

1. Description of Methods

Simple 1 Function:

For the simple 1 function, I used the quadratic penalty method and the Hooke-Jeeves algorithm to solve the newly formed unconstrained optimization problem. I chose quadratic penalty (as opposed to simple count) for a smoother objective function and opted to use Hooke-Jeeves as an interesting derivative free, direct method to account for my lack of immediate information about the gradient of the constraint functions.

To select alpha, I used a similar value to what I started with for nesterov momentum methods with the idea being that the step size I want to take should remain the same (since with direct methods we're just using specified search directions instead of gradient information for our **d**). Gamma was picked with some hand tuning. If the algorithm wasn't converging in time, I cranked up the penalty to get out of potential local minima or fruitless search directions.

I believe this combination of quadratic penalty and Hooke-Jeeves worked so well because the penalty function helped to cover a weakness of Hooke-Jeeves, a propensity to get stuck in local minima.

A definite pro to Hooke-Jeeves is its ability to find minima without information about the gradient of the objective function. Because of this, I didn't have to use function evaluations for automatic differentiation methods. As mentioned above, a con to Hooke-Jeeves is its tendency to get stuck in local minima (I believe primarily as a side-effect of its rigid orthogonal search direction set it's confined to exploring as opposed to General Pattern Search's arbitrary search directions).

Simple 2 Function:

Simple 2 was solved with the exact same algorithm, penalty method, and hyper-parameters as Simple 1, so for this section, see above. My reasoning remained the same.

Simple 3 Function:

Simple 3 used the same algorithm and penalty method but required a different gamma that was 4 times higher than those of the previous 2 problems. I think the higher penalty was required to account for the increased dimensionality with fewer constraints. Hooke-Jeeves

probably needed more nudging with less constraints and more dimensions to explore, which was accomplished with a higher gamma to push pro along further with fewer iterations.

Secret 1 Function:

Secret 1 used the same algorithm, penalty method, and hyper-parameters as Simple 3. However, I believe the higher gamma was needed in this problem because of the many local minima that Hooke-Jeeves can get stuck in (as opposed to the issue being with the dimensionality of the data and number of constraint functions like Simple 3).

Secret 2 Function:

My first attempt at Secret 2 involved using the same algorithm, penalty method, and hyper-parameters as above. Many failed attempts at tuning alpha and gamma later (including cranking up gamma to be incredibly high to help Hooke-Jeeves through the local minima of the highly non-convex objective function), I went back to the textbook to search for a more forgiving direct method. After some digging, I nearly chose Generalized Pattern Search to further vary my search directions but opted to try out Simulated Annealing since it's well regarded for avoiding getting trapped in the many local minima that I imagine a 22-dimension non-convex objective function might have.

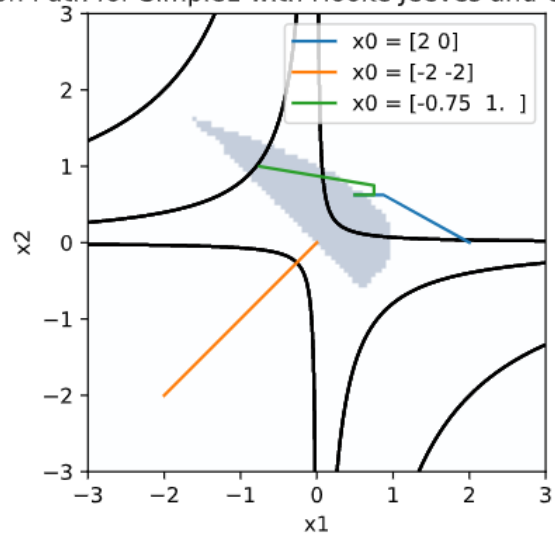
With a quadratic penalty still in place (same gamma as Simple 1), I initially tried a uniform distribution for my transition distribution T , but this didn't converge in time with a fixed annealing schedule (so t isn't changing over time, it sits at 1). Interestingly enough, rather than alter the acceptance probability, changing to a standard normal distribution (0 mean, diagonal covariance) worked pretty well. Something tells me this may have to do with the many nice properties that Gaussians present when working with various dynamical systems (at least in state estimation methods they're typically viewed as nicer and more robust to work with). I think I've been able to get away with a fixed temperature because the acceptance probability I've chosen is adequate when paired with T 's distribution and my gamma value for the penalty.

