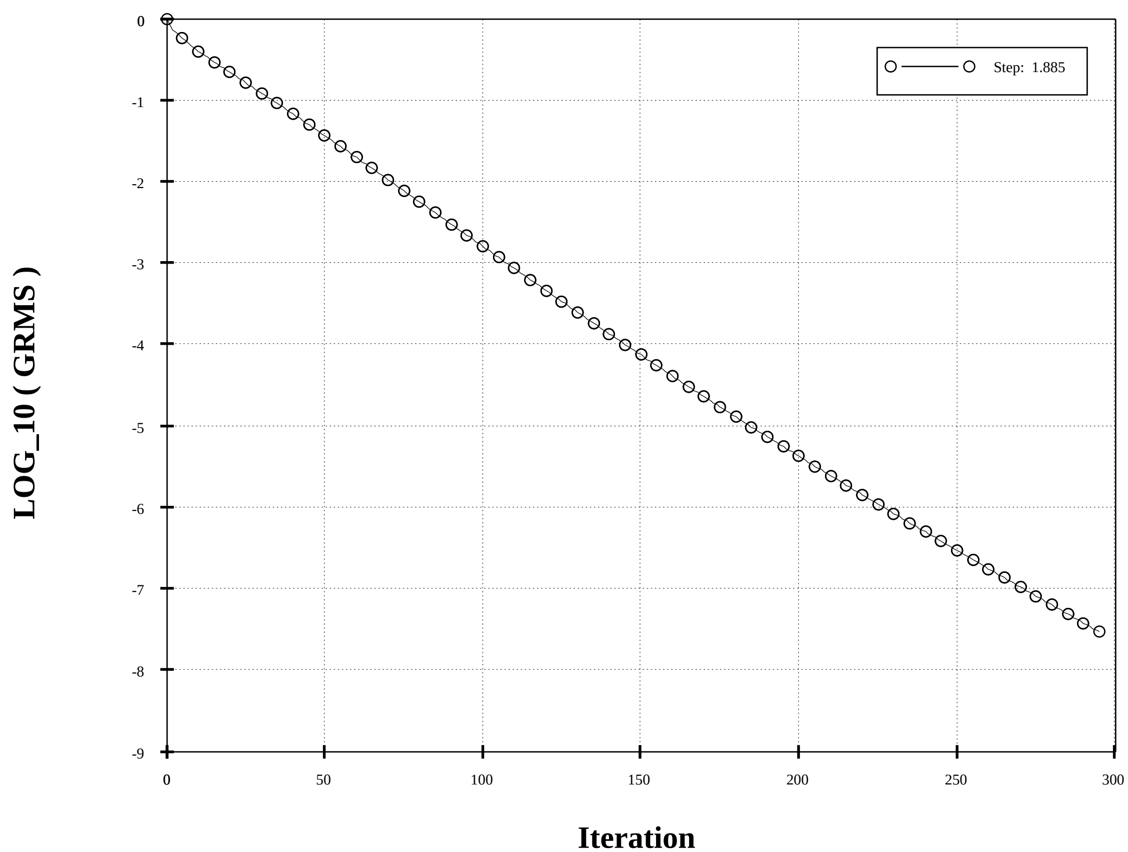
$$I^0 = (1 + 2\sqrt{40} + \sqrt{5}) \approx 15.88518$$

SPIDER-FLY INITIAL PATH



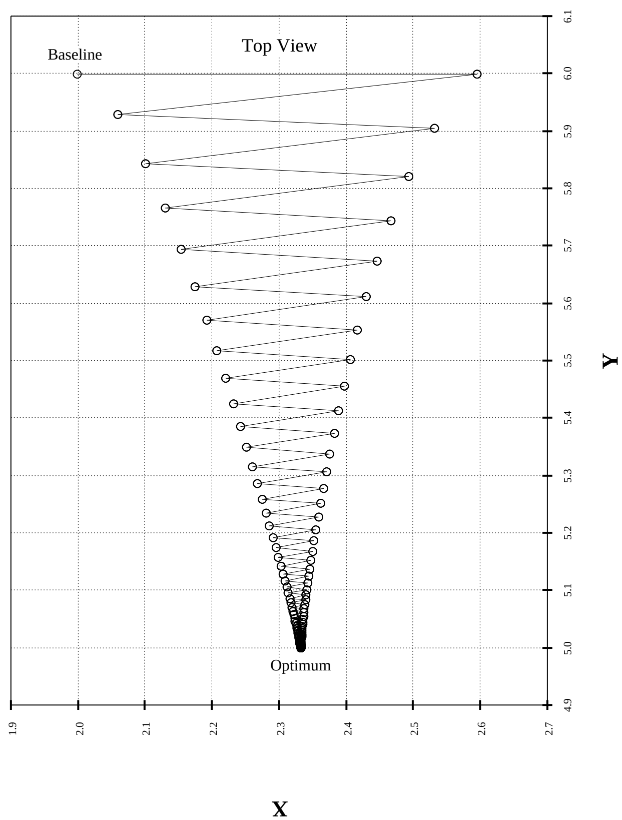
Initial Path between Spider and Fly.

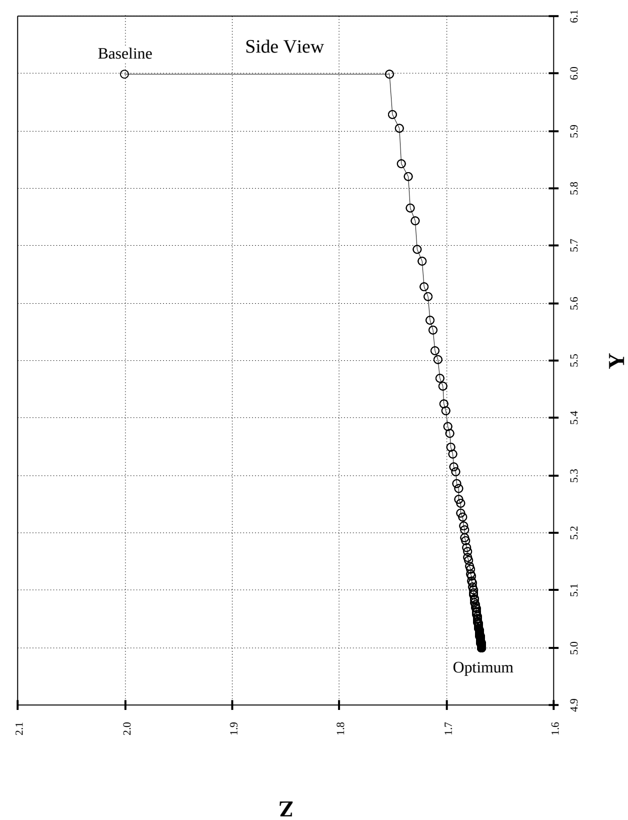
SPIDER-FLY STEEPEST DESCENT



Convergence of Gradient for Steepest Descent.

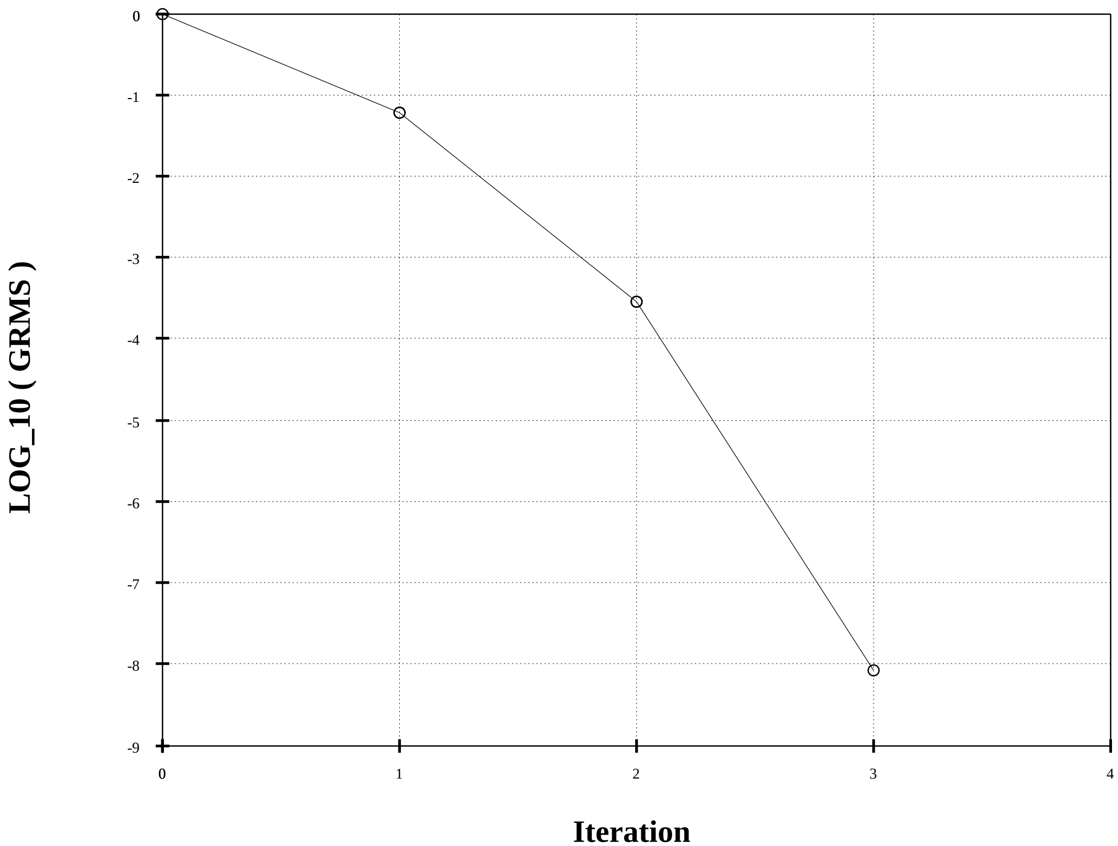
SPIDER-FLY STEEPEST DESCENT





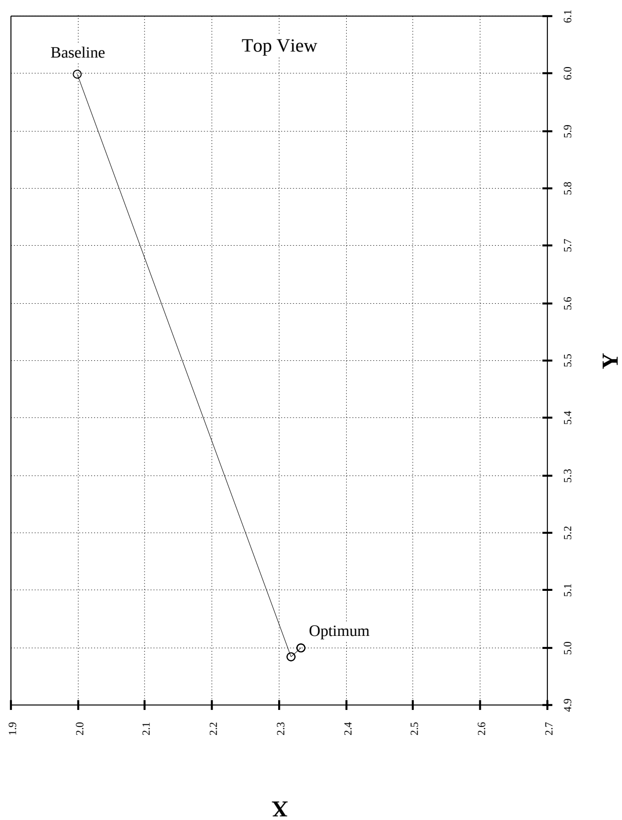
Steepest-Descent Trajectory through Design Space.

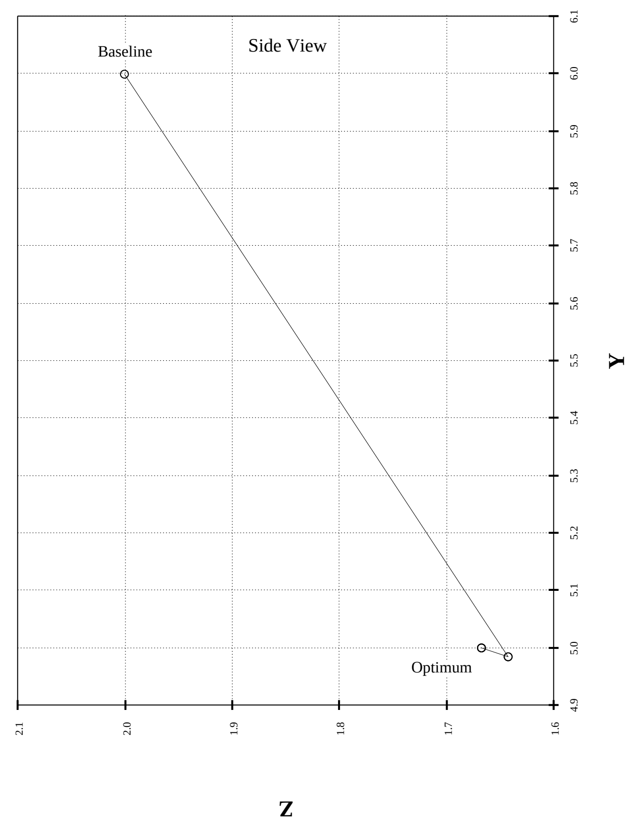
SPIDER-FLY NEWTON ITERATION



Convergence of Gradient for Newton Iteration.

SPIDER-FLY NEWTON ITERATION





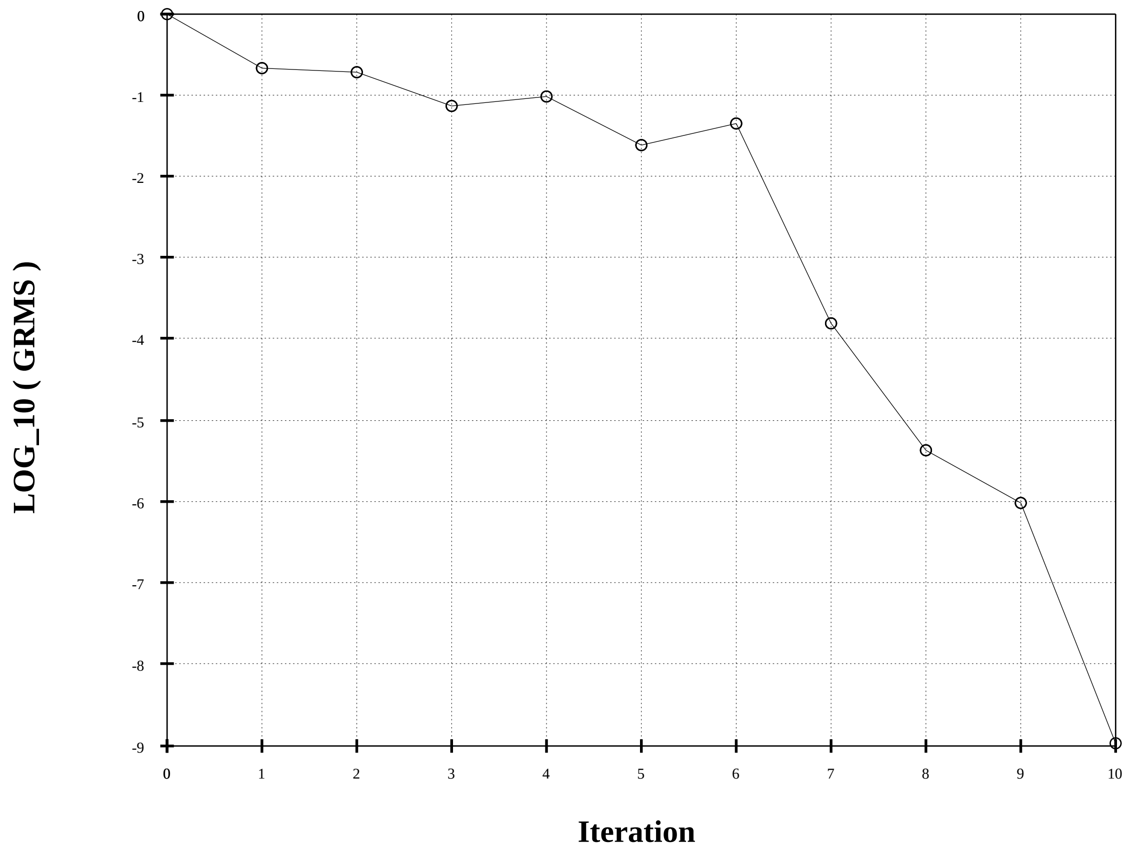
Newton-Iteration Trajectory through Design Space.

SPIDER-FLY NEWTON ITERATION

n	X^n	Y^n	Z^n	I^n
0	2.000000	6.000000	2.000000	15.88518
1	2.319023	4.984009	1.641696	15.81167
2	2.333268	4.999744	1.666556	15.81139
3	2.333333	5.000000	1.666667	15.81139

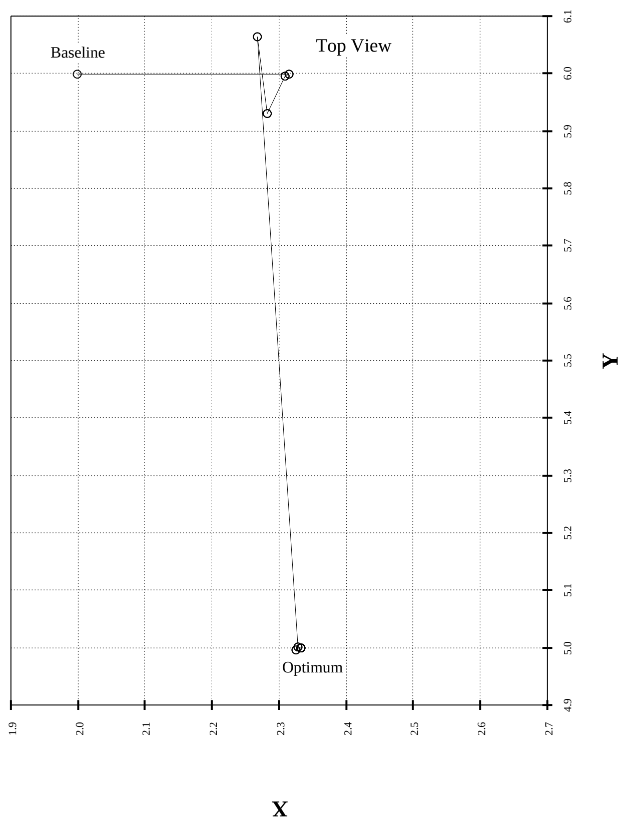
Convergence of Newton Iteration on the Spider-Fly Problem.

