Journal of
Marine Science and Engineering

MDPI

Article

Solving the Inter-Terminal Truck Routing Problem for Delay Minimization Using Simulated Annealing with Normalized Exploration Rate

Muhammad Hanif Ramadhan [1], Imam Mustafa Kamal [2], Dohee Kim [1] and Hyerim Bae [1,*]

[1] Industrial Data Science and Engineering, Department of Industrial Engineering, Pusan National University, Busan 46241, Republic of Korea; mhanifr@pusan.ac.kr (M.H.R.); kimdohee@pusan.ac.kr (D.K.)
[2] Institute of Intelligent Logistics Big Data, Pusan National University, Busan 46241, Republic of Korea; imamkamal@pusan.ac.kr
* Correspondence: hrbae@pusan.ac.kr

Abstract: The growth in containerized shipping has led to the expansion of seaports, resulting in the emergence of multiple terminals. While multi-terminal systems increase port capacity, they also pose significant challenges to container transportation, particularly in inter-terminal movements. Consequently, the transportation delay of containers in inter-terminal operations demands crucial attention, as it can adversely affect the efficiency and service levels of seaports. To minimize the total transportation delays of the inter-terminal truck routing problem (ITTRP), we introduce simulated annealing with normalized acceptance rate (SANE). SANE improves the exploration capability of simulated annealing (SA) by dynamic rescaling of the transportation delay objective to modify the acceptance probability. To validate the quality of solutions provided by SANE, we have developed a mathematical model that provides a set of linear formulations for ITTRP constraints, avoiding the known set-partitioning alternative. Experimental results showed that for small-scale ITTRP instances, SANE achieved a solution close to the optimal. In larger instances with 100–120 orders, SANE found feasible suboptimal solutions within 15–21 seconds, which is unattainable using the exact solver. Further comparison with baselines indicates that SANE provides considerable improvements compared to both SA and Tabu search in terms of the objective value.

Keywords: inter-terminal transport; inter-terminal truck routing; simulated annealing; tabu search; mixed integer programming

check for updates

Citation: Ramadhan, M.H.; Kamal, I.M.; Kim, D.; Bae, H. Solving the Inter-Terminal Truck Routing Problem for Delay Minimization Using Simulated Annealing with Normalized Exploration Rate. *J. Mar. Sci. Eng.* **2023**, *11*, 2103. https://doi.org/10.3390/jmse11112103

Academic Editor: Mihalis Golias

Received: 5 October 2023
Revised: 27 October 2023
Accepted: 30 October 2023
Published: 2 November 2023

1. Introduction

The growth of the global economy has constantly led to higher demand for containerized shipping. With the recovery from the COVID-19 pandemic, international maritime trade soared by an estimated 3.2%, with a total of 11 billion tons of shipments in 2021 [1]. To keep up with the increase in demand, large ports are being expanded and constructed with multiple terminals [2]. Thus, decision making related to efficient container transport in a multi-terminal system has become an increasingly important research topic.

Typically, container terminals within seaports must routinely serve vessels, barges, and other hinterland transportation modes daily. There are two main processes by which container terminals operate: loading and unloading of the containers. Upon discharge from the vessel, the container is expected to be delivered directly to the customer. However, in most cases, a container is transported to the stacking area, transferred between terminals and logistics facilities, or transferred to different modes of transportation to meet all logistics requirements [3]. This container movement is known as inter-terminal transport (ITT) [2,4]. ITT serves as a crucial operational problem for seaports as it compensates for infrastructure differences between terminals. If not handled correctly, ITT might lead to a significant delay, which in consequence, influences the competitiveness and sustainability of the seaport [5].

Previous studies related to ITT were typically concerned with the efficient pick-up and delivery of containers between separated areas within the seaports [2,3,6–10]. While the mathematical model of the specific ITT problem of interest is typically presented, in most cases the problem is solved either heuristically or approximately by using metaheuristics due to the computational complexity of the problem [8–10]. The proposed metaheuristic algorithms in this case may include modification, adjustment, or both to the known metaheuristic method for each of the specific ITT problems [3,6,8–11]. A study of the loading and unloading of trains at the landside of container terminals was performed along with the performance analysis of different neighbor functions of SA [9]. Heilig et al. [4] proposed a hybrid simulated annealing (SA) solution to inter-terminal truck routing problems using heuristics as an initialization algorithm [6]. Their study was followed by the adjusted use of SA for the multi-objective optimization case [7]. The SA algorithm presented by Oudani [10] emphasizes an adjustment to the algorithm to always generate feasible solutions in an incomplete network of intermodal transportation. To the best of our knowledge, the impact regarding the modification of acceptance probability of SA in the context of an ITT-related field has not been explored. This study, therefore, attempts to explore the gap with the proposed method.

