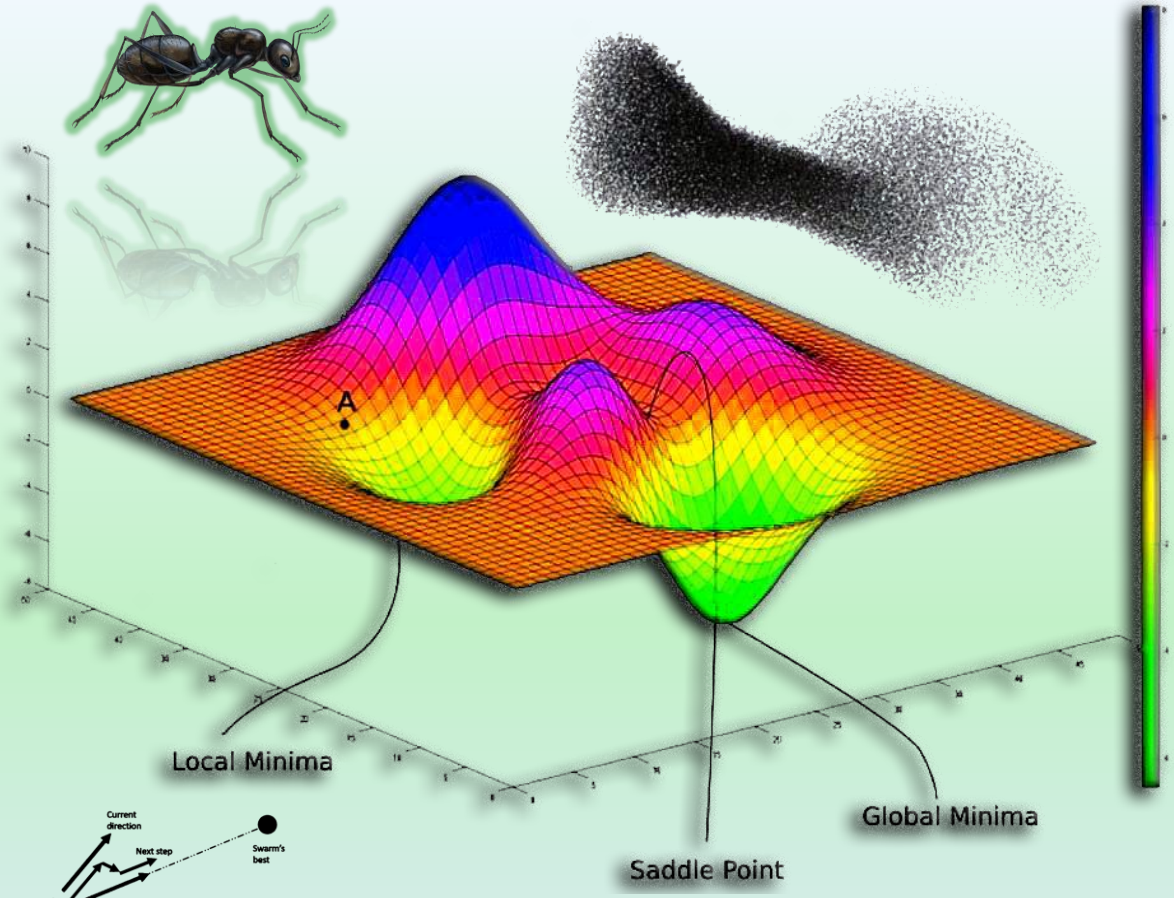
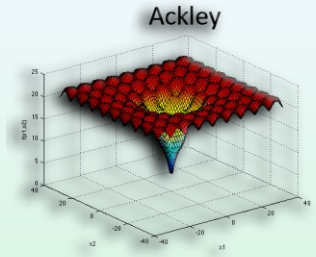
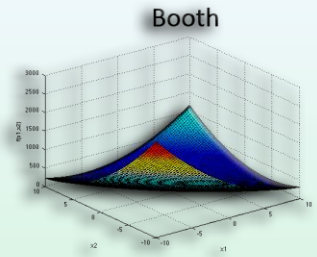
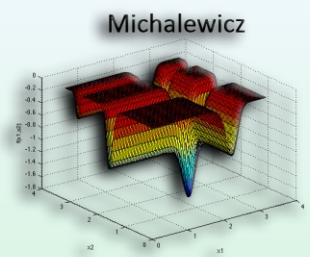
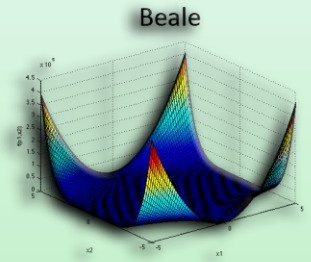
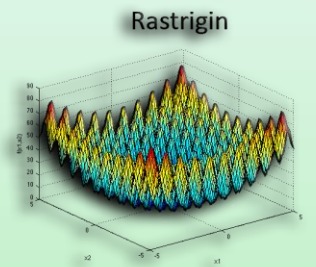


