Integer Particle Swarm Optimization (IPSO)

1. Initialization:
   - m (number of variables (candidates)) 20
   - n (population size) try 20 to 50
   - w (inertia weight) 0.8
   - c_1 (cognitive acceleration factor) 2.0
   - c_2 (social acceleration factor) 2.0
   - *maxite* (maximum number of iterations) 1000
   - *UB – Upper bounds of the positions*
   - *LB – Lower bound of the positions*

2. Define all possible connections based on measurements and network structure.
   a. If you identify one candidate as outflow then the possible match candidate should be an inflow.
   b. A candidate cannot **be matched with a match candidate from its own bus/ node.**

Table 1 Data Sample in pu, angle in radiant

Data Sample			
BUS5_01_V_ang	0.874805	BUS9_01_V_ang	0.076763
BUS5_01_V_mag	1.007121	BUS9_01_V_mag	0.969135
BUS5_02_I_ang	0.741158	BUS9_02_I_ang	3.486742
BUS5_02_I_mag	2.384393	BUS9_02_I_mag	0.674885
BUS6_01_V_ang	0.695549	BUS9_03_I_ang	3.486742
BUS6_01_V_mag	0.979073	BUS9_03_I_mag	0.674886
BUS6_02_I_ang	3.876711	BUS9_04_I_ang	3.309073
BUS6_02_I_mag	2.385065	BUS9_04_I_mag	4.736705
BUS6_03_I_ang	0.607188	BUS9_05_I_ang	0.019694
BUS6_03_I_mag	4.648791	BUS9_05_I_mag	6.028491
BUS7_01_V_ang	0.55186	BUS10_01_V_ang	0.222771
BUS7_01_V_mag	0.960054	BUS10_01_V_mag	0.983465
BUS7_02_I_ang	3.747569	BUS10_02_I_ang	0.168681
BUS7_02_I_mag	4.648883	BUS10_02_I_mag	4.736814
BUS7_03_I_ang	0.445898	BUS10_03_I_ang	3.425418
BUS7_03_I_mag	3.281877	BUS10_03_I_mag	2.436016
BUS7_04_I_ang	0.521691	BUS11_01_V_ang	0.40586
BUS7_04_I_mag	0.689644	BUS11_01_V_mag	1.008554
BUS7_05_I_ang	0.521691	BUS11_02_I_ang	0.289767
BUS7_05_I_mag	0.689644	BUS11_02_I_mag	2.435626
BUS8_01_V_ang	0.312094	BUS12_01_V_ang	0.55186
BUS8_01_V_mag	0.947523	BUS12_01_V_mag	0.960054
BUS8_02_I_ang	3.574894	BUS12_02_I_ang	3.58749
BUS8_02_I_mag	0.686882	BUS12_02_I_mag	3.281877
BUS8_03_I_ang	3.574894	BUS13_01_V_ang	0.076763
BUS8_03_I_mag	0.686883	BUS13_01_V_mag	0.969135
BUS8_04_I_ang	0.433306	BUS13_02_I_ang	3.161287
BUS8_04_I_mag	0.686857	BUS13_02_I_mag	6.028491
BUS8_05_I_ang	0.433306		
BUS8_05_I_mag	0.686857		

Table 2 Candidate composition from the data

Candidate 1			
BUS5_01_V_ang	BUS5_01_V_mag	BUS5_02_I_ang	BUS5_02_I_mag
Candidate 2			
BUS6_01_V_ang	BUS6_01_V_mag	BUS6_02_I_ang	BUS6_02_I_mag
Candidate 3			
BUS6_01_V_ang	BUS6_01_V_mag	BUS6_03_I_ang	BUS6_03_I_mag
:::::::::::::::			
Candidate 20			
BUS13_01_V_ang	BUS13_01_V_mag	BUS13_02_I_ang	BUS13_02_I_mag

Table 4 Current/ Power flow direction summary

Candidate	Flow	Candidate	Flow
Candidate 1	outflow	Candidate 11	outflow
Candidate 2	inflow	Candidate 12	inflow
Candidate 3	outflow	Candidate 13	inflow
Candidate 4	inflow	Candidate 14	inflow
Candidate 5	outflow	Candidate 15	outflow
Candidate 6	outflow	Candidate 16	outflow
Candidate 7	outflow	Candidate 17	inflow
Candidate 8	inflow	Candidate 18	outflow
Candidate 9	inflow	Candidate 19	inflow
Candidate 10	outflow	Candidate 20	inflow

Table 3 All possible candidates for connection

Candidate	All possible Candidates for connections
Candidate 1	Candidate 2, 4, 8, 9, 12, 13, 14, 17, 19, 20
Candidate 2	Candidate 1, 5, 6, 7, 10, 11, 15, 16, 18
::::	::::::::::::::::::::::::::::::::::
Candidate 20	Candidate 1, 3, 5, 6, 7, 10, 11, 15, 16, 18

3. IPSO Particle Initialization:
   - Initialize the population of particles (x_0)
   - Make sure there is no repetition while initialization as one candidate can match only one other candidate. This can be achieved by excluding the already selected candidate from the list of selection.

4. IPSO Particle Encoding:
   - Encode the particles into a suitable format for optimization.
     As after defining all the possible connections, it might not be continuous so encoding is necessary.

Table 5 Encoding

Candidate	All possible Candidates for connections	Encoded Values
Candidate 1	Candidate 2, 4, 8, 9, 12, 13, 14, 17, 19, 20	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Candidate 2	Candidate 1, 5, 6, 7, 10, 11, 15, 16, 18	1, 2, 3, 4, 5, 6, 7, 8, 9
::::	::::::::::::::::::::::::::::::::::::	::::::::::::::::::::::::::::::::
Candidate 20	Candidate 1, 3, 5, 6, 7, 10, 11, 15, 16, 18	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

5. Evaluate Initial IPSO Population:
   - Compute the fitness of the initial population.

6. Initialization of Personal Best ($pbest$) and Local Best ($lbest$):
   - Set the initial personal best ($pbest$) and local best ($lbest$).

7. Start of Main PSO Loop

IPSO Algorithm

- Update particle velocities using the following equation:

$$v_{ij}^{k+1} = w \cdot v_{ij}^{k} + c_1 \cdot \mathrm{rand}() \cdot (pbest_{ij} - x_{ij}^{k}) + c_2 \cdot \mathrm{rand}() \cdot (lbest_{ij} - x_{ij}^{k}) \quad (1)$$

- Handle boundary conditions and velocity normalization.
  After updating the velocity, the code checks if any components of the velocity exceed the maximum bounds or fall below the minimum bounds. If so, they are clipped to the respective bounds to ensure they remain within the allowable range.
  Additionally, the velocities are normalized to the range [0, 1] using min-max normalization to facilitate better control over the particle's movement.

$$v_{ij}^{k+1} = \begin{cases} -\frac{\mathrm{UB}(j)}{2} & \text{if } v_{ij}^{k+1} < -\frac{\mathrm{UB}(j)}{2} \\ \frac{\mathrm{UB}(j)}{2} & \text{if } v_{ij}^{k+1} > \frac{\mathrm{UB}(j)}{2} \\ v_{ij}^{k+1} & \text{otherwise} \end{cases} \tag{2}$$

$$v_{\text{normalized}}^{k+1}(i,j) = \frac{|v^{k+1}(i,j)|}{v_{\max}} \tag{3}$$

$$v_{\max} = \frac{\mathrm{UB}(j)}{2} \tag{4}$$

- Update particle positions iteratively using the following equation:

$$x_{ij}^{k+1} = \begin{cases} lbest_{ij} & \text{if } v_{\text{normalized},ij}^{k+1} < \text{Flipping}(10) \\ pbest_{ij} & \text{if } 1 - v_{\text{normalized},ij}^{k+1} < \text{Flipping}(10) \\ x_{ij}^{k} & \text{otherwise} \end{cases} \tag{5}$$

This equation describes the logic for updating the position of each particle based on its normalized velocity and the best positions found so far.

- Evaluate fitness for the updated positions.
- Update personal best (*pbest*) and local best (*lbest*).
- Apply mutation to a certain percentage of particles.
  Mutation Process:
  - In the given code snippet, mutation is applied to each particle's position matrix.
  - The mutation process can be expressed as follows:
    - Mutation Probability Assignment: A random number between 0 and 1 is generated. Based on this random number, a mutation probability is assigned to determine the type of mutation to be performed on each selected dimension.
    - If random number<0.33: Increase the value of the dimension by 50%.
    - Else if 0.33<random number<0.66: Decrease the value of the dimension by 50%.

$$x_{ij}^{k+1} = \begin{cases} \text{round}(1.5 \cdot x_{ij}^{k+1}) & \text{if rand} < 0.33 \\ \text{round}(0.5 \cdot x_{ij}^{k+1}) & \text{if } 0.33 < \text{rand} < 0.66 \\ x_{ij}^{k+1} & \text{otherwise} \end{cases} \tag{6}$$

- Boundary Constraint Handling: After mutation, the values are checked against the lower and upper bounds. If a mutated value exceeds these bounds, it is adjusted to fall within the permissible range.

$$x_{ij}^{k+1} = \begin{cases} \mathrm{LB}(j) & \text{if } x_{ij}^{k+1} < \mathrm{LB}(j) \\ \mathrm{UB}(j) & \text{if } x_{ij}^{k+1} > \mathrm{UB}(j) \\ x_{ij}^{k+1} & \text{otherwise} \end{cases} \tag{7}$$

8. <u>End of Main PSO Loop after maximum iterations.</u>
9. <u>Decode the best results to normal form.</u>

<u>Flow Chart</u>

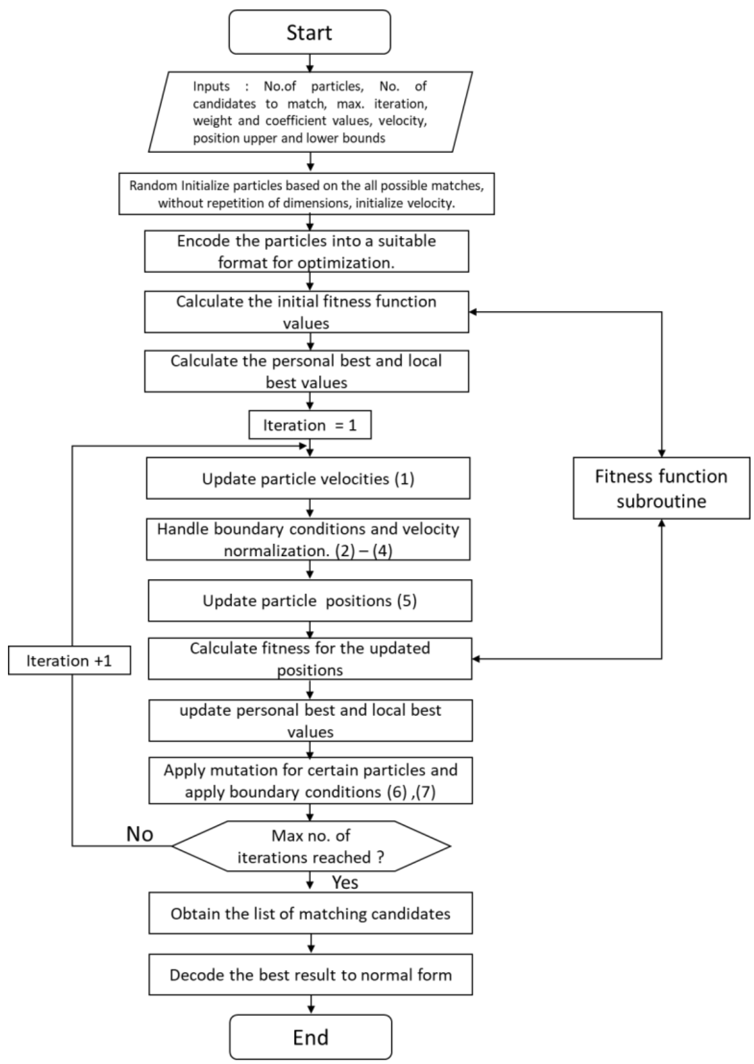

Figure 1 Flowchart of the IPSO algorithm

Functions

- Flipping simulates a coin toss and calculates the percentage of times the result is heads. For example, Flipping (10) would return the percentage of heads obtained out of 10 coin tosses.

Fitness Evaluation

1. Initialize variables:
   - Fitness 1 to track constraint violations penalty.
   - Fitness 2 to calculate fitness based on power flow differences and mean power loss.
   - Total Fitness to sum up the fitness values calculated for each candidate pair**.**
2. Decoding the particle to obtain the network candidates.
   Eg:

Before	1	3	8	1	9	6	4	6	3	4	6	7	6	2	3	4	9	2	5	9
After	2	6	19	1	20	14	12	15	5	13	17	11	10	3	8	9	18	4	7	16

3. Repeating Constraint:
   - This is to ensure that one candidate is paired with only one other candidate.
   - If the number of unique elements in x is less than 20, calculate the constraint violation c as the difference between 20 and the number of unique elements.
4. Apply a penalty on constraint violations:
   - Set the penalty very small otherwise conversion will not happen (try 0.1 to 0.00001pu) per constraint violation.
5. Calculate **Fitness1** as the product of penalty and c.

$$Fitness\ 1 = c \times penalty \tag{8}$$

6. Extract the steady-state data form given data at a time sample.
7. Formulate 20 candidate pairs based on the extracted data.
8. Calculate the fitness for each candidate pair and sum it up:
   - Calculate the power flow difference (DF_PM) between the candidate pairs.
   - Calculate the mean power loss (MPL) between the candidate pairs.
   - Fitness is calculated as the absolute difference between DF_PM and MPL.
   - Sum up the fitness values for all candidate pairs to obtain Fitness 2.
9. Calculate the overall fitness value as the sum of Fitness1 and Fitness2.

Example for fitness evaluation

Decoding

Before	1	3	8	1	9	6	4	6	3	4	6	7	6	2	3	4	9	2	5	9
After	2	6	19	1	20	14	12	15	5	13	17	11	10	3	8	9	18	4	7	16

- No repeats so no penalty here, **Fitness 1 = 0**

So the pairs for fitness are

Default order	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Decoded particle	2	6	19	1	20	14	12	15	5	13	17	11	10	3	8	9	18	4	7	16

Pair 1 **Pair 20**

Note: The default order never changes the decoded particle will change according to find the best match throughout the IPSO algorithm.

- let's take the fitness of pair 1 which is Candidate 1 – Candidate 2

Power Flow Calculation

Apply

$$P_{a_1} = 3 \times |V_a| \left|I_{a_1}\right| \cos(\theta_{a_1} - \alpha_{a_1})$$

P_1=3× BUS5_01_V_mag X BUS5_02_I_mag X **COS**(BUS5_01_V_ang - BUS5_02_I_ang)

= 7.1399 pu

P_2=3× BUS6_01_V_mag X BUS6_02_I_mag X **COS**(BUS6_01_V_ang - BUS6_02_I_ang)

= -7.001 pu

Note: As Candidate 2 is an inflow the flow is negative here.

$\underline{Absolute\ Power\ Flow\ Difference}$

$$DF_{P_{1,P2}} = ||P_{a_1}| - |P_{b_1}||$$

$$DF_{P_{1,P2}} = ||7.1399| - |7.001|| = 0.1398\ pu$$

Mean Power Loss

$$I_{mean_{1,2}} = \frac{I_1 + I_2}{2}$$

$$I_{mean_{1,2}} = \frac{BUS5\_02\_I\_mag\measuredangle BUS5\_02\_I\_ang + (-BUS6\_02\_I\_mag\measuredangle BUS6\_02\_I\_ang)}{2}$$

$$I_{mean_{1,2}} = \frac{2.3843\measuredangle 0.74115 + (-2.38506\measuredangle 3.87671)}{2}$$

$$I_{mean_{1,2}} = 2.384669 \measuredangle 0.738133\, pu$$

$$\Delta V_{a,b} = V_a - V_b$$

$$\Delta V_{1,2} = BUS5\_01\_V\_mag\measuredangle BUS5\_01\_V\_ang - BUS6\_01\_V\_mag\measuredangle BUS6\_01\_V\_ang$$

$$MPL_{1,2} = \left|3 * real\left(\Delta V_{1,2} \times \left(I_{mean_{1,2}}\right)^{*}\right)\right|$$

$$MPL_{1,2} = 0.1398\, pu$$

As this matches $\left|DF_{P_{1,P2}} - MPL_{1,2}\right|$ will be a very small value. ES will be big as

$$ES_{a_1,b_1} = \frac{1}{\left|DF_{P_{a_1,b_1}} - MPL_{a_1,b_1}\right|}$$

- Likewise all the values has to be calculate and summed up to get the fitness for a particle that have **20 pairs** in it. This will be **Fitness 2**

$$J = \sum_{i=1}^{n} \frac{1}{ES_i}$$

Fitness value = Fitness 1 + Fitness 2