In this study, we consider a model of inter-terminal truck routing problem (ITTRP) inspired by Heilig et al. [4]. However, in contrast to their study, which uses a set partitioning formulation, we consider a mixed integer programming (MIP) model that linearizes the formulated constraints to have a straightforward solving implementation through commercial solvers. To efficiently solve the model, we propose a metaheuristic method that modifies the acceptance probability of SA. The contributions of this study can be summarized as follows: (1) We developed simulated annealing with normalized acceptance rate (SANE), which improves the exploration capability of SA through dynamic rescaling of the transportation delay objective to modify the acceptance probability. (2) To evaluate the quality of the solution provided by SANE, we developed a mathematical model that formally linearizes some of the constraints in Heilig et al. [4] without the need for set-partitioning. (3) We further demonstrate from the experimental results that SANE yields a better performance than the original SA and Tabu Search (TS).

The remainder of this paper is structured as follows: Section 2 explores previous works related to ITT, ITTRP, and previous modifications to SA. Section 3 discusses the formulation of the ITTRP and the working mechanism of the SANE. Section 4 presents the experimental results and a comparison with the baselines, as well as a further discussion. Finally, Section 5 presents the conclusions of this study and directions for future work.

2. Literature Review

2.1. Routing Problems and Inter-Terminal Transportation

Many of the routing problems, including the routing of ITT, can be seen as an extension of the vehicle routing problem (VRP), which has been extensively studied in the context of logistics [12–17]. In some cases, the given routing problem may be directly assigned into a well-defined problem. Šedivý et al. [12] demonstrated the application of a commercial solver optimization module to solve a beer distribution problem that can be directly translated into the traveling salesman problem (TSP). They solved the case with up to 200 instances by an evolutionary algorithm built into the commercial solver. Similarly, Dedović et al. [17] utilized the GNU linear programming toolkit to address the distribution of vehicle routing of consumer goods, providing an enhancement of performance compared to the previously used route from the study case. A study by Stopka [13] investigated the use of constructive OR methods, such as Clarke–Wright, Mayer, and nearest neighbor algorithm to model distribution routes in city logistics that can be formulated into a capacitated VRP. Further use of the exact method and heuristics for VRP cases has also been observed in [15] to solve the XpressBees logistic problem, where Branch and Bound, Dijkstra, Dynamic Programming, and Clarke–Wright were used to solve the routing problem of a fleet of vehicles.

Reflecting on the studies related to VRP, Weerasinghe et al. [11] mentioned the extensive use of OR methods including MIP, heuristics, and metaheuristics in container terminal operations and ITT-related research. An important study of ITT by Tierney et al. [2] introduced a mathematical formulation of multimodal ITT in the form of a time–space graph to model the ports of Hamburg and Rotterdam. This study used a two-step solution that first solved the relaxation of the ITT problem before reaching a final feasible solution. Following this study, a similar formulation of ITT problem was addressed by Hu et al. [8] by extending the approach using TS. TS could rapidly find the solution in large-scale transport demand, whereas the exact solver failed to obtain the result in a reasonable amount of time. Jin and Kim [18] proposed a mathematical model by approaching ITT from a collaborative perspective to minimize the total cost by sharing the delivery orders throughout different trucking companies, thereby improving overall profit. Cao et al. [19] investigated the impact of inland container depots (ICD) and their influence on the efficiency of ports. The discussed study provided insights for making strategic decisions in regard to the use of ICD.

Decisions involving ITT encompass multimodal transportation such as barges, rails, and trucks. However, because of the flexibility and common use of trucks for inland transportation, it is important to model truck-specific routing problems in inter-terminal operations [3,6,20]. Heilig et al. [4] introduced the ITTRP to model a container pick-up delivery problem of trucks by considering truck service time and constant delay per-container penalty into the optimization objective while putting an emphasis on reducing empty truck trips (ETT). The study solves the problem using constructive heuristics and SA to find a satisfactory solution within a short amount of time. A multi-objective perspective of ITTRP was also further studied by Heilig et al. [7], by including the minimization of truck emission as an addition to the total cost minimization. Further study by Adi et al. [3] proposed a learning-based approach by applying deep reinforcement learning to ITTRP. Moreover, this study extends the model to cooperative multi-agent deep reinforcement learning [20] where each truck represents a single agent. Closely related to ITTRP is the recent study by Baals et al. [14] in minimizing earliness–tardiness costs in supplier networks for a just-in-time truck routing problem (TRP-JIT). TRP-JIT shares similarity to ITTRP by considering the use of a hard time window for pick-ups and a soft time window for delivery. Distinctive to ITTRP, the model proposed in TRP-JIT only involved a single depot, whereas in ITTRP, each truck can start at any given container terminal location within the seaport.

