# Particle Swarm Optimization

April 16, 2026

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Steven Dao
Laura Smith
Jonathan Tran

1. What is a possible downside of high exploitation?

2. What problem does Guaranteed Convergence PSO (GCPSO) attempt to solve?

3. Which field is PSO research currently mostly concentrated in?

# Quiz

THE UNIVERSITY OF TENNESSEE KNOXVILLE

# Introductions

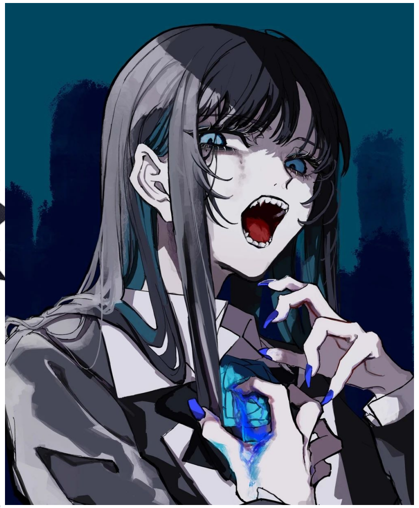
# Jonathan Tran

- Born in Chattanooga, TN
  - Known as the Scenic City and Gig City
- Vietnamese and Chinese
- Attended Ooltewah High School
- BS in Computer Science at UTK 2025
  - Minor in Cybersecurity
- Completing 5-year MS/BS in CS

# Jonathan Tran



- Hobbies
  - Cycling, going to conventions, sleeping…
  - Looking to get more into chess, cooking, learn an instrument
- Games
  - Persona 5, SSBU
- Anime
  - Mob Psycho 100, A Silent Voice
- Reading
  - Manga, Manhwa, Light Novels
- Music
  - Hoshimachi Suisei, Ado

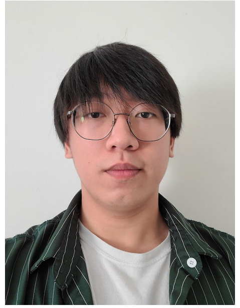


# Steven Dao

- Born and raised in Nashville, TN
  - First graduating class of my high school

- Finishing up 5YR CS MS/BS @ UTK 2026
  - Cybersecurity Minor

- CS PhD coming UTK Fall 2026
  - Advisor: Michela Taufer w/ Global Computing Lab

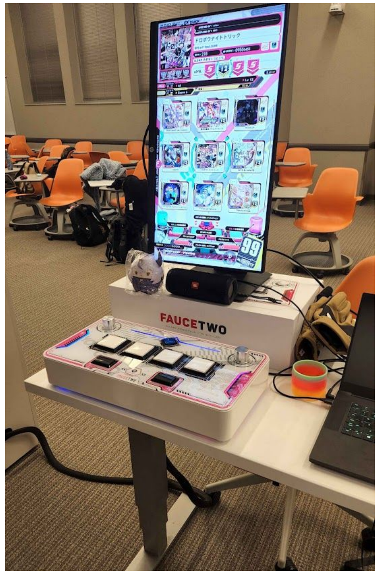




# Steven Dao

- Vietnamese Name: Minhtrung
  - Loosely means “Truthful Soul”
- Plays Shamisen
  - Shamisen Knoxville
- I was born at 5:45PM and my brother at 5:45AM.
- Hobbies: Hacky Sack, Karaoke, Rhythm Games
  - I have karaoke equipment stored in my car right now.

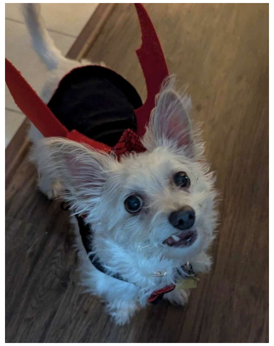


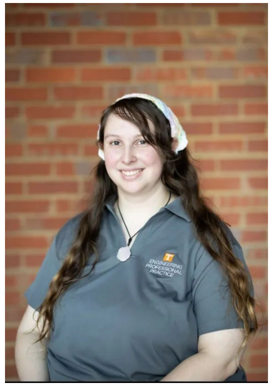






THE UNIVERSITY OF TENNESSEE KNOXVILLE

# Laura Smith



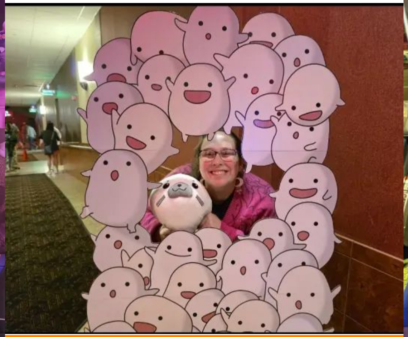

- Born in Dallas, Texas but moved to Colorado at a young age
- Did the IB program in high school at Douglas County High School

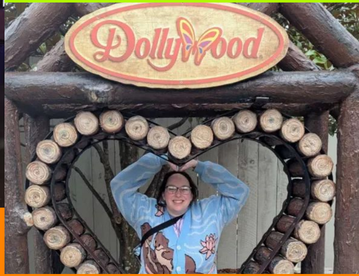
- Bachelors of science in Computer Science here at UTK 2025
  - minors in cybersecurity and machine learning and a concentration in scientific computing
- Current 5-year Master’s student
  - August 2026
- Continuing to the PhD in the fall
  - Advisor: Dr. Rachael Burns in the STARI lab
- Research in Socially Assistive Robotics - Quad4Elders
- Helping out with the FujiView research as well

# Laura Smith

- My main hobbies are tennis, art, playing video games, reading books and manga, and going to the theatre
- I love trying new things and have had so many hobbies
  - Sewing, knitting, crocheting, cross stitching, jewelry making, metal work, pottery, making stained glass art, 3D printing, MTG, Pokemon, Beyblade, Yugioh, cooking, fencing, making costumes, prop making, leather work, musical theatre, martial arts, etc.
- I run the manga reading club every week



# Outline

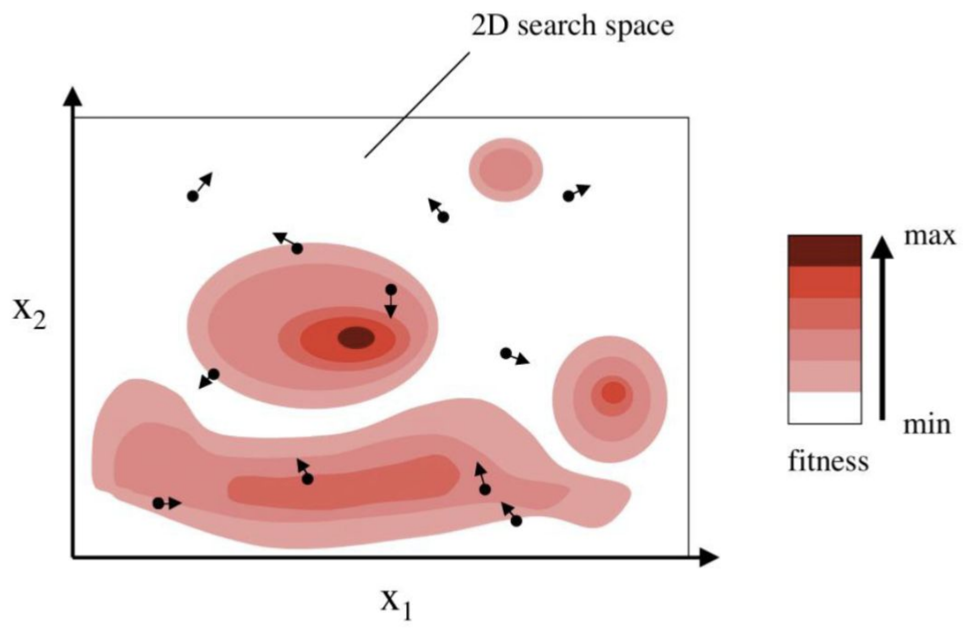
- Overview and Definitions
- History of Particle Swarm Optimization
- Algorithm Specifications
- Deep Dive into Applications
- Implementations of PSO
- Open Issues
- Discussion
- References

# Overview and Definitions - Birds

- Particle Swarm Optimization (PSO) is an optimization algorithm inspired by **bird flocking behavior**
  - Continuous optimization search
- Model for social information sharing
  - Cooperative search style
- Combines
  - Public knowledge
  - Private knowledge

# Overview and Definitions - PSO

- Search space
- Particles
  - Position
  - Velocity and momentum
  - Hyperparameters
    - Social bias
    - Cognitive bias
    - Inertia
- Neighbor relations
  - How is information shared?

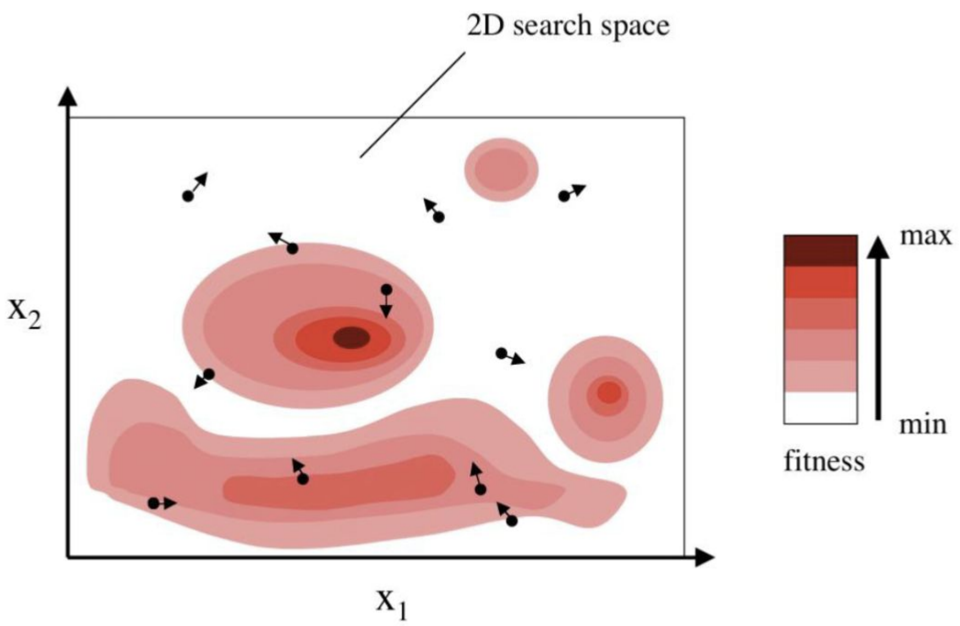


# Overview and Definitions - Search

- Exploration
  - Global Search
  - High exploration may never settle on a final answer

