Particle Swarm Optimization: Two Methods Using Implicit Filtering

George Kuliner

Dr. Ahmad Almomani

Agenda

1. What is Optimization?
2. Derivative Free Optimization
3. Particle Swarm Optimization (PSO)
4. Implicit Filtering Method (IF)
5. PSOIF Hybrid Method 1
6. PSOIF Hybrid Method 2
7. Results
8. Future Work
9. References

What is Optimization?

- More basic: Find the minimum or maximum of a function

- Optimization deals with two main questions:
  1.) What **Quantity** are we trying to optimize?
  2.) What is the **Domain** of optimization?

- More Formally:

We want to find x^* such that we min $f(x^*)$ or $f(x^*)$ for $x^* \epsilon D$ where f is the objective function and D is the given domain.

What is Optimization? Global vs Local

- Global Optimization: Searching for the minimum or maximum over the entire functions domain

- Local Optimization: Searching for the minimum or maximum over a neighborhood of points in the domain

Applications

- Data Analysis
- Finance and Economics
- Physics

Optimization Methods Overview

	Derivative Free	**Gradient Based**
Local	Implicit Filtering, Nelder-Mead Pattern Search	Gradient Descent, Newton Methods
Global	Particle Swarm, Genetic Algorithm, Simulated Annealing	Interval Analysis, Direct Method

Reasons for Derivative Free Optimization

- The derivative of the objective function is hard to find or non-existent
- Noise in function values
- The objective function is computed via Black Box

Particle Swarm Optimization (PSO)

- PSO was developed in 1995 by Dr. Eberhart and Dr. Kennedy
- Based on social behavior of bird flocks and schools of fish
- Particles have three components to movement at each iteration: inertia, cognitive, social

Velocity of i^{th} particle at time $t+1$

$$\vec{V}_i^{t+1} = \omega \vec{V}_i^t + c_1 r_1 (\vec{P}_i^t - \vec{X}_i^t) + c_2 r_2 (\vec{G}_i^t - \vec{X}_i^t)$$

