

Optimization Assignment 1

Kim Paolo Laberinto

2021-02-xx

Contents

1 Q1.	3
1.1 Example 1D Optimization Trajectories	3
1.2 Comparision between Swann's and Powell's Methods	3
2 Q2.	3
2.1 Example Optimization Trajectories	3
2.2 Example 1D Optimizations in Trajectory	3
2.3 (EXTRA) Hooke-Jeeves Result on Rosenbrock	3
3 Q3.	3
3.1 Gradient Descent	3
3.1.1 Objective Function Definition and Gradient Derivation	3
3.2 Hooke-Jeeves	3
3.3 (EXTRA) Automatic Differentiation Gradient Descent	3
4 Q4. (Graduate Student)	3
4.1 Proving B_1 and B_2 are both a basis of V	4
4.1.1 Proof: B_1 is Linearly Independent	4
4.1.2 Proof: B_1 spans V	4
4.1.3 Proof: B_2 is Linearly Independent	4
4.1.4 Proof: B_2 spans V	4
4.2 Transformation between B_1 and B_2	4
4.3 D Derivative Operator	4
4.3.1 Proof: D is linear	4
4.3.2 Matrix Representation of D in the bases B_1 and B_2	4
5 Q5. (Graduate Student)	4
5.1 x in full \mathbb{R}^3 standard basis	4
5.2 Coordinates of x in B_X	4
5.3 Coordinates of y in B_Y using L ($c_y = Lc_x$)	4
5.4 y in full \mathbb{R}^4 standard basis	4
6 Q6. (Graduate Student)	4
6.1 Df derivation	4
6.2 $D(\ Cx\)$ Derivation	5
6.3 $D(\ Ax - b\)$ Derivation	6
A Extra Plots	6
B Source Code	6
B.1 All Optimization Algorithms (A1Module.jl)	6
B.2 Plot Generation Script (makeplots.jl)	6
B.3 (EXTRA) Automatic Differentiation for Gradient	6

1 Q1.

1.1 Example 1D Optimization Trajectories

1.2 Comparision between Swann's and Powell's Methods

2 Q2.

2.1 Example Optimization Trajectories

2.2 Example 1D Optimizations in Trajectory

2.3 (EXTRA) Hooke-Jeeves Result on Rosenbrock

3 Q3.

3.1 Gradient Descent

3.1.1 Objective Function Definition and Gradient Derivation

$$L = \sum_i (y_{\text{data}} - y_{\text{model}})^2 \quad (1)$$

$$L = \sum_i (y_i - (x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i}))^2 \quad (2)$$

We can now derive the gradient.

$$\nabla L = \nabla \left(\sum_i (y_i - (x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i}))^2 \right) \quad (3)$$

$$\nabla L = \sum_i (\nabla ((y_i - (x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i}))^2)) \quad (4)$$

$$\nabla L = \sum_i (2(y_i - (x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i})) \cdot \nabla(y_i - x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i})) \quad (5)$$

$$\nabla L = \sum_i (2(y_i - (x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i})) \cdot \nabla(x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i})) \quad (6)$$

$$\nabla(x_1 + x_2 e^{x_4 t_i} + x_3 e^{x_5 t_i}) = \begin{bmatrix} 1 \\ e^{x_4 t_i} \\ e^{x_5 t_i} \\ x_2 t_i e^{x_4 t_i} \\ x_3 t_i e^{x_5 t_i} \end{bmatrix} \quad (7)$$

3.2 Hooke-Jeeves

3.3 (EXTRA) Automatic Differentiation Gradient Descent

4 Q4. (Graduate Student)

Let V be the vector space of polynomial functions from $\mathbb{R} \rightarrow \mathbb{R}$ which have degree less than or equal to 2. That is, for all $a, b, c \in \mathbb{R}$ the following polynomial $p(x)$ is in V

$$p(x) = a + bx + cx^2$$

4.1 Proving B_1 and B_1 are both a basis of V

4.1.1 Proof: B_1 is Linearly Independent

- Implication of only way to get 0 vector.

4.1.2 Proof: B_1 spans V

- Show all $v \in V$ can be represented in basis

4.1.3 Proof: B_2 is Linearly Independent

4.1.4 Proof: B_2 spans V

4.2 Transformation between B_1 and B_2

4.3 D Derivative Operator

4.3.1 Proof: D is linear

4.3.2 Matrix Representation of D in the bases B_1 and B_2

5 Q5. (Graduate Student)

5.1 x in full \mathbb{R}^3 standard basis

5.2 Coordinates of x in B_X

5.3 Coordinates of y in B_Y using L ($c_y = Lc_x$)

5.4 y in full \mathbb{R}^4 standard basis

6 Q6. (Graduate Student)

6.1 Df derivation

Let $f : \mathbb{R}^4 \rightarrow \mathbb{R}$ be defined as $f(x) = \|Ax - b\| + \lambda\|Cx\| + \gamma\|Eb\|$ where $A, C, E \in \mathbb{R}^{4 \times 4}$, $x, b \in \mathbb{R}^4$ and $\lambda, \gamma \in \mathbb{R}$.

Let D denote the differential operator.

$$\begin{aligned} Df(x) &= D(\|Ax - b\| + \lambda\|Cx\| + \gamma\|Eb\|) \\ &= D(\|Ax - b\|) + \lambda D(\|Cx\|) + \gamma D(\|Eb\|) \quad [\text{using linearity of } D] \\ &= D(\|Ax - b\|) + \lambda D(\|Cx\|) + 0 \\ &= D(\|Ax - b\|) + \lambda D(\|Cx\|) \\ &= \frac{(Ax - b)^T A}{\|Ax - b\|} + \lambda \frac{(Cx)^T C}{\|Cx\|} \quad [\text{using derivations from next sections}] \end{aligned}$$

The full derivation of $D(\|Ax - b\|)$ and $D(\|Cx\|)$ can be found in the following sections.

6.2 $D(\|Cx\|)$ Derivation

$$\begin{aligned}
 D(\|Cx\|) &= D\left(\sqrt{(Cx)^T(Cx)}\right) \\
 &= D\left(\left((Cx)^T(Cx)\right)^{\frac{1}{2}}\right) \\
 &= \frac{1}{2}\left((Cx)^T(Cx)\right)^{-\frac{1}{2}} D\left((Cx)^T(Cx)\right) \quad [\text{using Chain Rule}] \\
 &= \frac{1}{2\sqrt{(Cx)^T(Cx)}} D\left((Cx)^T(Cx)\right) \\
 &= \frac{1}{2\|Cx\|} D\left((Cx)^T(Cx)\right) \\
 &= \frac{\left((Cx)^T D(Cx)\right) + \left((Cx)^T D(Cx)\right)}{2\|Cx\|} \quad [\text{using Product Rule}] \\
 &= \frac{2\left((Cx)^T D(Cx)\right)}{2\|Cx\|} \\
 &= \frac{(Cx)^T C}{\|Cx\|}
 \end{aligned}$$

6.3 $D(\|Ax - b\|)$ Derivation

$$\begin{aligned}
 D(\|Ax - b\|) &= D\left(\sqrt{(Ax - b)^T(Ax - b)}\right) \\
 &= D\left(\left((Ax - b)^T(Ax - b)\right)^{\frac{1}{2}}\right) \\
 &= \frac{1}{2}\left((Ax - b)^T(Ax - b)\right)^{\frac{-1}{2}} D\left((Ax - b)^T(Ax - b)\right) \quad [\text{using Chain Rule}] \\
 &= \frac{1}{2\sqrt{(Ax - b)^T(Ax - b)}} D\left((Ax - b)^T(Ax - b)\right) \\
 &= \frac{1}{2\|Ax - b\|} D\left((Ax - b)^T(Ax - b)\right) \\
 &= \frac{\left((Ax - b)^T D(Ax - b)\right) + \left((Ax - b)^T D(Ax - b)\right)}{2\|Ax - b\|} \quad [\text{using Product Rule}] \\
 &= \frac{2(Ax - b)^T D(Ax - b)}{2\|Ax - b\|} \\
 &= \frac{2(Ax - b)^T A}{2\|Ax - b\|} \\
 &= \frac{(Ax - b)^T A}{\|Ax - b\|}
 \end{aligned}$$

A Extra Plots

B Source Code

- B.1 All Optimization Algorithms (A1Module.jl)
- B.2 Plot Generation Script (makeplots.jl)
- B.3 (EXTRA) Automatic Differentiation for Gradient