- Exploitation
  - Local Search
  - High exploitation may cause premature convergence

- Performance Metrics
  - Convergence Time
  - Fitness
  - Success Rate



# Benefits of PSO

- Relatively simple to implement
- Doesn’t require gradient information or continuity
  - Good for Black-box problems
- Few hyperparameters
- Highly efficient and parallelizable calculations
- Relatively fast convergence
  - Finds a “good enough” answer quickly

# Drawbacks

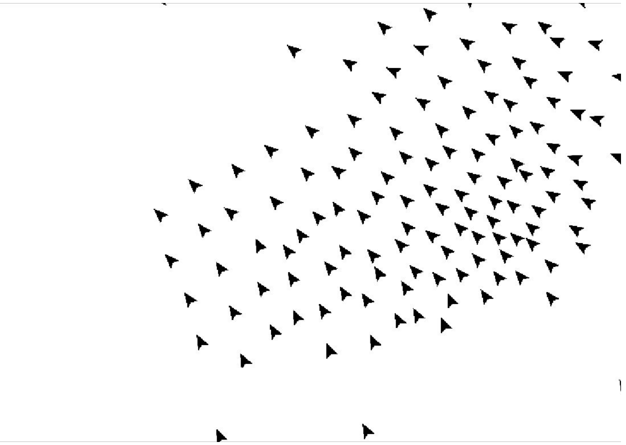
- Can be difficult to transform the parameters of a problem to be solved so that it can be encoded and searched by particles

- Risk of premature convergence (local optima)
- High hyperparameter sensitivity

# History

# History

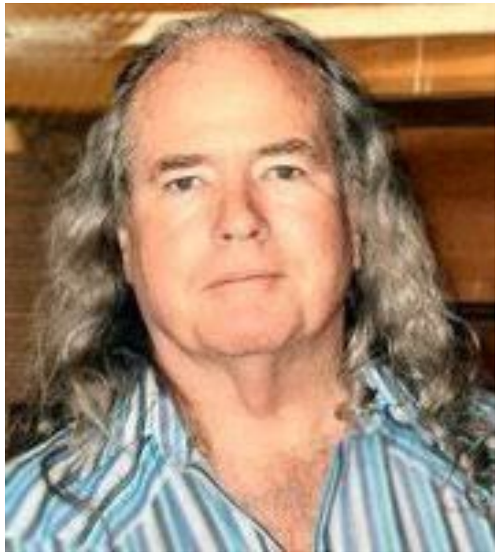
- **James Kennedy and Russel Eberhart** developed PSO in **1995**
- Inspired by **Craig Reynold's Boids model** showing flocking and collective motion
- Kennedy and Russel first started to add **food sources** to the model, then realized it could be used for **optimization**



Flocking Boids

Craig W. Reynolds
Symbolics Graphics Division

Dr. Russel C. Eberhart
Purdue School of Engineering and Technology



Dr. James Kennedy
Bureau of Labor Statistics

1995

**James Kennedy & Russell Eberhart** introduce **PSO** based on social behavior and flocking

1998

**Eberhart** works with **Yuhui Shi** to introduce **inertia to stabilize convergence**

$$\omega > \frac{1}{2}(\varphi_1 + \varphi_2) - 1.$$

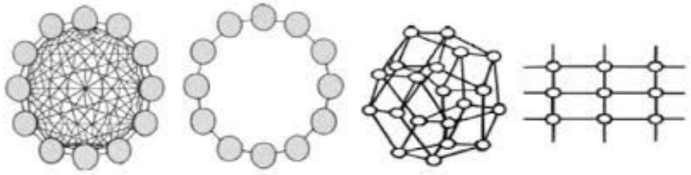
1999

**Maurice Clerc** introduces a **constriction factor** to help **PSO solve faster**

$$K = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|}$$

1999

**Kennedy** tests **static neighborhood topologies** (ring, star, von neumann,



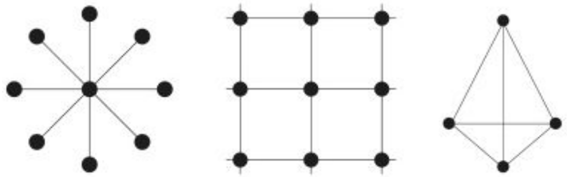
(a) Fully-connected (b) Ring (c) Von Neumann

2000

**P.N. Suganthan** proposes a **dynamic neighborhood** where particles become neighbors when they are close to each other

2000

**Kennedy** introduces **social stereotyping** using clustering behavior



(a) Start (b) Von Neumann (c) Pyramid

**2002**

**Kennedy & Ricardo Mendes** compared swarm topologies and **found von neuman performed best**

**2002**

**Frans Van den Bergh** identified **the stagnation problem** and **proposed Guarenteed Convergence PSO (GCPSO)**

**2003**

$$FDR = \frac{f\left(\vec{P}^{\,j}_{t}\right) - f\left(\vec{x}^{\,i}_{t}\right)}{\left|\left(\vec{P}^{\,j}_{t}\right)_d - \left(\vec{x}^{\,i}_{t}\right)_d\right|}$$

**Kalyan Veermachaneni et al.** introduced the **fitness-distance ratio (FDR)** to choose neighbors based on **solution quality and distance**

**2004**

**Mendes et al.** developed **Fully Informed Particle Swarm (FIPS) Optimization Algorithm** using **all neighbors instead of one best one**

**2004-2007**

**Brits, Parsopoulos, and Vrahatis** introduced **niching, multi-swarm, and multimodal optimization methods**

**2009**

**Zhan et al.** proposed **adaptive PSO (APSO)** with dynamic parameter tuning

$$D^{i}_{t} = \frac{1}{s-1}\sum_{j=1, j\neq i}^{s}\sqrt{\sum_{i=1}^{d}\left(\vec{x}^{\,i}_{t} - \vec{x}^{\,j}_{t}\right)^2}$$

# Modern Work

2010-2020

- **Adaptive PSO** (APSO), dynamic parameter tuning
- **Multi-objective PSO** (MOPSO) and Pareto Optimization
- **Cooperative PSO**, multi-swarm Methods
- **Niching**, **multimodal PSO**, **species-based PSO**
- **Parallel PSO**, **GPU** impelentations, **big data PSO**

2020-Today

- **Adaptive and learning-based PSO**
- **Niching**, **multimodal**, and **multi-objective PSO** improvements
- **Parallel**, **distributed**, and **multi-agent PSO**
- **Speciation** and **diversity-preserving PSO**

# Algorithm Description

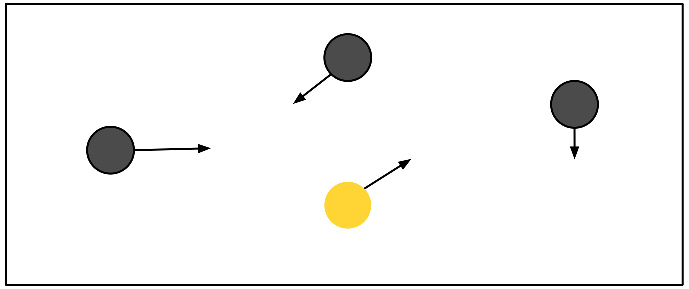
# Overview

PSO is stochastic, iterative, and evolutionary.

1. Initialize
2. Step
   a. Determines new solution candidates and new fitness scores
   b. Update personal bests and global best
3. Repeat until some stop condition

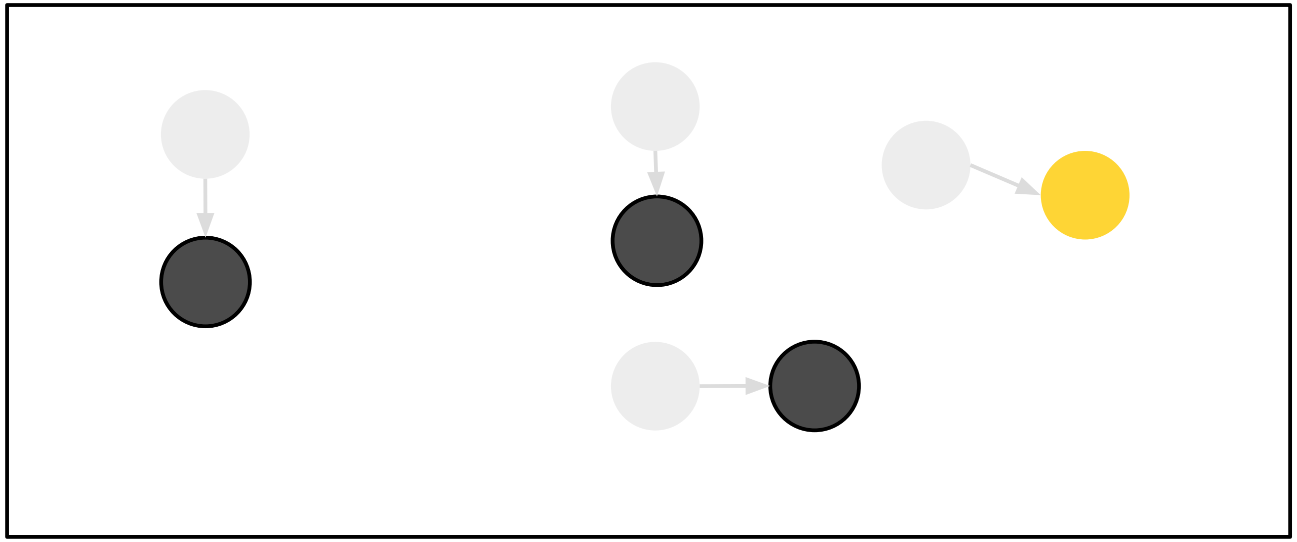
# Initialization

1. We start with some number of particles, each representing a solution candidate.
2. Randomly initialize each particle's position and velocity in the solution space.
3. Calculate each particle's fitness.
    a. This also becomes each particle's "personal best fitness".
4. Determine which particle has the best fitness to find current best solution candidate.



