

Solving Prize Collecting Travelling Salesman Problem with Time Windows by Particle Swarm Optimization

指導老師: 丁慶榮 教授 學生: 司大衛, 湯維斯

Introduction

Background

A Company must collect seafood from several coastal villages in Indonesia. The Company has one vehicle with existing fixed routes causing unnecessary travel and higher costs. This problem is modelled as a Prize Collecting Travelling Salesman Problem with Time Windows.

Motivation

Logistics requires cost and revenue under resource limits. Visiting all locations would be costly, hence, selective routing is needed. While minimizing cost with respect to constraints. Higher-value fish village would be prioritized.

Research Scope

This research addresses the inefficient fixed routing strategies used by Indonesian seafood distributors, which result in high costs. The problem is modeled as the Prize Collecting Travelling Salesman Problem with Time Windows (PCTSPTW). The core objective is to find an optimal route that minimizes the total cost minus the collected prize (Profit) by implementing a new Particle Swarm Optimization (PSO) algorithm.

Mathematical Model

Sets

K Set of Seafood Types
N Sets of Nodes

T_i Service start time at node i
 u_i Order variable (for MTZ subtour elimination)

$x_{ij} = \begin{cases} 1 & \text{If visit node } j \text{ immediately after node } i \\ 0 & \text{Otherwise} \end{cases}$

$y_i = \begin{cases} 1 & \text{If visited node } i \\ 0 & \text{Otherwise} \end{cases}$

Parameters

a_i Open time at node i
 b_i Close time at node i
 c Travel cost
 d_{ij} Travel distance from node i to j
 M Large constant for big-M constraint
 p_{ik} Profit for seafood type k at node i
 q_{ik} Seafood availability of type k at node i
 Q Vehicle capacity
 t_{ij} Travel time from i to j

Objective Function

$$\text{Min} \sum_{i \in N} \sum_{j \in N} cd_{ij}x_{ij} - \sum_{i \in N} \sum_{k \in K} q_{ik}p_{ik}y_i \quad (1)$$

$$\text{S.T.} \sum_{j \in N} x_{ij} = \sum_{j \in N} x_{ji} = y_i \quad \forall i \in N \setminus \{0\} \quad (2)$$

$$ay_i \leq T_i \leq by_i \quad \forall i \in N \quad (3)$$

$$T_i + s_i + t_{ij} - T_j \leq M(1 - x_{ij}) \quad \forall i \in N, \forall j \in N \quad (4)$$

$$\sum_{i \in N} \sum_{k \in K} q_{ik}y_i \leq Q \quad (5)$$

$$u_0 = 0 \quad (6)$$

$$1 \leq u_i \leq n-1 \quad \forall i \in N \setminus \{0\} \quad (7)$$

$$u_i - u_j + 1 \leq (n-1)(1 - x_{ij}) \quad \forall i \in N, \forall j \in N \quad (8)$$

$$T_i \geq 0 \quad \forall i \in N \quad (9)$$

$$x_{ij} \in \{0,1\} \quad \forall i \in N, \forall j \in N \quad (10)$$

$$y_i \in \{0,1\} \quad \forall i \in N \quad (11)$$

Experiment

PSO FRAMEWORK

The PSO is combined with some constructive heuristics based initial solutions (Nearest Neighbor, Nearest Insertion, Prize to Distance Ratio, Prize to supply ratio, Savings Heuristic).

Step 1 Parameter Initialization:

$c_1, c_2, w, \text{swarm size, max iterations, } V_{max}, (V_{min}, V_{max})$

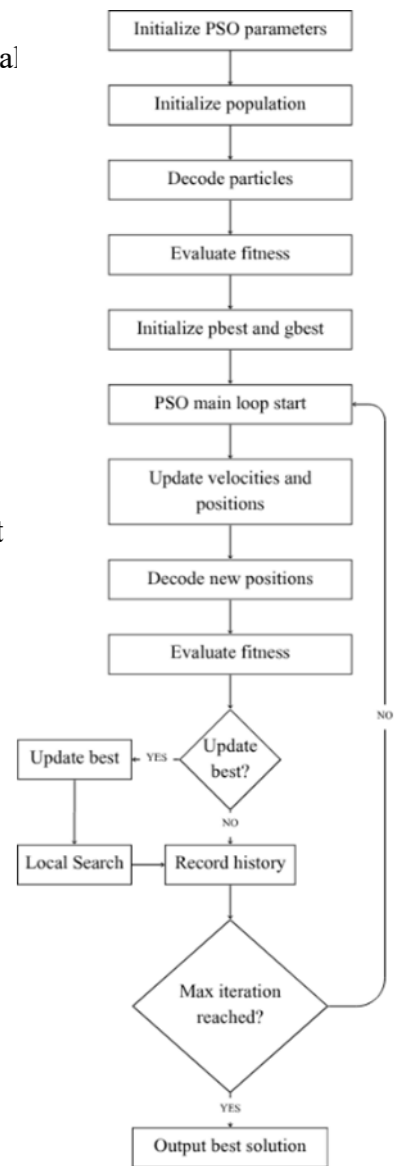
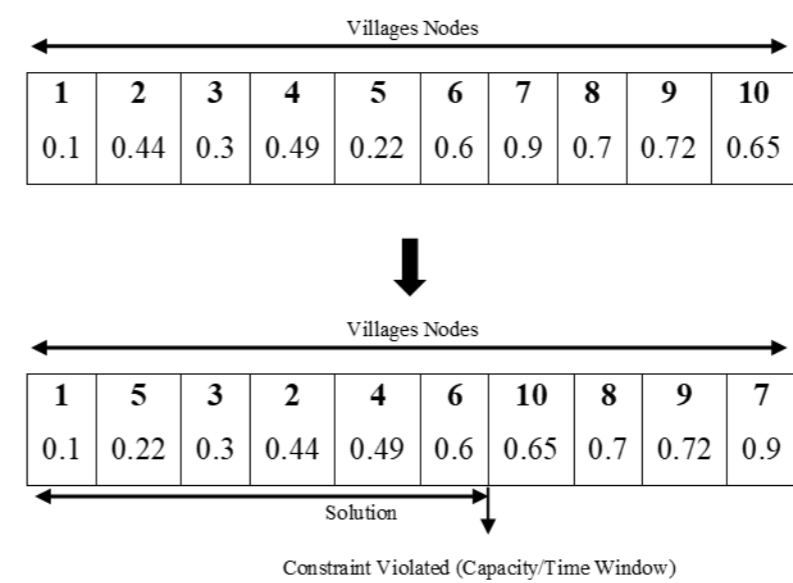
Step 2 Generate initial solutions

Step 3 For each iteration:

Velocity update, position update, evaluate fitness, update pbest and gbest

Step 4 Stop criterion satisfied, then move to step 5. Otherwise go back to step 3

Step 5 Output best solution



RESULTS

Table 4.4 Result of 10 Nodes

Instance	Gurobi		PSO	
	Optimal	CPU	Best	CPU
C10-1	-3,359,366.85	0.04	-3,359,366.85	0.045/0.00046*
C10-2	-3,359,366.85	0.06	-3,359,366.85	0.044/0.0003*
C10-3	-3,359,366.85	0.06	-3,359,366.85	0.046/0.0014*
C10-4	-3,324,385.38	0.04	-3,324,385.38	0.045/0.001*
C10-5	-3,333,664.02	0.05	-3,333,664.02	0.046/0.0004*
C20-1	-3,773,598.42	0.07	-3,773,598.42	0.077/0.0071*
C20-2	-3,773,598.42	0.16	-3,773,598.42	0.076/0.014*
C20-3	-3,773,598.42	0.18	-3,773,598.42	0.077/0.0018*
C20-4	-3,765,213.71	0.06	-3,765,213.71	0.073/0.0151*
C20-5	-3,767,287.30	0.06	-3,767,287.30	0.074/0.0021
C30-1	-4,059,525.77	0.11	-4,059,525.77	0.107/0.0076*
C30-2	-4,059,525.77	0.17	-4,059,525.77	0.112/0.0060*
C30-3	-4,059,525.77	0.2	-4,059,525.77	0.109/0.0147*
C30-4	-4,059,525.76	0.13	-4,059,525.77	0.104/0.004*
C30-5	-4,059,525.76	0.12	-4,059,525.77	0.109/0.024*
C100-1	-4,527,542.5	1.17	-4,527,542.5	0.367/0.1184*
C100-2	-4,527,542.5	0.96	-4,527,542.5	0.37/0.1528*
C100-3	-4,527,542.5	1.21	-4,527,542.5	0.377/0.2127*
C100-4	-4,524,785.72	1.16	-4,524,785.72	0.357/0.2278*
C100-5	-4,524,900.49	1.01	-4,524,900.49	0.369/0.1849*

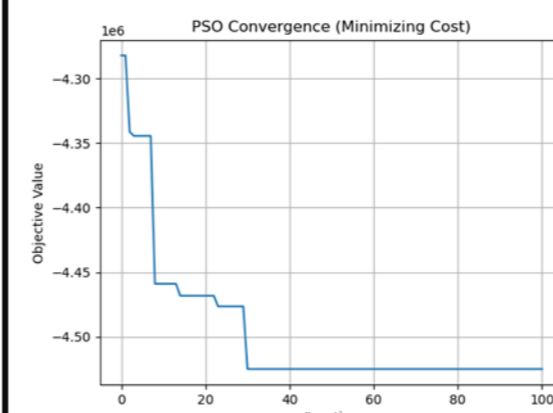


Figure 4.1 Example of PSO Convergence Plot

Conclusion & Future Work

➤ Conclusion

• Formulated PCTSPTW to seafood distribution
• A PSO framework with constructive heuristics was developed
• It show that PSO provides high-quality solutions, with slight faster time.
• Highlight the value of metaheuristics for large, realistic routing problems

➤ Future Work

• Multi vehicle extensions
• Stochastic Fish availability and price
• Hybrid optimization