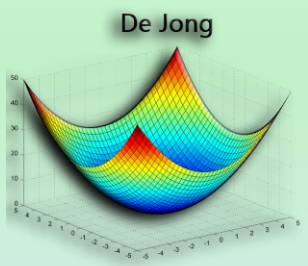


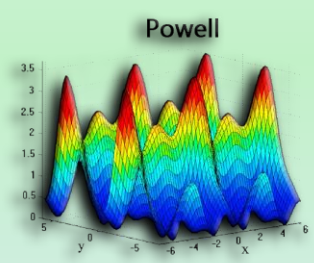


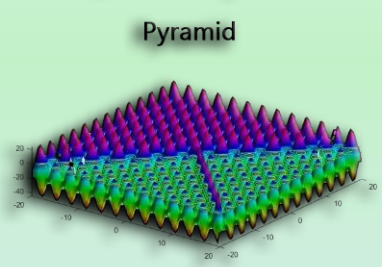


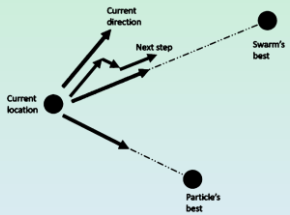


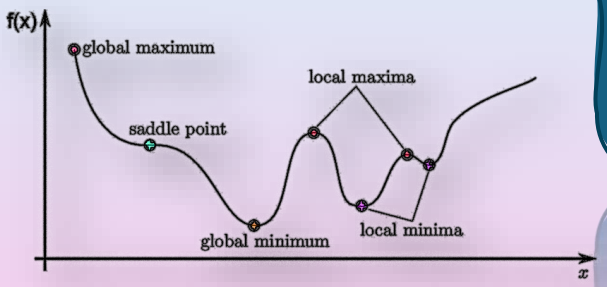










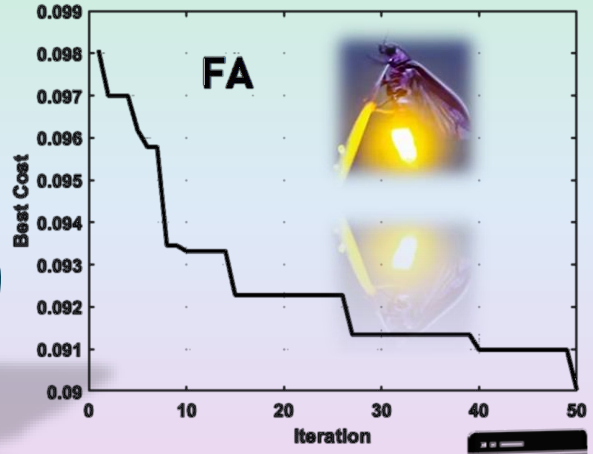


Metaheuristic Optimization: 4 Cutting-Edge Applications

Protein Structure Prediction by Differential Evolution algorithm (DE)

By: Seyed Muhammad Hossein Mousavi

2025



Outline:

- **Optimization**
  - ❖ **Optimization Problems**
    - ➢ **Protein Structure Prediction (Protein Folding) by Differential Evolution algorithm (DE)**
    - ➢ **Space-Time Warping by Firefly Algorithm (FA)**
    - ➢ **Exoplanetary Adaptation Simulation by Genetic Algorithm (GA)**
    - ➢ **Evolved Antenna Design by Particle Swarm Optimization algorithm (PSO)**

- # Optimization Problems

❖ Protein Folding by Differential Evolution algorithm (DE)

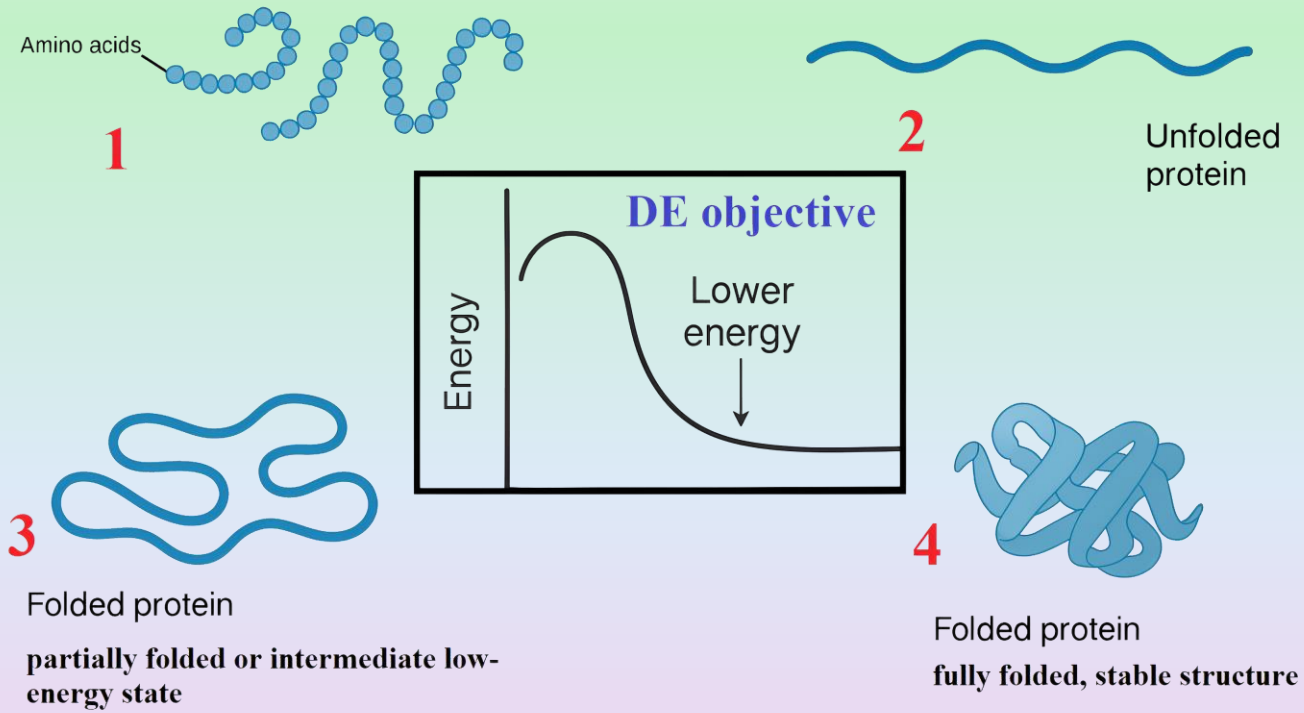
- **Amino acids** are the basic building blocks of proteins, linked together in chains.
- A **protein** is a large biomolecule made of amino acids that performs vital functions in living organisms.
- A **peptide bond** is the chemical link between two amino acids in a protein chain.
- **Protein folding** is the process by which a linear chain of amino acids folds into a functional 3D shape.
- **Protein structure** refers to the 3D arrangement of atoms in a protein, determining its function.
- **Protein energy minimization** is the process of finding the most stable shape of a protein by minimizing its total energy.
- **Why is protein folding important?**
  - Protein folding is essential in nature because only the correctly folded shape allows a protein to perform its biological function inside living cells.
  - In case of **misfolding** diseases such as Alzheimer's, Parkinson and more appear.
  - To simulate folding in the lab to understand, predict, fix, or design new proteins to find cures to diseases.