# Step

1. Calculate new velocity.
    a. Depends on old velocity, personal bests, and global best.
2. Add position and new velocity to determine new position.
3. Recalculate fitness
4. Update personal bests.
5. Update global best.



# Calculating New Velocity

$$V_{new} = \omega V_{old} + c_1 r_1 (p_{best} - x) + c_2 r_2 (gbest - x)$$

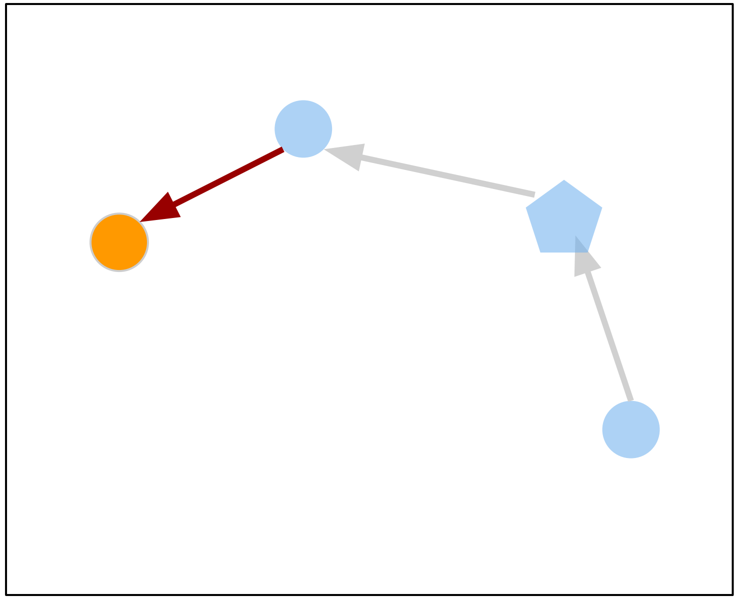
Hyperparameters:

- omega: inertia weight
- c1: cognitive weight
- c2: social weight

# Calculating New Velocity

$$V_{new} = \boxed{\omega V_{old}} + c_1 r_1 (p_{best} - x) + c_2 r_2 (gbest - x)$$

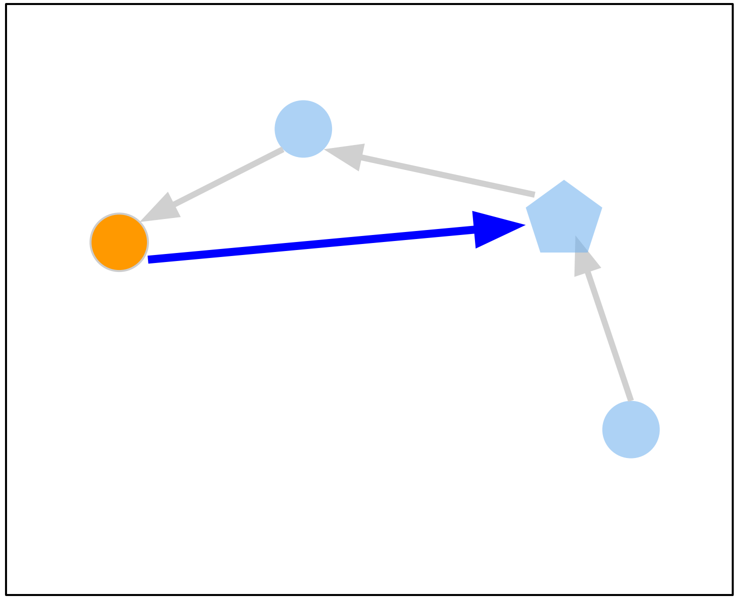
- Original random direction of exploration.
- omega can be described as a dampening factor.



# Calculating New Velocity

$$V_{new} = \omega V_{old} + \boxed{c_1 r_1 (p_{best} - x)} + c_2 r_2 (gbest - x)$$

- r1 is randomized between [0, 1] each step.
- c1, the cognitive bias, scales tendency to return back to known personal best



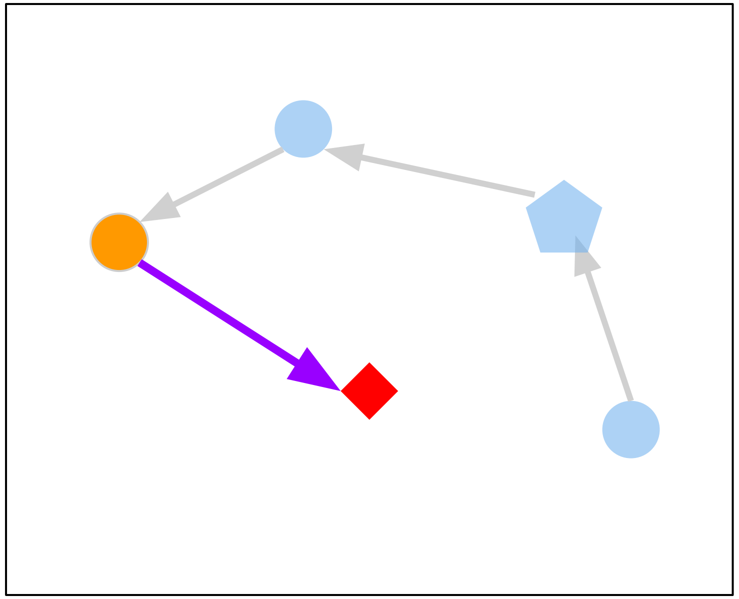


THE UNIVERSITY OF TENNESSEE KNOXVILLE

# Calculating New Velocity

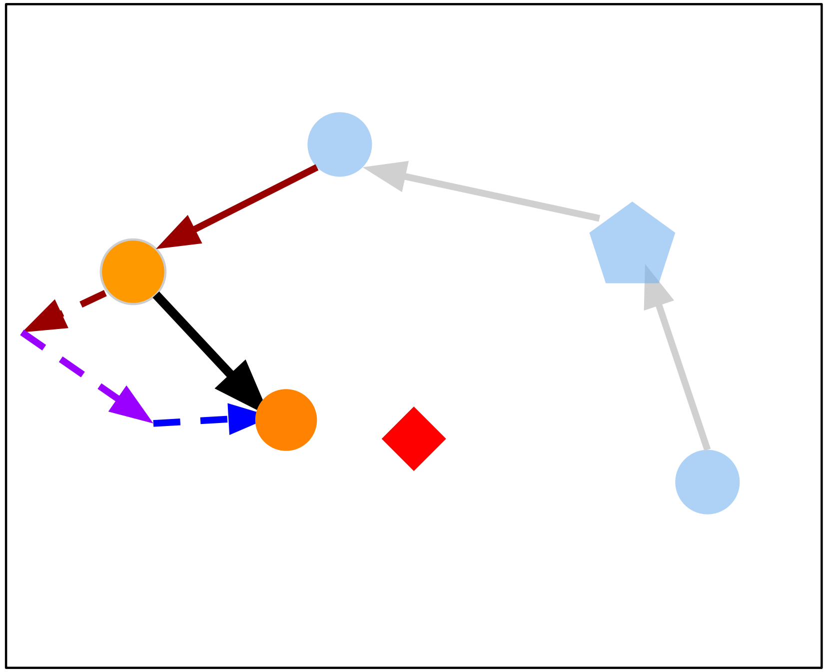
$$V_{new} = \omega V_{old} + c_1 r_1 (p_{best} - x) + \boxed{c_2 r_2 (gbest - x)}$$

- r2 also randomized between [0, 1] each step.
- c1, the social bias, scale tendency to go to globally known best



# Calculating New Position

$$P_{new} = P_{old} + V_{new}$$



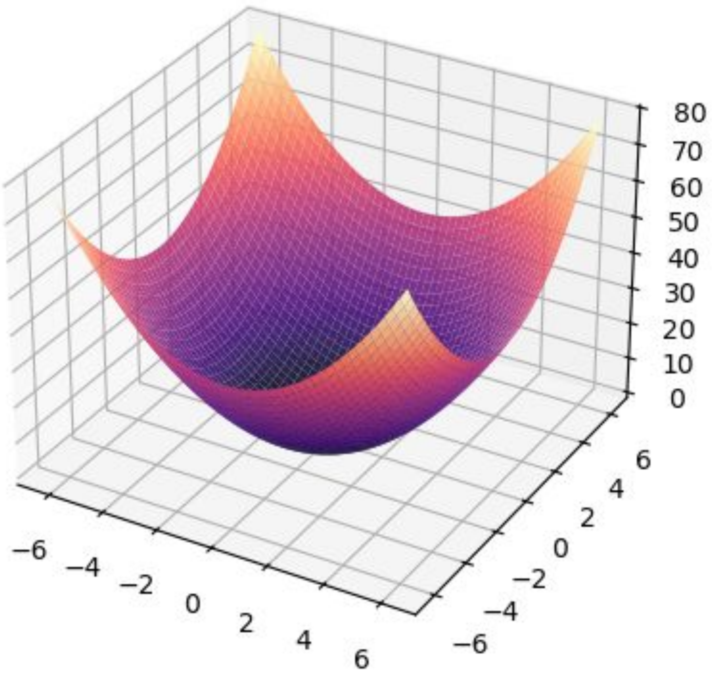
# Stopping Conditions

- Fixed Iteration Count
- Global Best Stagnation
- Inter-swarm Variance

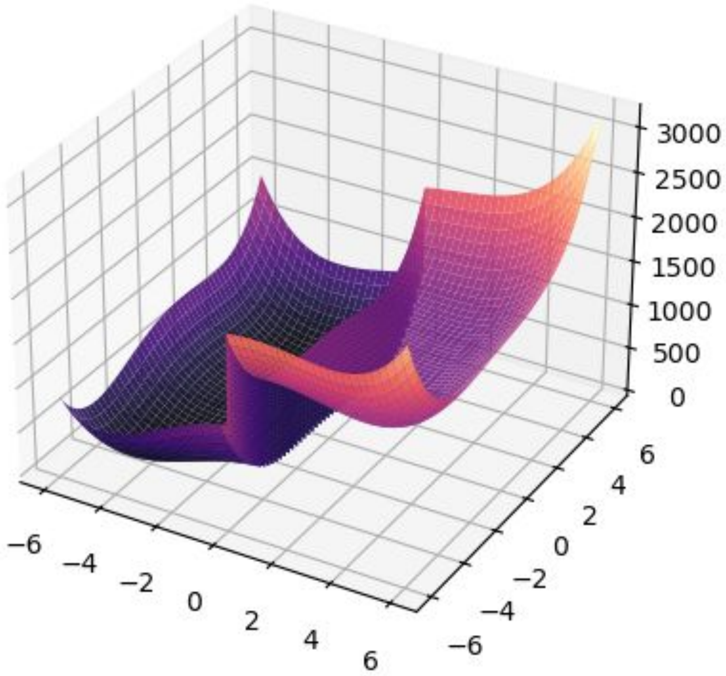
Which do we use? It depends on the application.