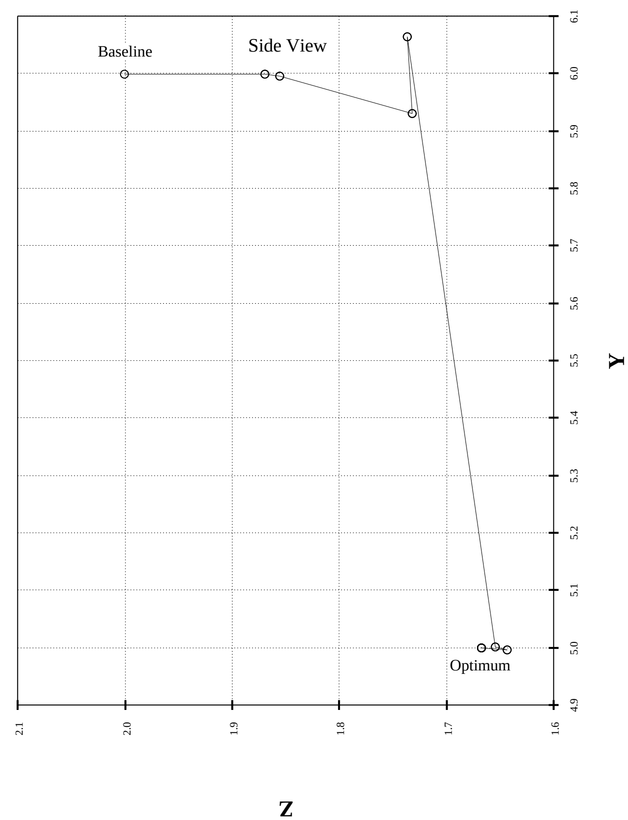
SPIDER-FLY RANK-1 QUASI-NEWTON



Convergence of Gradient for Rank-1 quasi-Newton Iteration.

SPIDER-FLY RANK-1 QUASI-NEWTON





Rank-1 quasi-Newton Trajectory through the Design Space.

SPIDER-FLY RANK-1 QUASI-NEWTON

n	X^n	Y^n	Z^n	I^n
0	2.000000	6.000000	2.000000	15.88518
1	2.316228	6.000000	1.869014	15.82842
2	2.309340	5.995497	1.854977	15.82729
3	2.283594	5.931327	1.731183	15.82250
4	2.268113	6.064459	1.736156	15.82602
5	2.329076	5.002280	1.654099	15.81144
6	2.325976	4.997523	1.643056	15.81157
7	2.333299	4.999719	1.666628	15.81139
8	2.333331	5.000017	1.666668	15.81139
9	2.333333	5.000002	1.666667	15.81139
10	2.333333	5.000000	1.666667	15.81139

Convergence of Rank-1 quasi-Newton on Spider-Fly.

SPIDER-FLY RANK-1 QUASI-NEWTON

n	G^n		
0	-0.3162278	0.0000000	0.1309858
1	0.0313201	0.0204757	0.0638312
2	0.0241196	0.0207513	0.0566640
3	-0.0051403	0.0239077	-0.0068202
4	-0.0156349	0.0272111	-0.0109428
5	-0.0042385	0.0003440	-0.0070051
6	-0.0076998	0.0004624	-0.0129071
7	-0.0000509	-0.0000095	-0.0000100
8	-0.0000014	0.0000003	0.0000003
9	-0.0000003	0.0000000	0.0000001
10	0.0000000	0.0000000	0.0000000

Convergence of Rank-1 quasi-Newton on Spider-Fly.

SPIDER-FLY RANK-1 QUASI-NEWTON

n	$\mathcal{P}^n$		
0	-0.0313201	-0.0204757	-0.0638312
1	-0.0027101	-0.0067548	-0.0130308
2	0.0032314	-0.0277891	-0.0010379
3	-0.0092369	0.1609362	0.0124328
4	0.0038082	0.0058428	0.0135643
5	-0.0061541	-0.0018454	-0.0198081
6	0.0000316	0.0002953	0.0000405
7	0.0000021	-0.0000171	-0.0000017
8	0.0000003	-0.0000015	-0.0000002
9	0.0000000	0.0000000	0.0000000
10			

Convergence of Rank-1 quasi-Newton on Spider-Fly.

SPIDER-FLY RANK-1 QUASI-NEWTON

n	H^n		
0	1.0000000	0.0000000	0.0000000
	0.0000000	1.0000000	0.0000000
	0.0000000	0.0000000	1.0000000
1	0.8602224	-0.0913802	-0.2848703
	-0.0913802	0.9402598	-0.1862351
	-0.2848703	-0.1862351	0.4194274
2	0.9263627	0.0734691	0.0331463
	0.0734691	1.3511333	0.6063951
	0.0331463	0.6063951	1.9485177
3	0.8366376	0.8450854	0.0619643
	0.8450854	-5.2845988	0.3585666
	0.0619643	0.3585666	1.9392619
4	0.9844233	-1.7298270	-0.1369557
	-1.7298270	39.5788215	3.8244048
	-0.1369557	3.8244048	2.2070087
5	0.7433885	-2.0996388	-0.9954916
	-2.0996388	39.0114314	2.5071815
	-0.9954916	2.5071815	-0.8509886

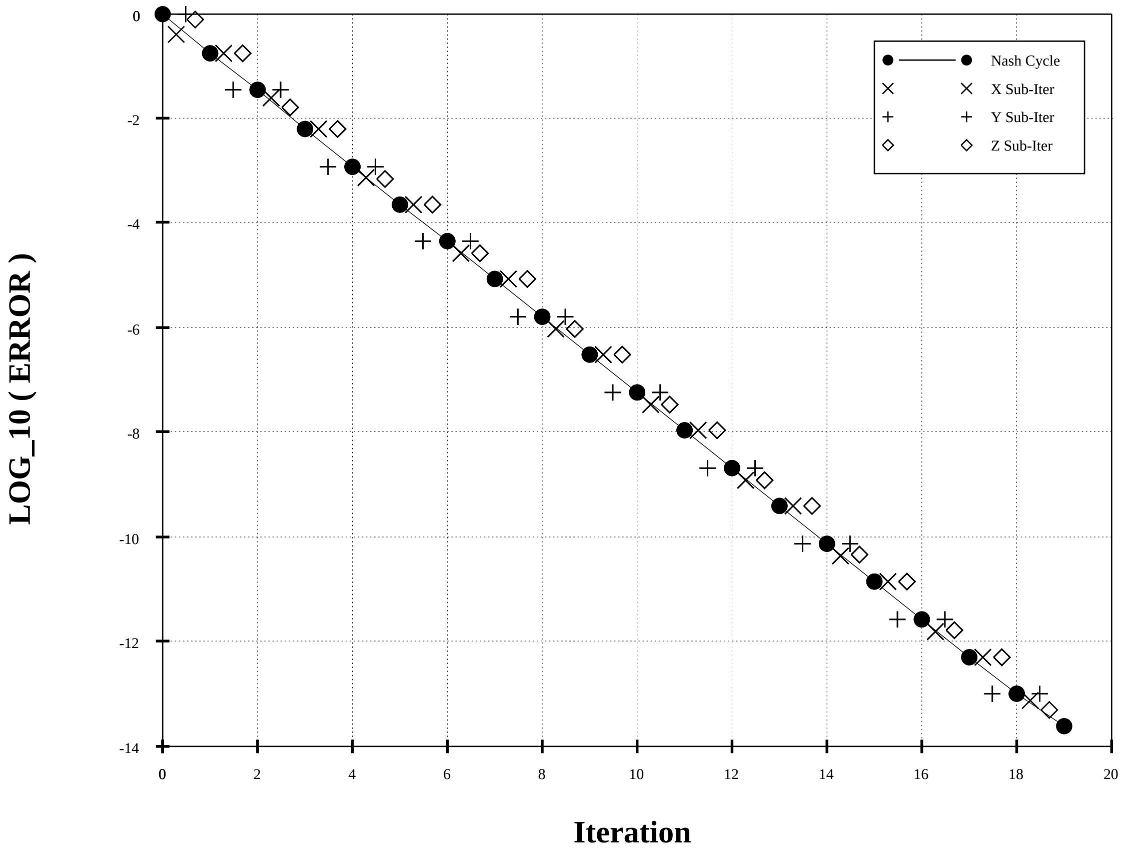
Convergence of Rank-1 quasi-Newton on Spider-Fly.

SPIDER-FLY RANK-1 QUASI-NEWTON

n	H^n		
6	1.0178515	-2.0173360	-0.1120774
	-2.0173360	39.0361115	2.7720896
	-0.1120774	2.7720896	1.9924568
7	1.0194495	-2.0023937	-0.1100296
	-2.0023937	39.1758307	2.7912373
	-0.1100296	2.7912373	1.9950809
8	0.9628110	-1.5479228	-0.0640429
	-1.5479228	35.5291298	2.4222374
	-0.0640429	2.4222374	1.9577428
9	1.0930870	-2.1085463	-0.1558491
	-2.1085463	37.9416902	2.8173117
	-0.1558491	2.8173117	2.0224391
10	1.0931086	-2.1081974	-0.1562477
	-2.1081974	37.9473320	2.8108673
	-0.1562477	2.8108673	2.0298003
∞	1.0931330	-2.1081851	-0.1561619
	-2.1081851	37.9473319	2.8109135
	-0.1561619	2.8109135	2.0301042

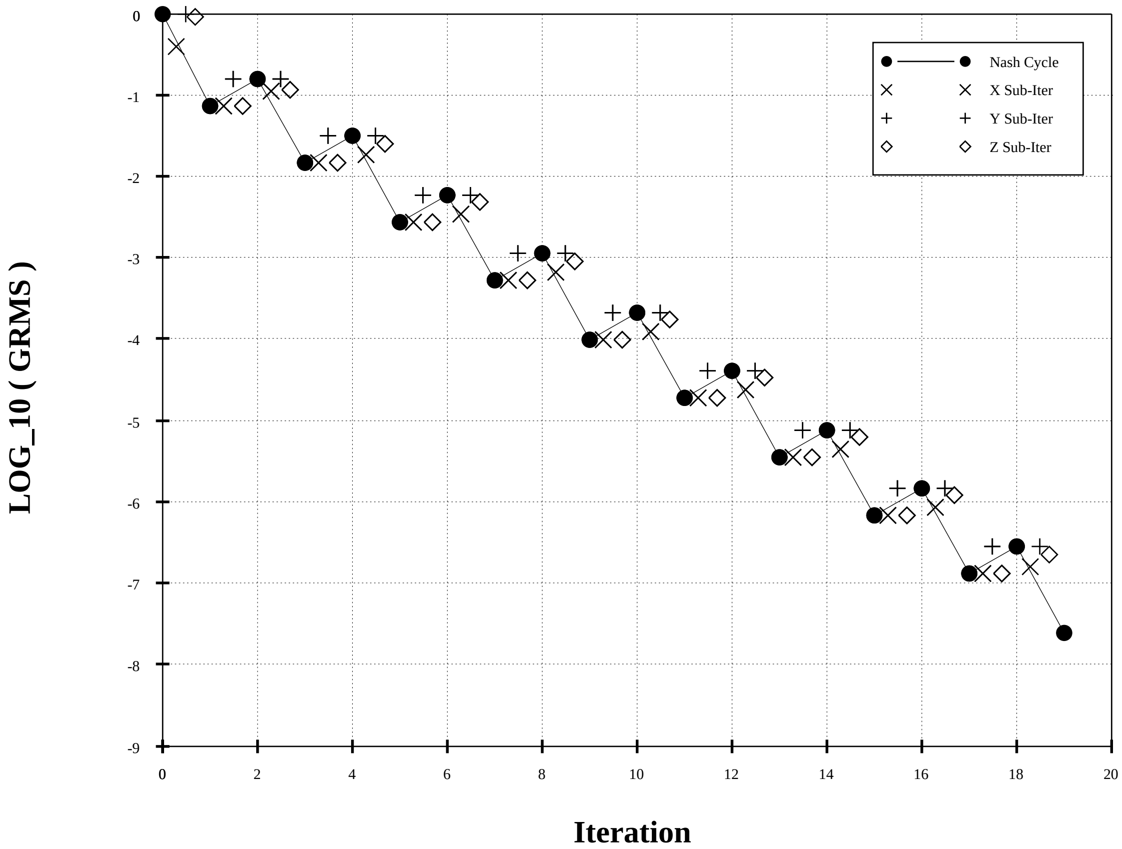
Convergence of Rank-1 quasi-Newton on Spider-Fly.

SPIDER-FLY NASH EQUILIBRIUM



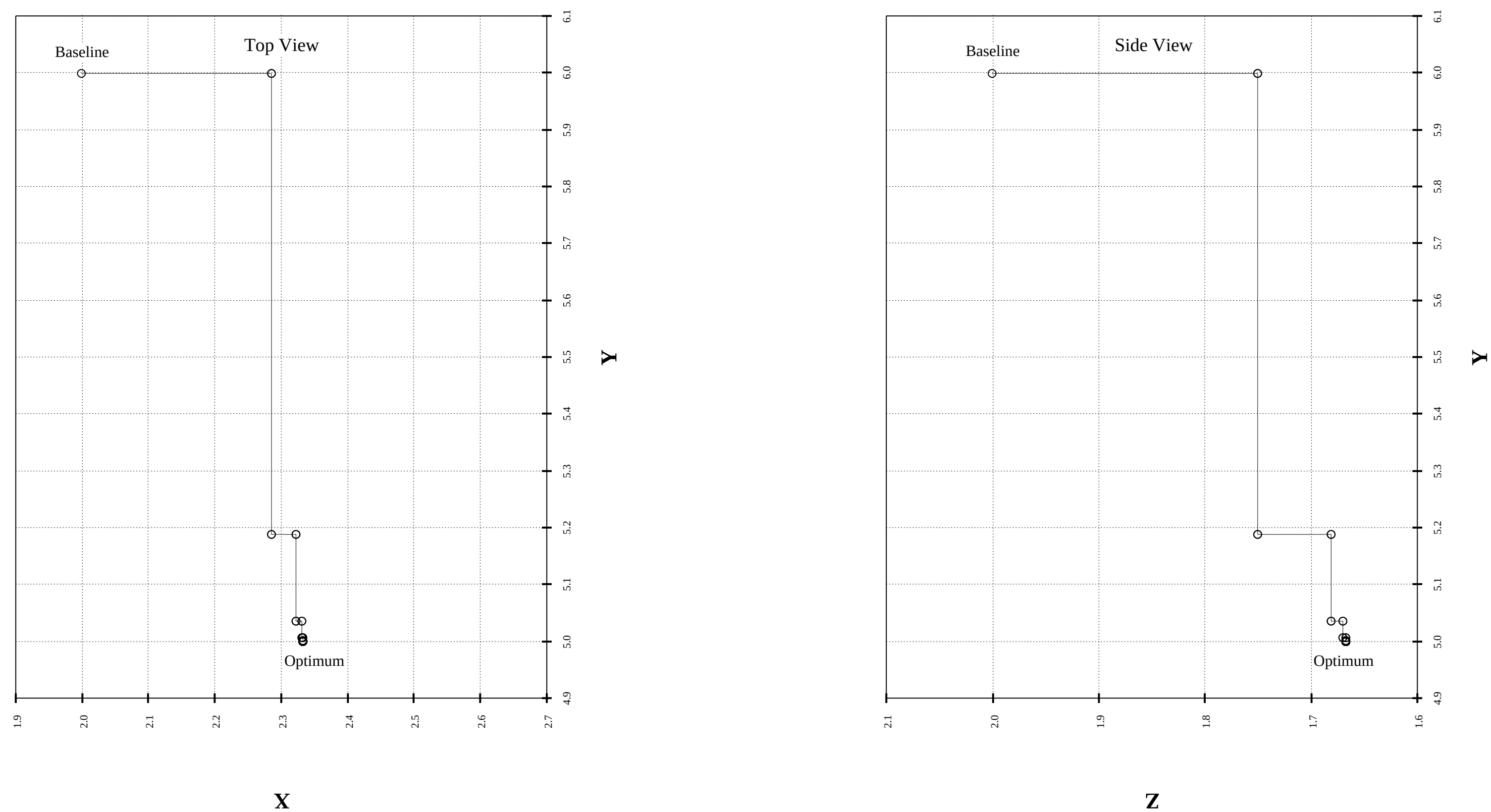
Convergence of Error for Nash Equilibrium.

SPIDER-FLY NASH EQUILIBRIUM



Convergence of Gradient for Nash Equilibrium.

SPIDER-FLY NASH EQUILIBRIUM



Nash Equilibrium Trajectory through the Design Space.