Example 1D Line Searches on Objective Function

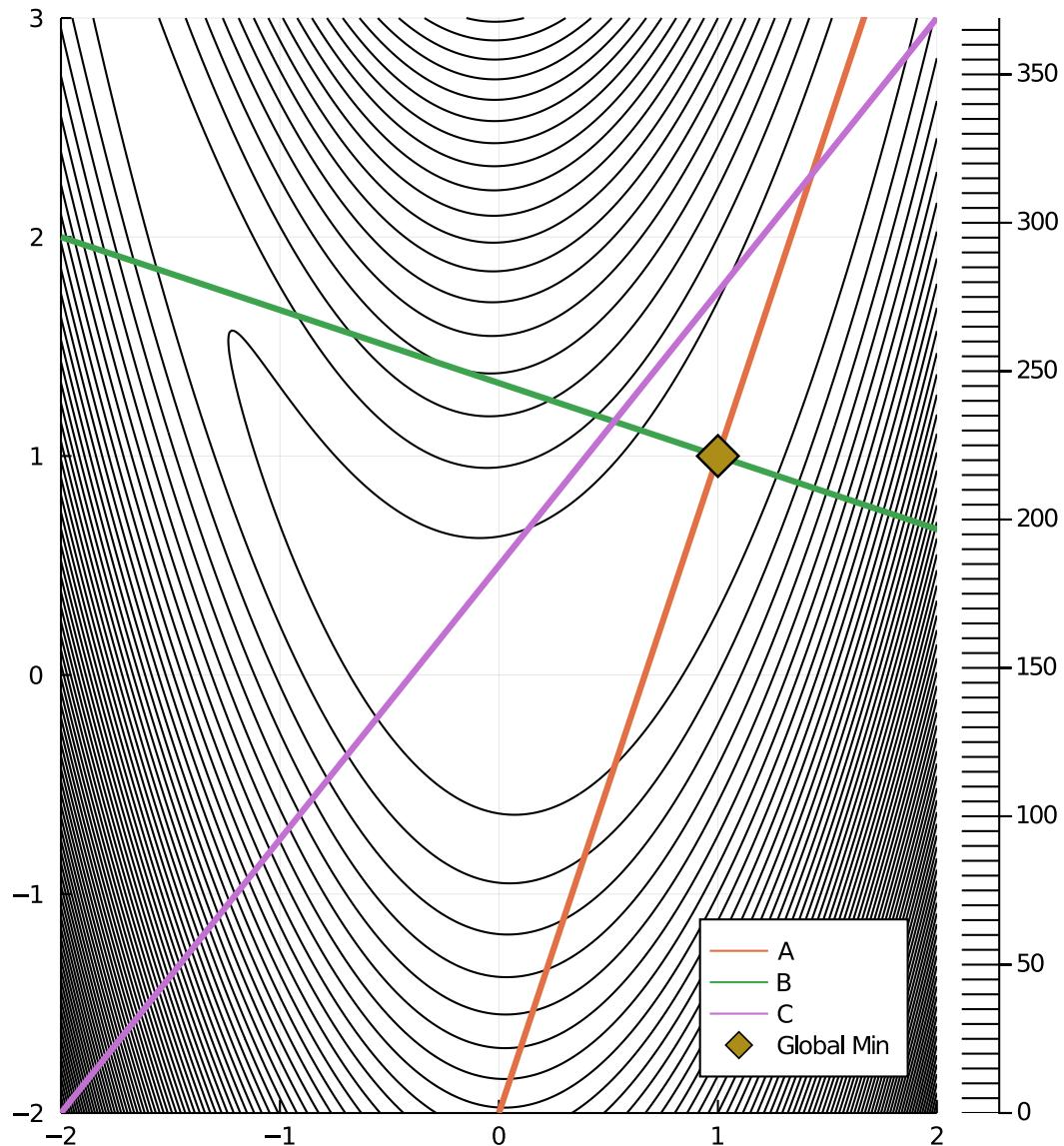


Figure 1: Caption

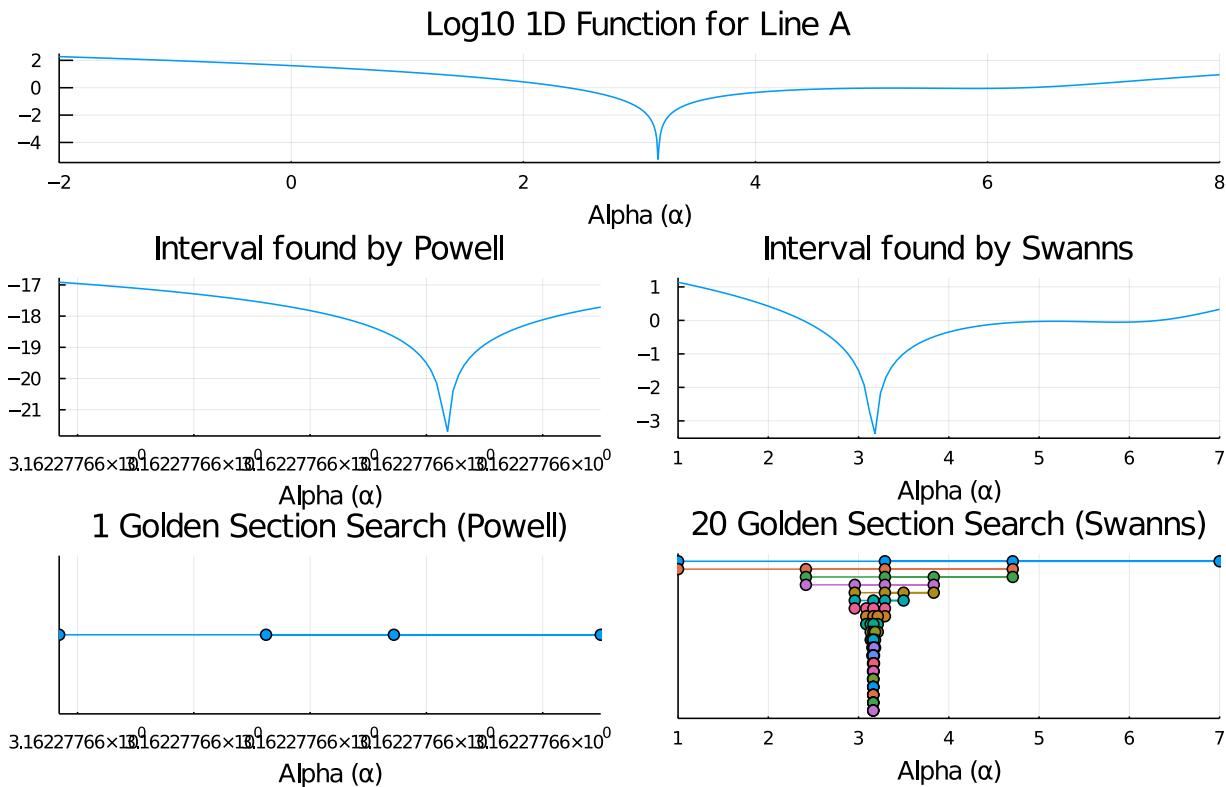


Figure 2: Caption