# Applications

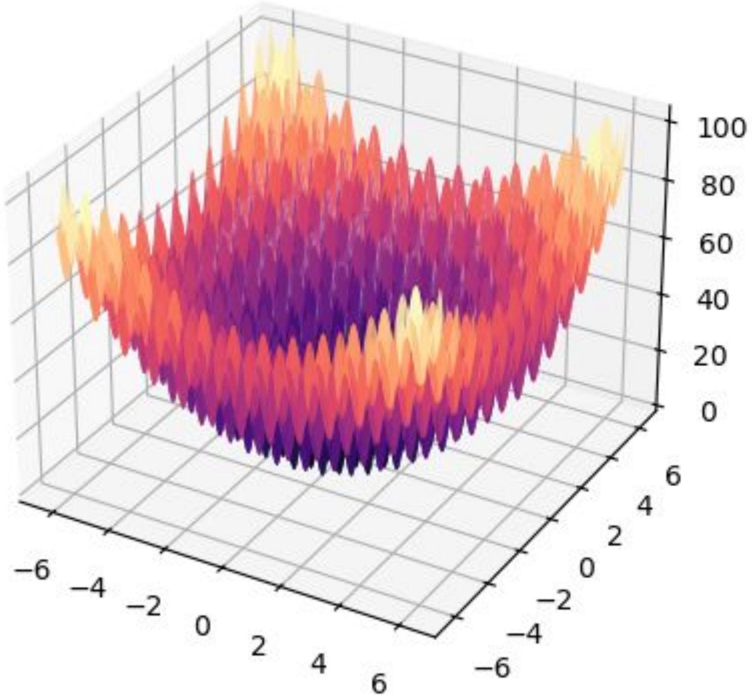
# Finding Optima of Functions




trivial




discontinuous



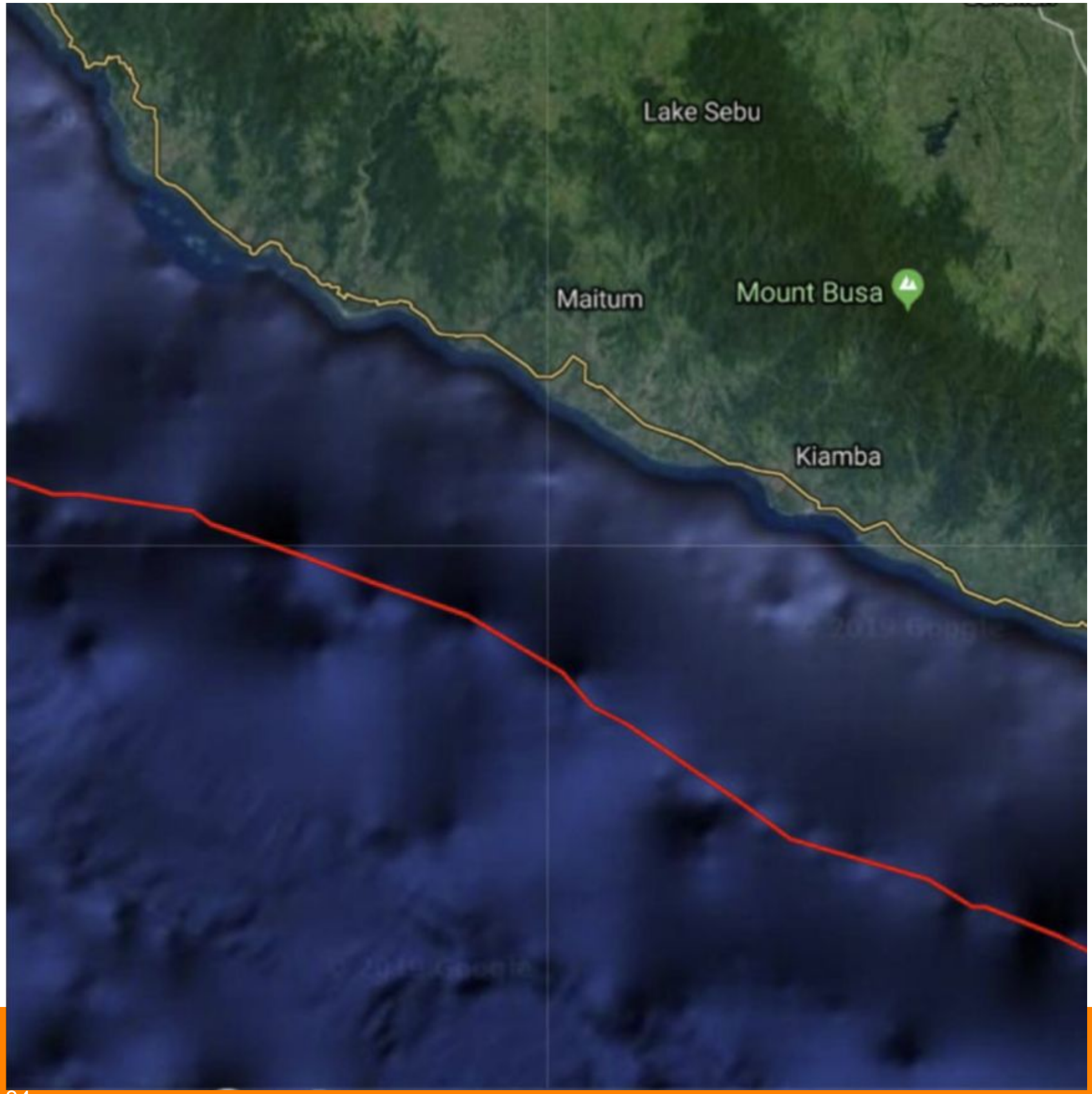

noisy

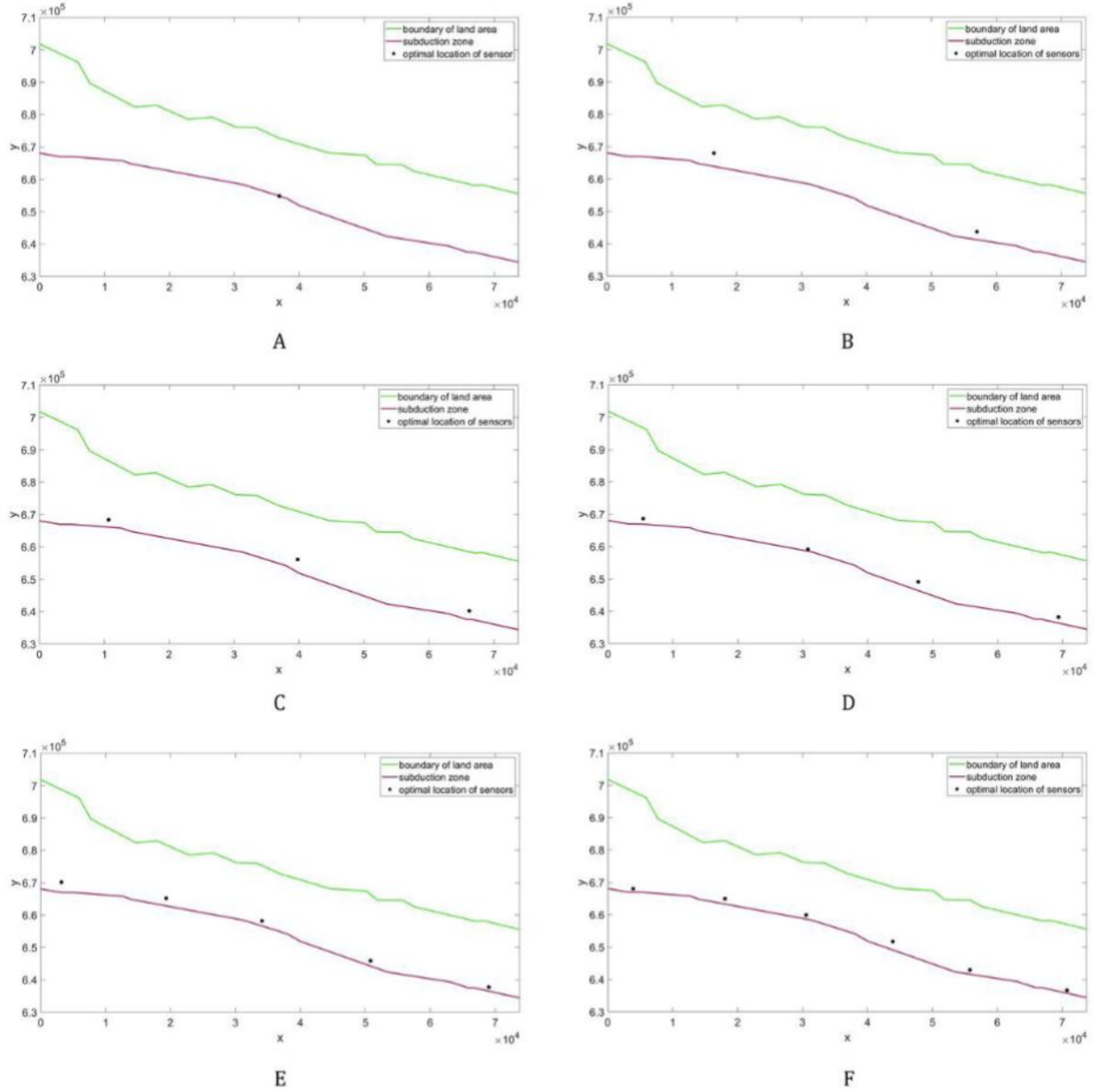
# Mobile Robot Navigation

- Automated mobile robots must navigate through terrain
  - Pathing can be difficult to determine
- Can be optimized with PSO