SPIDER-FLY NASH EQUILIBRIUM

n	X^n	Y^n	Z^n	I^n
0	2.000000	6.000000	2.000000	15.88518
1	2.285714	6.000000	1.750000	15.82411
2	2.285714	5.189189	1.750000	15.81388
3	2.323144	5.189189	1.680982	15.81186
4	2.323144	5.035762	1.680982	15.81148
5	2.331358	5.035762	1.669326	15.81141
6	2.331358	5.006782	1.669326	15.81139
7	2.332957	5.006782	1.667169	15.81139
8	2.332957	5.001287	1.667169	15.81139
9	2.333262	5.001287	1.666762	15.81139
10	2.333262	5.000244	1.666762	15.81139
11	2.333320	5.000244	1.666685	15.81139
12	2.333320	5.000046	1.666685	15.81139
13	2.333331	5.000046	1.666670	15.81139
14	2.333331	5.000009	1.666670	15.81139
15	2.333333	5.000009	1.666667	15.81139
16	2.333333	5.000002	1.666667	15.81139
17	2.333333	5.000002	1.666667	15.81139
18	2.333333	5.000000	1.666667	15.81139
19	2.333333	5.000000	1.666667	15.81139

Convergence of Nash Equilibrium on Spider-Fly.

SPIDER-FLY GEODESIC

Super Ellipsoid Surface:

$$\left[\frac{|x-2|}{2}\right]^p + \left[\frac{|y-6|}{6}\right]^p + \left[\frac{|z-2|}{2}\right]^p = 1, \quad p \geq 2$$

Spider Initial Position:

$$\begin{aligned} XS &= 2 \\ YS &= 6\left[1 - \left[1 - \frac{1}{2^p}\right]^{\frac{1}{p}}\right] \\ ZS &= 3 \end{aligned}$$

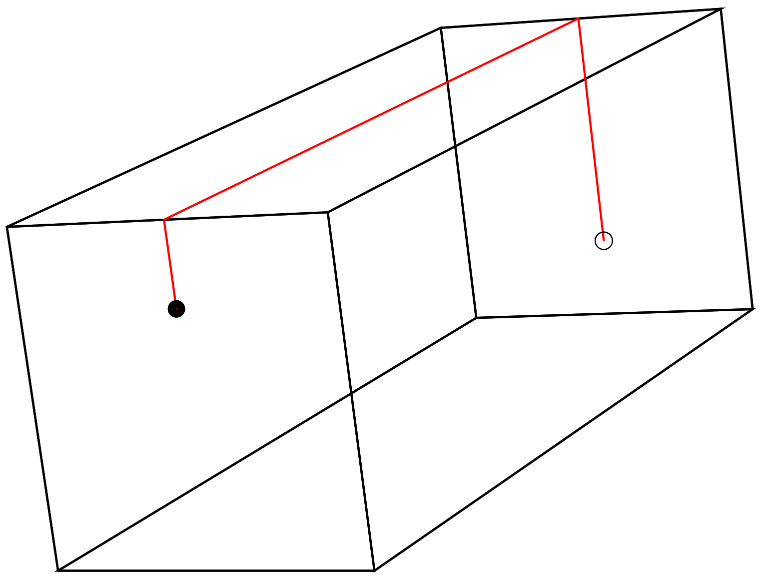
Trapped Fly Position:

$$\begin{aligned} XF &= 2 \\ YF &= 12 - YS \\ ZF &= 1 \end{aligned}$$

SPIDER-FLY OBSERVATIONS

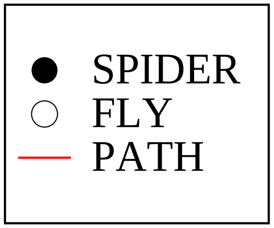
- **CHOICE OF PATH**
  - WOODEN BLOCK *vs* SUPER ELLIPSOID
  - DEFINES COST FUNCTION & DESIGN SPACE
  - DISCRETE *vs* CONTINUUM
- **CHOICE OF SEARCH METHOD**
  - N.I. $3(1+3) << 295$ S.D. $\rightarrow$ GOOD TRADE
  - HESSIAN COST $= \mathcal{O}(N) *$GRADIENT COST
  - LARGE $N \rightarrow$ AVOID NEWTON ITERATION

SPIDER-FLY EXACT SOLUTION



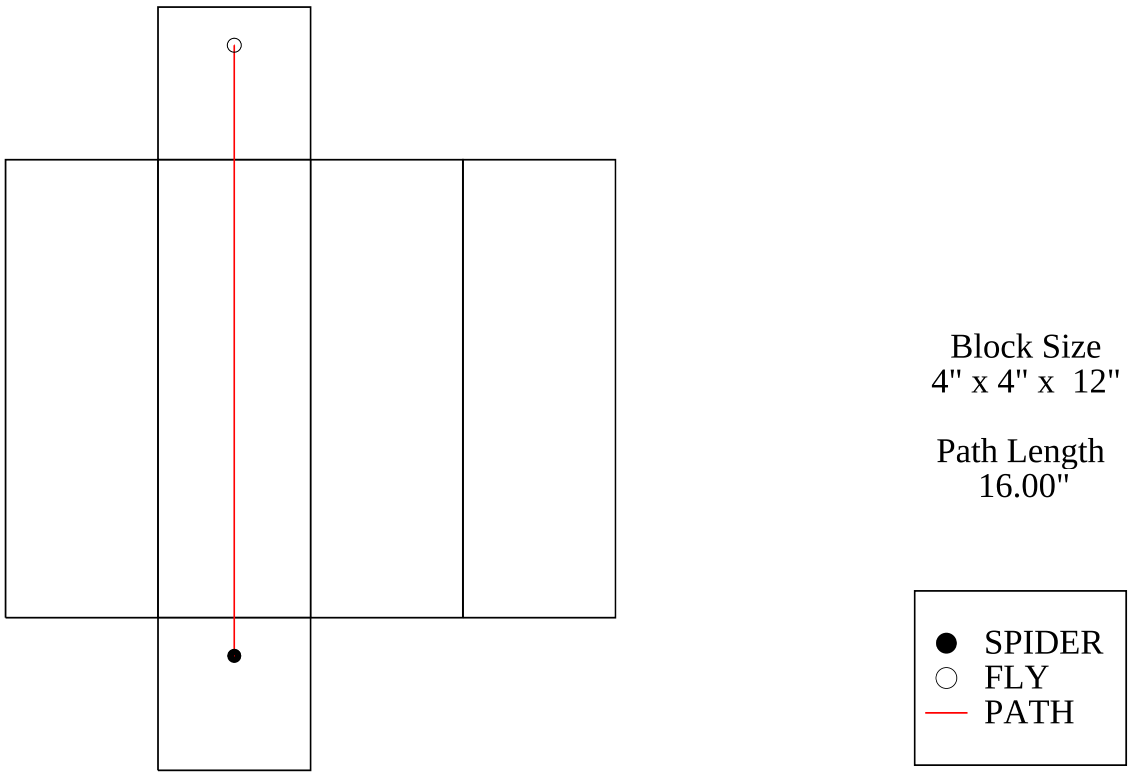
Block Size
4" x 4" x 12"

Path Length
16.00"



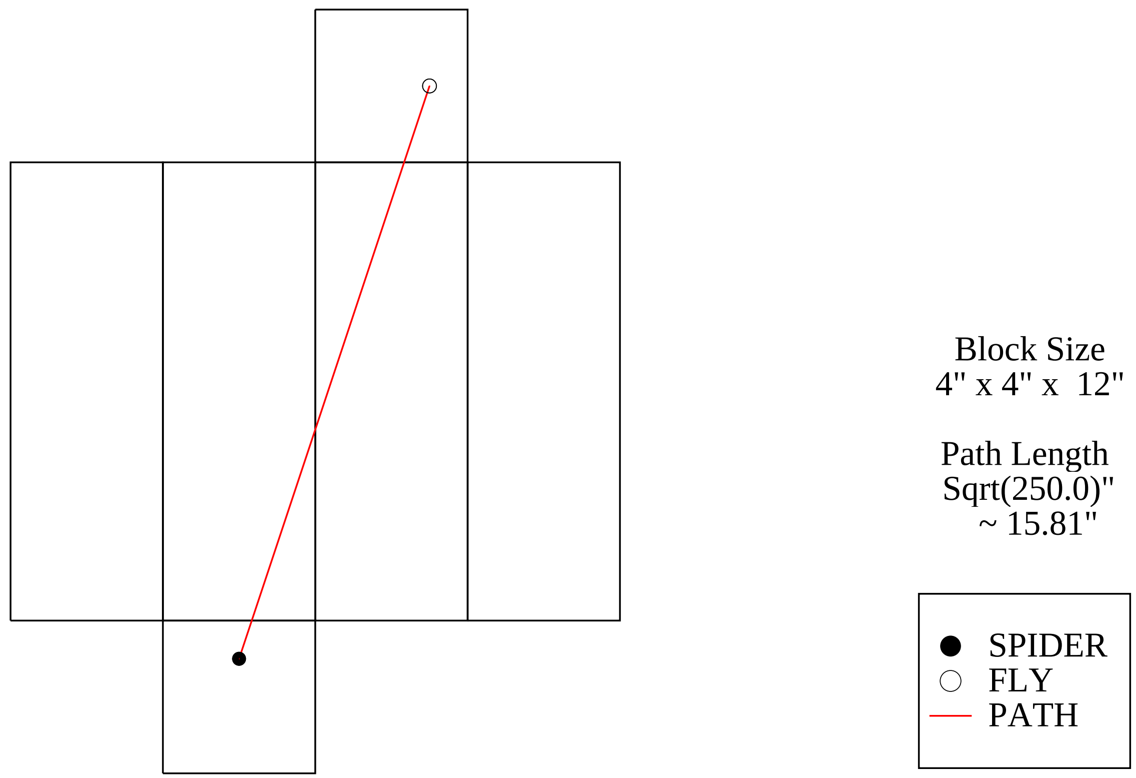
Obvious Local-Minimum Path between Spider and Fly.

SPIDER-FLY EXACT SOLUTION



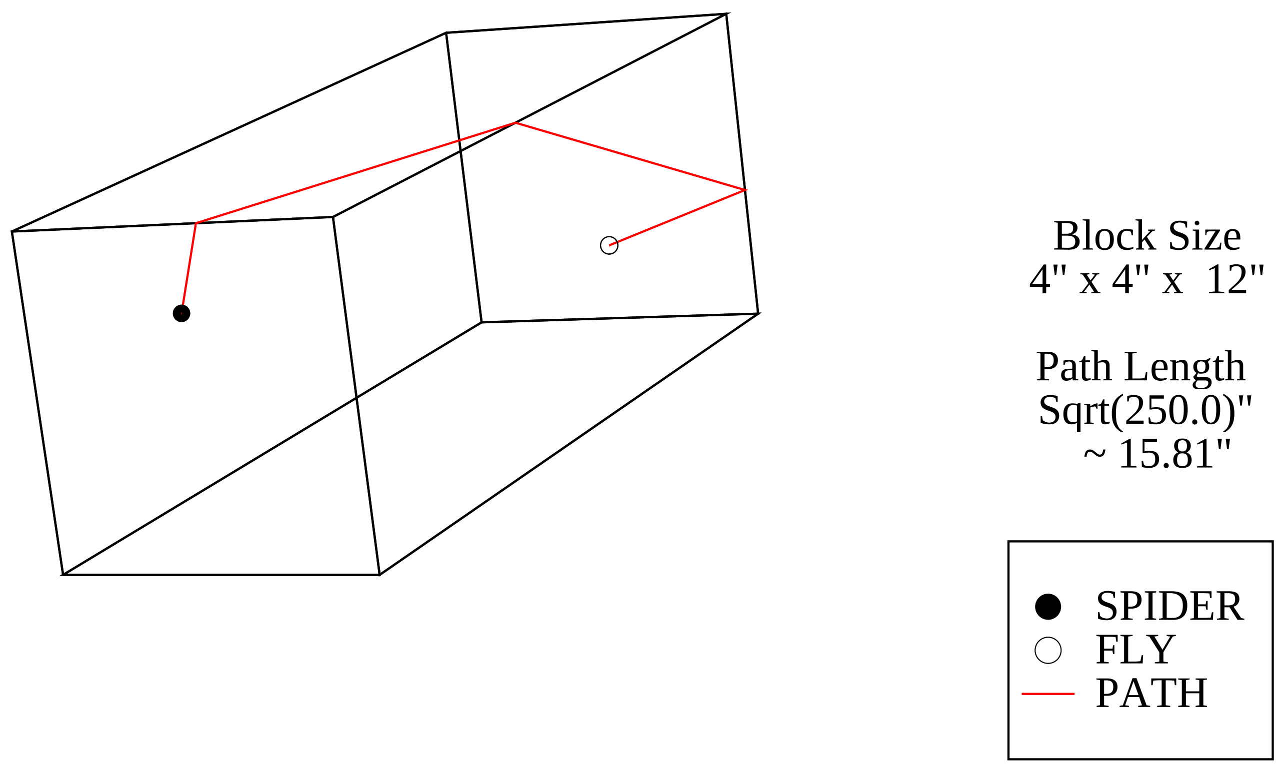
Obvious Local-Minimum Path on Flattened Box.

SPIDER-FLY EXACT SOLUTION



Non-Obvious Global-Minimum on Flattened Box.

SPIDER-FLY EXACT SOLUTION



Non-Obvious Global-Minimum Path between Spider and Fly.

BRACHISTOCHRONE PROBLEM

- GRADIENT & HESSIAN
- BRACHISTOCHRONE
- GRADIENT CALCULATIONS
- SEARCH METHODS
- RESULTS
- SUMMARY

GRADIENT & HESSIAN

Consider the class of optimization problems with cost function

$$I = \int_{x_0}^{x_1} F(x, y, y')\, dx \tag{1}$$

where F is an arbitrary, twice-differentiable function, and $y(x)$ is the trajectory between fixed end points to be optimized.

The first variation of the cost function is

$$\delta I = \int_{x_0}^{x_1} \mathcal{G}\; \delta y\; dx. \tag{2}$$

Under a variation δy, the resulting variation in I is

$$\delta I = \int_{x_0}^{x_1} \left(\frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y' \right) dx.$$

GRADIENT & HESSIAN

Integrating the second term by parts with fixed end points gives

$$\delta I = \int_{x_0}^{x_1} \mathcal{G}(x)\delta y(x)dx$$

where

$$\mathcal{G} = \frac{\partial F}{\partial y} - \frac{d}{dx}\frac{\partial F}{\partial y'}. \tag{3}$$

Also,

$$\delta\mathcal{G} = A\,\delta y$$

where A is the Hessian.

GRADIENT & HESSIAN

The first variation of the gradient can be written as

$$\delta\mathcal{G} = \frac{\partial\mathcal{G}}{\partial y}\delta y + \frac{\partial\mathcal{G}}{\partial y'}\delta y' + \frac{\partial\mathcal{G}}{\partial y''}\delta y''.$$

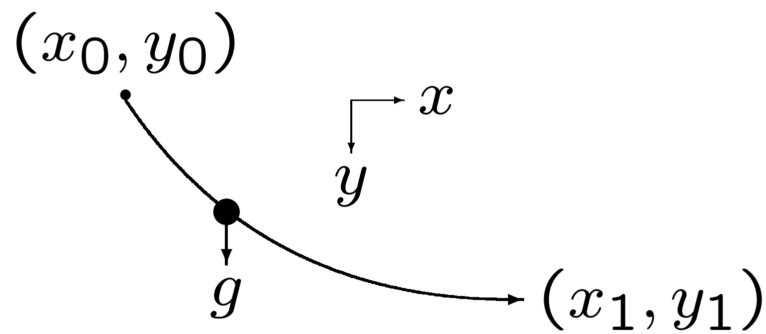
The Hessian can be represented as the local differential operator

$$A = \frac{\partial\mathcal{G}}{\partial y} + \frac{\partial\mathcal{G}}{\partial y'}\frac{d}{dx} + \frac{\partial\mathcal{G}}{\partial y''}\frac{d^2}{dx^2}. \tag{4}$$

One might also represent the Hessian by the integral operator

$$\delta\mathcal{G}(x) = \int_{x_0}^{x_1} a(x,\xi)\ \delta y(x)\ d\xi. \tag{5}$$

BRACHISTOCHRONE



The brachistochrone problem is the determination of path $y(x)$ connecting points (x_0, y_0) and (x_1, y_1) such that the time taken by a particle traversing this path, subject only to the force of gravity, is a minimum. The total time is given by

$$T = \int_{x_0}^{x_1} \frac{ds}{v}$$

where the velocity of a particle falling under the influence of gravity, g, and starting from rest at $y = 0$, is $v = \sqrt{2gy}$.