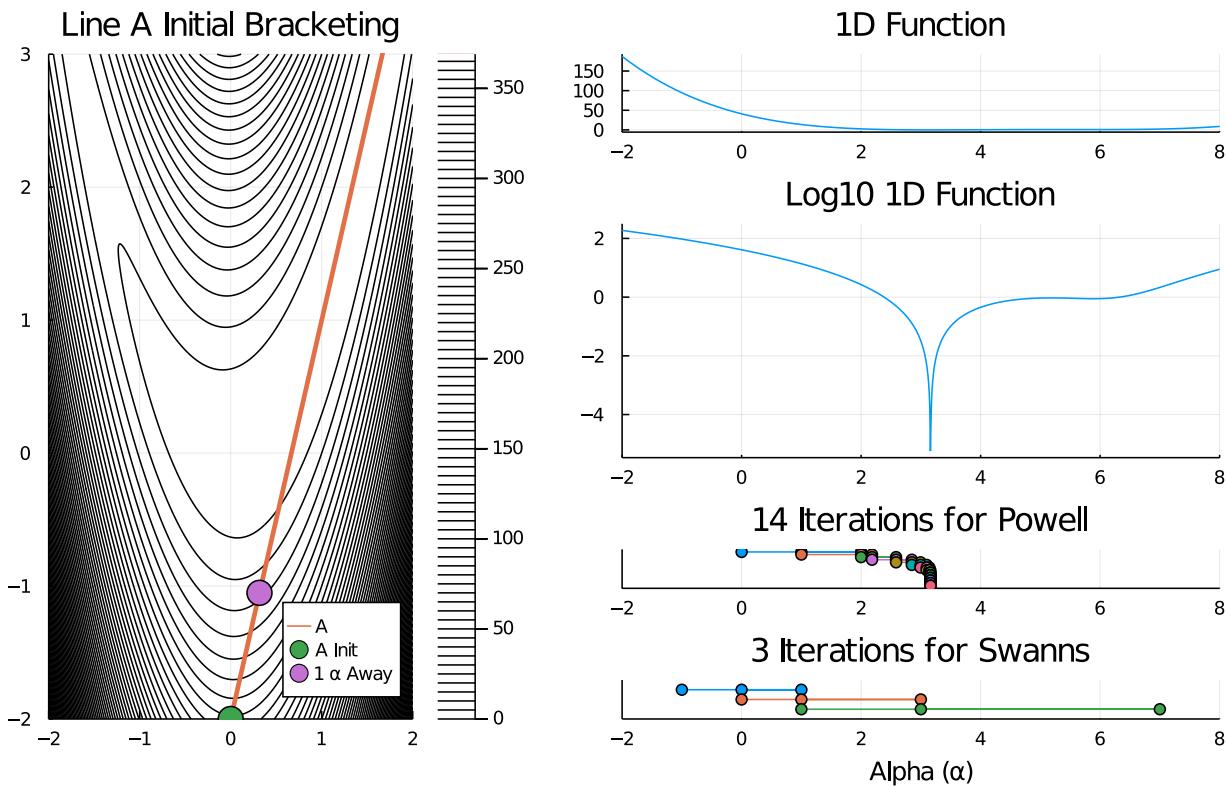


Figure 3: Caption

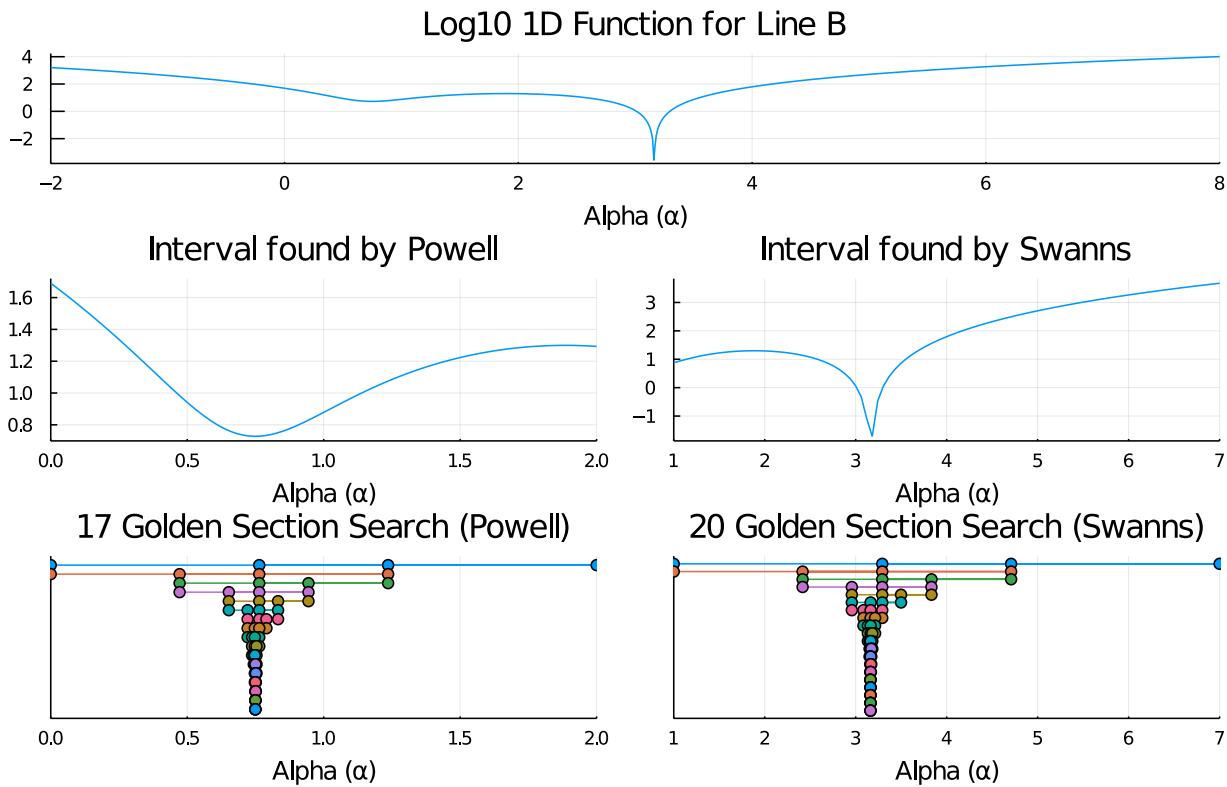


Figure 4: Caption

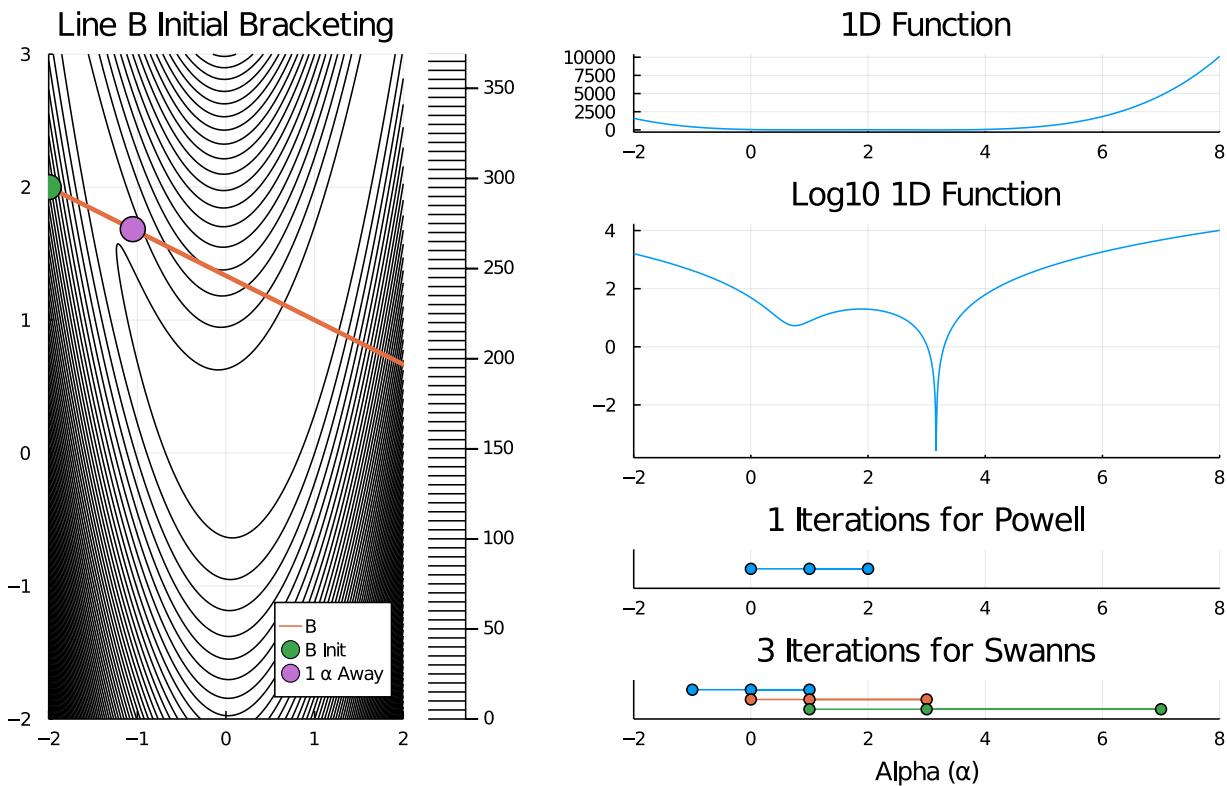