ω, c_1, c_2 - weights

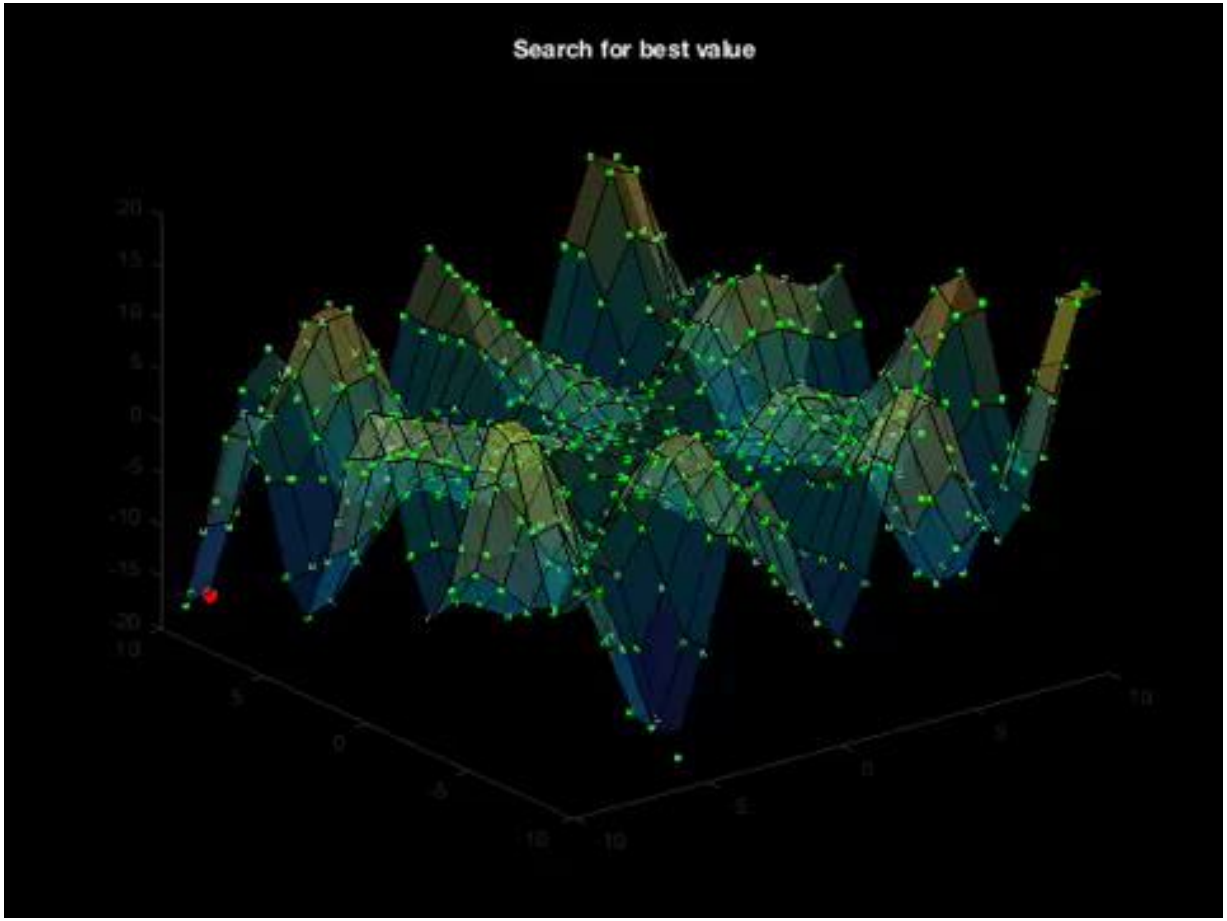
r_1, r_2- random variable generated

Particle Swarm Optimization (PSO)

Algorithm: PSO

Initialize population randomly
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
While (stopping criteria not met) **Do**
$V \leftarrow$ update velocities
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
end While
Return G*

PSO in Action



Implicit Filtering (IF)

- IF was developed by Dr. C.T. Kelly
- IF is a deterministic, derivative free, local optimization algorithm
- IF starts with sampling points to create a stencil
- Samples are arranged on a stencil to approximate the gradient
- We test points on the stencil and then chooses best location
- Uses quasi-Newton method to approximate Hessian
- Then compares these values to choose the best point and repeats until termination criterion is met

Our Main Goal

- Implement IF with PSO creating a hybrid solver improving on PSO
- Increased accuracy or a lower amount of function evaluations

Hybrid PSO with IF Version 1 (HPSOIFv1)

- This algorithm has PSO run first then uses IF to finish the search once the space is small enough
- The global best found by PSO is used as the initial iterate in IF
- The furthest particle is used for the bounds
- Also have the option to run IF with previous global bests for stencil directions

Algorithm: HPSOIFv1

Initialize population randomly
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
While (stopping criteria not met)
Do
$V \leftarrow$ update velocities
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
end While
Run Implicit Filter (G, Bounds)
Return G*

Hybrid PSO with IF Version 2 (HPSOIFv2)

- This algorithm has the ability to run PSO at every iteration
- The global best found by the run of PSO is used as the initial iterate in IF
- The furthest particle is used for the bounds
- Also have the option to run IF with previous global bests for stencil directions

Algorithm: HPSOIFv1

Initialize population randomly
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
While (stopping criteria not met) **Do**
$V \leftarrow$ update velocities
$P \leftarrow$ update personal best
$G \leftarrow$ update global best
If (Bounds has grown)
If ($G_{IF} \leq G$), $G \leftarrow$ update global best
Else If ($G_{IF} \leq P_{worst}$),
$P_{worst} \leftarrow$ update worst personal best
end While
Return G*