- Particle: a potential path or coordinate sequence
- Search space: physical environment

# Optimal placement of tsunami sensors



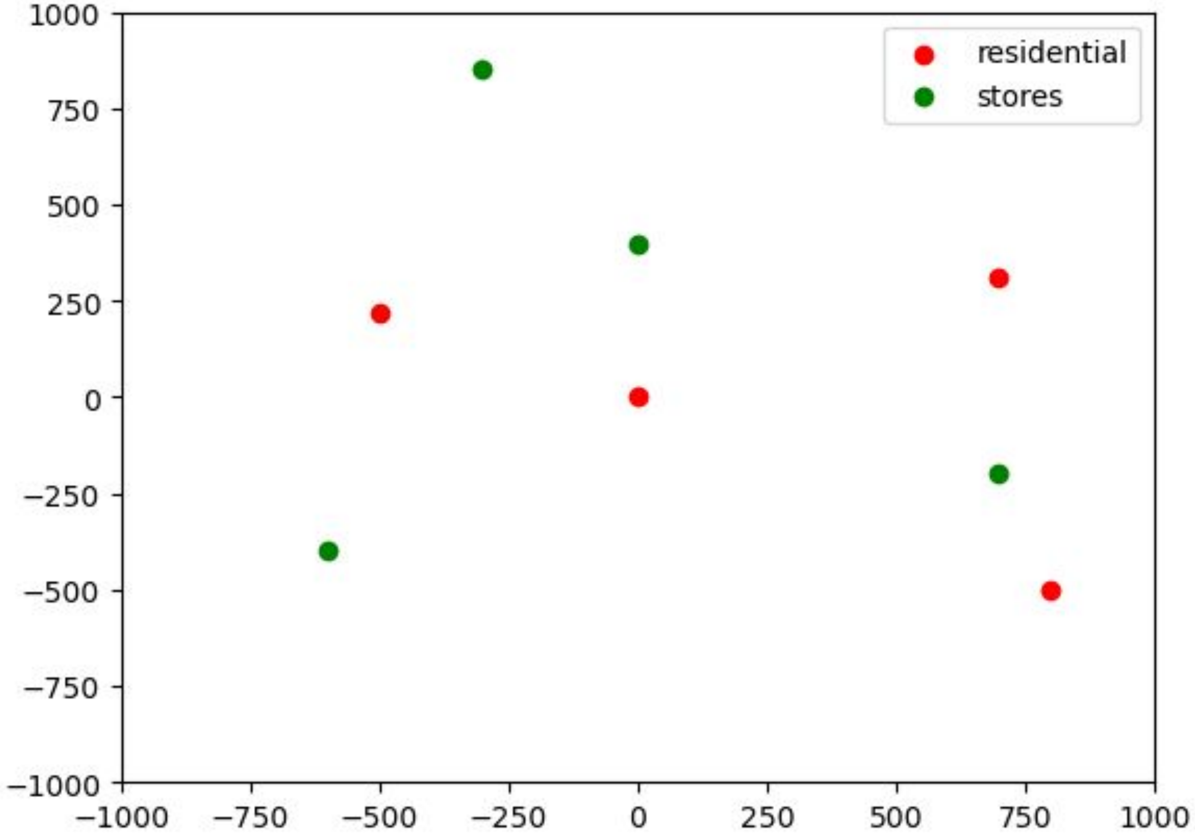
# Toy Example - Warehouse Placement

Supposed we have an area of land with residential areas and stores. We want to choose locations for 3 warehouses that…

- are close to stores
- are far from residential areas
- spreads the warehouses out
  - We don't want all the warehouses in one spot

# Implementation

# Warehouse Placement




- Continuous 2000x2000 area
- 4 stores
- 4 residential areas

# Warehouse Placement

- Converting it to a PSO problem:
  - Encode each candidate solution as a 6D vector

$$[x_1, y_1, x_2, y_2, x_3, y_3]$$

  - Let's start by designing a fitness function for our goals
    - Maximize min distance from residential areas.
    - Minimize max distance to stores.
    - Maximize min distance between warehouses
    - We will MAXIMIZE

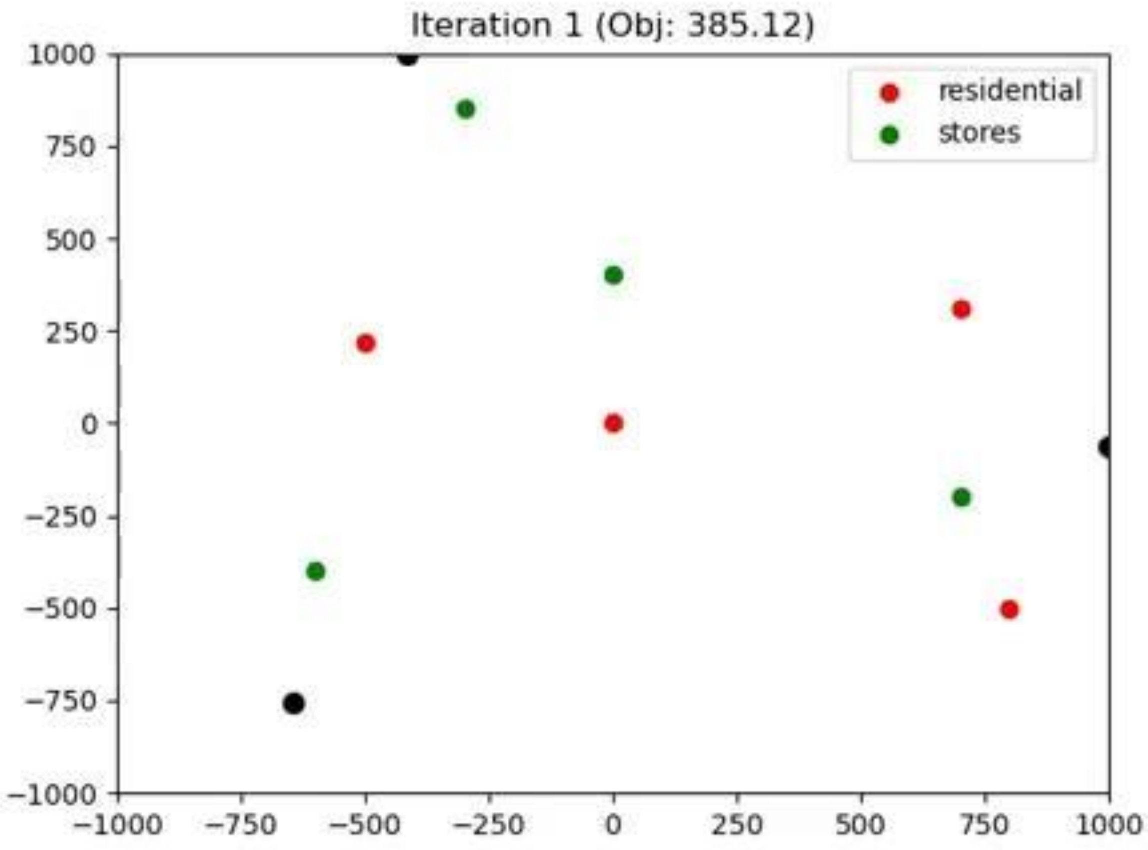
$$fitness = w_1 res_{min} - w_2 store_{max} + w_3 intra_{min}$$