While the studies by Heilig et al. [4] and Adi et al. [20] both proposed algorithms for solving the ITTRP, the constraints of the problem were described not with a mathematical formulation, but rather with a set of requirements of feasibility restrictions. Given such an informal description of the constraints, the IP problem was briefly formulated as a set-partitioning formulation and can be solved by column generation [4]. However, because the column size corresponding to the feasible routes may be extremely large, it is difficult to solve the formulation with an exact solution. Consequently, the specific procedure for the column generation for solving the exact solution has not yet been determined by Heilig et al. [4]; hence, requiring an additional procedure to be implemented.

2.2. Simulated Annealing and Modifications

Owing to its simplicity and effectiveness, the SA is widely used to find approximate solutions for many container logistics and transportation problems. Heilig et al. [4] used a hybrid SA to solve ITTRP using a constructive heuristic as the initialization method to SA. Oudani [10] proposed simulated annealing to solve the intermodal transportation problem on incomplete networks that is sufficiently efficient for real-life applications. In a more recent study [9], an SA was developed to minimize the total delay of trains for loading and unloading operations.

Despite the success of SA in the field, certain modifications to its internal procedures can significantly improve its performance. Suarez et al. [21] proposed an improvement to SA by using a modified sigmoid function to reformulate the acceptance probability of a given candidate solution. The improved method strictly outperformed the original SA in several baseline functions, such as Ackley, Dixon Price, and Rosenbrock. Similar to the previous study, a modified SA was found in a study of the supplier selection and order quantity allocation problem with nonlinear freight rates by Gonzalez-Ayala et al. [22]. In comparison to Suarez et al. [21], only sigmoid function was used to calculate the acceptance probability. Alnowibet et al. [23] developed a guided hybrid gradient-based SA to solve constrained global optimization problems. This method outperformed four other state-metaheuristic baselines. However, unlike our study, this application requires a continuous differentiable space to obtain the gradient components. In this study, we modified the SA by modifying the acceptance probability to handle a large negative delta energy function at the initial stage of optimization.

3. Problem Formulation and Methodology

3.1. Inter-Terminal Truck Routing Problem Formulation

For clarity, the complete mathematical notations in this section are described in Tables 1–3. The ITTRP comprises a set of container terminal locations, a set of trucks, and a set of orders. The task of each order $i \in O$ is to assign a truck to deliver a container from the pick-up location to the delivery location within a specified time window. Each order $i \in O$ is associated with a tuple of source/pick-up and delivery locations (s_i, d_i). To fulfill an order, an assigned truck must travel from its current location to the source location to pick-up a container, and finally deliver the container to its destination. The travel time of a given truck to fulfill order $j \in O$ given the previous order $i \in O$ is given by Equation (1):

$$D_{ij} = dist(d_i,\ s_j) + dist(s_j,\ d_j) \tag{1}$$

where $dist(a, b)$ is a distance function that returns the travel time from pick-up location $a \in L$ to destination location $b \in L$. Hence, $dist(d_i,\ s_j)$ is the empty truck trip time from the destination of order i to the pick-up location of order j, and $dist(s_j,\ d_j)$ is the time taken from the pick-up location to the delivery location of order j. Furthermore, each truck $k \in T$ can start at any location with an initial position $ip_k \in L$. Hence, we have a set of initial positions of trucks, IP. We can therefore define a time distance matrix $D = (D_{ij})\ \forall\ i, j \in IP \cap O$, where we calculate D_{ij} using Equation (1), if $j \notin IP$. In the case where i is an initial position, i.e., $i \in IP$, we assign order i as a dummy order, $d_i \in IP$ is the starting location of the truck, and s_i is undefined. If $j \in IP$, we set $D_{ij} = 0$, given that the truck does not need to return to its initial position. Moreover, the time window of order i $[a_i,\ b_i]$ is represented similarly to [18], specifically in terms of a_i. In this case, while a container is available at the source location at a specific time, a_i is the time adjusted from the time of the container arrival by adding a travel time from the source to the destination location. Hence a_i is the earliest possible time of delivery, instead of the time of container arrival.

Finally, we redesign some of the constraints in Heilig et al. [4] into a set of linear constraints that can be easily solved using an MIP solver. We further examine the model under a different objective function that minimizes the total transport delay under a piecewise linear function, where for each delayed order, a penalty is given by $p(t_i - b_i)$, leading to the objective function of (2). Hence, the mathematical model is given by

$$minimize\ p\textstyle\sum_{i\in O} y_i(t_i - b_i) \tag{2}$$

Subject to,

$$t_j - D_{ip,\ j}x_{ip,\ j} \geq 0,\ \forall ip \in IP,\ \forall j \in O \tag{3}$$

$$t_i \geq a_i,\ \forall\ i \in O \tag{4}$$

$$t_i - t_j + (M + D_{ij})x_{ij} \leq M,\ i \neq j,\ \forall i, j \in O \tag{5}$$