One pro to Simulated Annealing is its ability to deal with local minima by starting with an expansive search space and "lowering the temperature" (or in my case picking a good temperature to start with) to home in on the appropriate search space to converge to a solution. I believe this is what helps the algorithm avoid local minima the most (outside of its inherent

stochasticity). One con is probably the lack of intuition that may be present for tuning the hyper-parameters. At least for me, correlating temperature to search spaces doesn't quite enable me to look at an objective function and say, "Yeah, once it's at 20 degrees it will converge".

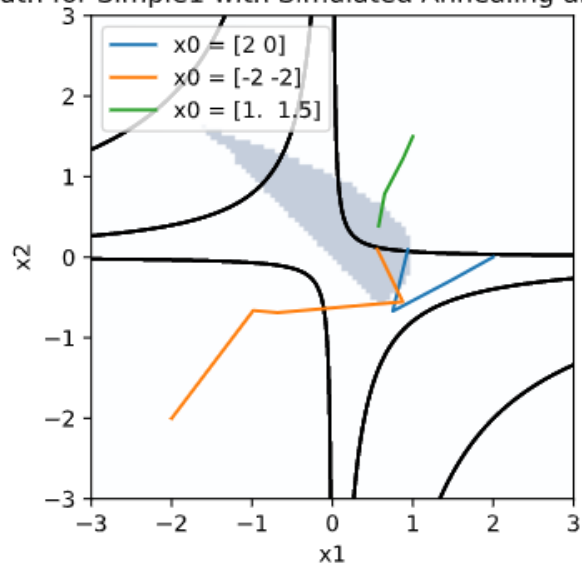
2. Simple1 Contour Plot with Optimization Paths (Hooke-Jeeves)

Optimization Path for Simple1 with Hooke Jeeves and Quadratic Penalty



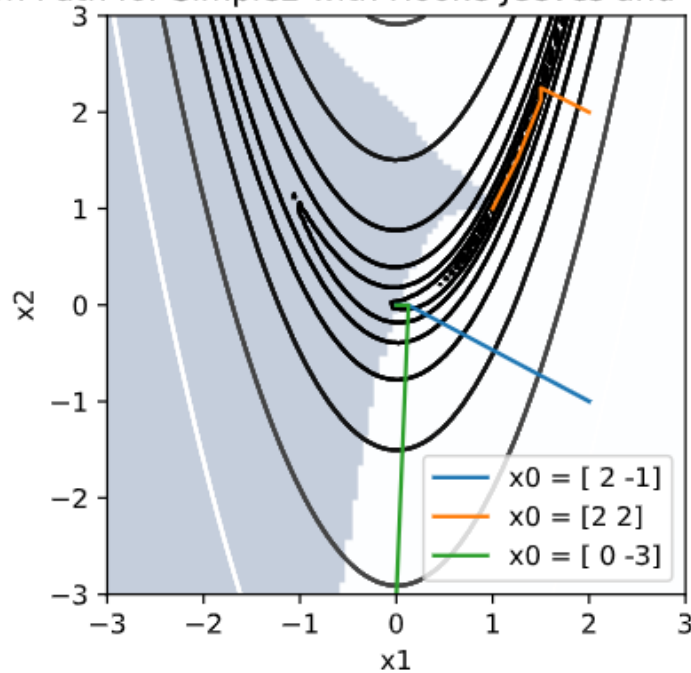
3. Simple1 Contour Plot with Optimization Paths (Simulated Annealing)