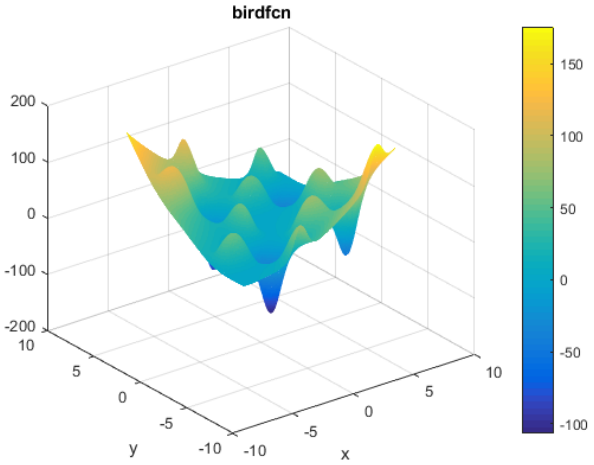
Test Functions

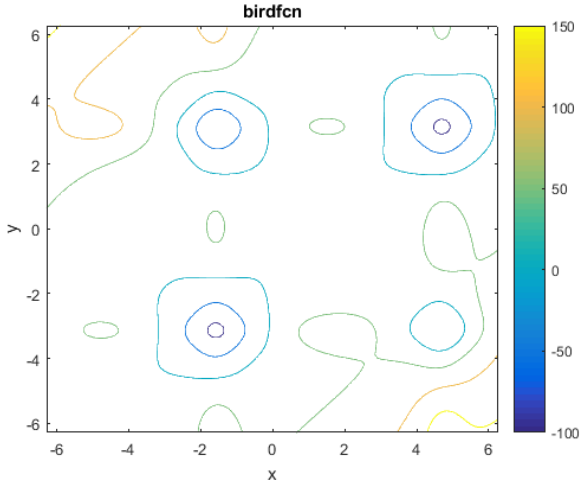
- Test functions are used to identify the types of problems the algorithm is able to solve
- We ran each of the algorithms 100 times, recording the results, and finding the mean function evaluation, mean minimum of the objective function, and the variance of the minimum
- The stopping criteria used was when the absolute value difference of the global best and previous global best is less than a tolerance value denoted *tol*
- For PSO and HPSOIFv2s we used $tol = 1^{-6}$
- For HPSOIFv1s we used $tol = 1^{-3}$ before letting IF run

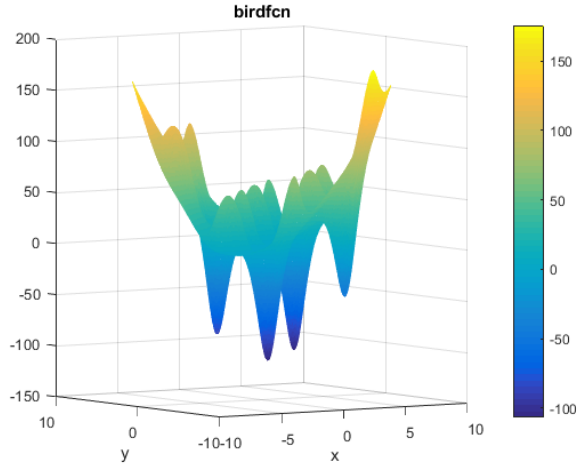
Bird Function:

$$f(x,y) = \sin(x)\, e^{(1-\cos(y))^2} + \cos(y)\, e^{(1-\sin(x))^2} + (x-y)^2$$

- Function Features: 2-Dimensional, Not Convex, Non-Separable, Differentiable
- Typically Evaluated on the domain $x, y \in [-2\pi, 2\pi]$
- Two Global Minima







Bird Function:

$$f(x,y) = \sin(x)\, e^{(1-\cos(y))^2} + \cos(y)\, e^{(1-\sin(x))^2} + (x-y)^2$$

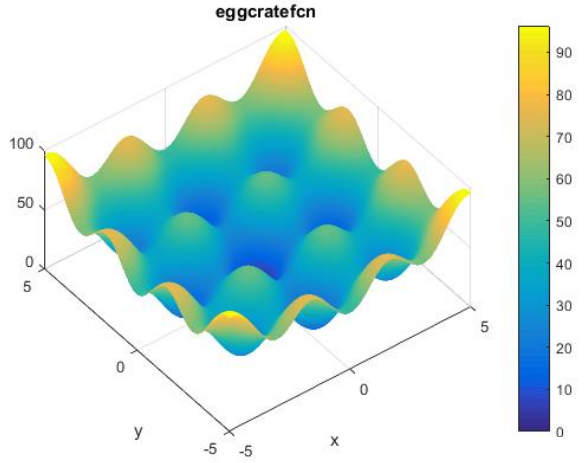
- Function Minima Location: $(4.70104, 3.15294), (-1.58214, -3.13024)$
- Minimum Function Value: $f(x^*, y^*) = -106.764537$

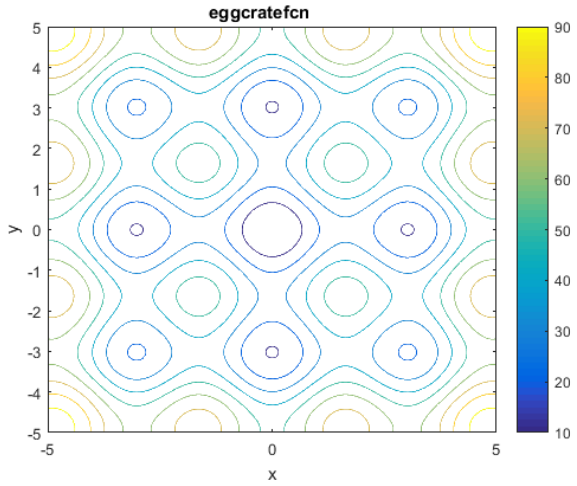
Algorithm	Average Function Evaluation	Average Minimum Found	Variance of Minimum Found
PSO	1665.3	-105.2082334	28.13498624
PSOIFv1	1881.4	-106.1783347	11.12374976
PSOIFv1.S	1905.09	-105.2045159	28.12531538
PSOIFv2	1697.7	-106.7645326	1.21E-10
PSOIFv2.S	1677.87	-105.0793079	28.84432231