BRACHISTOCHRONE

Setting $ds = \sqrt{(1 + y'^2)}dx$, one finds that

$$T = \frac{I}{\sqrt{2g}}$$

where

$$I = \int_{x_0}^{x_1} F(y, y')dx \tag{6}$$

with

$$F(y, y') = \sqrt{\frac{1 + y'^2}{y}}.$$

BRACHISTOCHRONE

Under a variation δy, the resulting variation in I is

$$\delta I = \int_{x_0}^{x_1} \left(\frac{\partial F}{\partial y} \delta y + \frac{\partial F}{\partial y'} \delta y' \right) dx.$$

Integrating the second term by parts with fixed end points

$$\delta I = \int_{x_0}^{x_1} \mathcal{G}(x) \delta y(x) dx$$

where

$$\mathcal{G} = \frac{\partial F}{\partial y} - \frac{d}{dx} \frac{\partial F}{\partial y'} = -\frac{\sqrt{1 + y'^2}}{2y^{\frac{3}{2}}} - \frac{d}{dx} \frac{y'}{\sqrt{y(1 + y'^2)}}.$$

BRACHISTOCHRONE

This may be simplified to

$$\mathcal{G} = -\frac{1 + y'^2 + 2yy''}{2(y(1 + y'^2))^{\frac{3}{2}}}. \tag{7}$$

In this case, since F is not a funciton of x,

$$\begin{aligned}\left(y'\frac{\partial F}{\partial y'} - F\right)' &= y''\frac{\partial F}{\partial y'} + y'\frac{d}{dx}\frac{\partial F}{\partial y'} - \frac{\partial F}{\partial y'}y'' - \frac{\partial F}{\partial y}y' \\ &= y'\left(\frac{d}{dx}\frac{\partial F}{\partial y'} - \frac{\partial F}{\partial y}\right) \qquad = -y'\mathcal{G}.\end{aligned}$$

BRACHISTOCHRONE

On the optimal path $\mathcal{G} = 0$ and hence

$$\left(y' \frac{\partial F}{\partial y'} - F \right) \; is \; constant.$$

It follows that $\sqrt{y(1 + y'^2)} = C$, where C is a constant.

The classical solution to the brachistochrone is a cycloid.

$$\begin{aligned} x(t) &= \frac{1}{2} C^2 (t - sin(t)) \\ y(t) &= \frac{1}{2} C^2 (1 - cos(t)) \end{aligned}$$

GRADIENT CALCULATIONS

- **CONTINUOUS GRADIENT**
  - Approximation of the Exact Gradient
- **DISCRETE GRADIENT**
  - Exact Derivative of Discrete Function

CONTINUOUS GRADIENT

The exact continuous gradient of Eqn (7) is approximated by

$$\mathcal{G}_j = -\frac{1 + y_j'^2 + 2y_j y_j''}{2(y_j(1 + y_j'^2))^{\frac{3}{2}}} \tag{8}$$

where

$$y_j' = \frac{y_{j+1} - y_{j-1}}{2\Delta x} \quad , \quad y_j'' = \frac{y_{j+1} - 2y_j + y_{j-1}}{\Delta x^2}.$$

DISCRETE GRADIENT

The exact cost function of Eqn (6) can be approximated by

$$I_R = \sum_{j=0}^{N} F_{j+\frac{1}{2}} \Delta x \tag{9}$$

where

$$F_{j+\frac{1}{2}} = \sqrt{\frac{1 + y'^2_{j+\frac{1}{2}}}{y_{j+\frac{1}{2}}}}$$

$$y_{j+\frac{1}{2}} = \frac{1}{2}(y_{j+1} + y_j) \qquad , \qquad y'_{j+\frac{1}{2}} = \frac{(y_{j+1} - y_j)}{\Delta x}.$$

DISCRETE GRADIENT

Differentiating Eqn (9) gives another approximate form for the gradient as

$$\mathcal{G}_j = \frac{\partial I_R}{\partial y_j} = \mathcal{B}_{j-\frac{1}{2}} - \mathcal{B}_{j+\frac{1}{2}} - \frac{\Delta x}{2}(\mathcal{A}_{j+\frac{1}{2}} + \mathcal{A}_{j-\frac{1}{2}}) \tag{10}$$

where

$$\mathcal{A}_{j+\frac{1}{2}} = \frac{\sqrt{1 + y'^2_{j+\frac{1}{2}}}}{2y^{\frac{3}{2}}_{j+\frac{1}{2}}} \quad , \quad \mathcal{B}_{j+\frac{1}{2}} = \frac{y'_{j+\frac{1}{2}}}{\sqrt{y_{j+\frac{1}{2}}(1 + y'^2_{j+\frac{1}{2}})}}.$$

SEARCH METHODS

- **STEEPEST DESCENT**
- **SMOOTHED STEEPEST DESCENT**
- **IMPLICIT DESCENT**
- **MULTIGRID DESCENT**
- **KRYLOV ACCELERATION**
- **QUASI-NEWTON METHODS**
  - Rank 1
  - Davidon-Fletcher-Powell (DFP)
  - Broyden-Fanno-Goldfarb-Shannon (BFGS)

STEEPEST DESCENT

Forward Euler step gives

$$y_j^{n+1} = y_j^n - \lambda \mathcal{G}_j^n \qquad , \qquad \lambda > 0$$

$$\delta y^n = -\lambda \mathcal{G}^n.$$

Then to first order the variation in I is

$$\delta I = \int_{x_0}^{x_1} \mathcal{G} \delta y dx = -\lambda \int_{x_0}^{x_1} \mathcal{G}^2 dx$$

and

$$\delta I \leq 0.$$

STEEPEST DESCENT

This may be regarded as a forward Euler discretization of a time dependent process with $\lambda = \Delta t$. Hence, $\frac{\partial y}{\partial t} = -\mathcal{G}$. Substituting for $\mathcal{G}$ from Eqn (7), y solves the nonlinear parabolic equation

$$\frac{\partial y}{\partial t} = \frac{1 + y'^2 + 2yy''}{2(y(1 + y'^2))^{\frac{3}{2}}}. \tag{11}$$

The time step limit for stable integration is dominated by the parabolic term $\beta y''$, where $\beta = \frac{y}{(y(1+y'^2))^{\frac{3}{2}}}$.

This gives the following estimate on the time step limit.

$$\Delta t^{\star} = \frac{\Delta x^2}{2\beta}.$$

SMOOTHED STEEPEST DESCENT

Define $\bar{\mathcal{G}}$ with the implicit smoothing equation.

$$\bar{\mathcal{G}} - \frac{\partial}{\partial x}\epsilon\frac{\partial\bar{\mathcal{G}}}{\partial x} = \mathcal{G} \tag{12}$$

Now set

$$\delta y = -\lambda\bar{\mathcal{G}}. \tag{13}$$

Then to first order the variation in I is

$$\delta I = \int_{x_0}^{x_1} \mathcal{G}\delta y dx = -\lambda\int_{x_0}^{x_1}\left(\bar{\mathcal{G}} - \frac{\partial}{\partial x}\epsilon\frac{\partial\bar{\mathcal{G}}}{\partial x}\right)\bar{\mathcal{G}}dx.$$

SMOOTHED STEEPEST DESCENT

Integrating by parts and noting that the end points are fixed,

$$\delta I = -\lambda \int_{x_0}^{x_1} \left(\bar{\mathcal{G}}^2 + \epsilon \left(\frac{\partial \bar{\mathcal{G}}}{\partial x} \right)^2 \right) dx.$$

Again,

$$\delta I \leq 0.$$

IMPLICIT DESCENT

If the gradient is dominated by a y'' term, the smoothed descent given by Eqn (13) can be made equivalent to an implicit scheme.

Consider the parabolic equation, $\frac{\partial y}{\partial t} = \beta \frac{\partial^2 y}{\partial x^2}$, where β is variable.

The system for an implicit scheme is

$$-\alpha \delta y_{j-1} + (1 + 2\alpha)\delta y_j - \alpha \delta y_{j+1} = -\Delta t \widehat{\mathcal{G}}_j \quad (14)$$

where δy_j is the correction to y_j,

$$\alpha = \frac{\beta \Delta t}{\Delta x^2} = \frac{\Delta t}{2\Delta t^\star} \quad (15)$$

IMPLICIT DESCENT

and

$$\hat{\mathcal{G}}_j = \frac{\beta}{\Delta x^2}(y^n_{j-1} - 2y^n_j + y^n_{j+1}).$$

Combining Eqns (12 & 13), the discrete smoothed descent method assumes the form of Eqn (14) with

$$\alpha = \frac{\epsilon}{\Delta x^2}. \tag{16}$$

Comparing Eqn (15) with Eqn (16), one can see using the smoothed gradient is equivalent to an implicit time stepping scheme if $\epsilon = \beta\Delta t$. Furthermore, a Newton iteration is recovered as $\Delta t \rightarrow \infty$.

MULTIGRID DESCENT

Consider a sequence of K meshes, generated by eliminating alternate points along each coordinate direction of mesh-level k to produce mesh-level $k+1$. Note that $k=1$ refers to the finest mesh of the sequence. In order to give a precise description of the multigrid scheme, subscripts may be used to indicate grid level. Several transfer operations need to be defined. First, the solution vector, y, on grid k must be initialized as

$$y_k^{(0)} = T_{k,k-1}\, y_{k-1} \quad , \quad 2 \leq k \leq K$$

where y_{k-1} is the current value of the solution on grid $k-1$, and $T_{k,k-1}$ is a transfer operator.

MULTIGRID DESCENT

It is also necessary to transfer a residual forcing function, P, such that the solution on grid k is driven by the residuals of grid $k-1$. This can be accomplished by setting

$$P_k = Q_{k,k-1}\, \mathcal{G}_{k-1}(y_{k-1}) - \mathcal{G}_k(y_k^{(0)}),$$

where $Q_{k,k-1}$ is another transfer operator. Now, $\mathcal{G}_k$ is replaced by $\mathcal{G}_k + P_k$ in the time-stepping such that

$$y_k^{+} = y_k^{(0)} - \Delta t_k \left[\mathcal{G}_k(y_k) + P_k\right]$$

where the superscript $^{+}$ denotes the updated value. The resulting solution vector, y_k^{+}, provides the initial data for grid $k+1$.

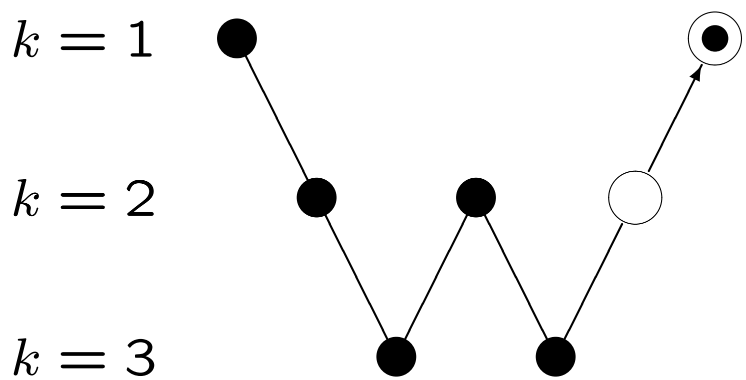
MULTIGRID DESCENT

Finally, the accumulated correction on grid k is transferred back to grid $k-1$ with the aid of an interpolation operator, $I_{k-1,k}$. Thus one sets

$$y_{k-1}^{++} = y_{k-1}^{+} + I_{k-1,k} \left(y_k^{++} - y_k^{(0)} \right)$$

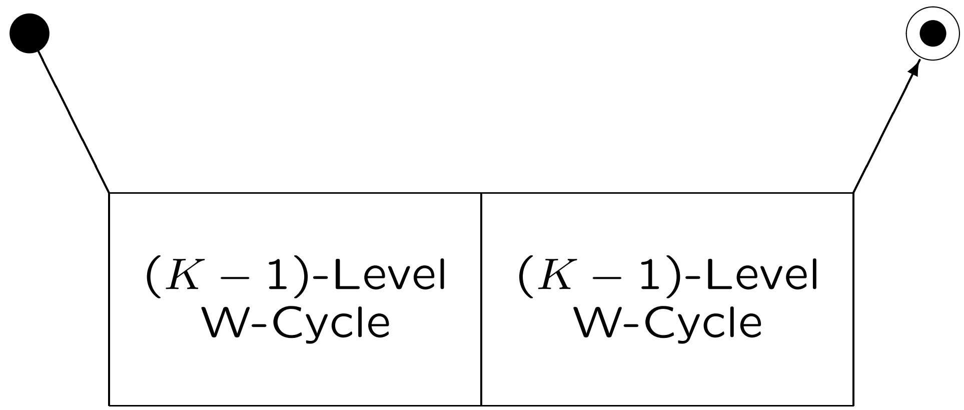
where the superscript $^{++}$ denotes the result of both the time step on grid k and the interpolated correction from grid $k+1$.

MULTIGRID DESCENT



Three-Level Multigrid W-Cycle

MULTIGRID DESCENT



Recursive Stencil for a
K-Level Multigrid W-Cycle

MULTIGRID DESCENT

In a three-dimensional setting, the number of cells is reduced by a factor of 8 on each coarser grid. By examination of the stencils, it can be verified that the work of one multigrid W-Cycle, in work units, is on the order of

$$1 + \frac{2}{8} + \frac{4}{64} + ... + \frac{1}{4^K} < \frac{4}{3}.$$

Hence, one multigrid W-Cycle only requires about $\frac{1}{3}$ more effort as that required for a fine-mesh iteration.

KRYLOV ACCELERATION

Given K linearly independent $(y, \mathcal{G})$ vectors, one can survey the K-dimensional subspace spanned by these vectors.

$$y^\star = \sum_{k=1}^{K} \gamma_k y^k \quad , \quad \mathcal{G}^\star = \sum_{k=1}^{K} \gamma_k \mathcal{G}^k \quad , \quad \sum_{k=1}^{K} \gamma_k = 1$$

Minimize the L_2 Norm of $\mathcal{G}^\star$ to determine the recombination coefficients γ_k.

Now,

$$y_j^{n+1} = y_j^\star - \lambda \mathcal{G}_j^\star$$

QUASI-NEWTON METHODS

Quasi-Newton methods estimate the Hessian, A, or its inverse A^{-1}, from changes $\delta\mathcal{G}$ in the gradient during the search steps. By the definition of A, to first order

$$\delta\mathcal{G} = A\delta y$$

Let H^n be an estimate of A^{-1} at the n^{th} step. Then it should be required to satisfy

$$H^n\delta\mathcal{G}^n = \delta y^n$$

This can be satisfied by various recursive formulas for H.

QUASI-NEWTON METHODS

Rank 1

$$H^{n+1} = H^n + \frac{\mathcal{P}^n(\mathcal{P}^n)^T}{(\mathcal{P}^n)^T\delta\mathcal{G}^n}$$

where

$$\mathcal{P}^n = \delta y^n - H^n\delta\mathcal{G}^n$$

QUASI-NEWTON METHODS

Davidon-Fletcher-Powell (DFP)

$$H^{n+1} = H^n + \frac{\delta y^n (\delta y^n)^T}{(\delta y^n)^T \delta \mathcal{G}^n} - \frac{H^n \delta \mathcal{G}^n (\delta \mathcal{G}^n)^T H^n}{(\delta \mathcal{G}^n)^T H^n \delta \mathcal{G}^n}$$

QUASI-NEWTON METHODS

Broyden-Fanno-Goldfarb-Shannon (BFGS)

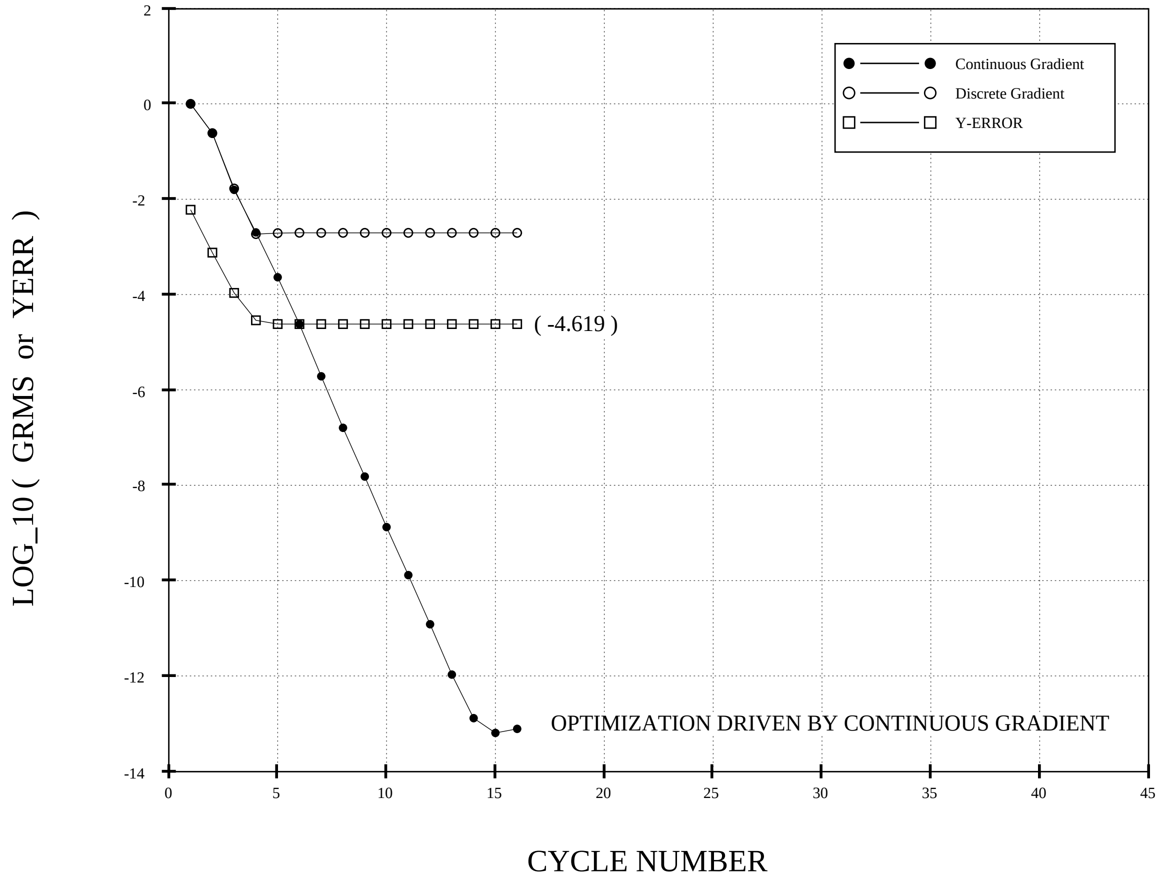
$$H^{n+1} \;=\; H^n + \left(1 + \frac{(\delta\mathcal{G}^n)^T H^n \delta\mathcal{G}^n}{(\delta\mathcal{G}^n)^T \delta y^n}\right) \frac{\delta y^n (\delta y^n)^T}{(\delta\mathcal{G}^n)^T \delta y^n}$$

$$- \frac{H^n \delta\mathcal{G}^n (\delta y^n)^T \;+\; \delta y^n (\delta\mathcal{G}^n)^T H^n}{(\delta\mathcal{G}^n)^T \delta y^n}$$