$$\sum_{i \in IP \cap O} x_{ij} = 1, \forall j \in IP \cap O \tag{6}$$

$$\sum_{j \in IP \cap O} x_{ij} = 1, \forall i \in IP \cap O \tag{7}$$

$$\sum_{i,j \in IP} x_{ij} = 0 \tag{8}$$

$$t_i - b_i - y_i M \leq 0, \forall i \in O \tag{9}$$

$$b_i - t_i - (1 - y_i)M \leq 0, \forall i \in O \tag{10}$$

$$x_{ij} \in \{0,1\}, \forall i,j \in IP \cap O \tag{11}$$

$$t_i \geq 0, \forall i \in O \tag{12}$$

$$y_i \in \{0,1\}, \forall i \in O \tag{13}$$

where decision variables in Equations (11)–(13) are described in Table 3. Constraint (3) ensures that the delivery time t_j is at least as short as the time of the first delivery if the transport order is performed from the initial position. Constraint (4) enforces each delivery time of order $i \in O$ to be at least as small as the earliest possible delivery time window a_i. If order j is performed immediately after order i, then $t_j \geq t_i$ must follow. Hence, constraint (5) is given, with M assigned a large positive number. Each order can only be performed once by a single truck; hence, constraints (6) and (7) are satisfied. A truck from the initial position $i \in IP$ could not go back to another initial position $j \in IP$, as expressed in Equation (8). Finally, the constraints (9) and (10) are given to force $y_i = 1$ if a transport order is delayed as in $t_i \geq b_i$ and $y_i = 0$ otherwise, thereby contributing to the objective function (2).

Table 1. Notation for sets.

Sets	
L	A set of all container terminal locations
T	A set of all trucks
O	A set of all orders
IP	A set of truck initial positions described by $IP = \bigcap_{k \in T} ip_k$, where $ip_k \in L$ is the initial position of truck $k \in T$

Table 2. Notation for parameters.

Parameters	
(i, j)	Index of an order or index of $IP \cap O$. To avoid ambiguity, we always state whether $i, j \in O$ or $i, j \in IP \cap O$ when such index is used.
k	Index of a truck, $k \in T$
ip	Index of initial position, $ip \in IP$
s_i	Source/pick-up location of order $i \in O$, where $s_i \in L$
d_i	Destination location of order $i \in O$, where $d_i \in L$
a_i	The earliest possible time of delivery of order $i \in O$
b_i	The time of delivery deadline
p	A penalty per-unit of time given for each late order, see the objective function (2) for more detail
D_{ij}	Travel time of delivering order $j \in O$ right after performing order $i \in O$
M	Big M notation to describe a large positive real number

Table 3. Notation for decision variables.

	Decision Variables
x_{ij}	Binary decision variable that returns 1 if order $j \in IP \cap O$ is performed immediately after order $i \in IP \cap O$ and return 0 otherwise
t_i	Continuous decision variable to represent the time of delivery of order $i \in O$
y_i	Binary decision variable that returns 1 if order $i \in O$ is late and returns 0 otherwise

3.2. Simulated Annealing with Normalized Acceptance Rate (SANE)

SA typically starts with a high exploration rate in the early stages of optimization and then performs more exploitation toward the end of the iteration. To construct a candidate solution s', we perform a random switch between the two index elements of solution s. With $f(\cdot)$ as the objective function (2) on a given solution input and given that $f(s') \geq f(s)$, the acceptance probability given a temperature c is typically calculated as in Equation (14) in the case of the original SA [24]. However, specific to our ITTRP formulation, the value of $f(s) - f$ from (s') (14) is often observed to have an extremely large negative value at the beginning of the iteration. Hence, the acceptance probability P (14) is small, even at the early stage of optimization, leading to an early high exploitation rate.

$$P = exp\left(\frac{f(s) - f(s')}{c}\right) \tag{14}$$

This study proposes a simple yet effective modification to calculate the acceptance probability by normalizing with respect to $max\{f(s), f(s')\}$. Instead of directly measuring the difference between the two objective values, we consider the relative difference $(f(s) - f(s'))/max\{f(s), f(s')\}$. Given that exploration of SA is performed when $f(s') > f(s)$, $max\{f(s), f(s')\}$ simplifies to $f(s')$ and we therefore calculate β' as in Equation (15). This increases the exploration rate and, hence, the variance of the objective value at the starting stage of optimization. We further extend the acceptance probability modification by adding a "dampening" term $I_{(-\beta' \leq \alpha)}$ with the parameter α as in Equation (16). In this case, α provides the intensity of the hard dampening. The intention of dampening is to deterministically prevent the acceptance of the candidate s', where $f(s') \gg f(s)$. Consequently, we use the alternative acceptance probability P' (17) instead of P.