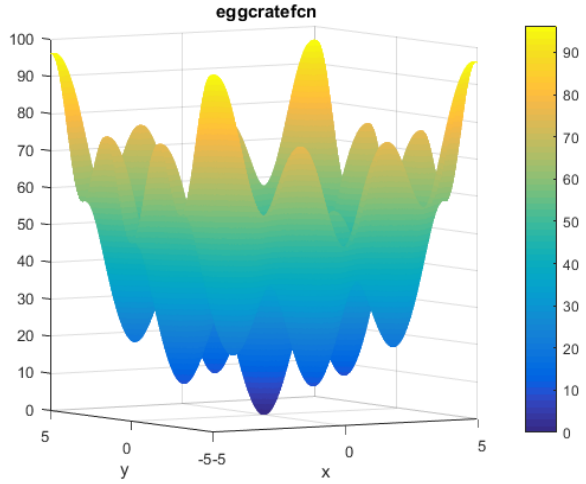
Egg Crate Function:

$$f(x, y) = x^2 + y^2 + 25(\sin^2(x) + \sin^2(y))$$

- Function Features: 2-Dimensional, Continuous, Not Convex, Separable, Multimodal, Differentiable
- Typically Evaluated on the domain $x, y \in [-5,5]$
- One Global Minimum







Egg Crate Function:

$$f(x,y) = x^2 + y^2 + 25(\sin^2(x) + \sin^2(y))$$

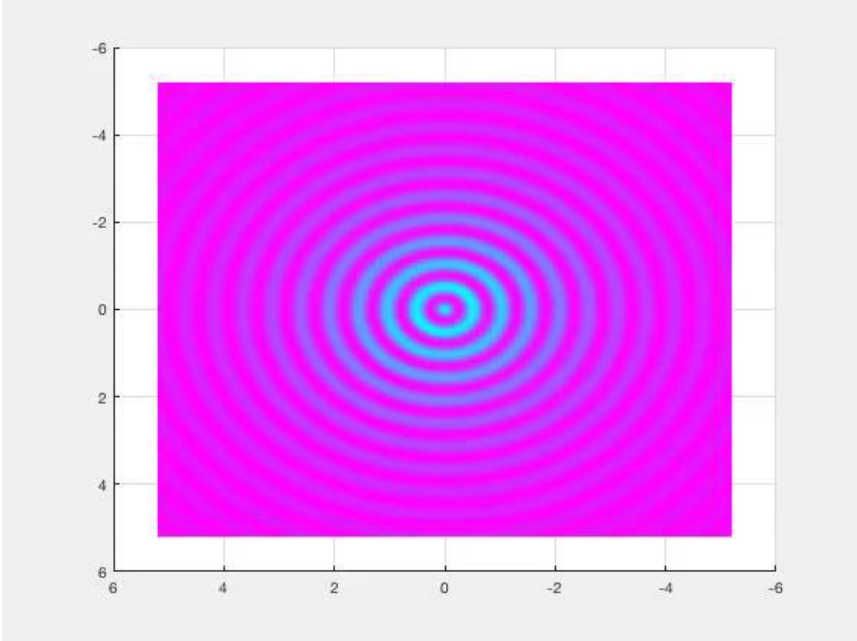
- Function Minimum Location: $(0,0)$
- Minimum Function Value: $f(x^*, y^*) = 0$

Algorithm	Average Function Evaluation	Average Minimum Found	Variance of Minimum Found
PSO	1429.5	0.094893	0.900257
PSOIFv1	1661.2	0.10364	0.94753
PSOIFv1.S	1711.08	0.002178	3.59E-05
PSOIFv2	1418.4	0.000506	2.44E-05
PSOIFv2.S	1390.24	0.126328	0.868686