RESULTS

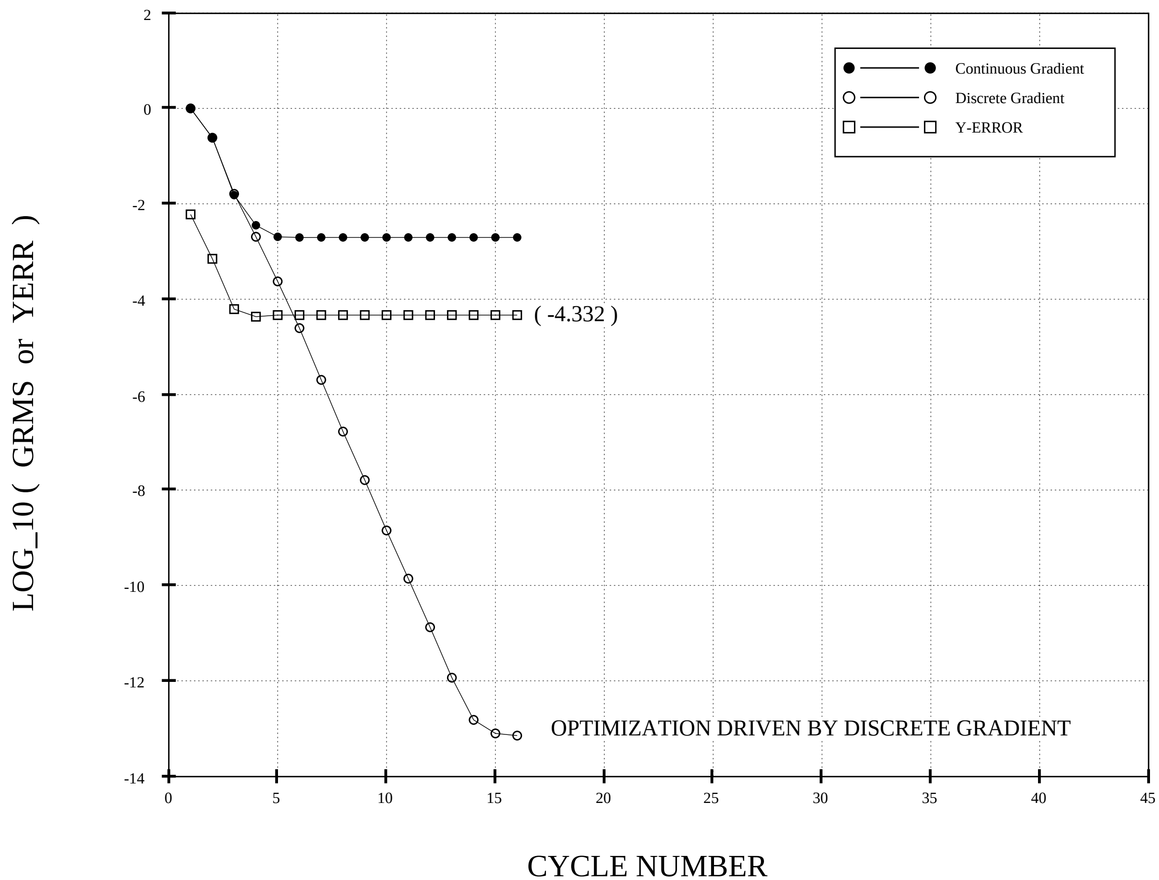
- **ACCURACY OF GRADIENTS**
  - Continuous vs. Discrete
  - Level of Accuracy
  - Order of Accuracy
- **PERFORMANCE OF SEARCH METHODS**
  - Build-up of Explicit Schemes
  - Comparison with Implicit Scheme
  - Grid-Independent Convergence
  - Tested with up to 8192 Design Variables
- **ROBUSTNESS**

ACCURACY: CONTINUOUS GRADIENT



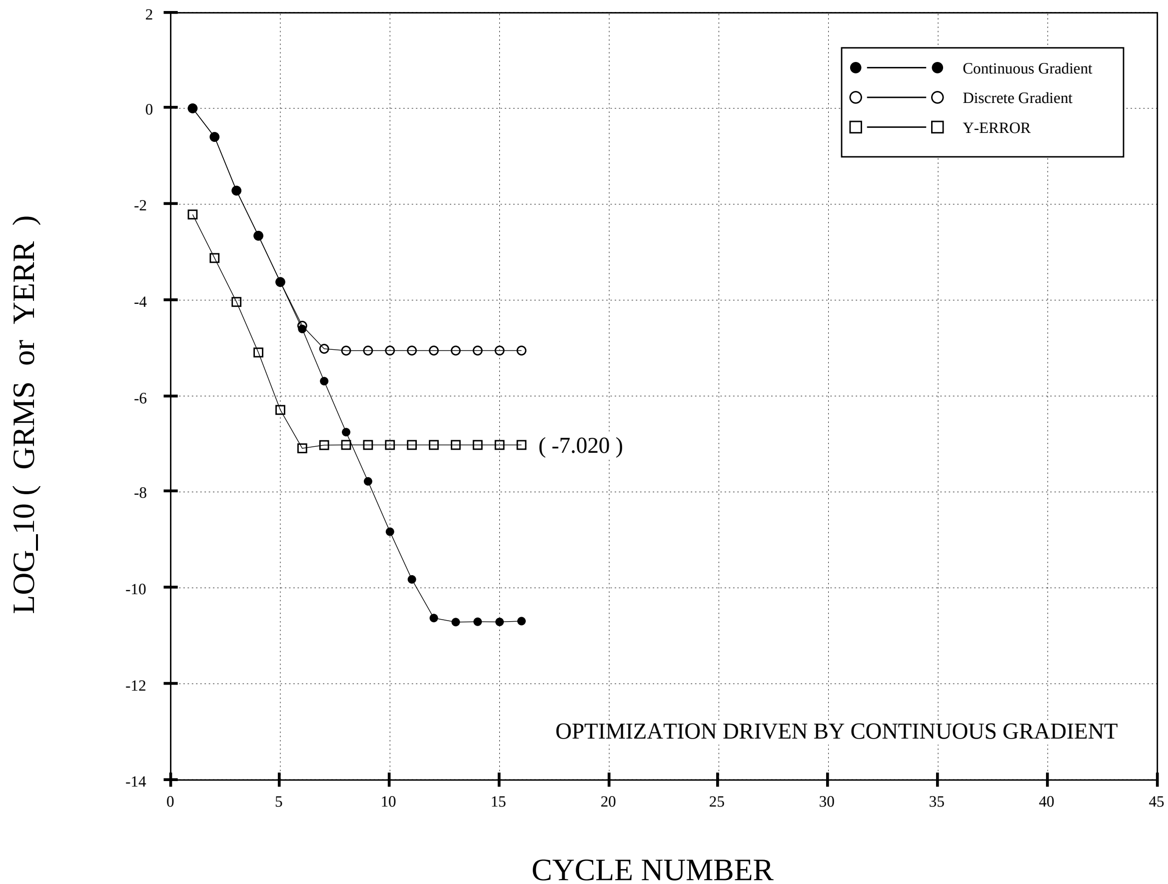
Convergence of continuous gradient, implicit scheme, N=31.

ACCURACY: DISCRETE GRADIENT



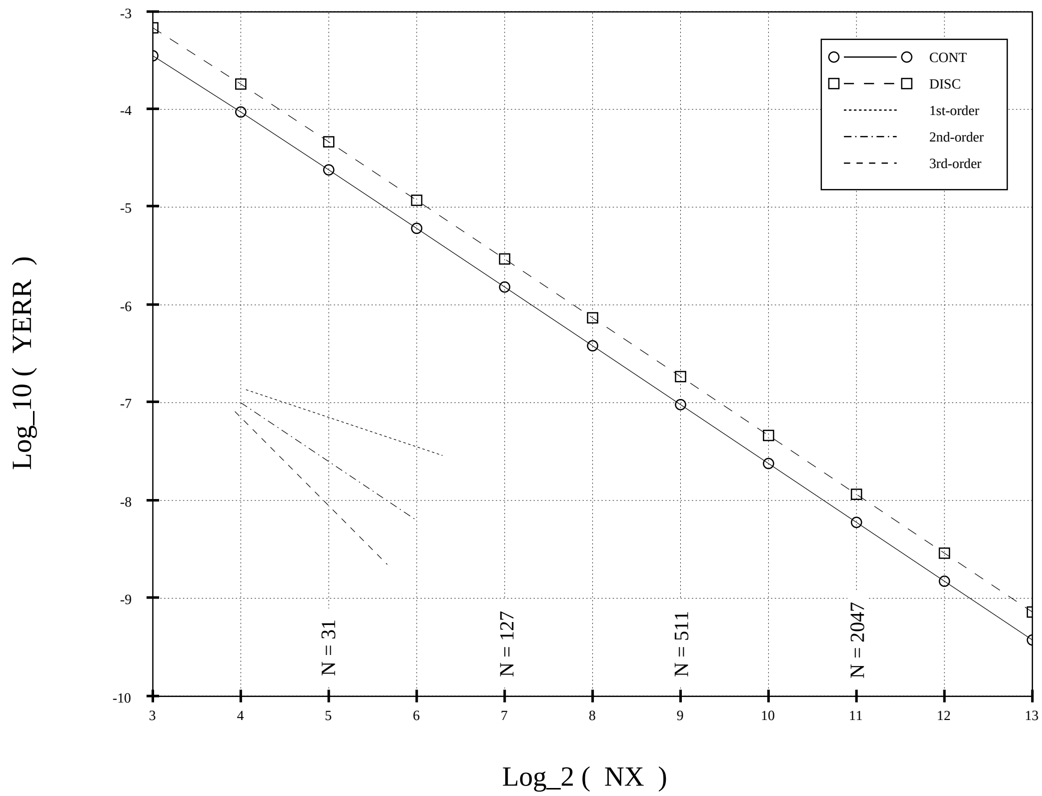
Convergence of discrete gradient, implicit scheme, N=31.

ACCURACY: CONTINUOUS GRADIENT



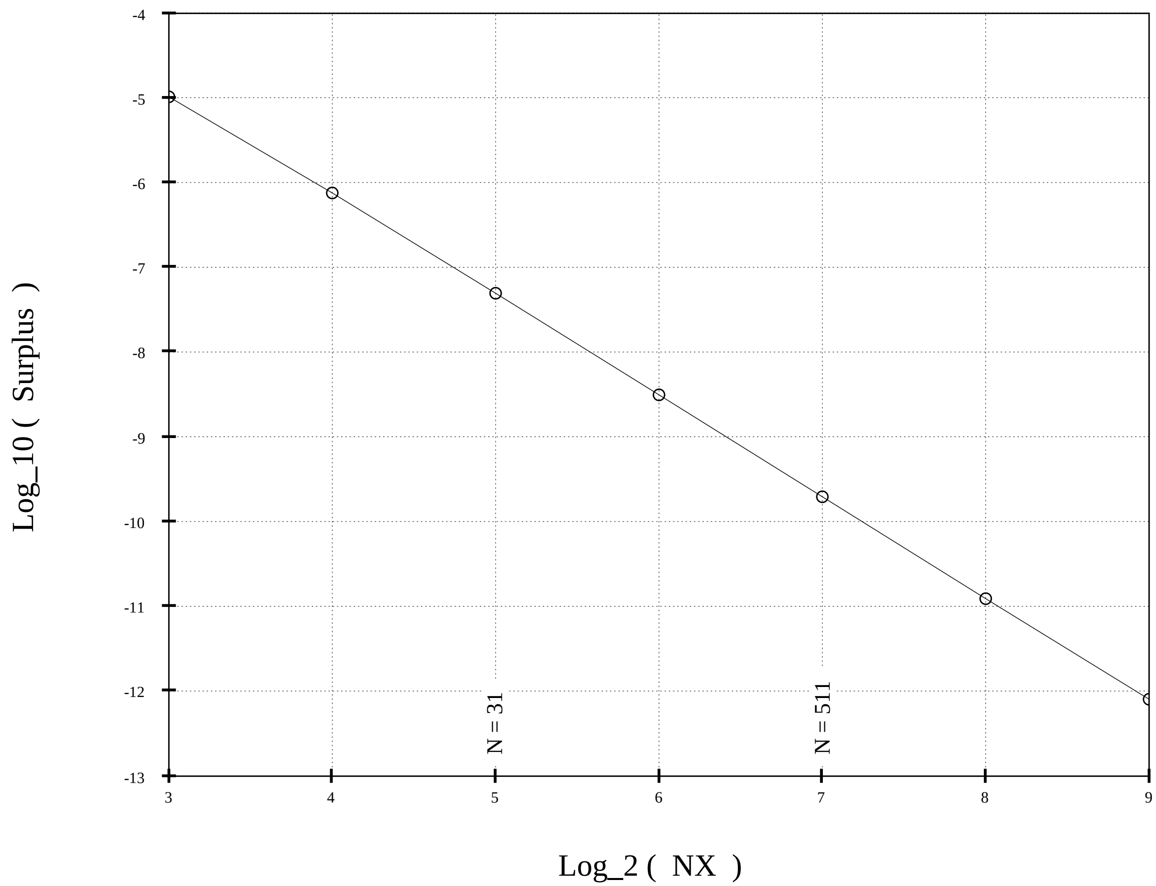
Convergence of continuous gradient, implicit scheme, N=511.

ACCURACY: CONTINUOUS vs. DISCRETE



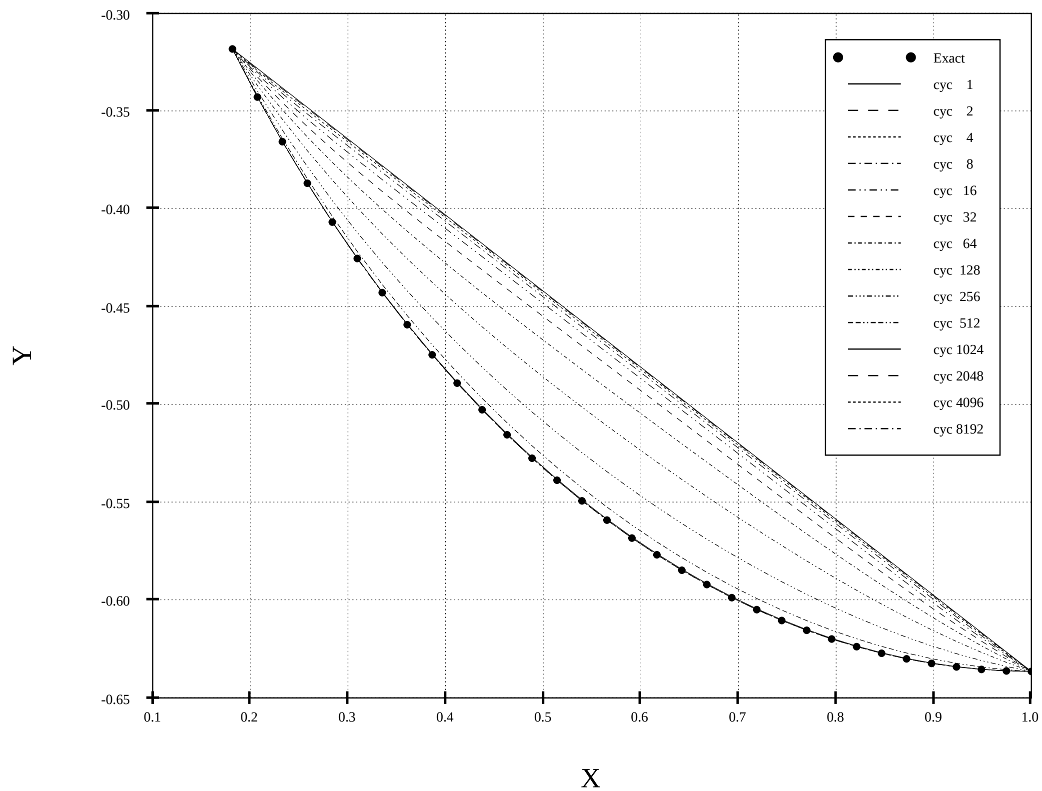
Computed path errors as a function of mesh size.