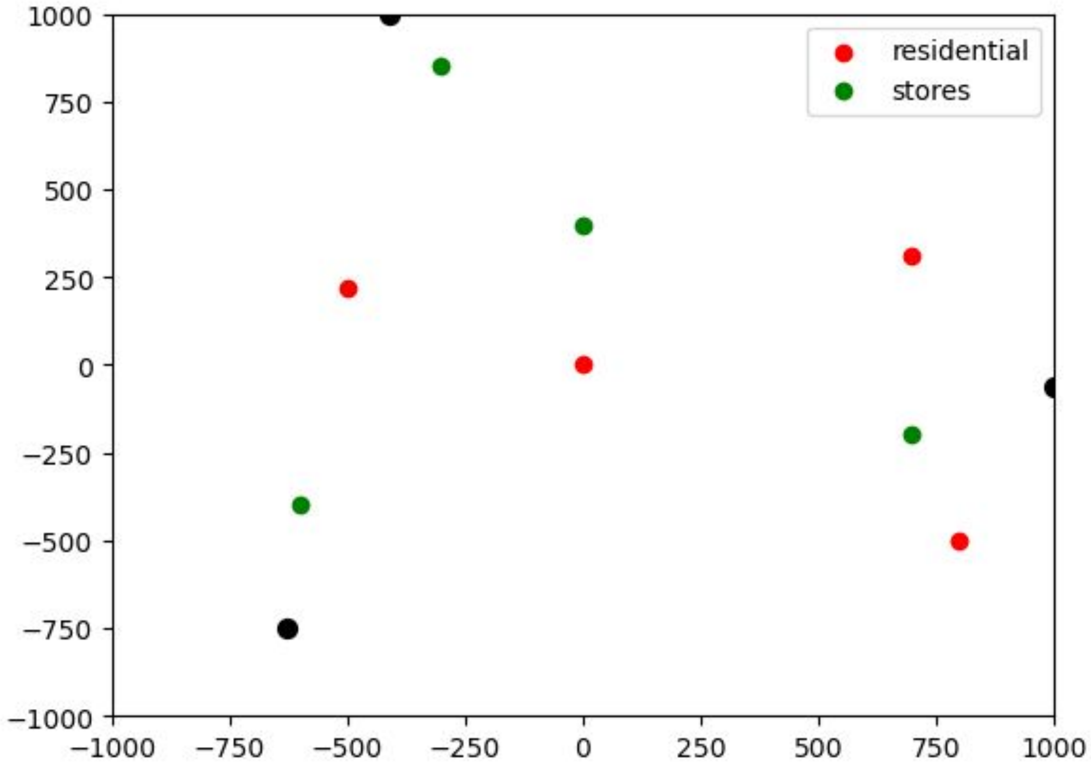


THE UNIVERSITY OF TENNESSEE KNOXVILLE

# Warehouse Placement

$$w_1, w_2, w_3 = 1$$



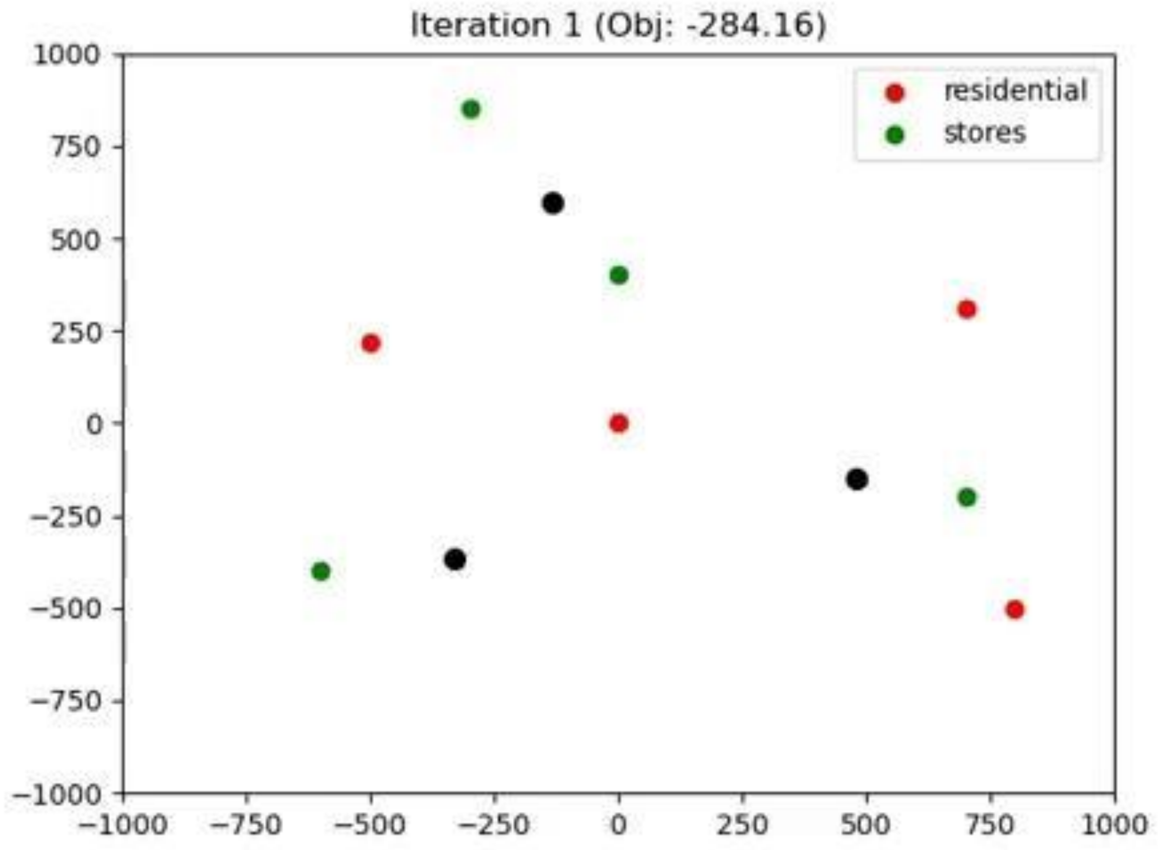



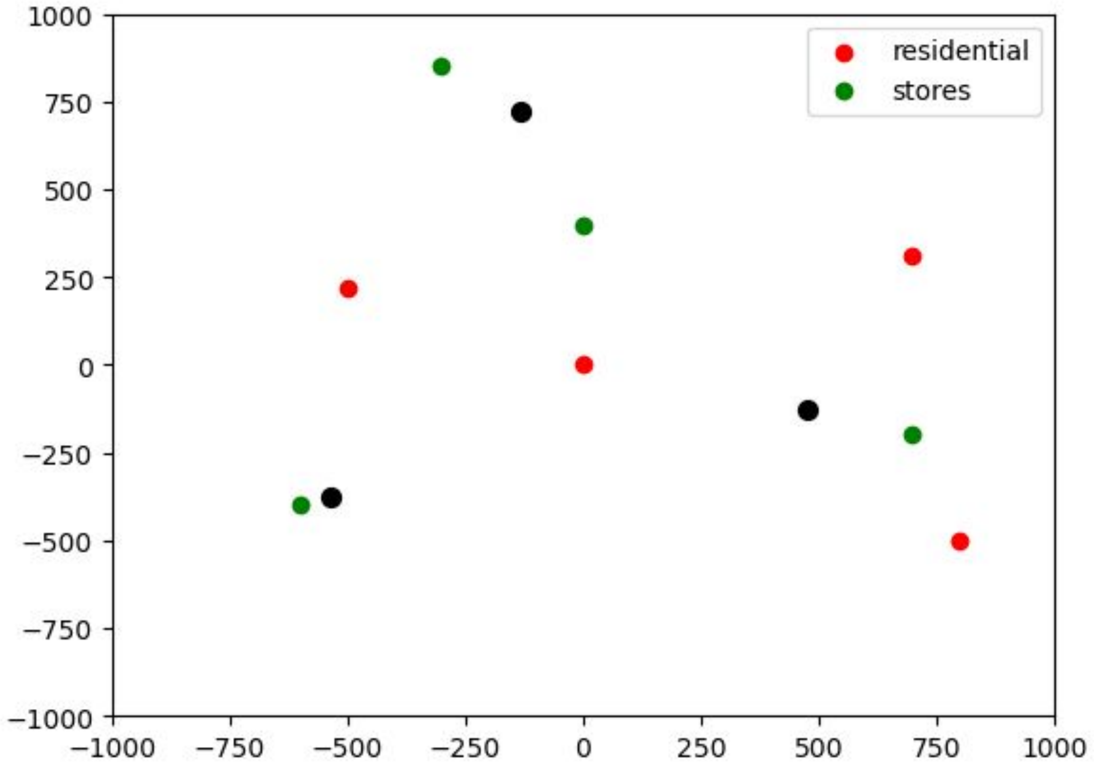



THE UNIVERSITY OF TENNESSEE KNOXVILLE

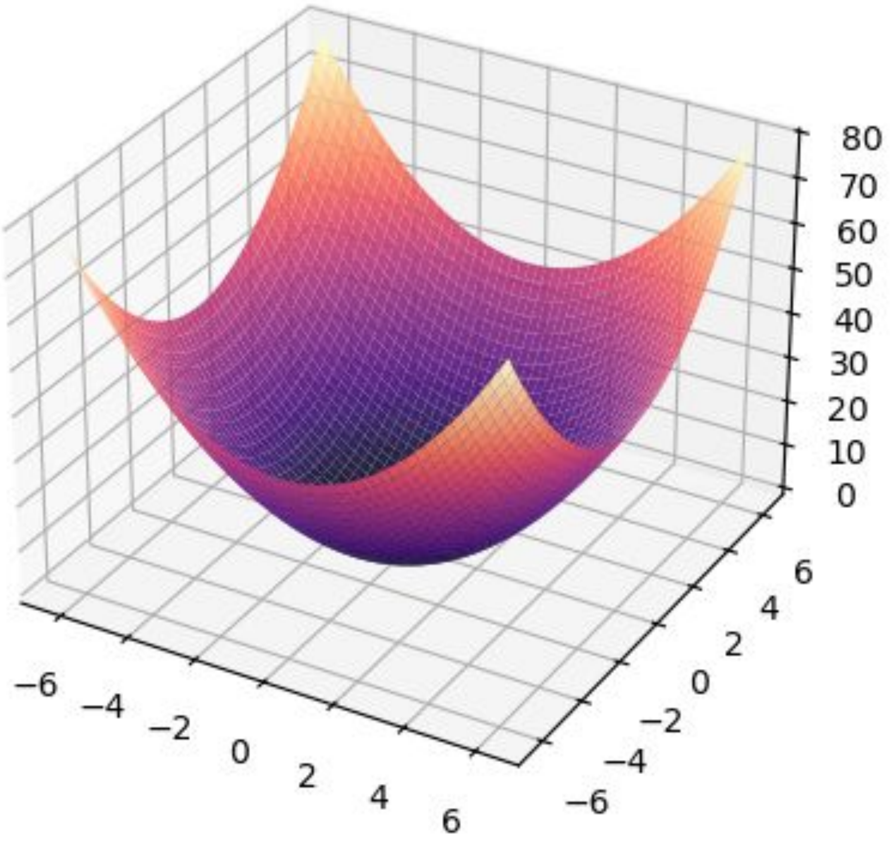
# Warehouse Placement

$w_1, w_2 = 1 \qquad w_3 = 0.6$



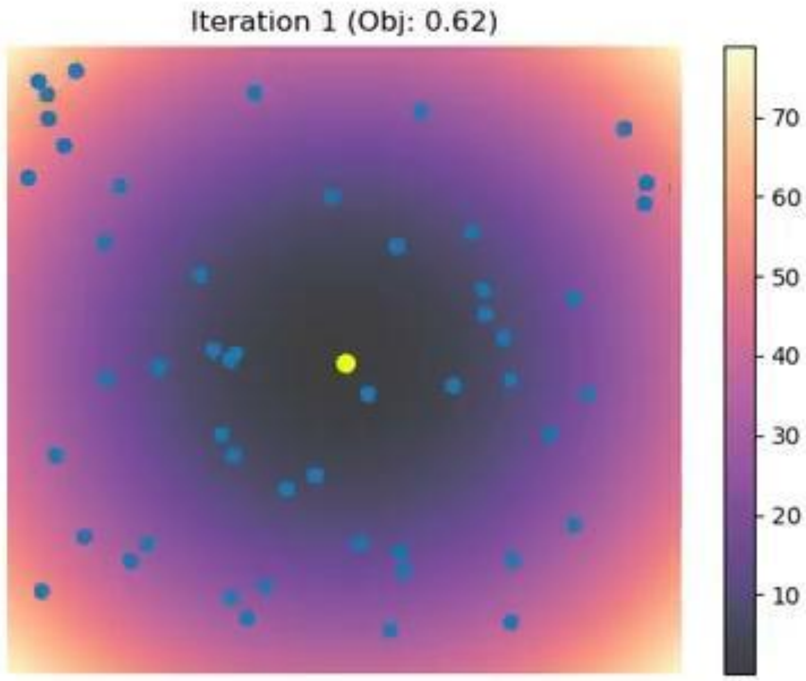



# Function Optimization




Sphere Function

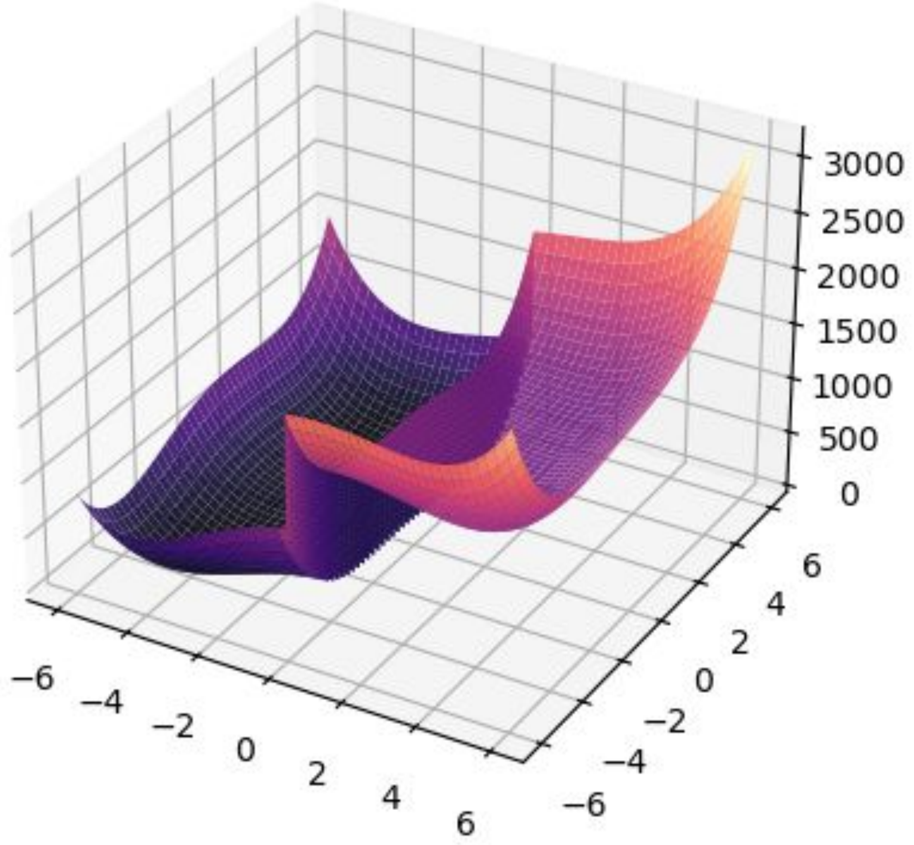
# Function Optimization



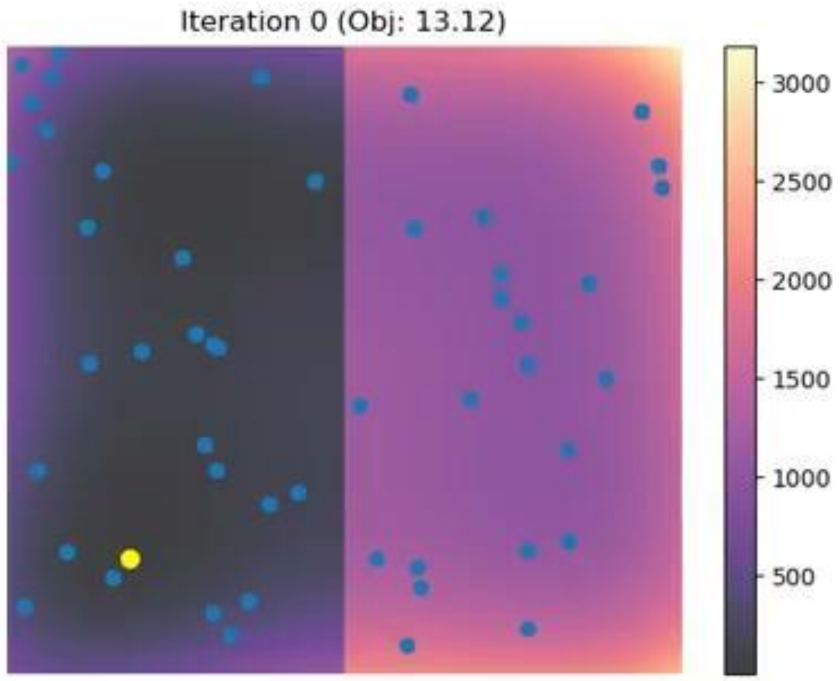



THE UNIVERSITY OF TENNESSEE KNOXVILLE

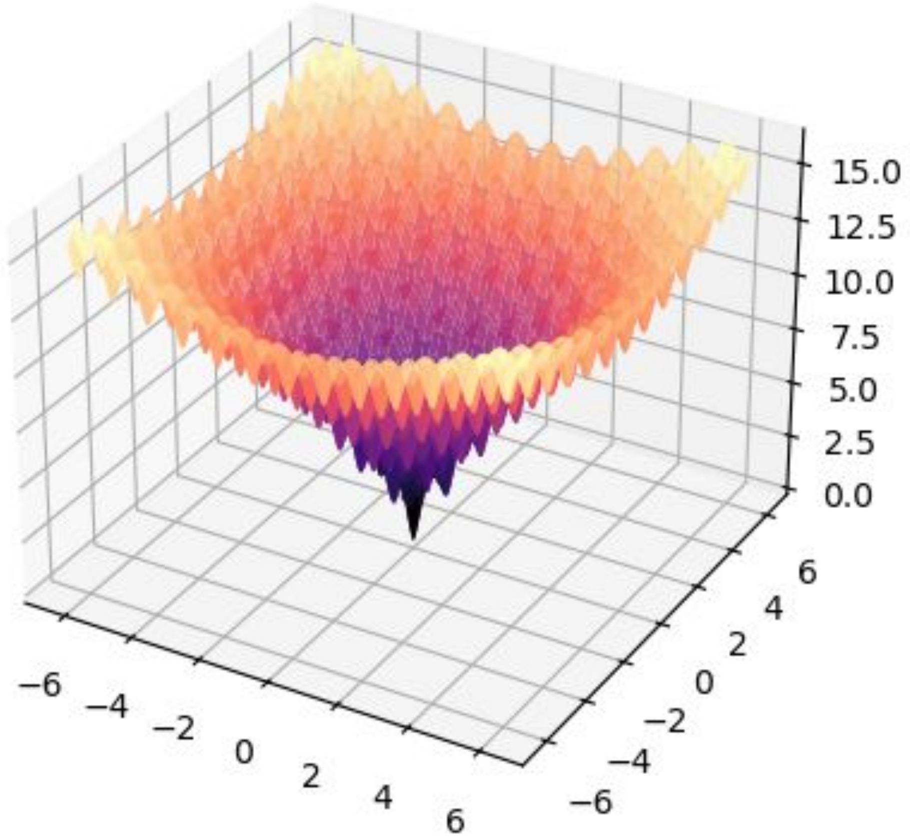
# Function Optimization



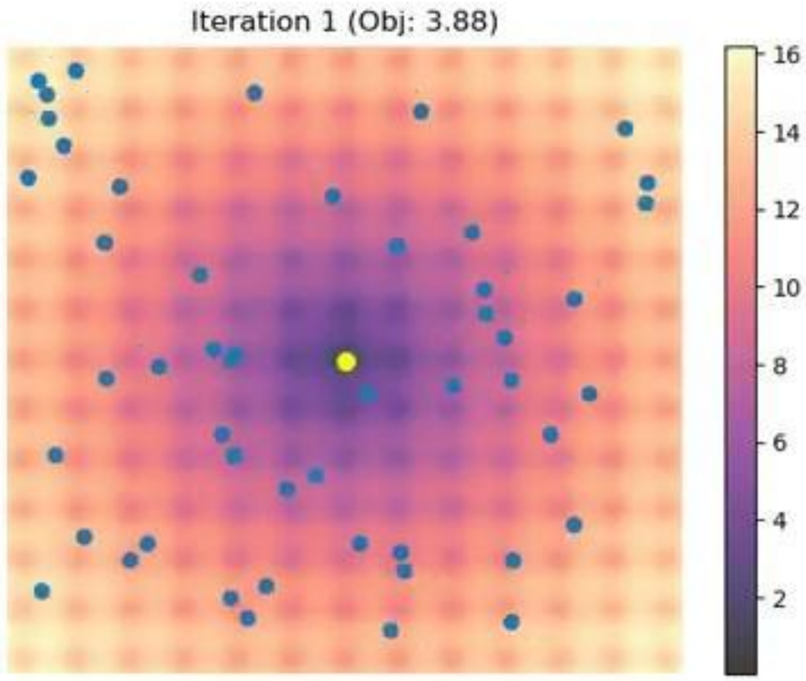

Discontinuous HimmelBlau



# Function Optimization




Ackley Function

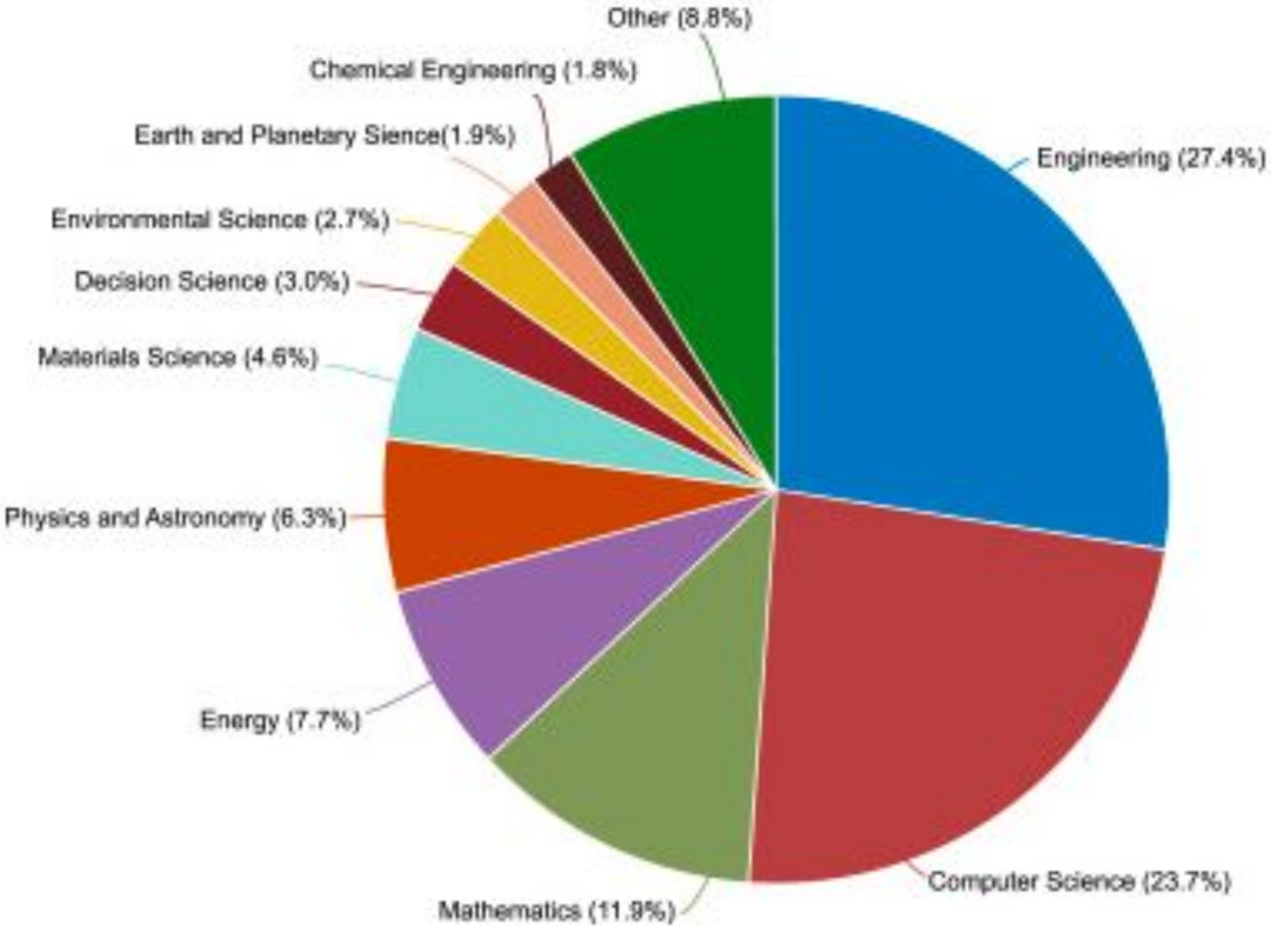


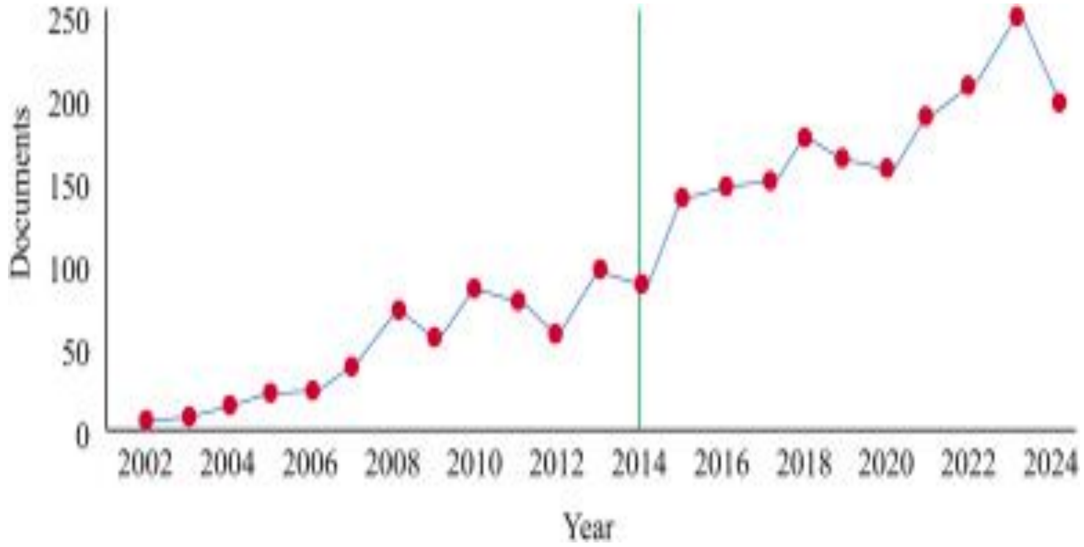
# Open Issues

# Algorithm Improvements

- Premature convergence
  - Local optimums
  - Especially bad in multimodal problems
- Exploration vs. Exploitation
  - Hard to balance
  - No universal setting works for all problems

- Parameter sensitivity
  - Required manual tuning
- Stagnation
- Scalability
  - Swarm is less effective as dimensions grow
- Loss of diversity
  - Reduces ability to explore new solutions




Images from: A. Casas-Ordaz *et al.*, “Particle swarm optimization: A survey of innovations over the last 10 years,” *Computer Science Review*, vol. 60, p. 100910, Feb. 2026, doi: https://doi.org/10.1016/j.cosrev.2026.100910.

## Machine Learning & AI

- Hyperparameter tuning
- Neural architecture search
- Feature selection

## Deep Learning Optimization

- Training neural networks as an alternative to gradient descent
- Weight initialization and tuning

## Internet of Things (IoT)

- Resource allocation
- Network optimization
- Energy-efficient routing

## Smart Grids & Energy Systems

- Power distribution optimization
- Renewable energy scheduling
- Load balancing

## Robotics & Autonomous Systems

- Path planning
- Multi-agent coordination
- Swarm robotics

## Bioinformatics & Healthcare

- Gene selection