Drop Wave Function:

$$f(x,y) = -\frac{1 + cos(12\sqrt{x^2 + y^2})}{(0.5(x^2 + y^2) + 2)}$$

- Function Features: 2-Dimensional, Continuous, Not Convex, multimodal
- Typically Evaluated on the domain $x, y \in [-5.2, 5.2]$
- One Global Minimum



Drop Wave Function:

$$f(x,y) = -\frac{1 + cos(12\sqrt{x^2 + y^2})}{(0.5(x^2 + y^2) + 2)}$$

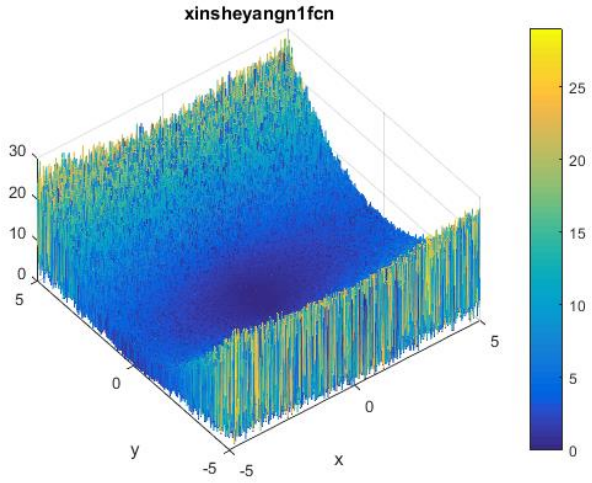
- Function Minimum Location: $(0,0)$
- Minimum Function Value: $f(x^*, y^*) = -1$

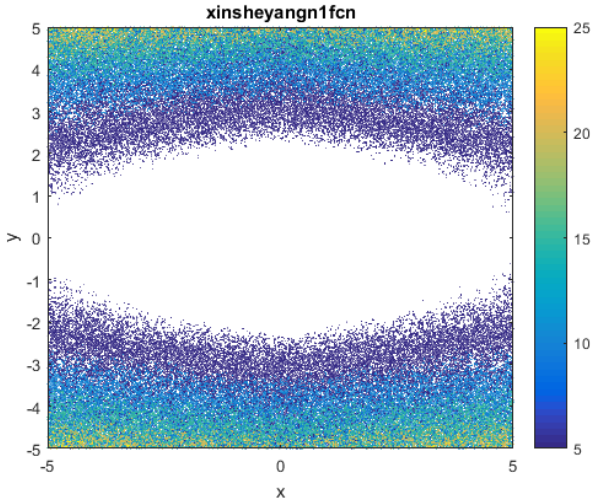
Algorithm	Average Function Evaluation	Average Minimum Found	Variance of Minimum Found
PSO	1881.3	-0.97513	0.000977
PSOIFv1	1113.7	-0.86327	0.012462
PSOIFv1.S	1119.19	-0.86732	0.012253
PSOIFv2	1500.9	-0.98339	0.00079
PSOIFv2.S	1481.42	-0.97876	0.001346

Xin-She Yang Function:

$$f(\mathbf{x}) = f(x_1, \ldots, x_n) = \sum_{i=1}^{n} \epsilon_i |x_i|^i$$

- ε is drawn from uniform distribution on [0,1]
- Function Features: 2-Dimensional, discontinuous, Not Convex, Separable, Not Differentiable
- Typically Evaluated on the domain $x, y \in [-10,10]$
- One Global Minimum





Xin-She Yang Function:

$$f(\mathbf{x}) = f(x_1, \ldots, x_n) = \sum_{i=1}^{n} \epsilon_i |x_i|^i$$

- Function Minima Location: $(0,0)$
- Minimum Function Value: $f(x^*, y^*) = 0$

Algorithm	Average Function Evaluation	Average Minimum Found	Variance of Minimum Found
PSO	8360	8.95E-06	6.03E-10
PSOIFv1	1256	0.426006	0.531639
PSOIFv1.S	1255.42	0.784851	2.327926
PSOIFv2	3307.2	9.57E-06	2.05E-09
PSOIFv2.S	4006.96	4.94E-06	9.91E-11

Future Work

- HPSO worked well with non-convex functions
- Want to test on higher dimensional objective functions
- Test algorithms in real applications
- Develop new criteria for when to run IF

Refrences

Implicit Filtering, CT Kelly

Particle Swarm Optimization, Maurice Clerc

A First Course in Optimization Theory, Rangarajan K. Sundaram

HPS and Optimization, Ahmad Almomani