ACCURACY: SURPLUS COST



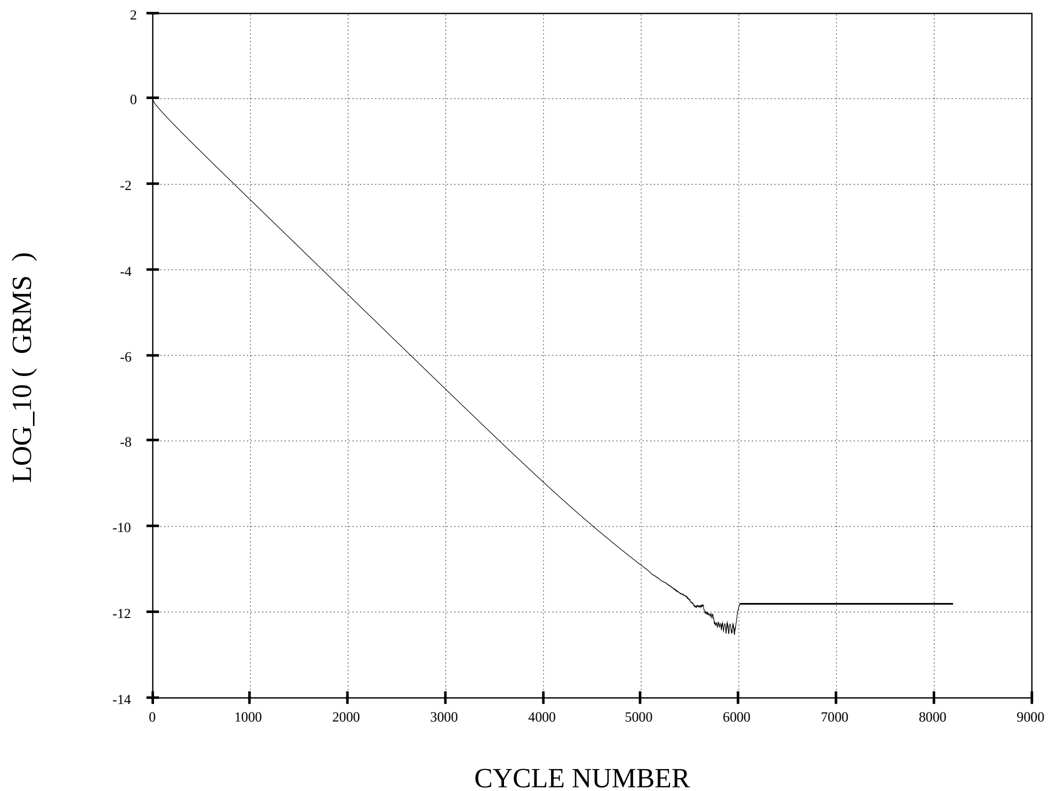
Difference of measurable cost function between gradients.

PERFORMANCE: STEEPEST DESCENT



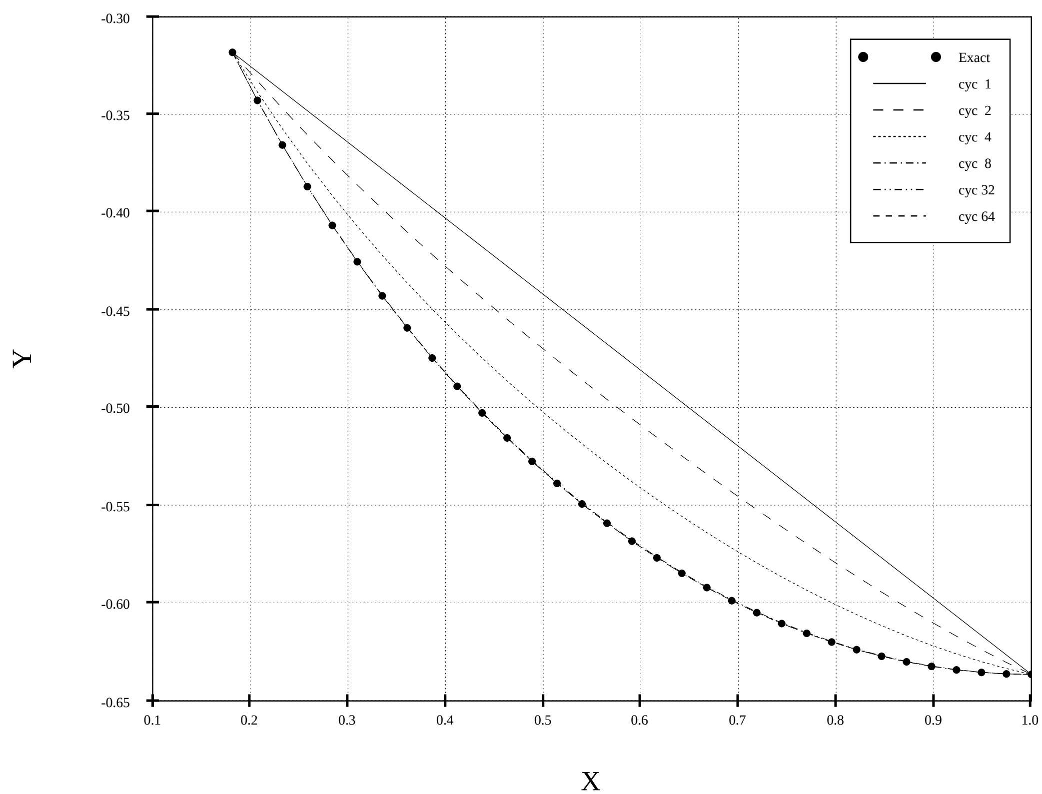
History of paths of steepest descent, N=31.

PERFORMANCE: STEEPEST DESCENT



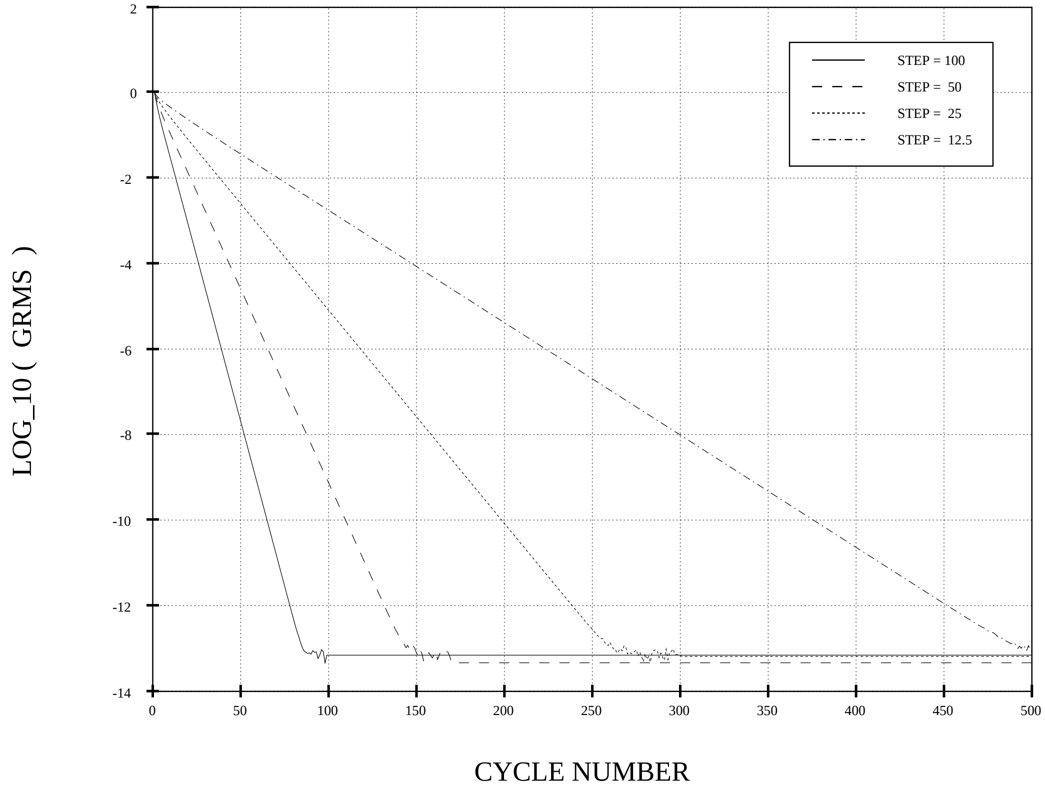
Convergence history of steepest descent, N=31.

PERFORMANCE: SMOOTHED DESCENT



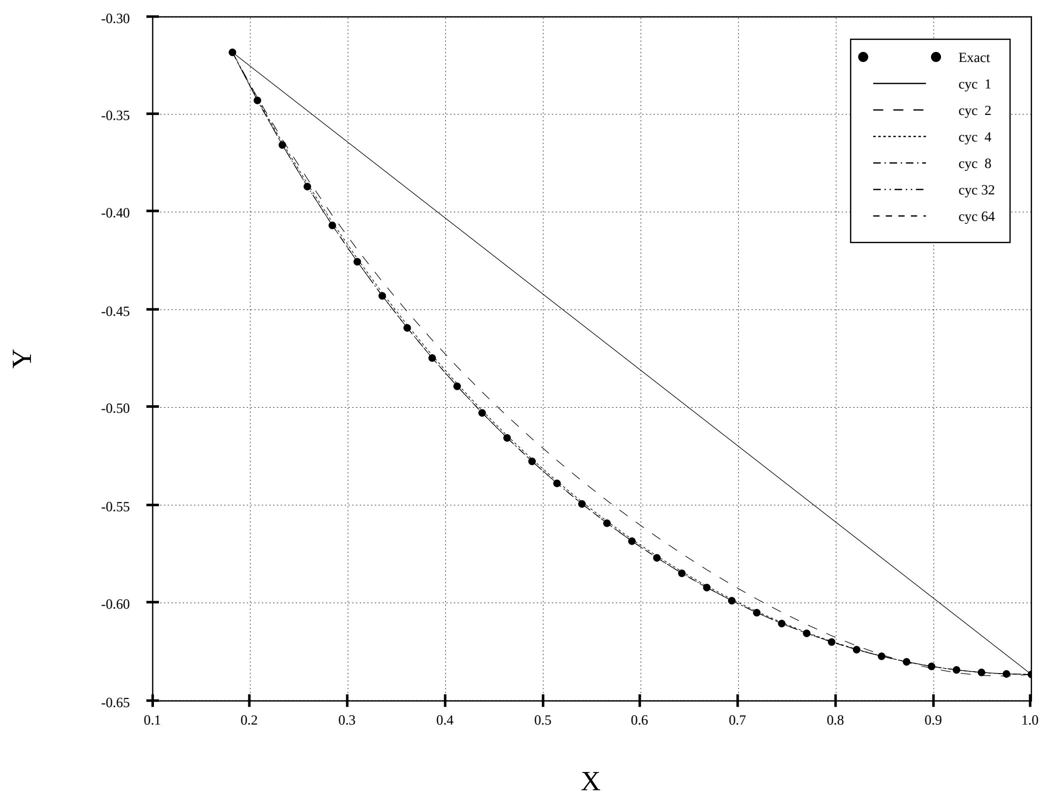
History of paths of smoothed descent, N=31 & STEP=100.

PERFORMANCE: SMOOTHED DESCENT



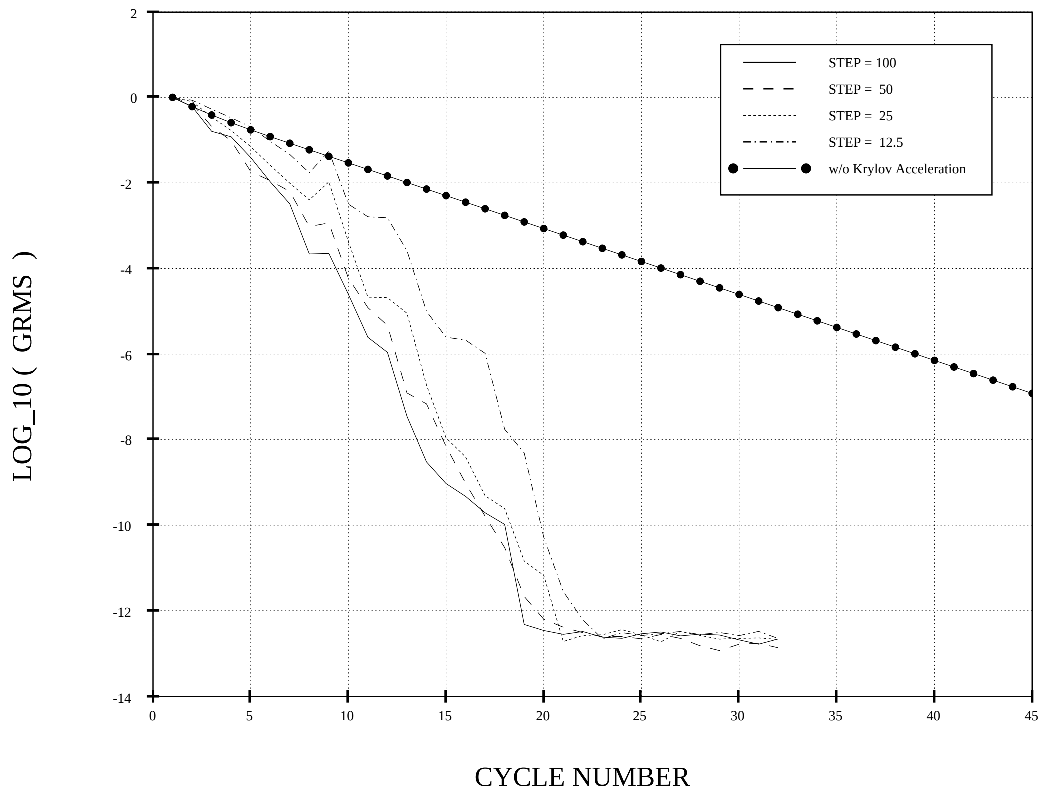
Convergence history of smoothed descent, N=31.

PERFORMANCE: KRYLOV ACCELERATION



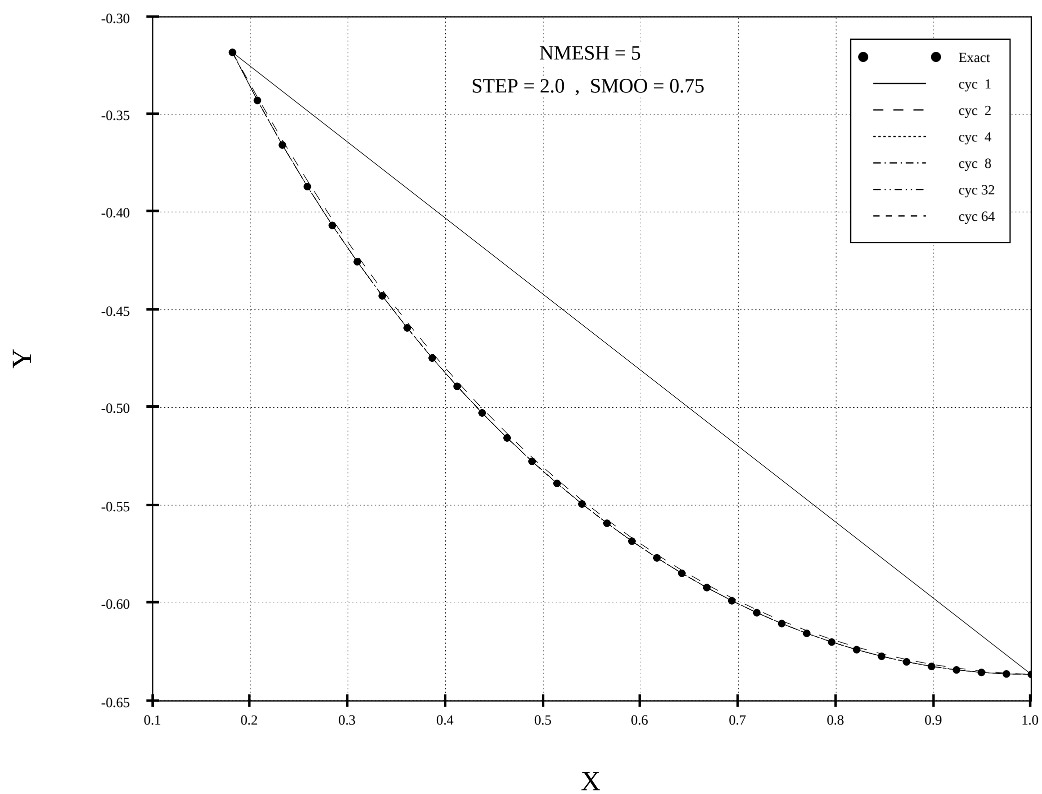
History of paths for Krylov acceleration, N=31 & STEP=100.

PERFORMANCE: KRYLOV ACCELERATION



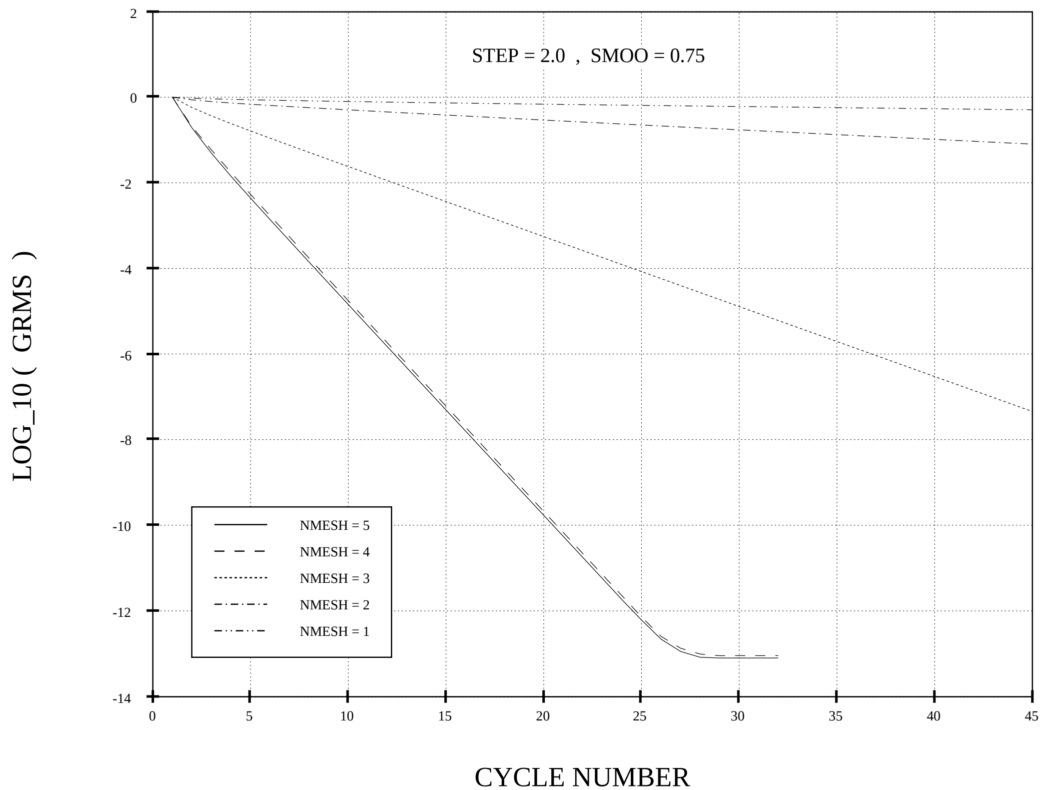
Convergence history of Krylov acceleration, N=31.