- Protein structure prediction
- Medical image analysis

## Cloud Computing

- Task scheduling
- Resource allocation
- Load balancing

## Combinatorial Optimization

- Traveling Salesman Problem
- Scheduling problems
- Logistics optimization

A. Casas-Ordaz *et al.*, “Particle swarm optimization: A survey of innovations over the last 10 years,” *Computer Science Review*, vol. 60, p. 100910, Feb. 2026, doi: https://doi.org/10.1016/j.cosrev.2026.100910.

# Discussion

What’s your favorite potential use case?

Why do you think swarm behavior is a good model for solving optimization problems?

Why do you think randomness is important in PSO? Could the algorithm work without it?

How might PSO behave differently if applied to a changing environment?

# Quiz

1. What is a possible downside of high exploitation?

2. What problem does Guaranteed Convergence PSO (GCPSO) attempt to solve?

3. Which field is PSO research currently mostly concentrated in?

THE UNIVERSITY OF TENNESSEE KNOXVILLE

# References


- Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. Proceedings of the IEEE International Conference on Neural Networks, 4, 1942-1948. http://dx.doi.org/10.1109/ICNN.1995.488968
- Y. Shi and R. Eberhart, "A modified particle swarm optimizer," 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA, 1998, pp. 69-73, doi: 10.1109/ICEC.1998.699146. keywords: {Particle swarm optimization;Nonlinear equations;Computational modeling;Evolutionary computation;Genetic programming;Genetic algorithms;Genetic mutations;Educational institutions;Birds;Collaboration}, https://ieeexplore.ieee.org/document/699146
- M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," in IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, pp. 58-73, Feb. 2002, doi: 10.1109/4235.985692. keywords: {Stability;Convergence;Multidimensional systems;Particle swarm optimization;Algorithm design and analysis;Testing;Control system synthesis;Optimization methods;Evolutionary computation;Stochastic systems},https://ieeexplore.ieee.org/document/985692