Figure 5: Caption

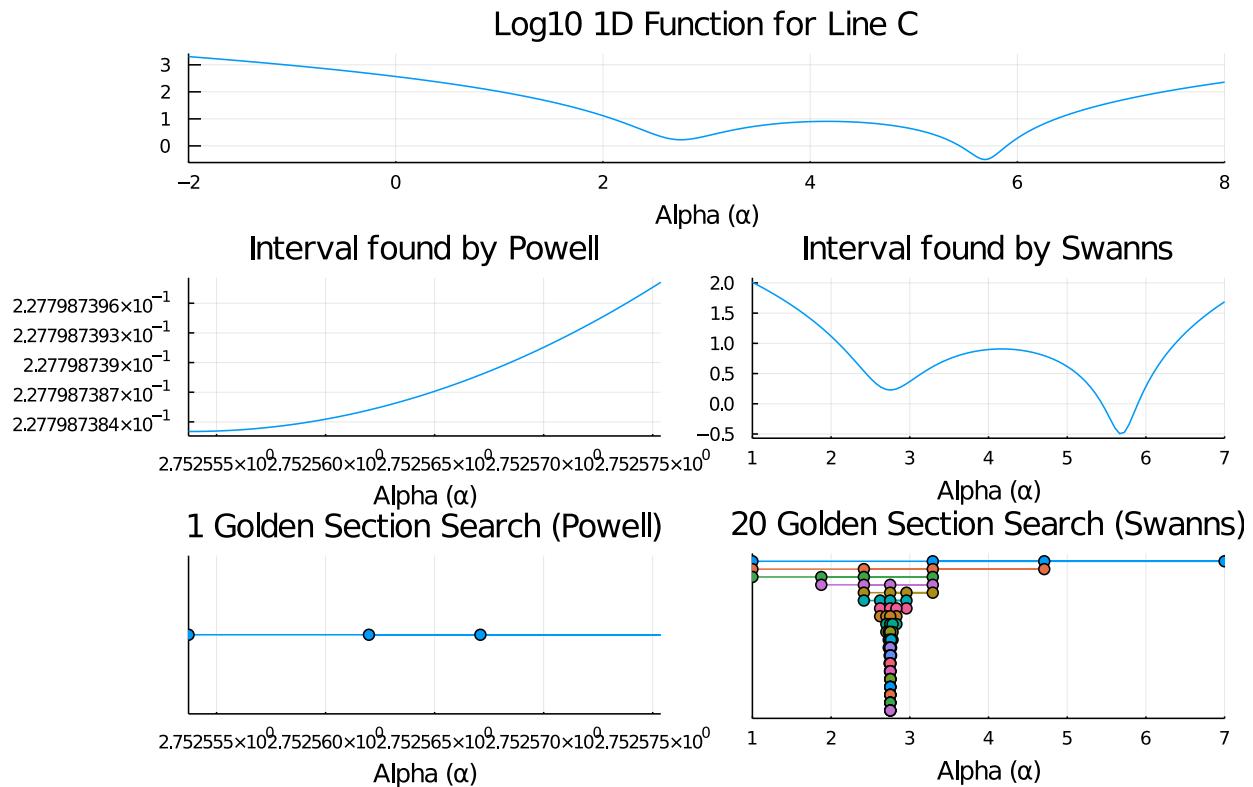


Figure 6: Caption

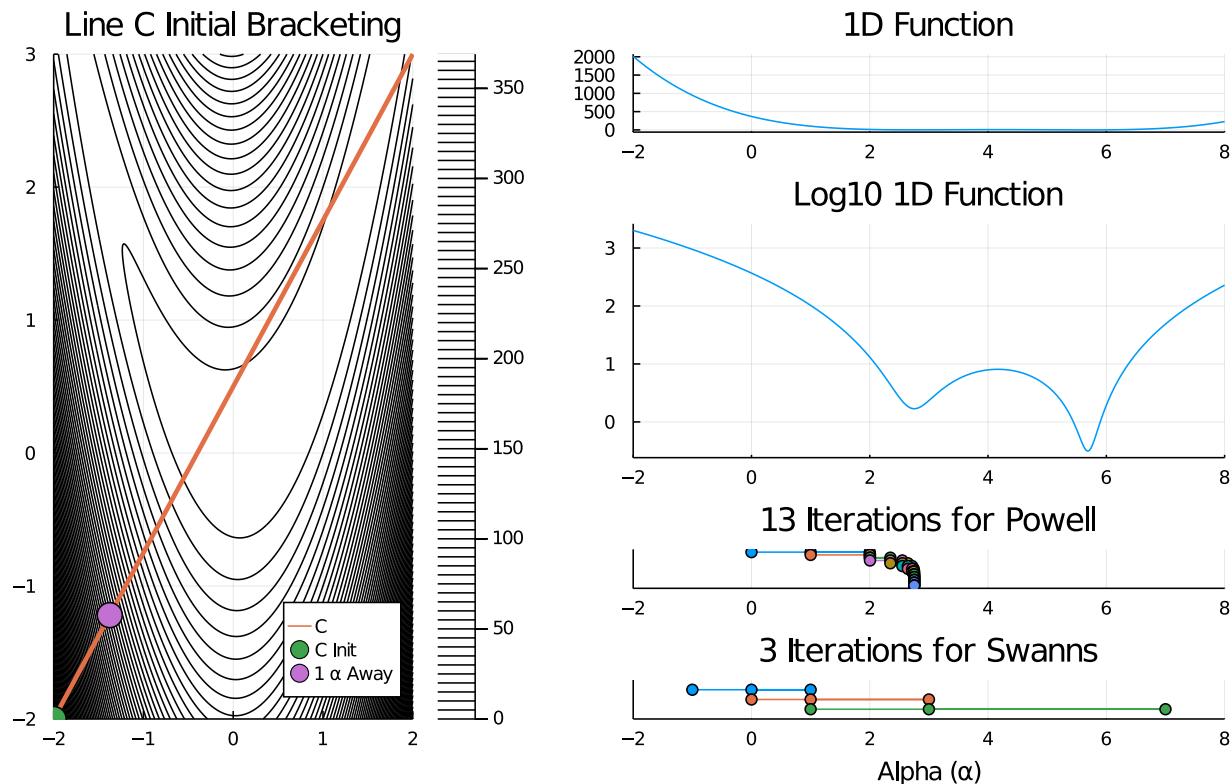
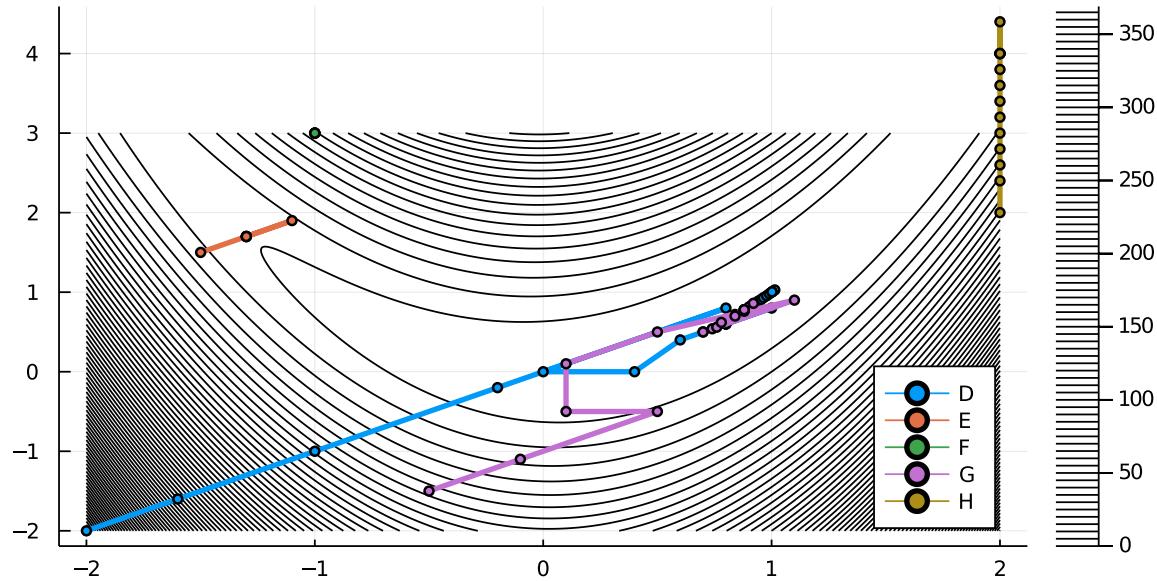