PERFORMANCE: MULTIGRID DESCENT



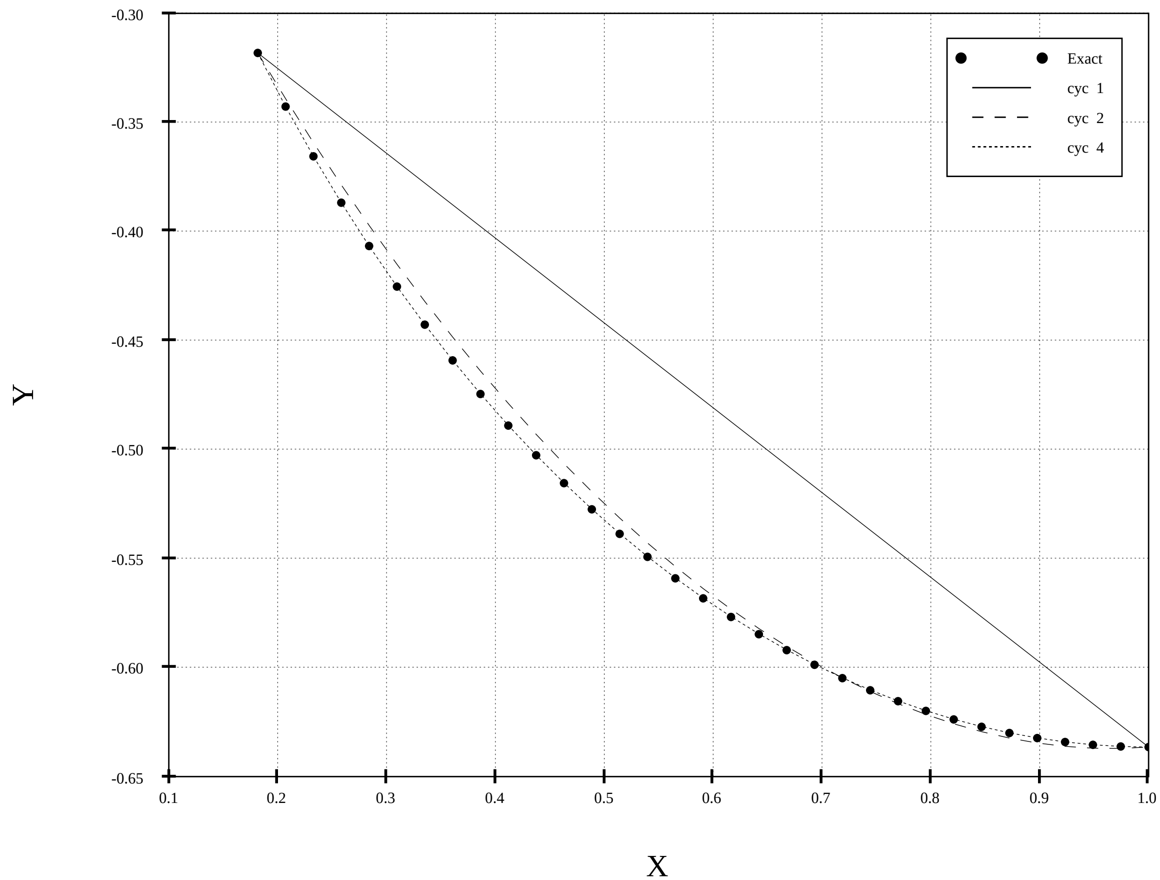
History of paths for multigrid acceleration, N=31.

PERFORMANCE: MULTIGRID DESCENT



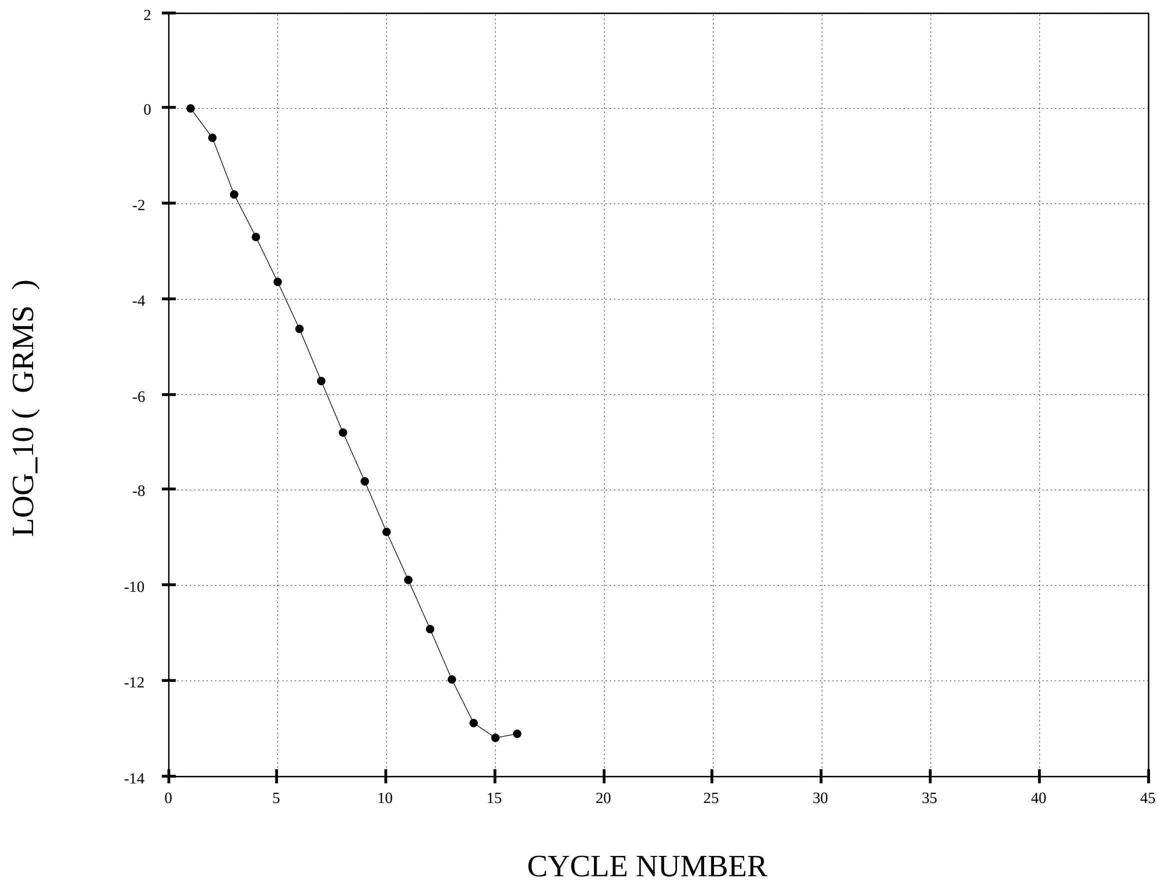
Convergence history of multigrid acceleration, N=31.

PERFORMANCE: IMPLICIT DESCENT



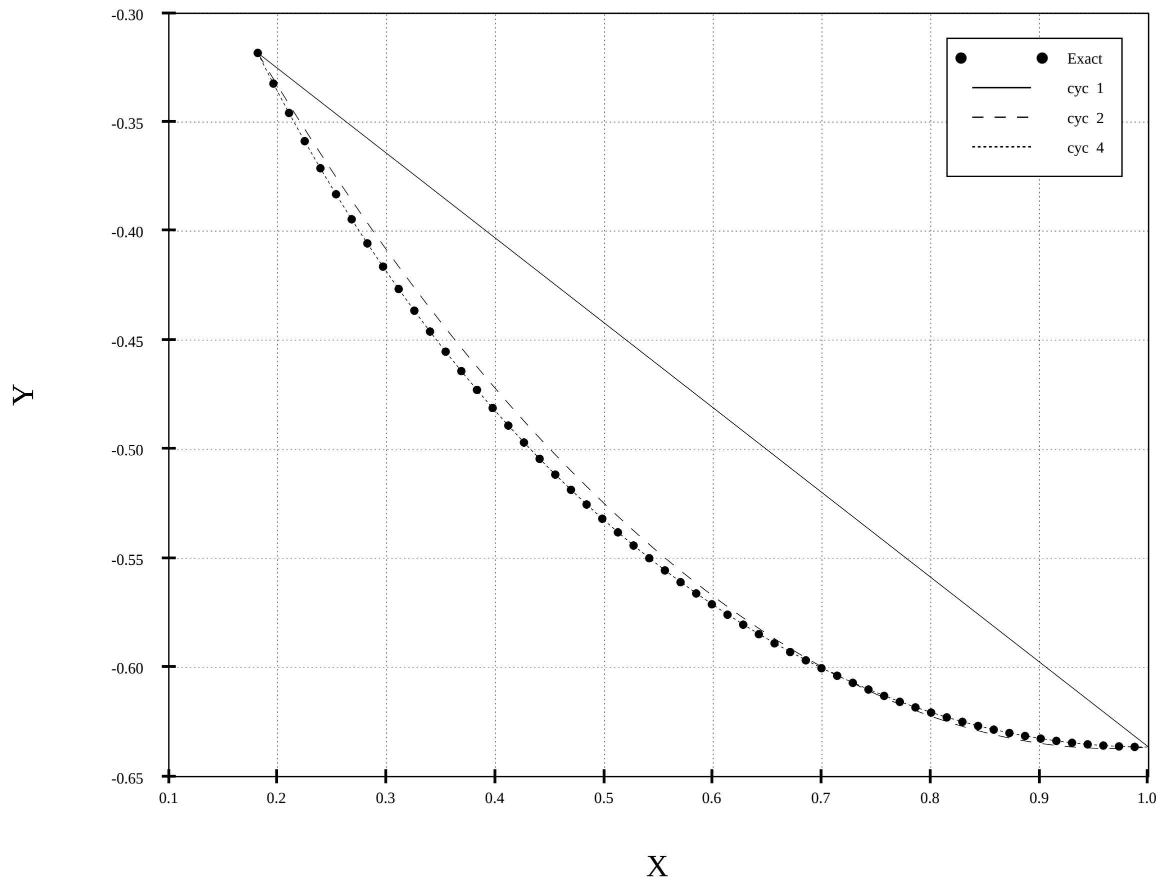
History of paths of implicit stepping, N=31.

PERFORMANCE: IMPLICIT DESCENT



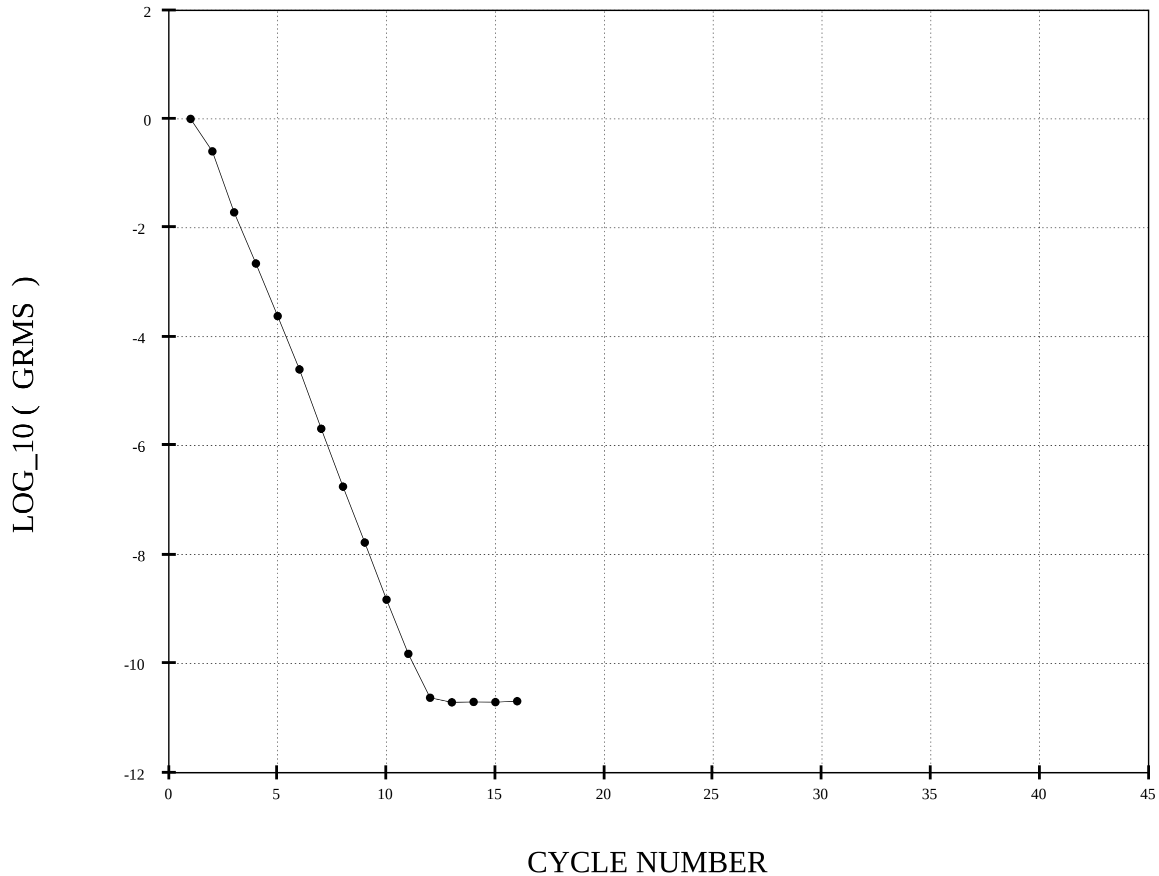
Convergence history of implicit stepping, N=31.

PERFORMANCE: IMPLICIT DESCENT



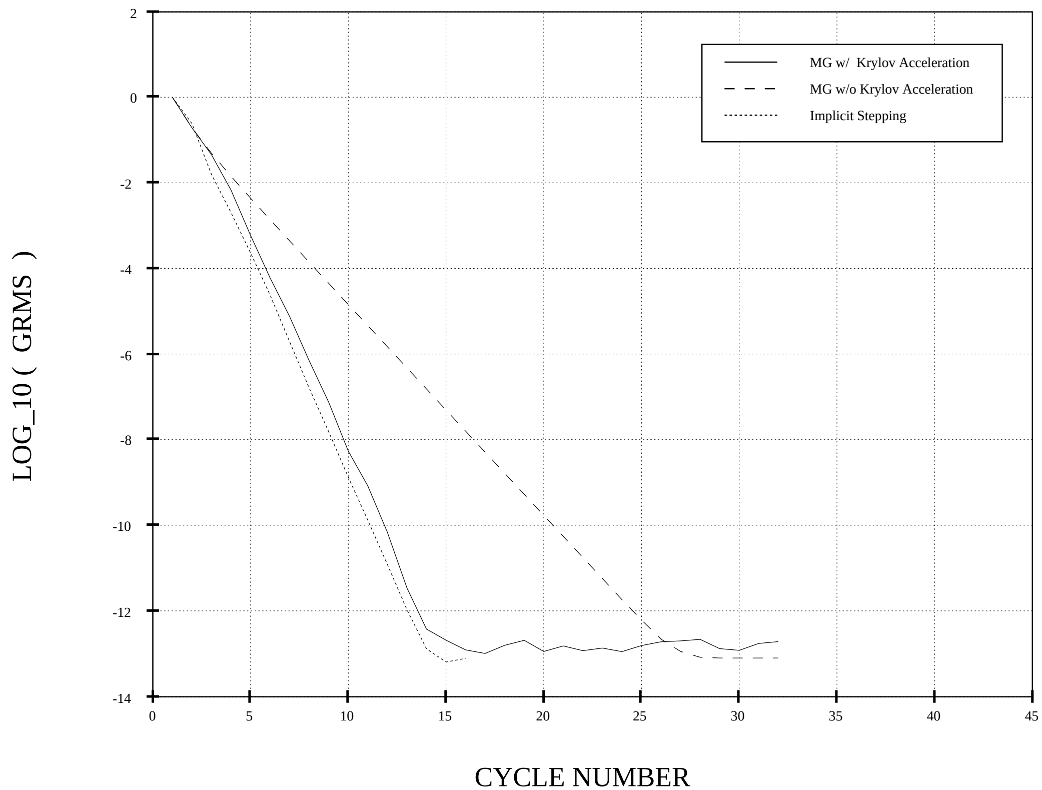
History of paths of implicit stepping, N=511.