$$\beta' = \frac{f(s) - f(s')}{c \times f(s')} \tag{15}$$

$$I_{(-\beta' \leq \alpha)} = \begin{cases} 1, & if\ -\beta' \leq \alpha c \\ 0, & otherwise \end{cases} \tag{16}$$

$$P' = I_{(-\beta' \leq \alpha)} exp(\beta') \tag{17}$$

The complete SANE algorithm for ITTRP transport delay minimization is described in Algorithm 1. The initialization step discussed in Section 3.3 is used to ensure solution feasibility. The procedure then enters the main loop, which terminates when a certain stopping criterion is satisfied. In the main loop, a candidate solution is selected according to a neighbor function. A candidate solution is then evaluated to obtain f'. If $f' \geq f$, then there is a probability of P' to accept the candidate. In the last line of the main loop, we update and track a certain stopping criterion that would eventually lead to convergence.

Algorithm 1 SANE for ITTRP Transport Delay Minimization

Initialize:

Truck indices $\gamma = \{0,\ 1, \ldots |T| - 1\}$

Transport order indices $\delta = \{|T|,\ |T| + 1, \ldots, |T| + |O| - 1\}$

Parameter: temperature c, dampening parameter α, decay rate θ

Generate initial solution $s_0 = 0 \cap\ \{\text{any permutation of} \gamma \cap \delta/0\}$ (see Section 3.3)

Let $s \leftarrow s_0$

Evaluate $f \leftarrow f(s)$

While stopping criterion is not met:

  pick a candidate s' by random swap: $s' \leftarrow random_{swap(s)}$

  evaluate $f' \leftarrow f(s')$

  if $f' \geq f$:

    calculate β' with Equation (15)

    if $-\beta' \leq \alpha$:

      assign $s' \leftarrow s$ with probability P' (17)

    else:

      keep solution s unchanged

  decay the temperature $c\ \leftarrow \theta\ x\ c$

  update stopping criterion

3.3. Solution Representation

To represent any arbitrary feasible ITTRP solution for metaheuristics, we first describe a set of truck indices $\gamma = \{0,\ 1, \ldots |T| - 1\}$ and a set of transport order indices $\delta = \{|T|,\ |T| + 1, \ldots, |T| + |O| - 1\}$. A solution to an ITTRP can be represented as any permutation of $\gamma \cap \delta/0$. For a given solution sequence s, truck k is assigned to all transport order indices placed after truck k until the next truck l. Concretely, an example of the solution representation with three trucks and six transport orders is illustrated in Figure 1. In this case, truck 0 is assigned to transport orders 4, 7, and 6 sequentially; similarly, trucks 1 and 2 are assigned the same orders. Furthermore, for every transport order assignment j, the time of delivery t_j is assigned, as shown in Equation (18).

$$t_j = \max\{a_j,\ x_{ij}t_{ij} + t_i\} \tag{18}$$

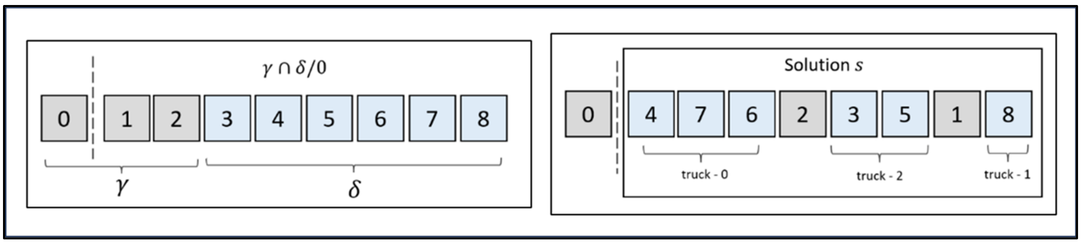


Figure 1. Solution representation of the ITTRP for metaheuristics implementation.

4. Experiments and Discussion

4.1. Experimental Settings

The experiments were conducted based on the environment and location of Busan Port. This study considers five main container terminal locations of Pusan New Port: PNIT, PNC, HJNC, HPNT, and BNCT. The average time required to travel between terminals was taken from [25]; hence, by randomly adding the set truck initial positions, the time distance matrix D can be constructed as described in Section 3, with Equation (1). To obtain the transport order data, we used a data generation procedure similar to that used previously [3,20]. A set of pick-up and delivery location pairs was obtained by sampling the container from a container throughput rate distribution adapted from [25]. An example of the throughput rate distribution table is given by Table 4. For a set of orders with the same pair of pick-up and delivery locations, the same time window is given by sampling the earliest delivery time uniformly at random from 24 h intervals and then randomly selecting the window size within 1–3 h. Therefore, we constructed datasets with six categories and varying parameters for the number of trucks and orders, as shown in Table 5. Each category consisted of ten instances.