Figure 7: Caption

Hooke-Jeeves Steps



Obj. Func. Progression
Hooke-Jeeves

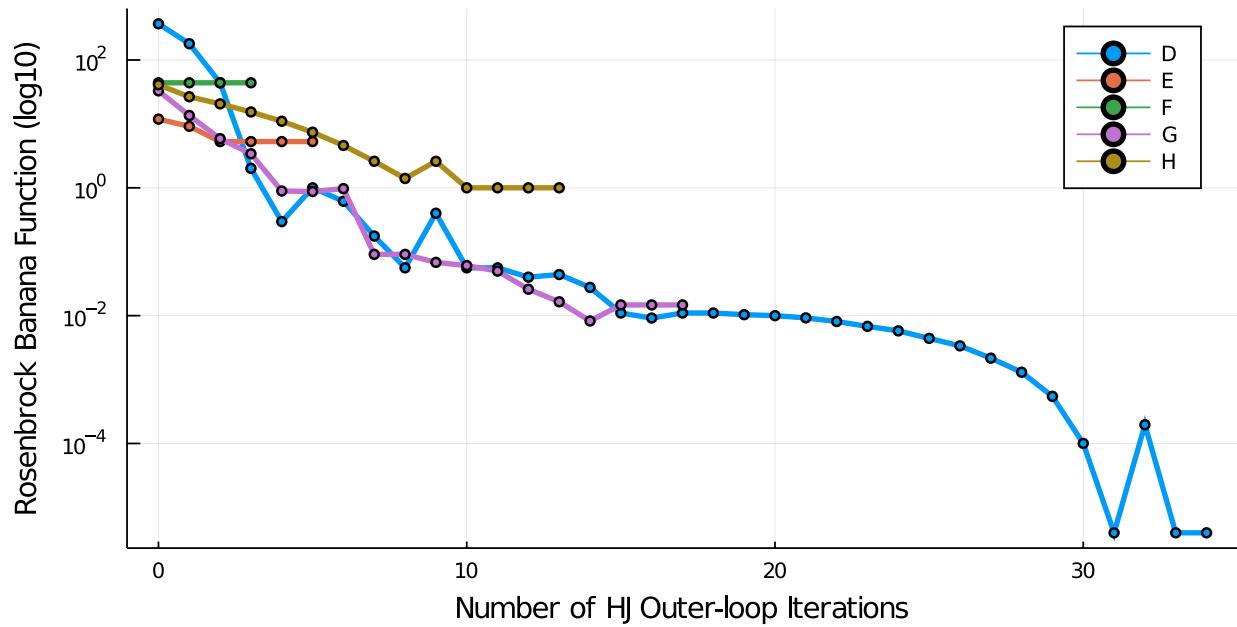


Figure 8: Caption

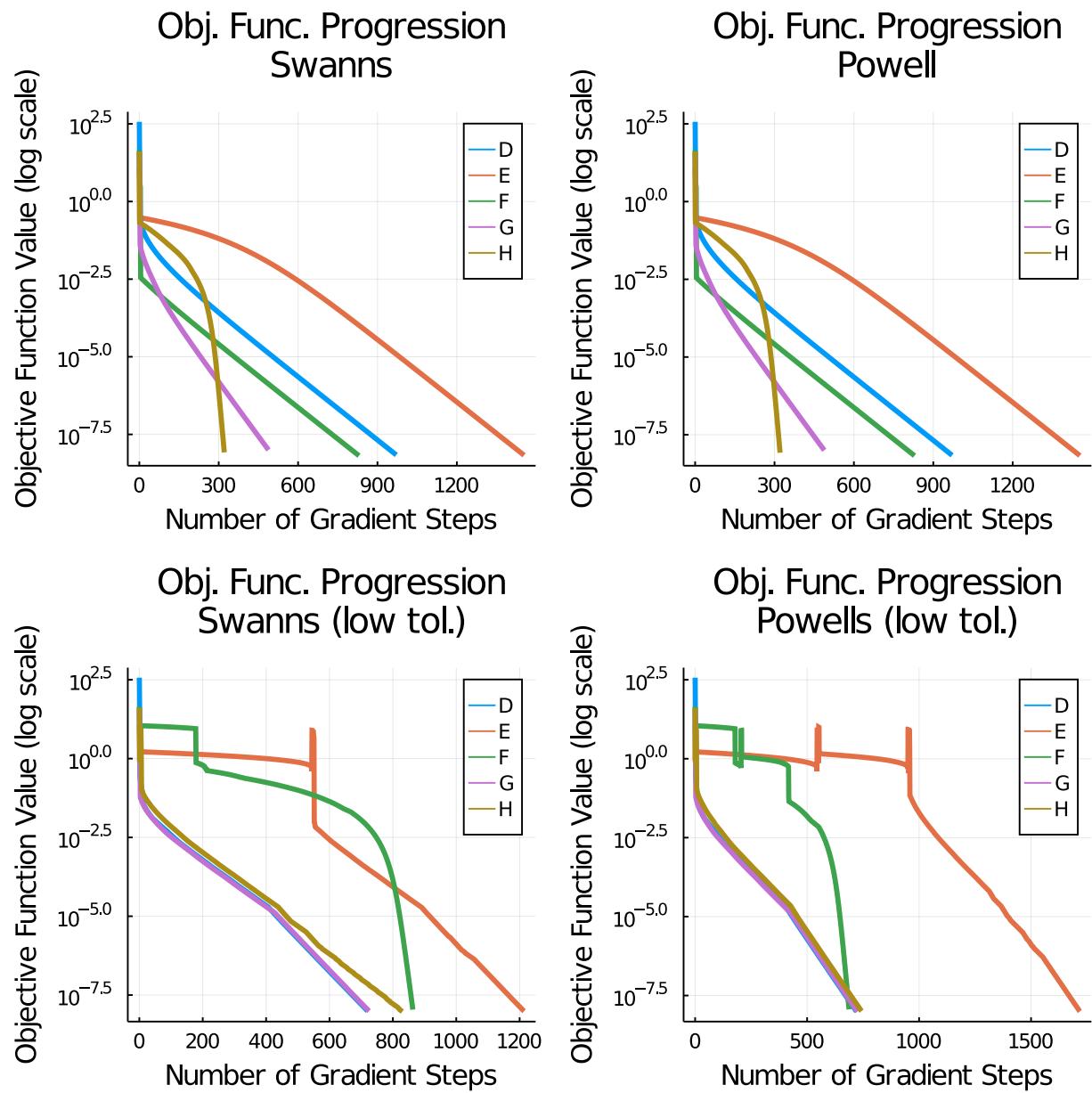


Figure 9: Caption

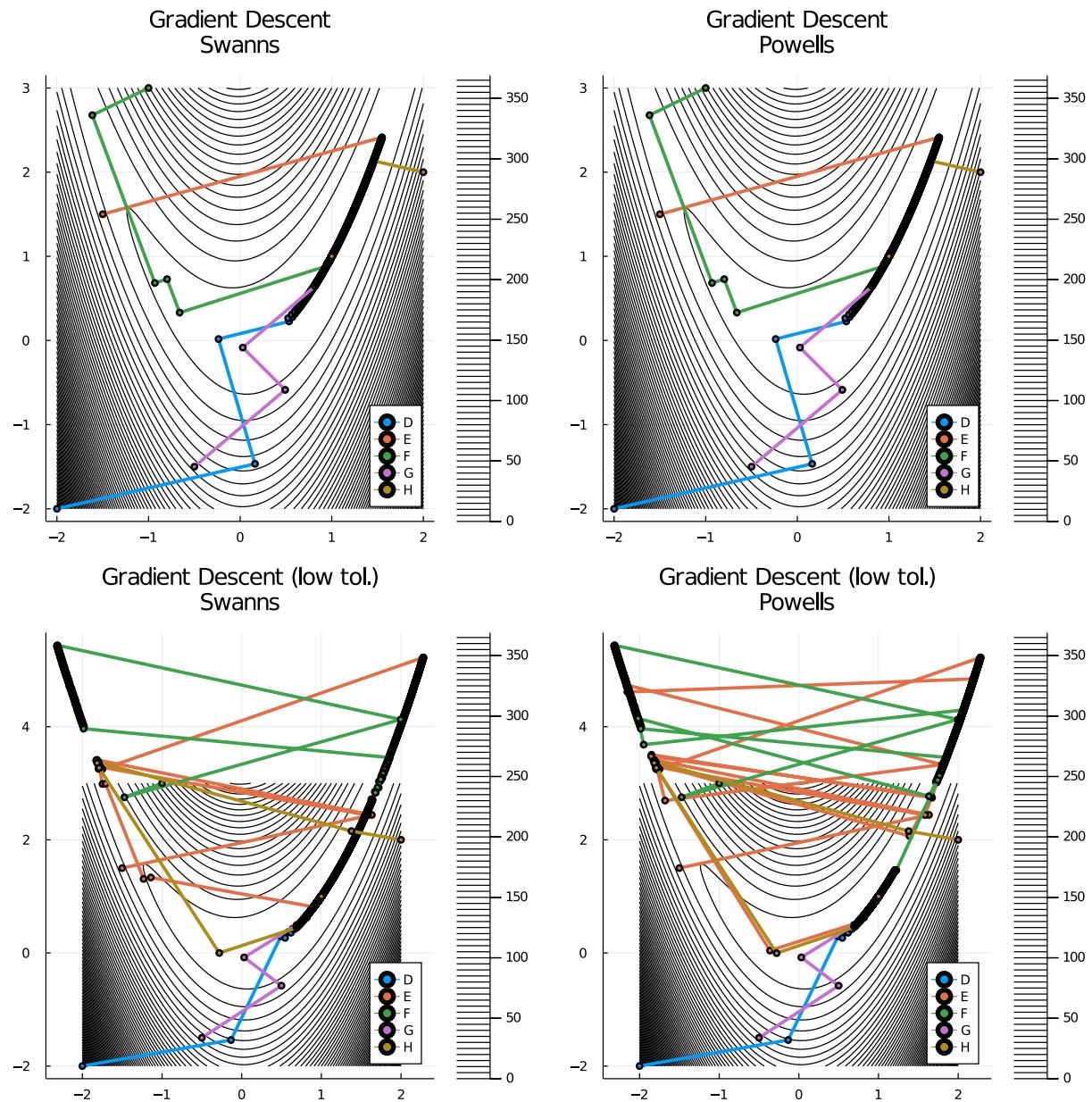


Figure 10: Caption

Manual vs Auto-Diff Gradient Descent Initial Parameters = [0.1, 0.1, 0.1, 0.1, 0.1]