PERFORMANCE: IMPLICIT DESCENT



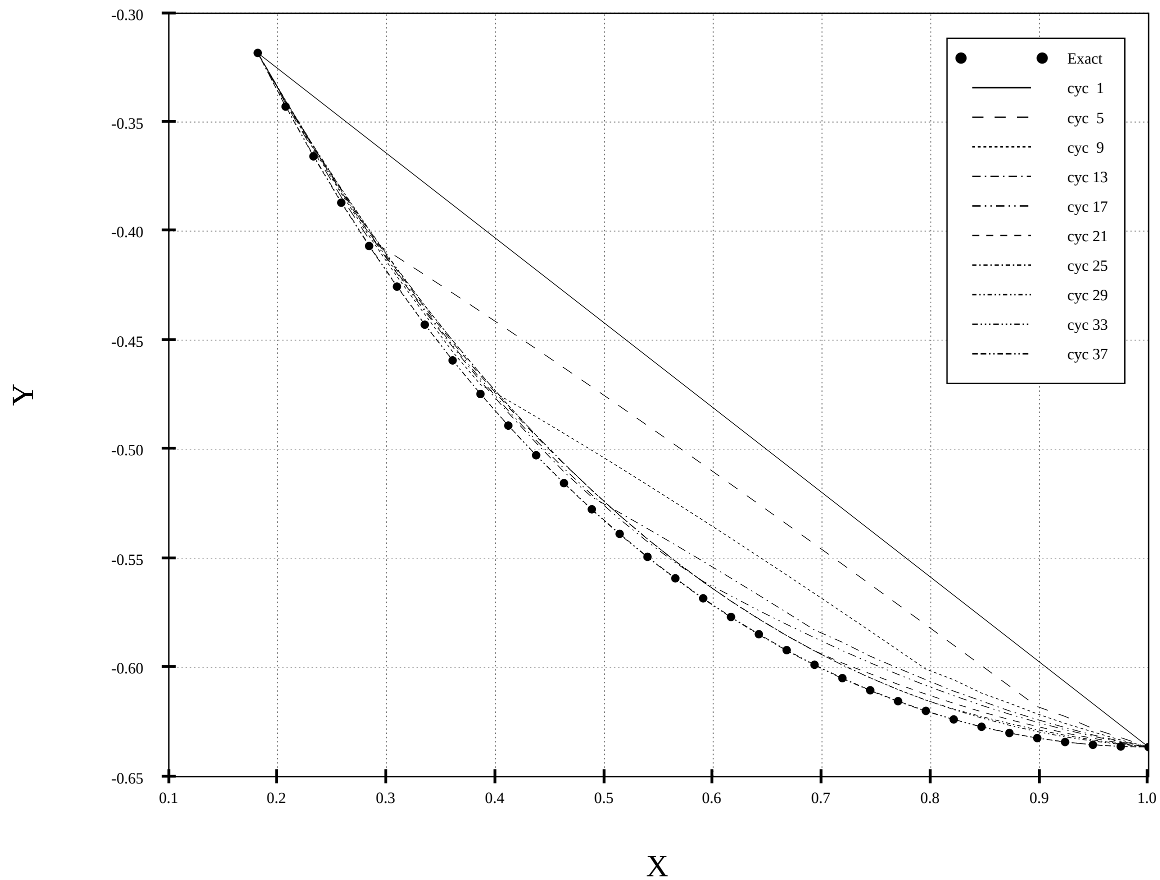
Convergence history of implicit stepping, N=511.

PERFORMANCE: MULTIGRID vs. IMPLICIT



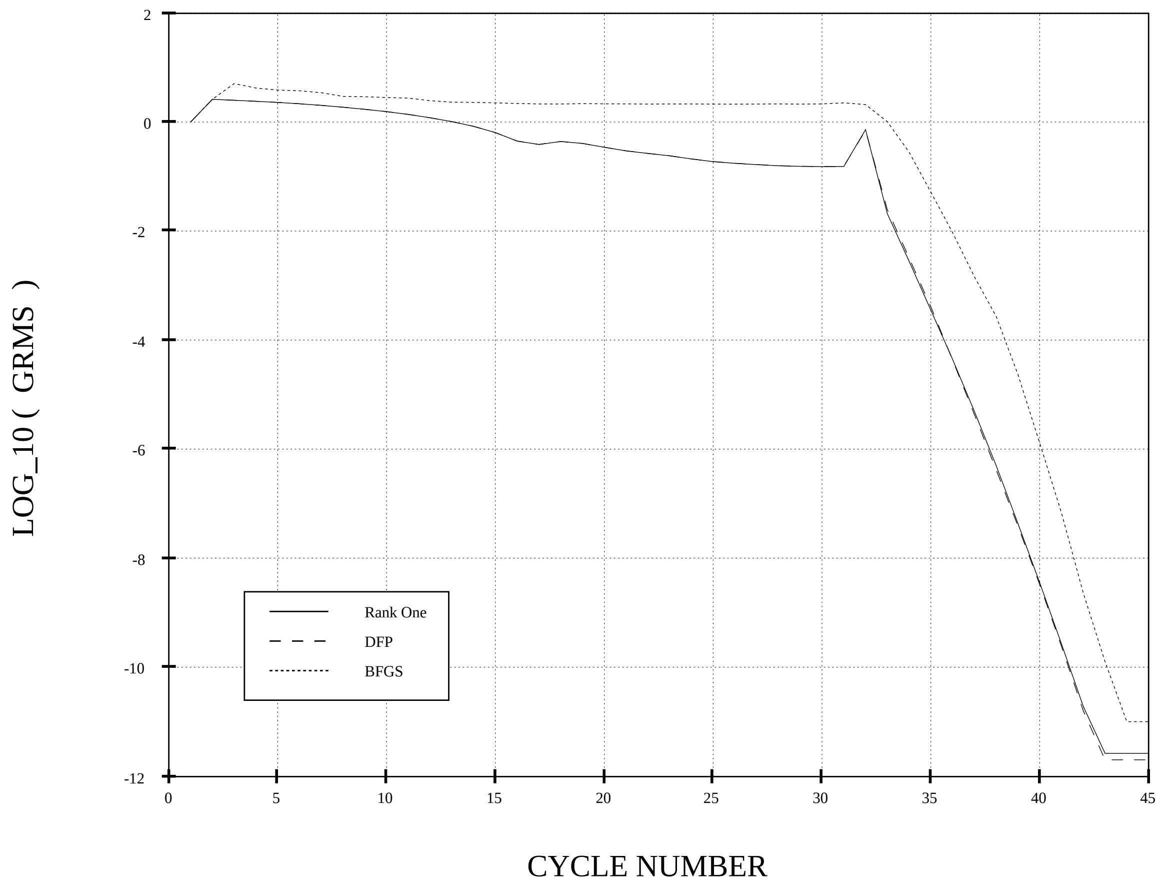
Comparison of grid-independent convergence histories.

PERFORMANCE: QUASI-NEWTON



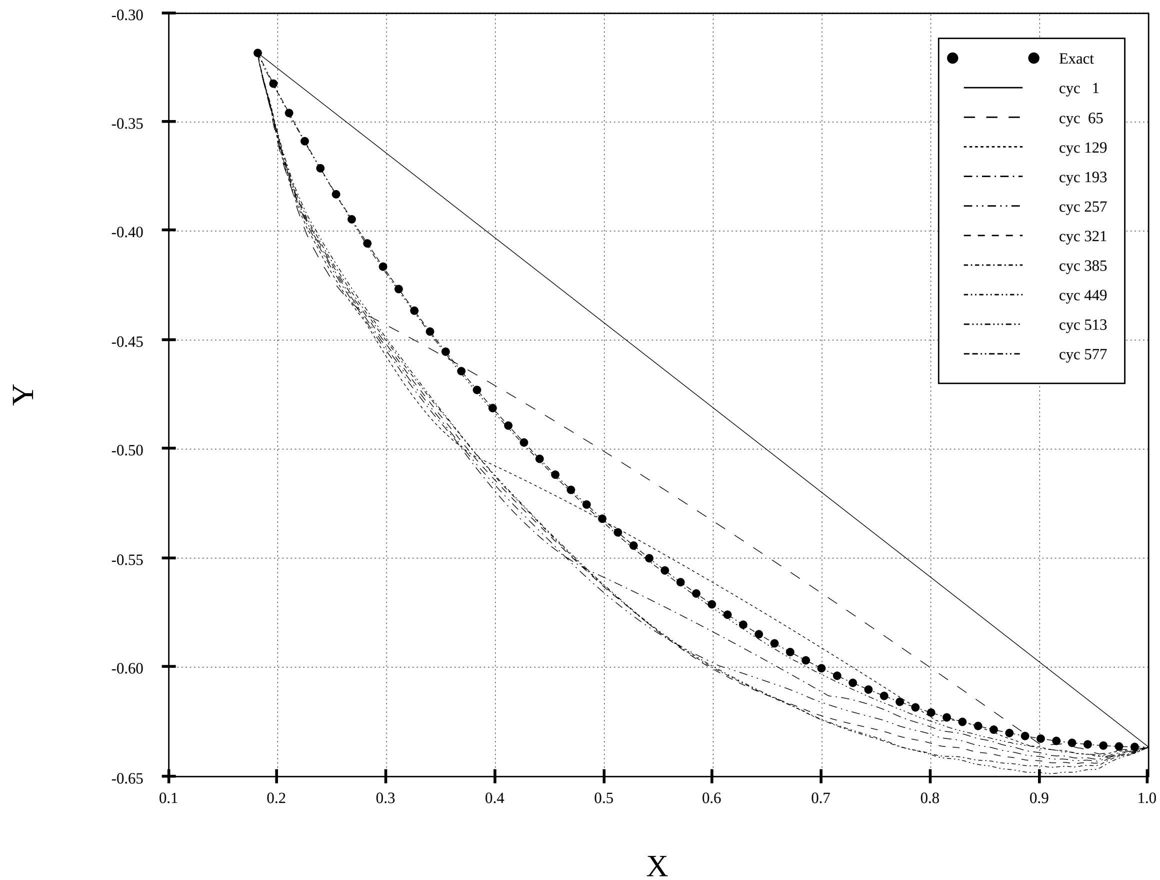
History of paths for Rank-1 quasi-Newton, N=31.

PERFORMANCE: QUASI-NEWTON



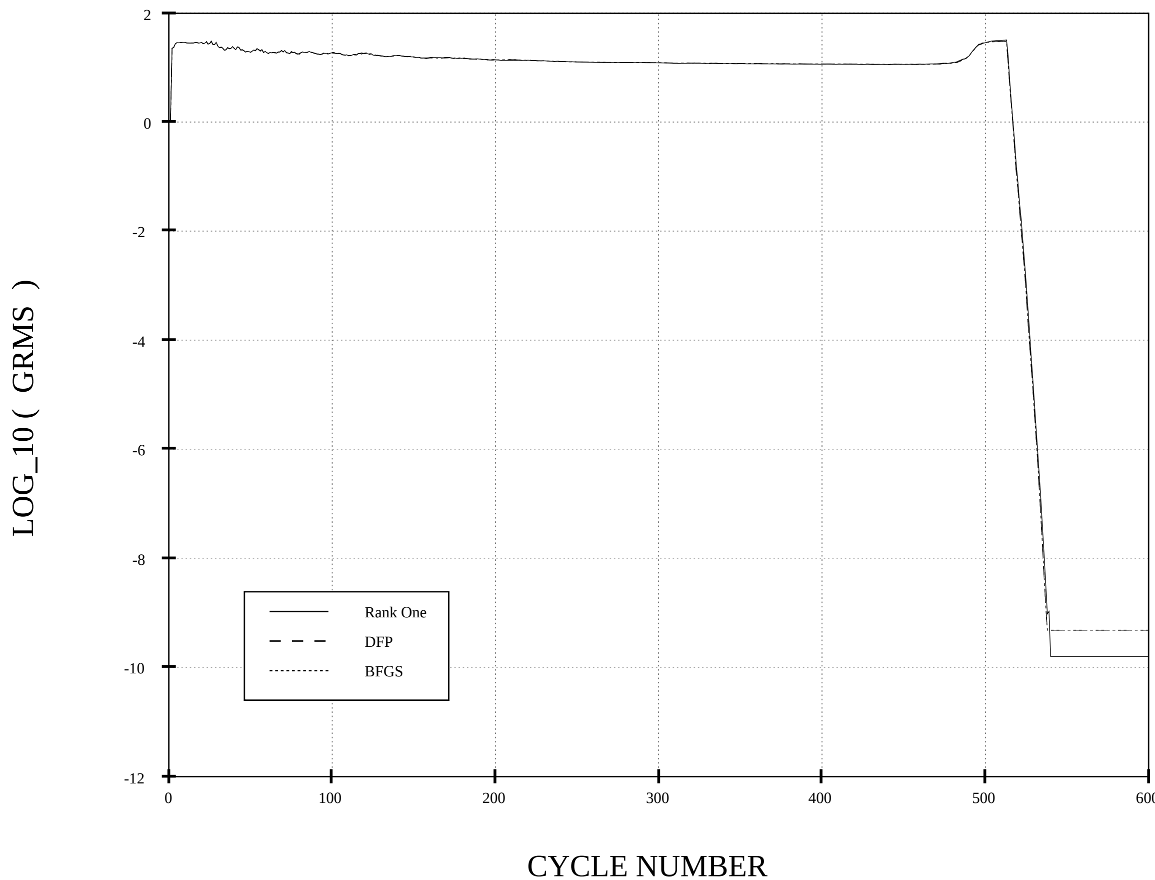
Comparison of quasi-Newton convergence histories, N=31.

PERFORMANCE: QUASI-NEWTON



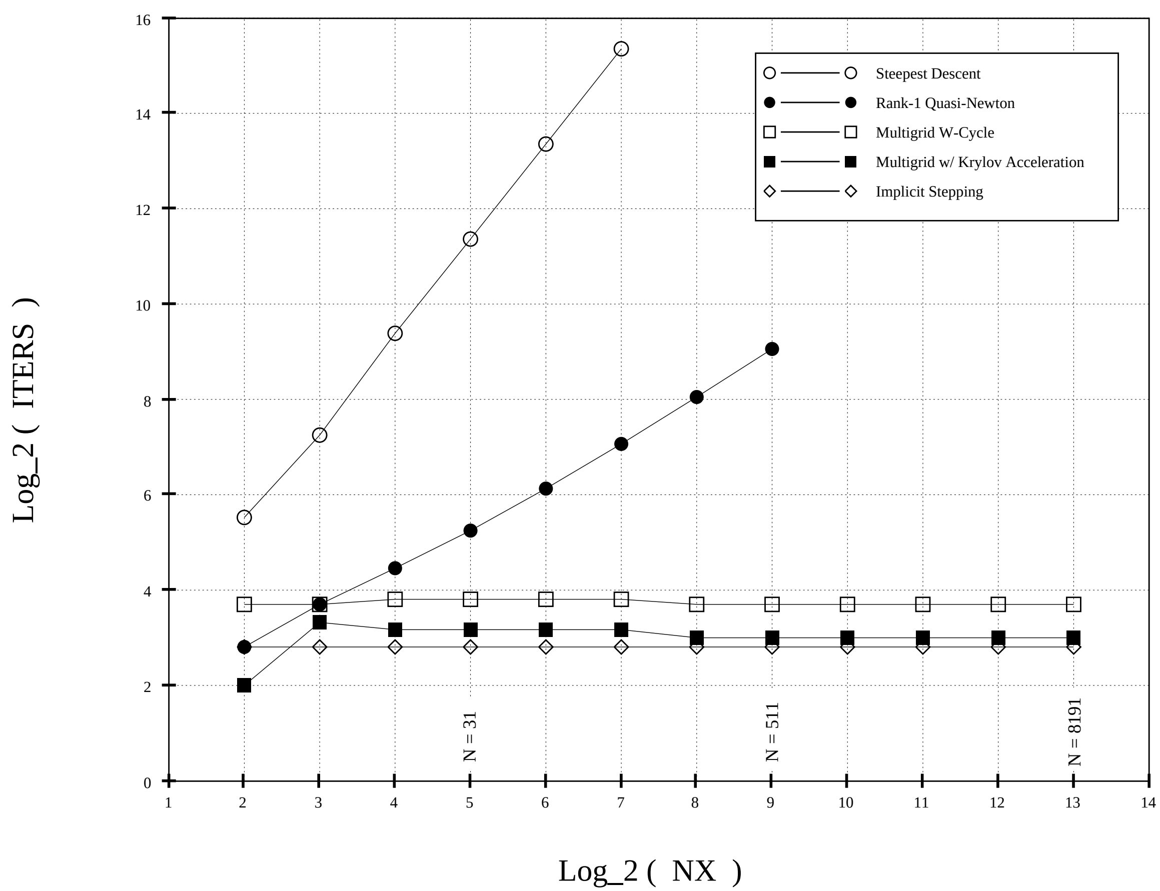
History of paths for Rank-1 quasi-Newton, N=511.

PERFORMANCE: QUASI-NEWTON



Comparison of quasi-Newton convergence histories, N=511.

PERFORMANCE: GRID DEPENDENCE



Comparison of convergence dependencies on dimensionality.

SUMMARY: BRACHISTOCHRONE STUDY

- **COMPARISON OF GRADIENTS**
  - Both Gradients Exhibited 2^{nd}-Order Accuracy
  - Continuous Gradient Slightly More Accurate
- **SEARCH METHODS**
  - Steepest Descent Scales with N^2
  - Quasi-Newton Methods Scale with N
  - Implicit Scheme Independent of N
  - Multigrid Descent Independent of N
  - Smoothed Descent Equivalent to Implicit Scheme

Theoretical Background for Aerodynamic Shape Optimization

John C. Vassberg

Boeing Technical Fellow
Advanced Concepts Design Center
Boeing Commercial Airplanes
Long Beach, CA 90846, USA

Antony Jameson

T. V. Jones Professor of Engineering
Dept. Aeronautics & Astronautics
Stanford University
Stanford, CA 94305-3030, USA

Von Karman Institute
Brussels, Belgium
7 April, 2014