Optimization Path for Simple1 with Simulated Annealing and Quadratic Penalty



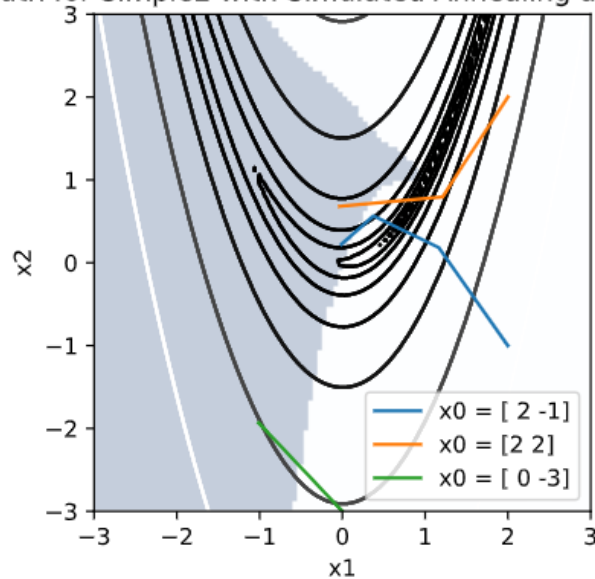
4. Simple2 Contour Plot with Optimization Paths (Hooke-Jeeves)