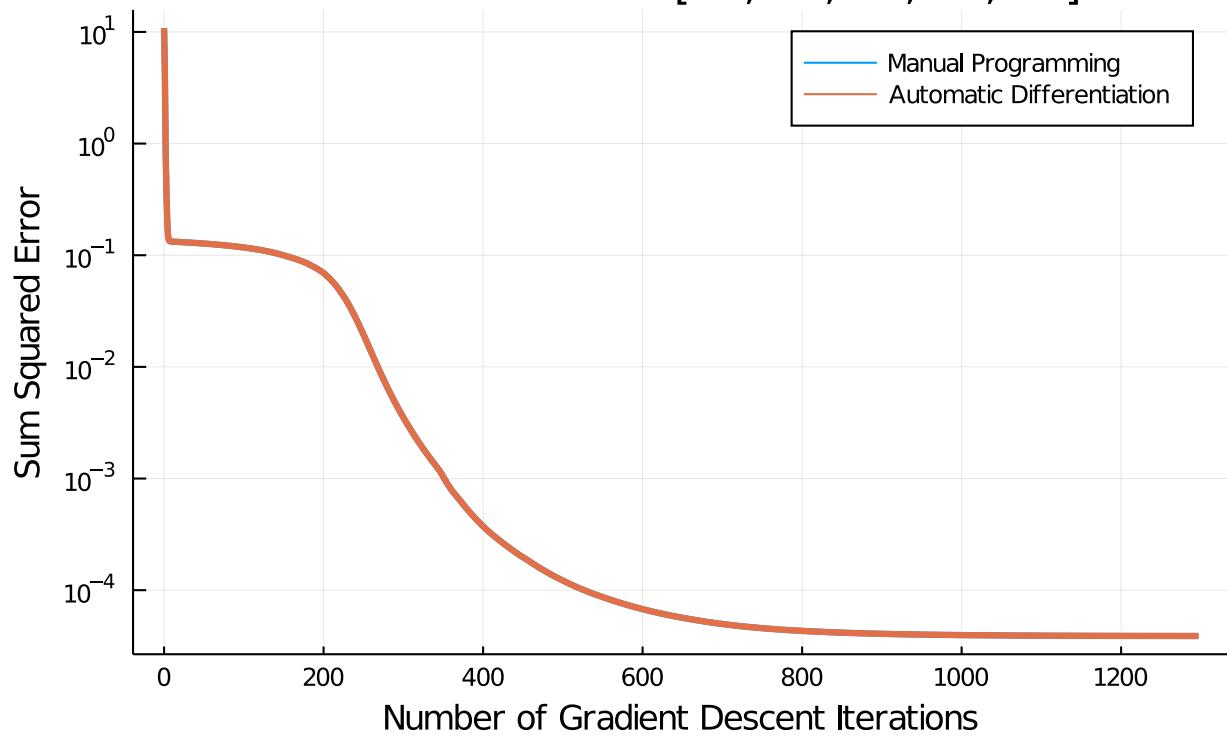


Figure 11: Caption

Best Fits from Optimization
Initial Parameters = [0.1, 0.1, 0.1, 0.1, 0.1]

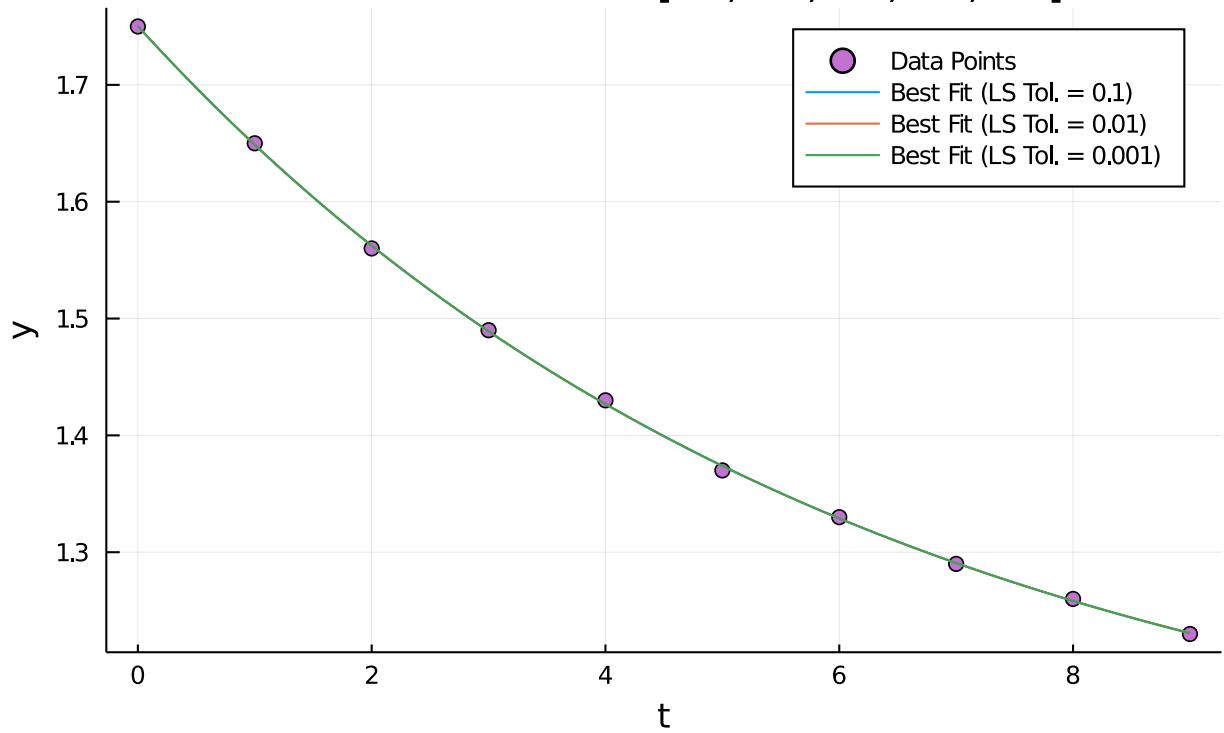


Figure 12: Caption

Best Fits from Optimization
Initial Parameters = [-0.1, -0.1, -0.1, -0.1, -0.1]

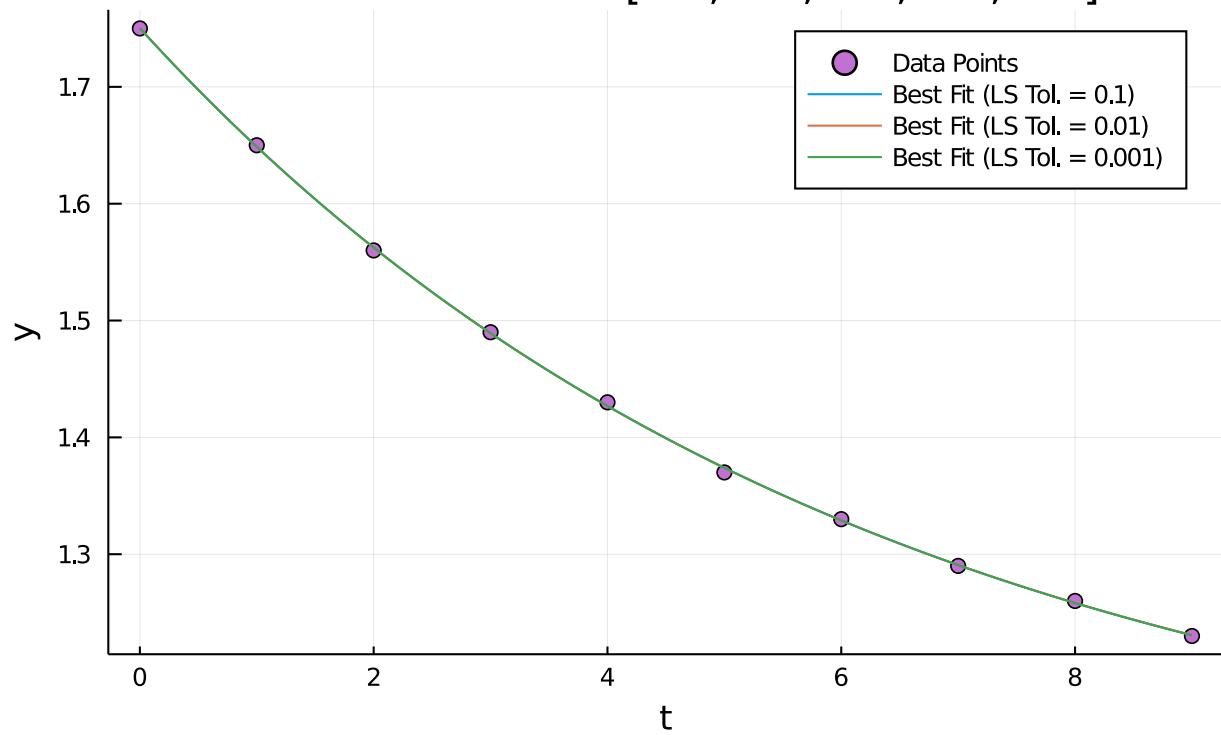


Figure 13: Caption

Best Fits from Optimization
Initial Parameters = [0.0, 0.1, -0.1, 0.0, 0.1]