Table 4. Example of the container throughput rate distribution.

From/To	PNIT	PNC	HJNC	HPNT	BNCT
PNIT	0.0%	6.6%	0.9%	3.3%	9.2%
PNC	9.3%	0.0%	9.1%	0.6%	8.2%
HJNC	4.3%	10.0%	0.0%	2.1%	7.8%
HPNT	1.7%	8.1%	2.6%	0.0%	5.2%
BNCT	6.3%	0.6%	2.0%	1.9%	0.0%

Table 5. Generated Busan Port transport order dataset.

Dataset	Number of Trucks	Number of Orders
ITT010-2	2	10
ITT015-3	3	15
ITT030-6	6	30
ITT060-9	9	60
ITT100-12	12	100
ITT120-15	15	120

We used the TS and original SA as baselines to further evaluate the SANE performance. We set a Tabu tenure of 10 as the required TS parameter, which corresponds to the number of iterations for which a Tabu solution is maintained as a Tabu. If the objective value did not change for the last 300 iterations, it was declared converged. Subsequently, aspiration was performed. In other words, if a Tabu solution yields the best solution thus far, it is still chosen as the newly updated solution. For the SA and SANE parameter settings, exponential decay was chosen as the temperature-cooling strategy. The initial temperature was one with a decay rate of 0.001. The dampening parameter α for SANE was set into 0.2. We declare that the algorithms converge if the objective value remains the same over the last 3000 iterations. A much larger convergence patience was given compared to that of TS because the iteration of SA is faster than that of TS, given that it only evaluates one neighbor at a time.

In the following experiments, the initial solutions were generated from a random permutation with the same seed rather than using a heuristic algorithm as an initial solution. However, it is possible to replace the initialization method with constructive heuristics, similar to the hybrid-SA approach from Heilig et al. [4]. Such hybridization would be applicable not only to SA, but also to all the metaheuristics presented here, including our proposed method. However, in this study, we aimed to determine the effect of modifying the acceptance probability of the SA without any hybrid interventions.

4.2. Performance Comparison to Baselines

We solved the instances generated under the datasets shown in Table 5 to obtain the performance of SANE and the baselines, as seen in Table 6. Each value of the averaged total delay was obtained by averaging the objective value (2) over each of the 10 instances generated for each dataset category. Similarly, the averaged solving time is obtained by averaging the CPU time to solve each instance for each category.

The MIP was solved by using CPLEX to provide the optimal solution for dataset categories of smaller instances, which includes three categories (ITT010-2, ITT015-3, ITT030-6). For a problem with 10–15 orders, the solver was able to provide the optimal solution quickly, even compared to baselines and the proposed method. However, we observed an escalation of average solving time from 0.42 s to 58.84 s once the number of orders reached

30. We further attempted to test the MIP to solve larger instances when the number of orders is larger or equal to 60. However, even for a single problem instance where the number of orders is 60, the solver fails to return a solution within the span of 2 days. Hence, no MIP result is provided for ITT060-9, ITT100-12, and ITT120-15 due to the time infeasibility.

Table 6. Performance comparison of SANE and baseline methods.

Dataset	Averaged Total Delay Cost (Unit Price)				Averaged Solving Time (s)			
	MIP	TS	SA	SANE	MIP	TS	SA	SANE
ITT010-2	**200.8**	209.2	270.4	**200.8**	**0.05**	0.33	0.38	0.80
ITT015-3	**233.2**	296.4	271.6	239.6	**0.42**	1.25	0.56	1.14
ITT030-6	**681.6**	842.4	726.0	717.6	58.84	11.997	**1.36**	2.57
ITT060-9	-	679.2	589.2	**444.8**	-	109.82	**3.57**	5.87
ITT100-12	-	2350.4	2224.4	**1944.8**	-	717.60	**12.83**	15.15
ITT120-15	-	3431.6	3273.2	**2674.4**	-	817.52	**18.97**	20.69

The best-performing results in each dataset category are highlighted in bold.

It was observed that SANE was able to outperform TS and SA in terms of minimizing the total transportation delay cost. With a dataset category of smaller instances where the optimal value can be observed from MIP solutions, we were able to obtain solutions that are either optimal or close to the optimal. Specifically, we reached the optimal solutions for all 10 generated cases of ITT010-2, thereby yielding the same average objective value as the MIP solver. In the case of ITT015-3 and ITT030-6, SANE was able to obtain a considerably small margin compared to the optimal solutions. While the optimal total transportation delay cost of the remaining ITT060-9, ITT100-12, and ITT120-15 could not be confirmed by MIP, we were able to obtain smaller averaged objectives compared to the baselines.

In terms of computational time, SA and SANE always converged faster than TS, particularly as the number of orders and trucks increased. This is because the Tabu structure size increases with $|s|(|s|-1)$ as the length of the solution increases. Hence, the algorithm requires the evaluation of many candidate solutions in a single iteration. With SA and SANE, a neighbor is selected randomly and only once for a given number of iterations, which greatly reduces the number of objective value evaluations per iteration.

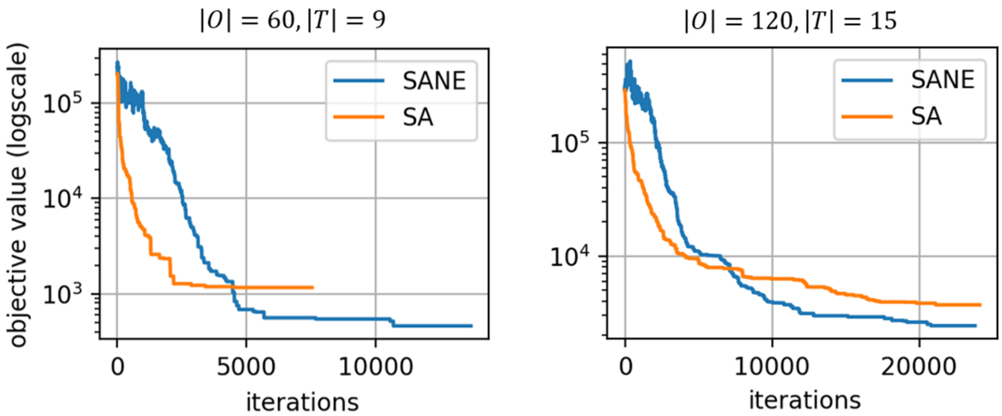
4.3. Exploration Behavior and Property of SANE

To contrast the behavior of SA and SANE, comparisons of both algorithms during the search for a suboptimal solution are given in Figure 2. In the SA case, it was observed that the beginning of the search barely indicates any fluctuations in the transportation delay. Such observations are obtained due to the extreme numerical difference in transportation delay of a candidate solution s', which typically occurs during an early stage of the search as observed in Figure 2. Exploration is considered whenever $f(s) < f(s')$. Hence a large negative value of the delta energy $f(s) - f(s')$ pushes the probability of exploration P (14) close to zero, resulting in the algorithm performing an exploitation intensive search.

SANE improved the condition by normalizing the delta energy into a relative difference $(f(s) - f(s'))/max\{f(s), f(s')\}$ to calculate β' (15). Given Equation (19), it is therefore possible to obtain the lower bound and upper bound value of the acceptance probability P' as Equation (20). Hence, it is easy to guarantee a nonzero acceptance probability regardless the magnitude of the delta energy $f(s) - f(s')$ by setting the temperature parameter c accordingly (e.g., setting temperature $c = 1$ would give P' a lower bound of 0.367). The decaying temperature c is therefore a crucial aspect of SANE as it is a procedure to slowly decrease the lower bound of P' toward zero.

$$-1 \leq \frac{f(s) - f(s')}{f(s')} < 0 \tag{19}$$

$$I_{(-\beta' \leq \alpha)} exp\left(-\frac{1}{c}\right) \leq P' < 1 \tag{20}$$



(**a**) (**b**)

Figure 2. Comparison of SA and SANE during the search for suboptimal solutions. (**a**) Simulation run with 60 orders and nine trucks. (**b**) Simulation run with 120 orders and 15 trucks. The different length of iterations for each algorithm and each run is due to the convergence criterion of 3000 iterations of unimproved objective values.

4.4. Algorithm Implications

Considering the property described by Equation (20), the effectiveness of SANE is obtained if the exploration strategy of SA fails due to acceptance probability vanishing at the early phase of search/optimization. Given that the performance may depend on the characteristics of the objective function, the potential for application is suggested toward cases with the similar or related objective functions. This includes other related ITT cases related to delay minimization [2,8] or scheduling problems that minimize tardiness-related costs [26].

5. Conclusions

Large ports are being expanded and constructed into multiple terminals to keep pace with the growth in demand. Hence, decision making related to efficient container transport in a multiterminal system is an important topic of research. ITT serves as an important operational issue for ports, as it compensates for the differences in infrastructure between terminals. An efficient ITT solution should minimize transportation delays. This study is concerned with minimizing the container transportation delay of an ITT system with trucks as the mode of transportation. We propose an alternative MIP model to the ITTRP that can optimally solve small-case problems. Consequently, the SANE approach was introduced to solve large-scale problems within a reasonable amount of time. Our method outperformed SA and TS in terms of transportation delay minimization.