- # Optimization Problems

❖ Protein Folding by Differential Evolution algorithm (DE)

- This section helps to **understand how a protein folds.**
- Useful for **identifying when it folds incorrectly**.
- **Detecting misfolded shapes**, related to diseases like Alzheimer's, Parkinson's, and more.
- We **minimize the total energy of the protein**, which includes:
  - **Lennard-Jones potential** (for non-bonded/adjacent amino acids) and
  - **Harmonic bond energy** (for adjacent amino acids)
- By **adjusting the 3D positions of amino acids** so that distances between them are physically ideal:
  - **Not too close** (causing strong **repulsion (**high energy)) and
  - **Not too far** (breaking bonds (high energy)),
- We use Differential Evolution (DE) to search for the **most stable, lowest-energy** arrangement.
- By finding not far nor close but **just the right distance.**
- We are looking for the **distance from one amino acid to another in 3D space.**
- Each amino acid has a 3D coordinate (x, y, z)
- The code calculates the distance between pairs of these coordinates.

- # Optimization Problems
  - ## ❖ Protein Folding by Differential Evolution algorithm (DE)
    - **Differential Evolution (DE)** is used because it's a powerful optimization algorithm that works well on complex, non-linear problems like protein folding.
    - Its advantage is that it doesn't need gradients and can efficiently explore the search space to find a global minimum even in rugged energy landscapes.



- # Optimization Problems

❖ Protein Folding by Differential Evolution algorithm (DE)

Objective Function is:

$$E_{\text{total}} = E_{\text{LJ}} + E_{\text{bond}}$$

Where:
(A) Lennard-Jones Potential for Non-Adjacent Residues:

$$E_{\text{LJ}} = \sum_{i<j} 4\left[\left(\frac{1}{r_{ij}}\right)^{12} - \left(\frac{1}{r_{ij}}\right)^{6}\right]$$

- r_{ij} = Euclidean distance between amino acid i and j
- Only applies to non-adjacent residues
- Models van der Waals forces:
- Too close → **large positive repulsion**
- Our Goal: Optimal spacing → **minimum energy**
- Too far → **weak negative attraction**

- # Optimization Problems
  - ## ❖ Protein Folding by Differential Evolution algorithm (DE)

(B) Harmonic Bond Energy for Adjacent Residues:

$$E_{\text{bond}} = \frac{1}{2}\sum_{i=1}^{N-1}\left(r_{i,i+1} - 1\right)^2$$

- $r_{i,i+1}$ = distance between adjacent amino acids
- Penalizes deviation from ideal bond length = $\mathbf{1.0}$ unit
- Acts like a **spring** between neighboring amino acids

• Optimization Problems

❖ Protein Folding by Differential Evolution algorithm (DE)

Input:

- Number of amino acids: num_amino_acids = 15
- Random 3D positions: Initial population of candidates (shape: [particles, 15, 3])
- **DE Process**:
- For each generation:

1. **Mutation**: Create a donor vector using 3 random candidates

$$v = a + F \cdot (b - c)$$

2. **Crossover**: Mix the new candidate vector with the current target to form a trial vector
3. **Evaluation**: Compute the energy of the trial using:

$$E = E_{\text{LJ}} + E_{\text{bond}}$$

4. **Selection**: If trial energy is better (lower), replace the current candidate
5. **Track best score**: Store best position and its energy

• Optimization Problems

❖ Protein Folding by Differential Evolution algorithm (DE)

Output:

- **Best 3D positions of all 15 amino acids**
- **Final minimized energy**
- **Convergence plot** showing energy decreasing over iterations
- **3D visualization** of the final **folded protein structure**