- Poli, Riccardo & Kennedy, James & Blackwell, Tim. (2007). Particle Swarm Optimization: An Overview. Swarm Intelligence. 1. 10.1007/s11721-007-0002-0. https://www.researchgate.net/publication/257722304_Particle_Swarm_Optimization_An_Overview
- A. Casas-Ordaz et al., “Particle swarm optimization: A survey of innovations over the last 10 years,” Computer Science Review, vol. 60, p. 100910, Feb. 2026, doi: https://doi.org/10.1016/j.cosrev.2026.100910.

# References


- Freitas D, Lopes LG, Morgado-Dias F. Particle Swarm Optimisation: A Historical Review Up to the Current Developments. Entropy (Basel). 2020 Mar 21;22(3):362. doi: 10.3390/e22030362. PMID: 33286136; PMCID: PMC7516836.
- Xin, J.; Chen, G.; Hai, Y. A particle swarm optimizer with multi-stage linearly-decreasing inertia weight. In Proceedings of the International Joint Conference on Computational Sciences and Optimization (CSO), Sanya, China, 24–26 April 2009; Volume 1, pp. 505–508.
- Eberhart, R.C.; Shi, Y. Particle swarm optimization: Developments, applications and resources. In Proceedings of the Congress on Evolutionary Computation (CEC), Seoul, Korea, 27–30 May 2001; Volume 1, pp. 81–86.
- Suganthan, P.N. Particle swarm optimiser with neighbourhood operator. In Proceedings of the Congress on Evolutionary Computation (CEC), Washington, WA, USA, 6–9 July 1999; Volume 3, pp. 1958–1962
- Ratnaweera, Asanga, Saman K. Halgamuge, and Harry C. Watson. "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients." IEEE Transactions on evolutionary computation 8.3 (2004): 240-255.
- Elsheikh, A. H., and M. Abd Elaziz. "Review on applications of particle swarm optimization in solar energy systems." International Journal of Environmental Science and Technology 16.2 (2019): 1159-1170.
- Jain, Meetu, et al. "An overview of variants and advancements of PSO algorithm." Applied Sciences 12.17 (2022): 8392
- Ferrolino, Angelie, Renier Mendoza, Ikha Magdalena, and Jose Ernie Lope. "Application of particle swarm optimization in optimal placement of tsunami sensors." PeerJ Computer Science 6 (2020): e333..


# Thank You for Listening!

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE