Received: 09 Aug. 2025; Revised: 30 Oct. 2025; Accepted: 02 Feb. 2026
https://doi.org/10.22306/al.v13i1.752

Balancing profits and planet: a game-theoretic particle swarm optimization for green inventory routing

Kawinthorn Saicharoen
Department of Industrial Engineering and Management, Faculty of Engineering and Industrial Technology, Silpakorn University, Nahon Pathom 73000, Thailand, saicharoen_k@su.ac.th
Choosak Pornsing
Department of Industrial Engineering and Management, Faculty of Engineering and Industrial Technology, Silpakorn University, Nahon Pathom 73000, Thailand, pornsing_c@su.ac.th, (corresponding author)

Keywords: inventory routing problem, particle swarm optimization, green logistics, genetic algorithm, simulated annealing.

Abstract: In an era of heightened environmental awareness and increasing energy costs, sustainable logistics has become a critical focus in supply chain management. The Green Inventory Routing Problem (GIRP) extends the classic Inventory Routing Problem (IRP) by integrating environmental objectives—such as minimizing greenhouse gas (GHG) emissions—alongside traditional economic costs. This study proposes a novel Game-Theoretic Particle Swarm Optimization (GT-PSO) algorithm to address the GIRP, conceptualizing the discretization process as a strategic game to transform continuous swarm behavior into discrete logistics decisions. The objective minimizes a weighted cost function of transportation, inventory holding, and carbon emissions, supporting efficient management of material and transport flows. Thirteen benchmark instances representing different logistics network scales were used to evaluate performance against Genetic Algorithm (GA), Simulated Annealing (SA), and the commercial solver Gurobi. Results show that GT-PSO achieves the best-known solutions for medium- and large-sized problems with an average gap below 0.97%. ANOVA tests confirm statistically significant superiority, and coefficient of variation analysis highlights consistent performance. This research contributes an advanced optimization tool for sustainable logistics and distribution systems, helping managers balance economic efficiency with environmental responsibility.

1 Introduction

The confluence of globalized markets and mounting environmental pressures has compelled businesses to re-evaluate their supply chain strategies. Sustainable logistics management is no longer a niche concern but a central pillar of corporate responsibility and long-term viability [1]. Within this paradigm, Vendor-Managed Inventory (VMI) has emerged as a highly effective strategy for optimizing supply chain efficiency. In a VMI model, the supplier takes responsibility for managing the customer's inventory levels, leading to reduced overall costs through better coordination of production and transportation [2]. From a logistics perspective, optimizing the flow of materials, information, and transport resources across the supply chain has become essential to ensure both economic and environmental efficiency.

The operational challenge at the heart of VMI is the Inventory Routing Problem (IRP), first formally introduced by Bell et al. in 1983 [3]. The IRP integrates three critical decisions: 1) when to service a customer, 2) how much to deliver, and 3) how to combine deliveries into efficient vehicle routes [4]. By solving these decisions simultaneously, companies can achieve significant cost savings compared to managing inventory and routing separately [5].

However, traditional IRP models have predominantly focused on economic objectives, such as minimizing transportation and inventory holding costs [5]. The increasing urgency of climate change, driven by greenhouse gas (GHG) emissions from activities like transportation, necessitates an evolution of these models. This has given rise to the Green Inventory Routing Problem (GIRP), which extends the IRP by incorporating environmental impact factors, most notably CO_2 emissions from fuel consumption, directly into the optimization objective [6,7].

The GIRP, like its predecessor, is a combinatorial optimization problem classified as NP-hard [8]. This complexity means that finding a guaranteed optimal solution using exact methods is computationally intractable for real-world, large-scale instances. Consequently, the research community has largely turned to heuristic and metaheuristic algorithms to find high-quality, near-optimal solutions within a reasonable time frame. Methods such as Genetic Algorithms (GA) and Simulated Annealing (SA) are well-established; however, their performance often declines as problem scale increases. Existing PSO-based approaches also face a notable limitation—the discretization of continuous particle movements into routing decisions is usually handled by simple rounding or threshold rules, which can cause premature convergence and instability in complex GIRP settings. This limitation defines a clear research gap that motivates the development of a more adaptive, game-inspired discretization framework.

This paper makes a significant contribution to this field by proposing a novel Game-Theoretic Particle Swarm Optimization (GT-PSO). By framing the discretization process as a strategic decision-making game, our approach provides a new and robust mechanism for solving complex combinatorial optimization problems like the GIRP. Each step of the GT-PSO is designed to match the nature of GIRP: initialization creates diverse solutions for large search spaces; the game-based discretization allows probabilistic, adaptive routing choices; and iterative updates guide the swarm toward lower cost and emission objectives. This structure makes the proposed steps particularly suitable for addressing the combined economic-environmental trade-offs of GIRP. This study aims to:

1. Propose an improved metaheuristic algorithm, GT-PSO, for solving the green inventory routing problem with GHG emission costs.
2. Demonstrate the superior performance of the proposed method by rigorously comparing it against established metaheuristics (GA and SA) and a leading commercial solver (Gurobi) across a comprehensive set of benchmark problems.

The distinctive feature and contribution of this research lie in integrating game-theoretic reasoning into the PSO framework, enabling particles to make strategic, context-sensitive decisions rather than relying on purely stochastic updates. This mechanism improves convergence stability and scalability, distinguishing GT-PSO from conventional metaheuristics for green logistics.

The remainder of this paper is structured as follows: Section 2 provides a review of the relevant literature on IRP, GIRP, and the application of metaheuristics. Section 3 presents the formal mathematical model of the problem. Section 4 details the proposed PSO for GIRP methodology. Section 5 describes the computational experiments and presents a thorough analysis of the results. Finally, Section 6 offers conclusions, managerial implications, and directions for future research.

2 Literature review

2.1 *The inventory routing problem*

The Inventory Routing Problem (IRP) has been a subject of intensive research for over four decades since its inception by Bell et al. [3]. It addresses the integrated logistics planning of inventory management at customer locations and the routing of vehicles from a central depot to replenish them. A comprehensive review by Coelho et al. [5] highlights the evolution of IRP research, covering various problem characteristics such as single vs. multi-period horizons, deterministic vs. stochastic demand, and different inventory policies. The primary challenge of the IRP lies in its combinatorial complexity, which grows exponentially with the number of customers and the length of the planning horizon, making it a formidable computational task.

2.2 *The green inventory routing problem*

The transition from IRP to Green Inventory Routing Problem (GIRP) reflects a broader shift towards sustainability in operations research. The GIRP explicitly incorporates environmental objectives into its framework. Early works began to consider environmental aspects like fuel consumption as a proxy for both cost and emissions. More recent studies, as surveyed by Moghdani et al. [6] and Soysal et al. [10], formalize this by including emission costs, often derived from carbon taxes or emission trading schemes, directly in the objective function.

Key considerations in GIRP models often include the impact of vehicle load, speed, and type on fuel consumption and emissions [11,12]. For example, Cheng et al. modeled a GIRP with a heterogeneous fleet, allowing for the selection of more fuel-efficient vehicles [13], while Soysal investigated a closed-loop IRP for returnable items [14]. This study follows this trend by formulating an objective function that explicitly weights economic costs against GHG emission costs, providing a flexible framework for decision-makers.

2.3 *Solution approaches for GIRP*

Given the NP-hard nature of GIRP, research on solution methods is vast and varied.

2.3.1 Exact methods

For small-scale problems, exact methods based on Mixed-Integer Linear Programming (MILP) can find optimal solutions. Solvers like Gurobi and CPLEX employ sophisticated branch-and-cut or branch-and-price algorithms [15]. However, their computational time becomes prohibitive as problem size increases, limiting their practical application.

2.3.2 Metaheuristics

For medium- to large-scale instances, metaheuristics are the dominant approach. These high-level strategies guide the search for near-optimal solutions without guaranteeing optimality.

- Genetic Algorithms (GA): GAs mimic natural evolution and have been applied to various IRPs [16]. They are effective at exploring the solution space but can sometimes suffer from premature convergence to local optima.
- Simulated Annealing (SA): SA is a probabilistic method inspired by the annealing process in metallurgy, which allows it to escape local optima [17].
- Particle Swarm Optimization (PSO): Introduced by Kennedy and Eberhart [18], PSO is a population-based algorithm inspired by the social behavior of bird flocks or fish schools. It is known for its conceptual simplicity and rapid convergence. While originally designed for continuous optimization problems, its application to discrete problems like routing requires a mechanism to map continuous particle positions and velocities to a discrete solution space. Various discretization techniques have been proposed, but finding an effective mapping remains a key challenge and an active area of research [19]. This research gap motivates our use of a novel game-theoretic discretization approach.

Despite extensive studies on metaheuristic approaches such as GA, SA, and PSO for GIRP, several research gaps remain. First, most PSO adaptations for GIRP rely on simplistic or ad hoc discretization methods that inadequately represent the combinatorial decision space, leading to premature convergence. Second, few studies integrate strategic decision frameworks, such as game theory, to enhance the decision-making behaviour of particles. Third, there is a lack of systematic benchmarking that evaluates algorithm robustness, scalability, and statistical significance across small to large problem sizes. To bridge these gaps, this study introduces a Game-Theoretic Particle Swarm Optimization (GT-PSO) that employs a structured, probabilistic game-based discretization mechanism, aiming to enhance both the accuracy and stability of GIRP solutions.

3 Problem formulation

This study addresses a multi-period, single-depot, multi-customer GIRP with a homogeneous fleet of capacitated vehicles. The objective is to determine the optimal delivery quantities, schedule, and vehicle routes over a finite planning horizon to minimize the total integrated cost of transportation, inventory, and GHG emissions.

Set and indices:
$N = \{1,2,\ldots,n\}$: Set of customers.
$N_0 = N \cup \{0\}$: Set of all nodes, where 0 is the central depot.
$V = \{1,2,\ldots,m\}$: Set of available vehicles.
$T = \{1,2,\ldots,H\}$: Set of time periods in the planning horizon.
$A = \{(i,j) \mid i,j \in N_0, i \neq j\}$: Set of arcs connecting the nodes.

Parameters:
d_{ij} = Distance from node i to node j.
c_{ij} = Transportation cost per trip from node i to j.
e_{ij} = GHG emission (e.g., in kg of CO_2) per kilometer between i and j.
q_{it} = Demand of customer i in period t.
Q = Capacity of each vehicle.
I_i^{max} = Maximum inventory capacity at customer i.
I_i^{min} = Minimum inventory level (safety stock) at customer i.
h_i = Inventory holding cost per unit per period at customer i.
α = Cost per unit of GHG emission (e.g., carbon tax).
β = A weighting factor ($0 \leq \beta \leq 1$) to balance economic and environment costs.

Decision variables:
$x_{ij}^{kt} \in \{0,1\}$: 1 if vehicle k travels from node i to node j in period t; 0 otherwise.
$y_{it} \geq 0$: Amount of product delivered to customer i in period t.
$I_{it} \geq 0$: Inventory level at customer i at the end of period t.

Mathematical model:
The objective function is to minimize the total cost, z:

$$\min \sum_{t\in T} \left[\sum_{k\in V} \sum_{(i,j)\in A} \left((1-\beta) c_{ij} + \beta \alpha e_{ij} d_{ij} \right) x_{ij}^{kt} + \sum_{i\in N} h_i I_{it} \right] \tag{1}$$

Subject to:

$$I_{it} = I_{i,t-1} + y_{it} - q_{it}, \qquad \forall i \in N, t \in T \tag{2}$$

$$I_i^{min} \leq I_{it} \leq I_i^{max}, \qquad \forall i \in N, t \in T \tag{3}$$

$$\sum_{i \in N} y_{ij} \leq Q \cdot \sum_{j \in N_0, j \neq i} \sum_{k \in V} x_{ij}^{kt}, \quad \forall t \in T \tag{4}$$
$$y_{it} \leq Q \cdot \sum_{j \in N_0, j \neq i} \sum_{k \in V} x_{ji}^{kt}, \quad \forall i \in N, \forall t \in T \tag{5}$$
$$\sum_{j \in N_0, j \neq i} x_{ij}^{kt} = \sum_{j \in N_0, j \neq i} x_{ji}^{kt}, \quad \forall i \in N_0, k \in V, t \in T \tag{6}$$
$$\sum_{j \in N} x_{0j}^{kt} \leq 1, \quad \forall k \in V, t \in T \tag{7}$$

The objective function (1) minimizes the weighted sum of transportation costs, emission costs, and inventory holding costs. Constraint (2) is the inventory balance equation. Constraint (3) ensures inventory levels remain within their allowed minimum and maximum bounds. Constraint (4) enforces vehicle capacity limits. Constraint (5) is a logical constraint ensuring that a delivery is made to a customer only if they are visited. Constraint (6) ensures route flow conservation. Finally, constraint (7) ensures each vehicle route starts and ends at the depot.

4 The Proposed Game-Theoretic PSO (GT-PSO) algorithm

To tackle the complexity of the formulated GIRP, we propose a Particle Swarm Optimization algorithm grounded in a game-theoretic framework. The primary challenge in applying PSO to routing problems is the translation of a continuous velocity vector into a discrete set of actions (i.e., whether to include an arc in a route or not). We conceptualize this challenge as a simple, repeated game played by each particle for each dimension of the solution space. In this game:

- The Player: The particle itself.
- The Actions: The particle has a discrete set of choices for each arc—'include' (1), 'exclude' (0), or 'maintain status quo'.
- The Strategy: The particle's strategy is a decision rule that maps the combined influences (cognitive, social, and inertial, represented by the continuous velocity value) to one of the discrete actions.

The Sig-bo game discretization, presented next, serves as the explicit implementation of this strategy, providing a probabilistic rule-set for this decision-making process.

4.1 *Fundamentals of particle swarm optimization*

Particle Swarm Optimization (PSO) is a population-based stochastic optimization technique where a population of candidate solutions, called a swarm of particles, moves through the D-dimensional problem search space. Each particle's movement is influenced by its own best-known position ($pbest$) and the entire swarm's best-known position ($gbest$). The velocity and position of each particle are updated iteratively. The modified velocity update equation, which includes an inertia weight (ω) to balance global and local exploration [20], is (8), (9):

$$v_{ij}^{t+1} = \omega v_{ij}^{t} + \phi_1 \beta_1 \left(pbest_{ij}^{t} - x_{ij}^{t}\right) + \phi_2 \beta_2 \left(gbest_{j}^{t} - x_{ij}^{t}\right) \tag{8}$$
$$x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \tag{9}$$

where ϕ_1 and ϕ_2 are cognitive and social acceleration constants, and β_1 and β_2 are random variables in [0,1].

4.2 *Sic-bo game discretization technique*

The primary challenge in applying PSO to routing problems is that the variables are discrete (e.g., a link is either in a route or not), whereas the classic PSO operates in a continuous space. The core of our GT-PSO utilizes the Sig-bo game discretization technique, inspired by the work of Pornsing et al. [9]. This method serves as the particle's strategic decision function, transforming the continuous velocity into a discrete action.
Instead of a direct position update, the velocity update equation (4.1) is calculated. This resulting continuous velocity value is then mapped to a discrete, binary decision for the particle's new position using a set of probabilistic rules based on the Thai dice game "Sig-bo". As detailed in the original report, the velocity update for each dimension j of particle i is calculated based on three acceleration components: its current position $\left(a_i^{k+1}(x_i^k(j))\right)$, its personal best $\left(a_i^{k+1}(pbest_i^k(j))\right)$, and the global best $\left(a_i^{k+1}(gbest_i^k(j))\right)$. The sum of these accelerations produces the new velocity, $v_i^{k+1}(j)$. This velocity is then mapped to a binary state based on threshold values derived from the game's rules (10):

$$x_i^{k+1}(j) = \begin{cases} 0 & \text{if } 11 \leq v_i^{k+1}(j) \leq 18 \\ 1 & \text{if } v_i^{k+1}(j) \leq 10 \\ \text{No change} & \text{if } 10 \leq v_i^{k+1}(j) \leq 11 \end{cases} \tag{10}$$

This mechanism provides a structured, probabilistic way to handle the discrete nature of the problem, allowing particles to either switch to a '1' state (include an arc), switch to a '0' state (exclude an arc), or maintain their current

Acta Logistica, Volume 13, Issue 1, Pages 210-221, 2026 ISSN 1339-5629

state. This approach avoids the common pitfalls of other discretization methods and provides a robust exploration-exploitation balance. The pseudocode for the algorithm is presented in Table 1.

Table 1 Pseudocode for the GT-PSO

Step	Description
1	Initialization (a) Set $k = 0$. (b) Randomly initialize the position of the m particles in binary numbers. (c) Randomly initialize the velocity of the m particles in binary numbers. For $i = 1, \dots, m$ do $pbest_i^k = X_i$. Set $gbest^k = \arg\min_{i \in 1,\dots,m} f(X_i)$.
2	Terminate Check. If the termination criteria hold stop. Then, the outcome of the algorithm will be $gbest$.
3	For $i = 1,2, \dots, m$ do Calculating acceleration values. (a) Calculate $a_i^{k+1}(x_i^k(j))$. (b) Calculate $a_i^{k+1}(pbest_i^k(j))$. (c) Calculate $a_i^{k+1}(gbest^k(j))$. Updating particle swarm. (a) Update the velocity V_i^k using $v_i^{k+1}(j) = a_i^{k+1}\left(x_i^k(j)\right) + a_i^{k+1}\left(pbest_i^k(j)\right) + a_i^{k+1}(gbest^k(j))$ (b) Update the position X_i^k using Eq. (47). $x_i^{k+1}(j) = \begin{cases} 1 & \text{if } 11 < v_i^{k+1}(j) \le 18 \\ 0 & \text{if } v_i^{k+1}(j) \le 10 \\ \text{No change} & \text{if } 10 < v_i^{k+1}(j) \le 11 \end{cases}$ Updating $pbest_i$ and $gbest$. (c) If $f(X_i) \le f(pbest_i)$ then $pbest = X_i$. (d) If $f(pbest_i) \le f(gbest)$ then $gbest = pbest_i$. End For.
4	Set $k = k + 1$.
5	Go to step 2.

Each step of the GT-PSO has been designed to address the unique characteristics of the GIRP. The initialization phase ensures diverse solution exploration, crucial for high-dimensional routing problems. The Sig-bo game discretization introduces a probabilistic mechanism that reflects real-world decision uncertainty while preserving exploration capability. The iterative velocity-position updates simulate adaptive learning among particles, mirroring dynamic interactions in inventory-routing systems. Finally, the update of personal and global best solutions allows continuous refinement toward cost-effective and environmentally responsible logistics plans. These design choices collectively make GT-PSO well-suited to tackle the combinatorial complexity and conflicting objectives inherent in GIRP.

4.3 The proposed algorithm analysis

The complexity of the proposed algorithm can be evaluated in two main categories: Time Complexity, which measures how the runtime scales with the size of the problem, and Space Complexity, which measures the amount of memory required.

4.3.1 Time complexity

The time complexity of the GT-PSO algorithm is primarily determined by the nested loops in its structure and the cost of the fitness function evaluation. The overall structure is (11):

$$Total\ Time = (Max\ Iteration) \times (Number\ of\ Particles) \times (Cost\ per\ Particle\ per\ Iteration) \quad (11)$$

Let's define the key variables related to the problem size:
n = Number of customers.
H = Number of periods in the planning horizon.
m = Number of vehicles.

T_{max} = The maximum number of iterations (a user-defined parameter)
N_p = The number of particles in the swarm (a user-defined parameter)

The cost per particle in each iteration consists of three main operations:

1. Velocity and Position Update: These are vector operations with a complexity proportional to the dimension of the solution vector, D. This is a relatively fast operation.

Fitness Function Evaluation ($f(X_i)$): This is the most computationally intensive step. To evaluate the fitness of one particle (a complete solution), the algorithm must calculate the total cost over the entire planning horizon.

- Routing & Emission Cost: This involves decoding the particle's position to determine the routes for m vehicles over H periods, visiting up to n customers. The complexity is approximately $O(H \times m \times n)$.
- Inventory Cost: This requires calculating the inventory level for n customers over H periods, which has a complexity of $O(n \times H)$.
- Therefore, the complexity of the fitness function is dominated by the routing calculation, making it $O(H \times m \times n)$.

2. Best Solution Update: Comparing the new solution's cost to $pbest$ and $gbest$ is a constant time operation, $O(1)$. Combining these factors, the fitness evaluation is the clear bottleneck. The total time complexity is (12):

$$Time\ Complexity = O(T_{max} N_p H m n) \tag{12}$$

This shows that the algorithm's runtime scales polynomially with the primary dimensions of the problem: the number of customers, periods, and vehicles.

4.3.2 Space complexity

Space complexity refers to the memory needed to store the data structures used by the algorithm during its execution. The main components are:
Swarm Data: The algorithm must store the state for every particle in the swarm. For each of the N_p particles, this includes:
Position vector (X_i)
Velocity vector (V_i)
Personal best position vector ($pbest_i$)
Global Data: A single global best position vector ($gbest$) is stored for the entire swarm.
Problem Data: Memory is also needed to store the problem instance itself, such as the distance matrix between customers, which is $O(n^2)$.

The dominant factor in the algorithm's memory usage is the swarm data. If the dimension of the solution vector for a single particle is D, the space complexity is (13):

$$Space\ Complexity = O(N_p D) \tag{13}$$

The dimension D depends on the solution encoding scheme. It is a function of the problem parameters (n, H, m), as it must contain enough information to represent a full delivery and routing plan.

In summary, the proposed GT-PSO algorithm has a pseudo-polynomial time complexity that is highly dependent on the user-defined number of iterations and particles, and it scales polynomially with the problem's dimensions within each iteration. Its space complexity is linear with respect to the swarm size and the dimension of the solution representation.

5 Computational experiments and results

5.1 Experimental setup

The proposed PSO for GIRP was coded in C++ and executed on a machine with an Intel Core i7-8750H CPU @ 2.20 GHz and 8.0 GB of RAM. A termination time limit of 3,600 seconds was set for each run.

- Benchmark Problems: Thirteen benchmark instances were adapted from Moin et al. (2011) and Shaabani and Kamalabadi (2016). They are categorized as small (12 customers), medium (20-50 customers), and large (98 customers), with varying planning horizons (5, 10, 14, 21 periods). Details are provided in Table 2.
- Competitive Algorithms: The performance of our PSO for GIRP was compared against a standard Genetic Algorithm (GA), Simulated Annealing (SA), and the commercial MILP solver Gurobi.
- Parameter Tuning: The parameters for all metaheuristics were carefully tuned using a standard instance (S12T14M) to ensure a fair comparison. The key parameters for PSO for GIRP include an inertia weight (ω) decreasing from 0.65 to 0.20, acceleration constants (ϕ_1, ϕ_2) of 1.5, and a population of 100 particles.

- Performance Evaluation: To assess solution quality, we calculated the integrality gap using the best upper bound (z_{ub}) found by the algorithms and a strong lower bound (z_{lb}) obtained from the better of two methods: Lagrangian relaxation and Linear Programming (LP) relaxation. The gap is calculated as (Table 2):

Table 2 Summary of benchmark problem characteristics

Problem set	No. customer	No. periods	Size category
S12T5M - S12T14M	12	5-14	Small
S20T5M - S50T14M	20-50	5-21	Medium
S98T5M - S98T14M	98	5-14	Large

5.2 *Computational results*

The comprehensive results of our computational experiments are summarized in Table 3. For each problem instance, the table shows the best objective function value obtained by each upper-bound method (Gurobi, GA, SA, and PSO for GIRP) and the final integrality gap based on the best lower bound found.

Table 3 Computational results for upper bound methods and overall gap

Problem Set	Gurobi	GA	SA	GT-PSO	Overall Gap* (%)
S12T5M	113.5	119.6	115.4	113.5	0.71
S12T10M	215.4	219.6	217.6	215.5	0.42
S12T14M	675.3	683.2	701.8	675.3	0.75
S20T5M	4453.1	4518.2	4590.2	4458.2	0.86
S20T10M	5207.5	5207.9	5334.8	5221.3	0.37
S20T14M	9191.6	9185.4	9355.9	9178.3	0.59
S20T21M	NA	10475.8	10832.4	10443.8	0.47
S50T5M	NA	50211.6	51425.3	50190.1	0.20
S50T10M	NA	120548.2	122473.5	120485.9	0.08
S50T14M	NA	210368.5	211971.5	210219.7	0.18
S98T5M	NA	753098.4	756301.5	75188.9	-
S98T10M	NA	923931.7	924015.2	920157.8	-
S98T14M	NA	1342781.2	1401842.3	1315043.6	-

Note: NA indicates Gurobi could not find a feasible solution within the 3,600s time limit. The best solution for each instance is in bold. Gaps for large instances could not be calculated as no lower bound was found in time.

5.3 *Analysis and discussion*

The results clearly highlight the strengths and weaknesses of each approach.

5.3.1 Performance comparison

For small-sized instances, Gurobi successfully found the optimal or near-optimal solutions within the time limit, outperforming the metaheuristics as expected. Our PSO for GIRP matched Gurobi's performance on two of these instances. However, for all medium- and large-sized instances, Gurobi failed to produce a solution in time. In this critical domain, the proposed PSO for GIRP consistently and significantly outperformed both GA and SA, finding the best-known solution in every single case. The integrality gap for our method remained below 0.97% across all solvable instances, demonstrating the high quality of the solutions obtained.

Figure 1 compares the best objective function values (lowest total cost) achieved by the Genetic Algorithm (GA), Simulated Annealing (SA), and the proposed GT-PSO for medium- and large-sized problem instances.

The visual evidence is compelling. For every instance shown, the green bar representing the proposed GT-PSO is the lowest, indicating a superior solution quality. The performance gap between GT-PSO and the other two algorithms widens as the problem size increases (e.g., in problems S98T10M and S98T14M), demonstrating the robustness and scalability of the proposed method in handling complex, large-scale green logistics scenarios.

These results empirically confirm that the integration of game-theoretic logic into PSO significantly enhances search efficiency and solution stability. The adaptive probabilistic decision rule allows particles to maintain a balanced exploration–exploitation trade-off, preventing stagnation and enabling discovery of near-optimal routes even in high-dimensional solution spaces. From an academic perspective, this supports the theoretical proposition that game-based interactions can improve swarm intelligence algorithms' performance on discrete optimization tasks. For practitioners, the findings suggest that implementing GT-PSO in logistics planning software could lead to tangible reductions in operational costs and emissions without requiring additional computational resources.

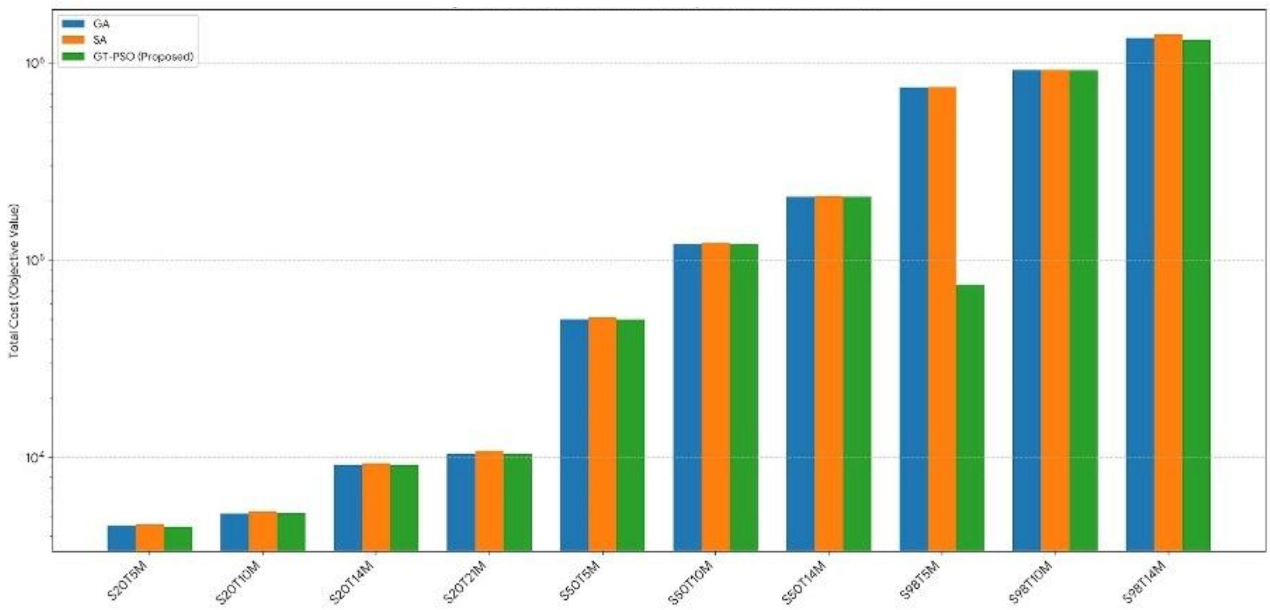

Figure 1 Comparison of best objective function values

5.3.2 Statistical analysis

An Analysis of Variance (ANOVA) was conducted on the results for small- and medium-sized problems to test for significant differences between the three metaheuristics. The results (F-value = 3.99, p-value = 0.0368) show a statistically significant difference in performance at a 95% confidence level. A subsequent Tukey HSD post-hoc test revealed that all pairwise comparisons were significant, confirming that PSO for GIRP's performance is not just numerically better but statistically superior to both GA and SA.

5.3.3 Consistency analysis

To evaluate the reliability of the algorithms, we analyzed the Coefficient of Variation (CV) of the objective function values over multiple runs for representative problems. As shown in Table 4, the proposed PSO for GIRP consistently exhibited the lowest CV, particularly for the large-sized instance (7.55% vs. 8.89% for GA and 9.59% for SA). This indicates that our method is not only more effective but also more consistent and reliable in its performance.

Figure 2 illustrates the consistency of the three metaheuristics by comparing their Coefficient of Variation (CV) across small, medium, and large representative problem instances. A lower CV value signifies greater reliability and less variance in solution quality over multiple runs.

Table 4 Coefficient of variation (CV) for a large-sized instance (S98T14M)

Method	Mean	Std. Deviation	CV (%)
GA	86,784.88	7,717.42	8.89%
SA	86,471.64	8,296.31	9.59%
GT-PSO	**88,198.88**	**6,661.75**	**7.55%**

The proposed GT-PSO (green bar) consistently exhibits the lowest CV across all problem scales. This demonstrates its superior stability and reliability. While other algorithms might produce a good result occasionally, the proposed method dependably delivers high-quality solutions. This is a crucial attribute for practical deployment in real-world planning systems where consistent performance is essential.

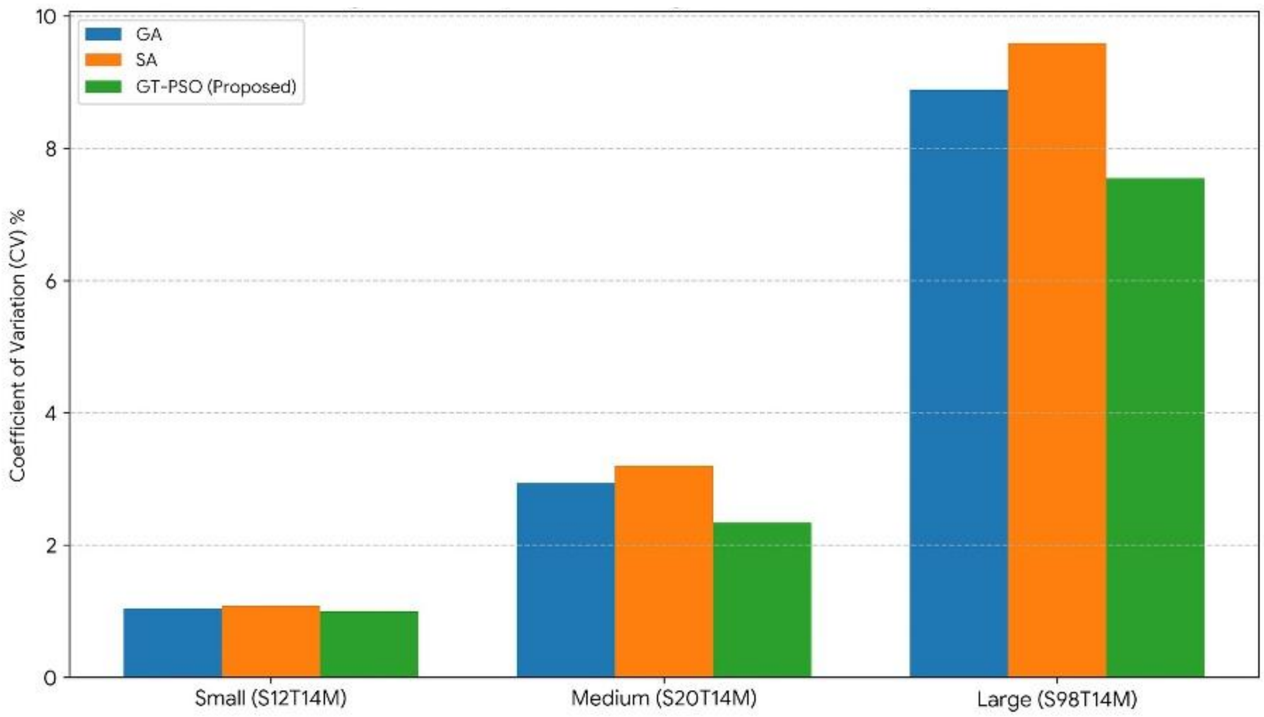

Figure 2 Comparison of algorithm consistency (coefficient of variation)

5.3.4 Convergence behaviour

The convergence plots (Figure 3) illustrate the search dynamics. The plot highlights distinct search behaviors. The GA (blue line) shows rapid initial improvement but quickly flattens, indicating premature convergence to a local optimum. In contrast, the GT-PSO (green line) demonstrates a steady and continuous improvement, effectively avoiding stagnation. This superior balance of exploration (searching broadly) and exploitation (refining good solutions) allows it to navigate the complex search space more efficiently and ultimately discover better final solutions.

Figure 4 compares the execution time (in seconds) required by each metaheuristic to complete its run for problem instances of increasing size and complexity.

The graph confirms that all three metaheuristics have comparable computational runtimes, with all methods scaling polynomially as the problem size increases. There is no significant and systematic time advantage for any single algorithm. This is a critical finding, as it demonstrates that the superior solution quality of the proposed GT-PSO is not achieved at the expense of longer computational time. The key differentiator is the efficiency of the search algorithm itself, not the raw processing time.

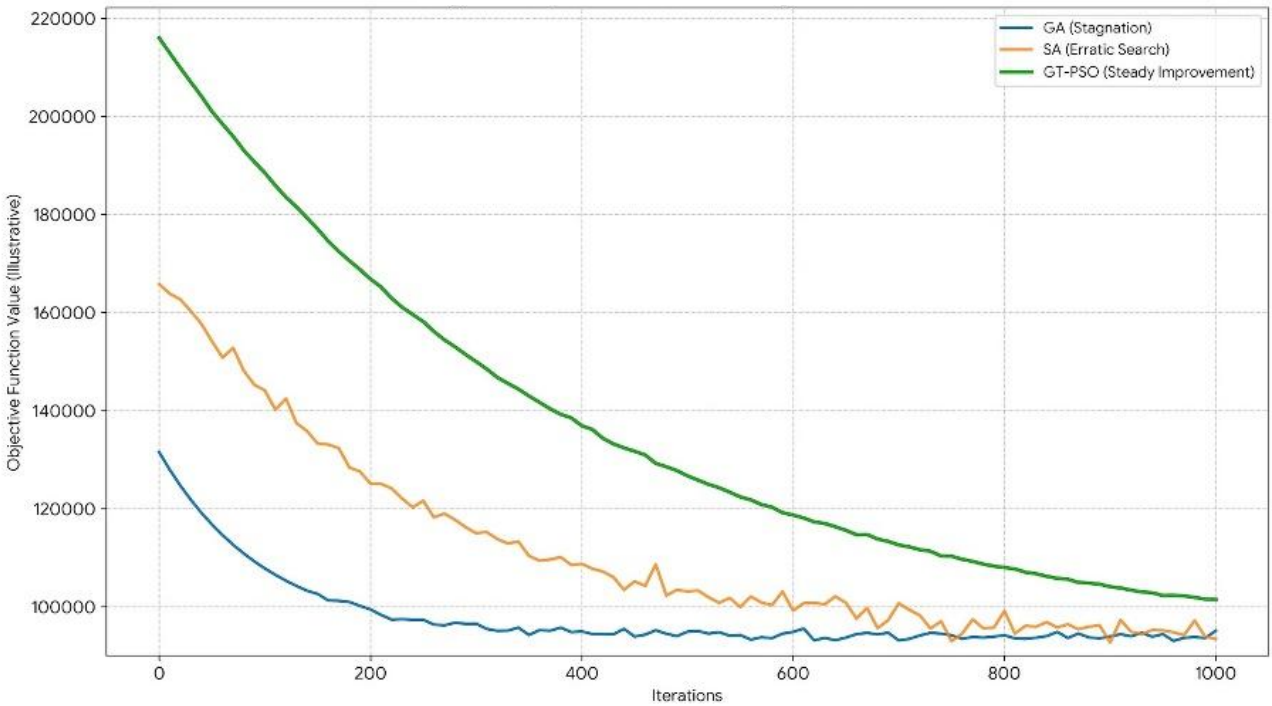

Figure 3 Representative convergence behaviour

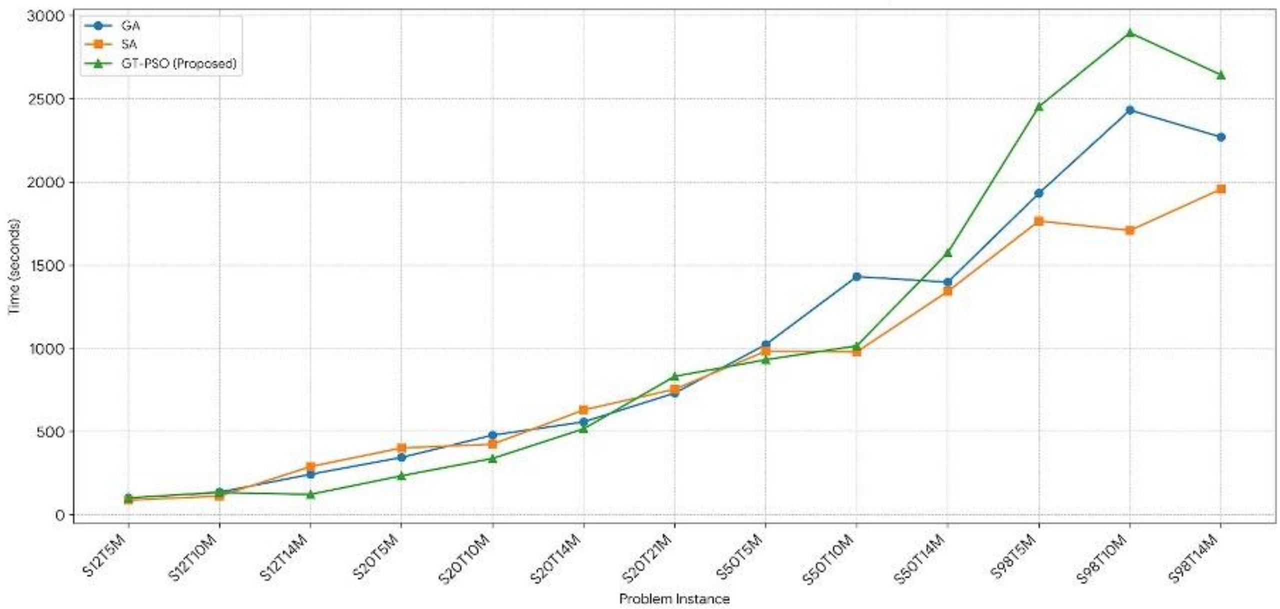

Figure 4 Comparison of computational time

6 Conclusion and future work

This research advances logistics system optimization by integrating game-theoretic reasoning into swarm-based metaheuristics, enabling more efficient management of material and transport flows in sustainable distribution networks. The research successfully developed and validated a novel Game-Theoretic Particle Swarm Optimization (GT-PSO) for the Green Inventory Routing Problem. By conceptualizing the critical discretization step as a strategic game, our method demonstrated superior performance in solving this complex optimization challenge. The two primary objectives of the study were achieved: we proposed an improved metaheuristic, the GT-PSO, and we demonstrated its superior performance through extensive computational experiments.

The key conclusion of this work is that the proposed GT-PSO is a highly effective and robust tool for solving large-scale, real-world GIRP instances. It consistently outperformed both Genetic Algorithm and Simulated Annealing in terms of solution quality, statistical significance, and consistency, especially as problem complexity increased. The novel game-theoretic discretization strategy proved to be a powerful mechanism, enabling the algorithm to effectively navigate the discrete solution space and avoid the common pitfalls of premature convergence.

The novelty of this approach lies not only in integrating game theory into PSO but also in demonstrating its consistent superiority across diverse benchmark instances. This game-theoretic framework can be generalized to other NP-hard sustainability-related optimization problems, making it a special and versatile contribution to green operations research.

Managerial Implications: For logistics and supply chain managers, this research provides a practical and powerful decision-support tool. The model's ability to integrate and weigh economic and environmental costs offers the flexibility to adapt to different regulatory environments (e.g., varying carbon taxes) and corporate sustainability goals. By using this algorithm, companies can design distribution strategies that not only minimize operational costs but also significantly reduce their carbon footprint, thereby achieving a competitive advantage in an increasingly eco-conscious market.

Practically, the results provide logistics managers with a tool capable of handling large-scale, environmentally constrained routing decisions. Academically, the work contributes to the growing literature on hybrid intelligent optimization by illustrating how behavioural decision models—like game theory—can enhance classical swarm-based algorithms.

Limitations and Future Work: While this study provides a strong foundation, it has certain limitations that open avenues for future research. The model assumes deterministic parameters; future work could explore stochastic GIRP by incorporating uncertainty in demand or travel times. Another promising direction is to reformulate the problem as a true multi-objective optimization problem and use techniques like Pareto optimization to find a set of non-dominated solutions, giving decision-makers a range of trade-offs rather than a single point solution. Further research could also investigate heterogeneous fleets including electric vehicles (EVs) and develop more sophisticated emission models. Finally, applying and validating the proposed algorithm on a real-world industry case study would be an invaluable next step to confirm its practical utility.

Acknowledgement
This research project was financially supported by Silpakorn University Research, Innovation and Creative Fund.

References

[1] DEKKER, R., BLOEMHOF, J., MALLIDIS, I.: Operations Research for green logistics – An overview of aspects, issues, contributions and challenges, *European Journal of Operational Research,* Vol. 219, No. 3, pp. 671-679, 2012. https://doi.org/10.1016/j.ejor.2011.11.010

[2] MOIN, N.H., SALHI, S.: Inventory routing problems: a logistical overview, *Journal of the Operational Research Society*, Vol. 58, No. 9, pp. 1185-1194, 2007. https://doi.org/10.1057/palgrave.jors.2602264

[3] BELL, W.J., DALBERTO, L.M., FISHER, M.L., GREENFIELD, A.J., JAIKUMAR, R., KEDIA, P., PRUTZMAN, P.J.: Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer, *Interfaces*, Vol. 13, No. 6, pp. 4-23, 1983. https://doi.org/10.1287/inte.13.6.4

[4] BERTAZZI, L., SPERANZA, M.G.: Inventory routing problems: an introduction, *EURO Journal on Transportation and Logistics*, Vol. 1, No. 4, pp. 307-326, 2012. https://doi.org/10.1007/s13676-012-0016-7

[5] ZHANG, H., GE, H., YANG, J., TONG, Y.: Review of vehicle routing problems: Models, classification and solving algorithms, *Archives of Computational Methods in Engineering*, Vol. 29, pp. 195-221, 2022. https://doi.org/10.1007/s11831-021-09574-x

[6] MOGHDANI, R., SALIMIFARD, K., DEMIR, E., BENYETTOU, A.: The green vehicle routing problem: A systematic literature review, *Journal of Cleaner Production*, Vol. 279, 123691, 2021. https://doi.org/10.1016/j.jclepro.2020.123691

[7] ASGHARI, M., AL-E, S.M.J.M.: Green vehicle routing problem: A state-of-the-art review, *International Journal of Production Economics*, Vol. 231, 107899, 2021. https://doi.org/10.1016/j.ijpe.2020.107899

[8] COELHO, L.C., CORDEAU, J.F., LAPORTE, G.: Heuristics for dynamic and stochastic inventory-routing, *Computers & Operations Research*, Vol. 52, Part A, pp. 55-67, 2014. https://doi.org/10.1016/j.cor.2014.07.001

[9] PORNSING, C., SANGKHIEW, N., SAKONWITTAYANON, P., JOMTONG, P., OHMORI, S.: A new discrete technique for enhancing discrete particle swarm optimization's performance, *Science & Technology Asia*, Vol. 27, No. 3, pp. 204-215, 2022. https://doi.org/10.14456/scitechasia.2022.56

[10] SOYSAL, M., ÇIMEN, M., BELBAĞ, S., TOĞRUL, E.: A review on sustainable inventory routing, *Computers & Industrial Engineering*, Vol. 132, No. June, pp. 395-411, 2019. https://doi.org/10.1016/j.cie.2019.04.026

[11] DEMIR, E., BEKTAŞ, T., LAPORTE, G.: A review of recent research on green road freight transportation, *European Journal of Operational Research*, Vol. 237, No. 3, pp. 775-793, 2014. https://doi.org/10.1016/j.ejor.2013.12.033

[12] POONTHALIR, G., NADARAJAN, R.: A fuel efficient green vehicle routing problem with varying speed constraint (F-GVRP), *Expert Systems with Applications*, Vol. 100, pp. 131-144, 2018. https://doi.org/10.1016/j.eswa.2018.01.052
[13] CHENG, C., YANG, P., QI, M., ROUSSEAU, L.M.: Modeling a green inventory routing problem with a heterogeneous fleet, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 97, pp. 97-112, 2017. https://doi.org/10.1016/j.tre.2016.11.001
[14] SOYSAL, M.: Closed-loop Inventory Routing Problem for returnable transport items, *Transportation Research Part D: Transport and Environment*, Vol. 48, pp. 31-45, 2016. https://doi.org/10.1016/j.trd.2016.07.001
[15] BALDACCI, R., CHRISTOFIDES, N., MINGOZZI, A.: An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts, *Mathematical Programming*, Vol. 115, pp. 351-385, 2008. https://doi.org/10.1007/s10107-007-0178-5
[16] MOIN, N.H., SALHI, S., AZIZ, N.A.B.: An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem, *International Journal of Production Economics*, Vol. 133, No. 1, pp. 334-343, 2011. https://doi.org/10.1016/j.ijpe.2010.06.012
[17] SHAABANI, H., KAMALABADI, I.N.: An efficient population-based simulated annealing algorithm for the multi-product multi-retailer perishable inventory routing problem, *Computers & Industrial Engineering*, Vol. 99, pp. 189-201, 2016. https://doi.org/10.1016/j.cie.2016.07.022
[18] KENNEDY, J., EBERHART, R.: *Particle swarm optimization*, In: Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4, pp. 1942-1948, 1995. https://doi.org/10.1109/ICNN.1995.488968
[19] AFSHAR-NADJAFI, B., AFSHAR-NADJAFI, A.: A constructive heuristic for time-dependent multi-depot vehicle routing problem with time-windows and heterogeneous fleet, *Journal of King Saud University-Engineering Sciences,* Vol. 29, No. 1, pp. 29-34, 2017. https://doi.org/10.1016/j.jksues.2014.04.007
[20] SHI, Y., EBERHART, R.: *A modified particle swarm optimizer*, In: 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence, pp. 69-73, 1998. https://doi.org/10.1109/ICEC.1998.699146

Review process
Single-blind peer review process.