Optimization Path for Simple2 with Hooke-Jeeves and Quadratic Penalty

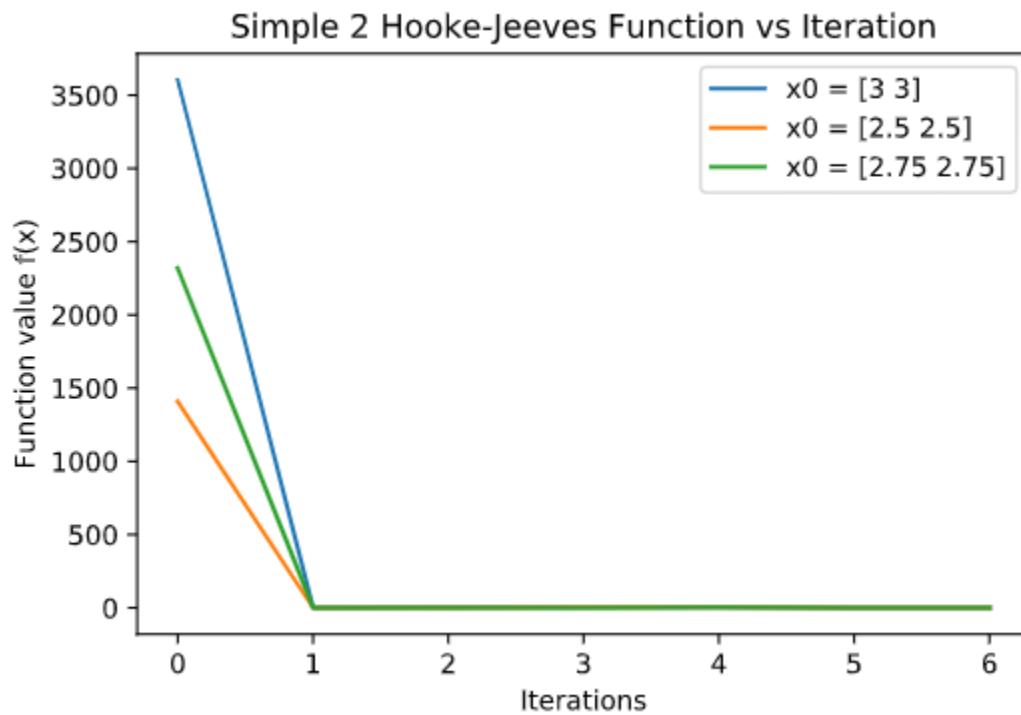


5. Simple2 Contour Plot with Optimization Paths (Simulated Annealing)

Optimization Path for Simple2 with Simulated Annealing and Quadratic Penalty



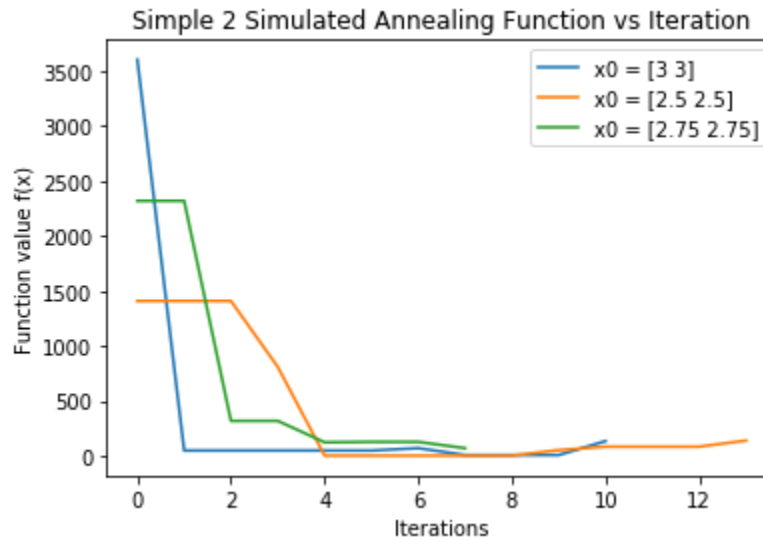
6. Simple2 Objective Function vs Iteration (Hooke-Jeeves)



```
[3604, 1.0, 2.5625, 2.5625, 2.3681640625, 0.453125, 0.0]
[1408.5, 0.25, 0.25, 1.8125, 1.8125, 0.453125, 0.0]
[2319.078125, 0.953125, 0.953125, 0.953125, 2.3681640625, 0.453125, 0.0]
```

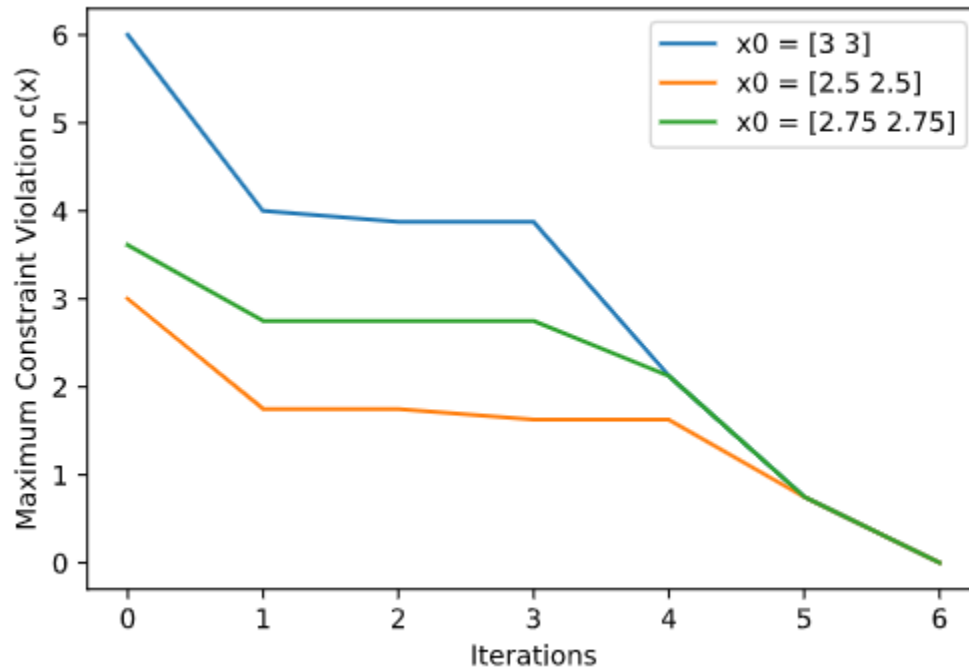
***function values for different I.Cs given above since the graph doesn't capture the small values as they evolve**

7. Simple2 Objective Function vs Iteration (Simulated Annealing)



8. Simple2 Cost Function vs Iteration (Hooke-Jeeves)

Simple 2 Hooke-Jeeves Maximum Constraint Violation vs Iteration



9. Simple2 Cost Function vs Iteration (Simulated Annealing)

Simple 2 Simulated Annealing Maximum Constraint Violation vs Iteration