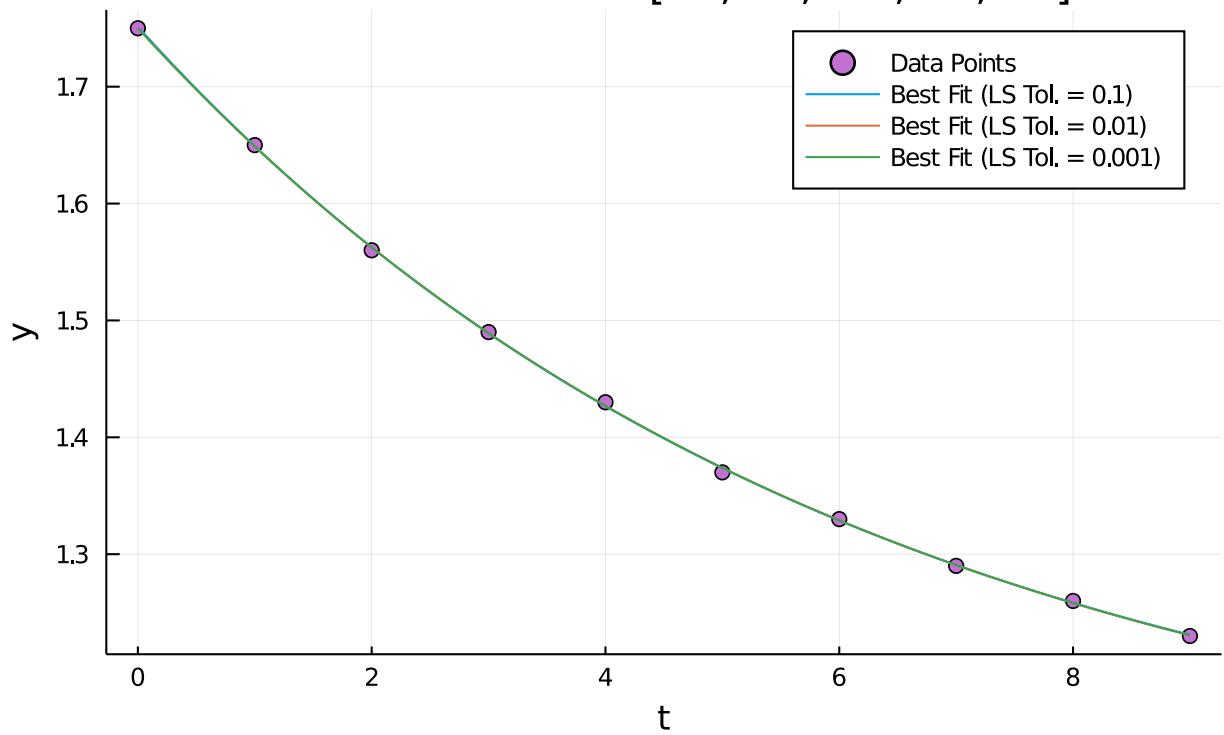


Figure 14: Caption

Loss vs Gradient Steps
Initial Parameters = [0.1, 0.1, 0.1, 0.1, 0.1]

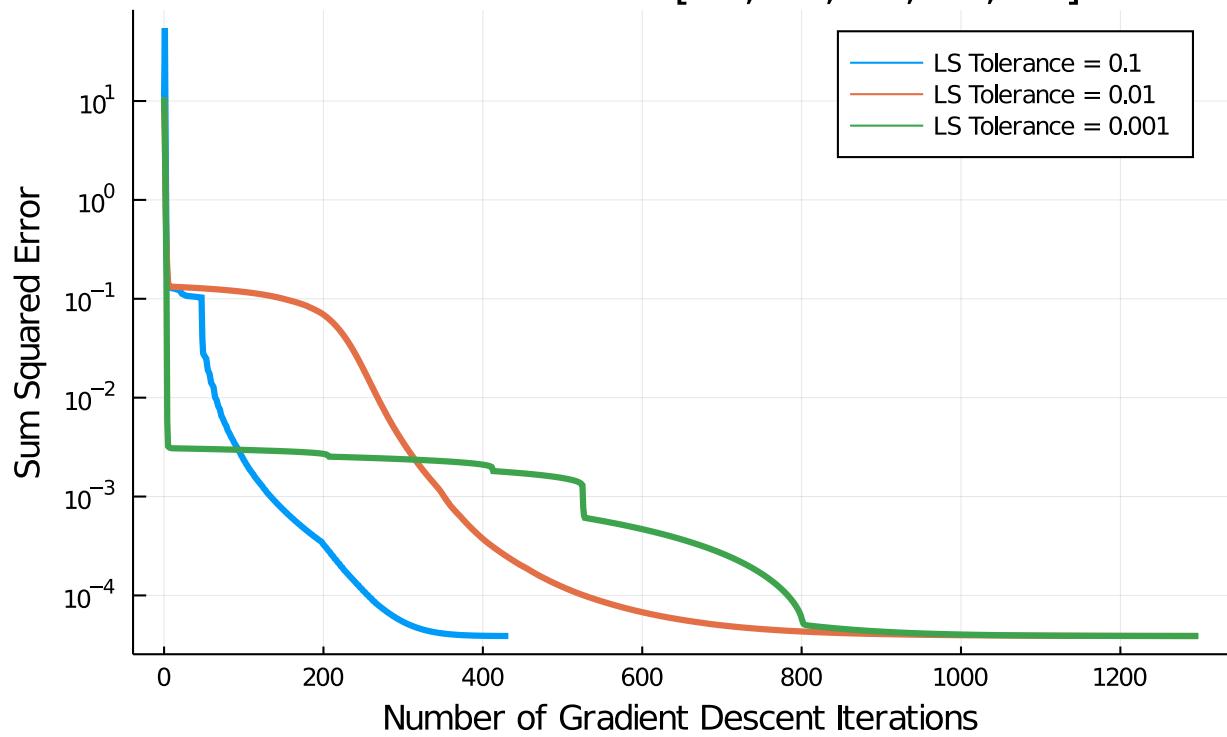


Figure 15: Caption

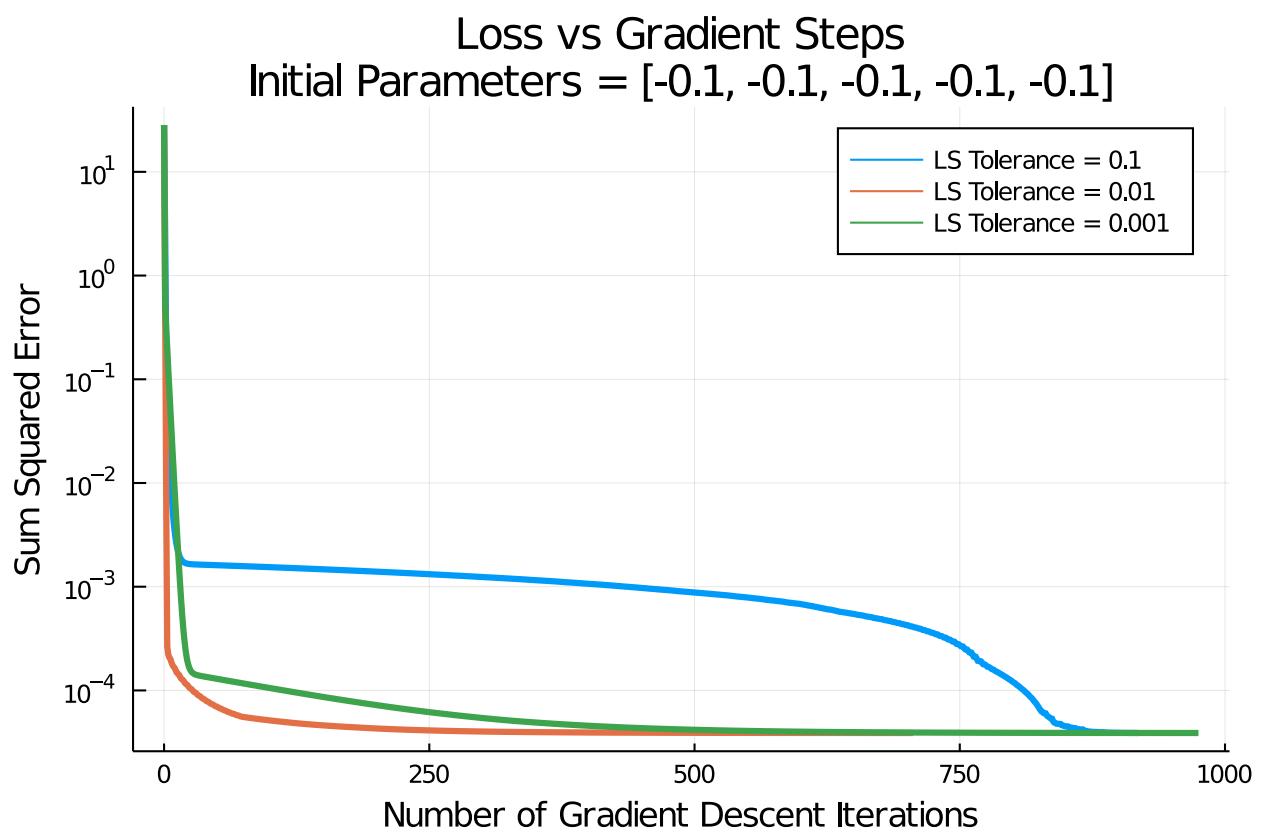


Figure 16: Caption

Loss vs Gradient Steps
Initial Parameters = [0.0, 0.1, -0.1, 0.0, 0.1]

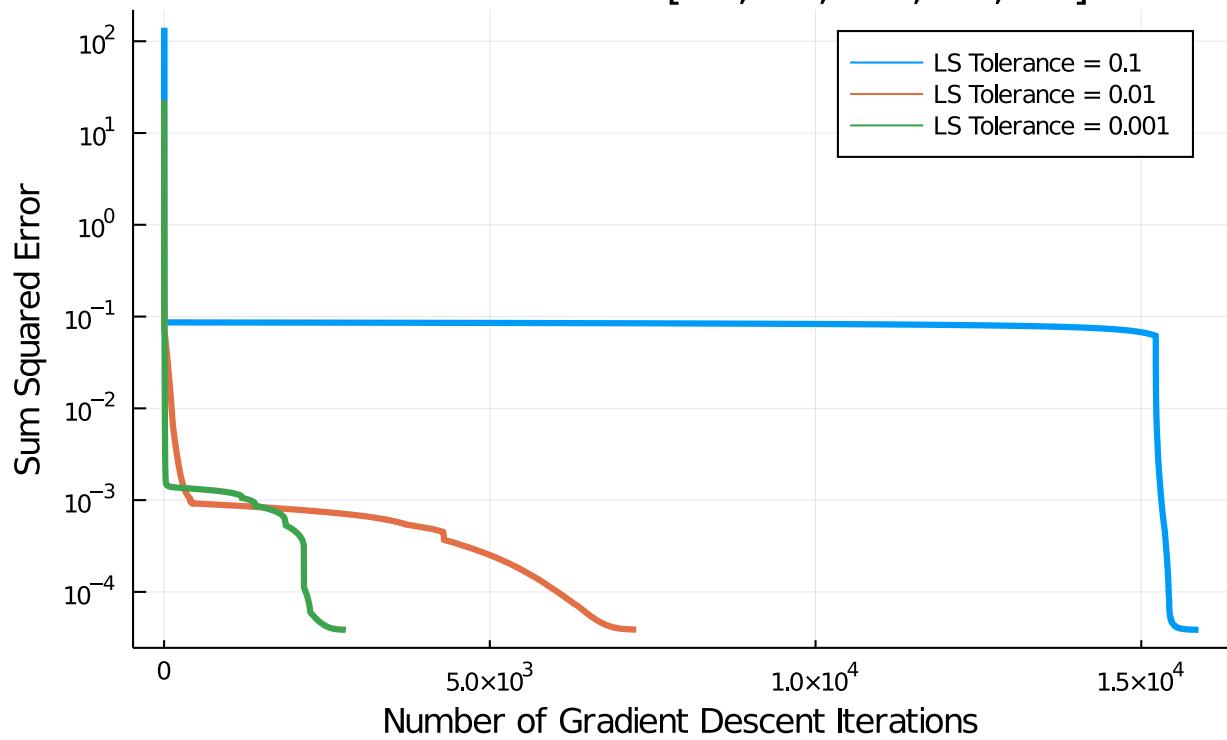


Figure 17: Caption

Best Fits from Optimization Hooke-Jeeves

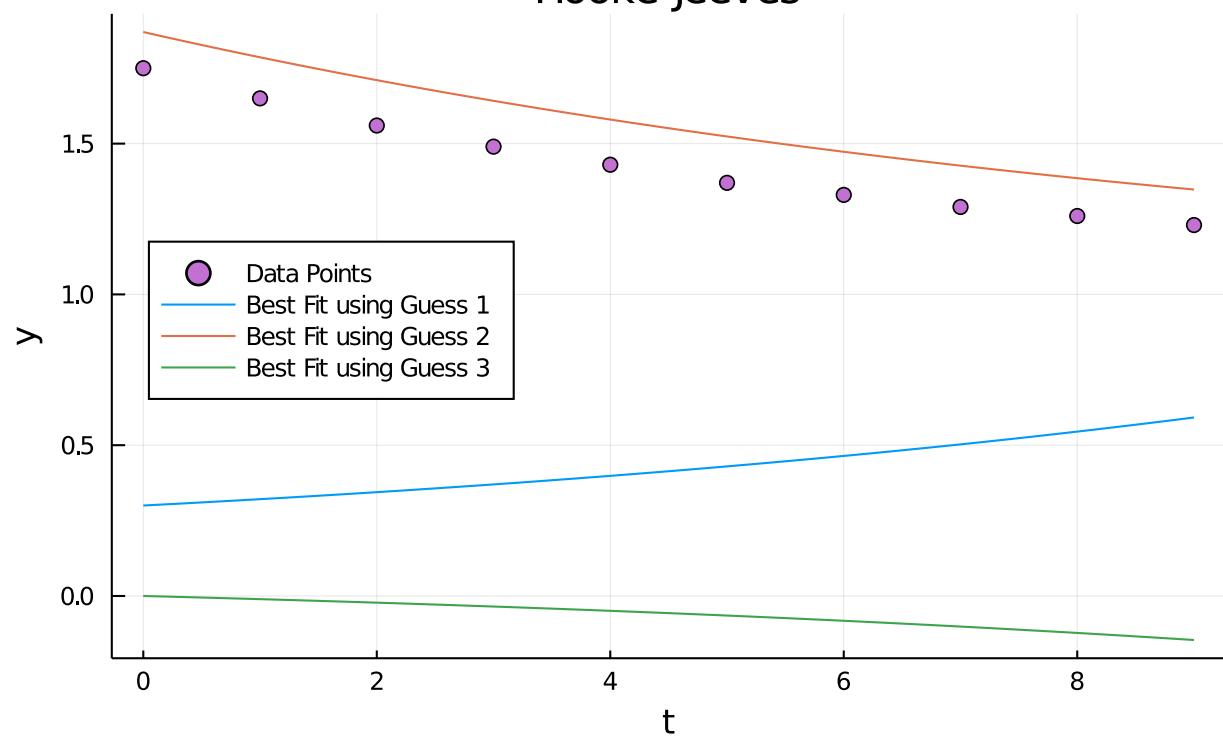


Figure 18: Caption

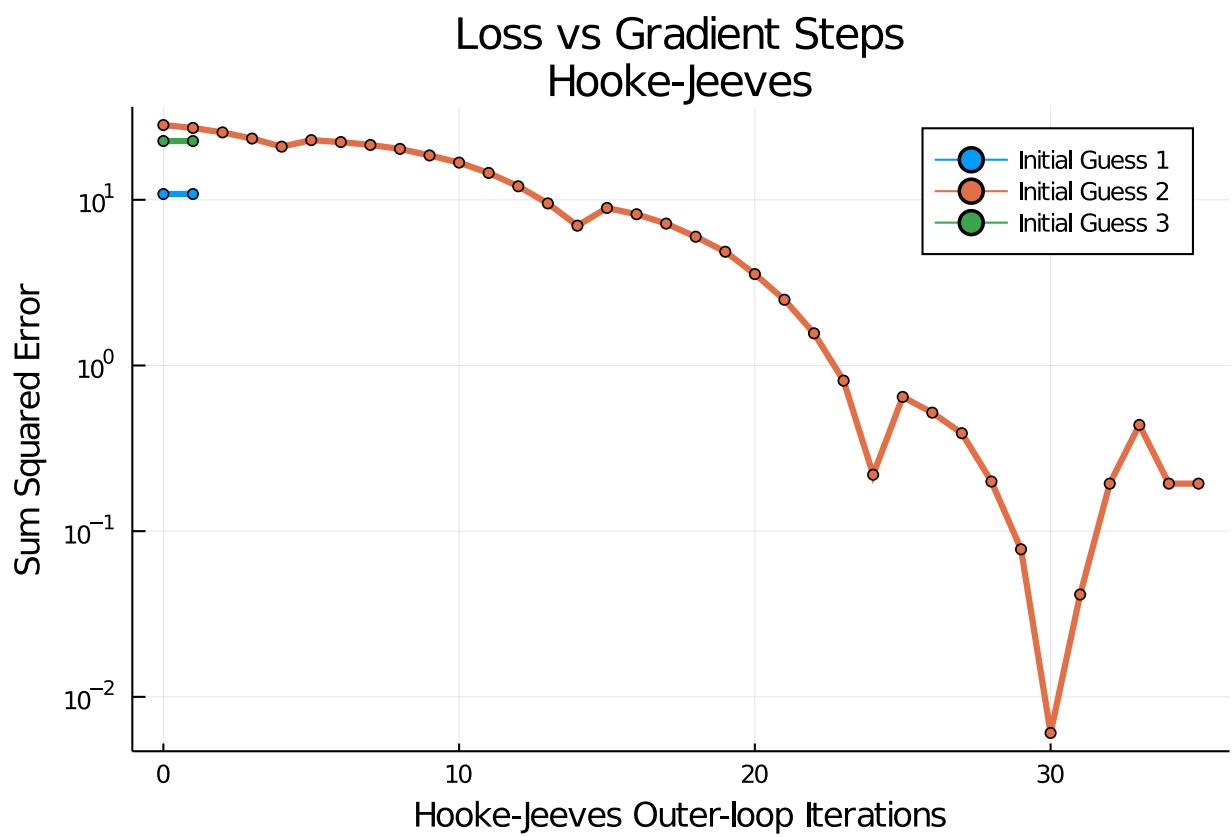


Figure 19: Caption