We found that the dynamic rescaling of the objective function performed in SANE leads to a well-defined lower bound of acceptance probability. While the SA acceptance probability may diminish early in an exploitation-intensive search, SANE can guarantee a nonzero acceptance probability during the early stage of search. Consequently, we find that SANE provides an improvement of the objective function over the baselines while still maintaining a relatively small computing time, hence improving the practicality of solving real ITTRP cases with large instances.

Despite the possibilities given by the method, some areas of improvement can be explored. This study only considers the swapping operation to define the neighborhood of a solution. However, an investigation toward formulating a neighbor solution that considers the characteristics of ITTRP is important as they could potentially improve performance stability and time to solve the solution [9,27].

The proposed MIP can solve only limited to small instances of orders up to 30 as obtained in Table 6. While this hinders the direct applicability of the MIP, several data driven studies related to deep reinforcement learning show the promising direction toward exploiting MIP for approximation methods [28,29]. Exploring these topics in the context of ITTRP, as well as the comparison to SANE, are subjects of future research.

Author Contributions: Conceptualization, M.H.R. and I.M.K.; methodology, M.H.R.; data curation, M.H.R.; writing—review & editing, H.B. and D.K.; software, M.H.R.; validation, M.H.R. and I.M.K.; supervision H.B.; project administration, D.K.; resources H.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Korea Institute of Marine Science & Technology Promotion (KIMST) funded by the Ministry of Oceans and Fisheries, Republic of Korea (202205382) and supported by the MSIT (Ministry of Science and ICT), Republic of Korea, under the Grand Information Technology Research Center support program (IITP-2023-2020-0-01791) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The generated transport order datasets are publicly available. Related data and results can be found here: https://github.com/muhhanifra/ITTRP-Data (accessed on 4 October 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. UNCTAD UNCTADstat. Available online: https://unctadstat.unctad.org/EN/Index.html (accessed on 5 January 2023).
2. Tierney, K.; Voß, S.; Stahlbock, R. A Mathematical Model of Inter-Terminal Transportation. *Eur. J. Oper. Res.* **2014**, *235*, 448–460. [CrossRef]
3. Adi, T.N.; Iskandar, Y.A.; Bae, H. Interterminal Truck Routing Optimization Using Deep Reinforcement Learning. *Sensors* **2020**, *20*, 5794. [CrossRef] [PubMed]
4. Heilig, L.; Lalla-Ruiz, E.; Voß, S. Port-IO: An Integrative Mobile Cloud Platform for Real-Time Inter-Terminal Truck Routing Optimization. *Flex. Serv. Manuf. J.* **2017**, *29*, 504–534. [CrossRef]
5. Kadłubek, M.; Thalassinos, E.; Noja, G.G.; Cristea, M. Logistics Customer Service and Sustainability-Focused Freight Transport Practices of Enterprises: Joint Influence of Organizational Competencies and Competitiveness. *J. Green Econ. Low-Carbon Dev.* **2022**, *1*, 2–15. [CrossRef]
6. Heilig, L.; Voß, S. Inter-Terminal Transportation: An Annotated Bibliography and Research Agenda. *Flex. Serv. Manuf. J.* **2017**, *29*, 35–63. [CrossRef]
7. Heilig, L.; Lalla-Ruiz, E.; Voß, S. Multi-Objective Inter-Terminal Truck Routing. *Transp. Res. Part E Logist. Transp. Rev.* **2017**, *106*, 178–202. [CrossRef]
8. Hu, Q.; Luan, X.; Corman, F.; Lodewijks, G. A Tabu Search Algorithm for Inter-Terminal Container Transport. *IFAC-PapersOnLine* **2016**, *49*, 413–418. [CrossRef]
9. Gharehgozli, A.; Roy, D.; Saini, S.; van Ommeren, J.K. Loading and Unloading Trains at the Landside of Container Terminals. *Marit. Econ. Logist.* **2022**, *25*, 549–575. [CrossRef]
10. Oudani, M. A Simulated Annealing Algorithm for Intermodal Transportation on Incomplete Networks. *Appl. Sci.* **2021**, *11*, 4467. [CrossRef]
11. Weerasinghe, B.A.; Perera, H.N.; Bai, X. Optimizing Container Terminal Operations: A Systematic Review of Operations Research Applications. *Marit. Econ. Logist.* **2023**, 1–35. [CrossRef]
12. Šedivý, J.; Cejka, J.; Guchenko, M. Possible Application of Solver Optimization Module for Solving Single-Circuit Transport Problems. *LOGI Sci. J. Transp. Logist.* **2020**, *11*, 78–87. [CrossRef]
13. Stopka, O. Modelling Distribution Routes in City Logistics by Applying Operations Research Methods. *Promet-Traffic Transportation* **2022**, *34*, 739–754. [CrossRef]
14. Baals, J.; Emde, S.; Turkensteen, M. Minimizing Earliness-Tardiness Costs in Supplier Networks—A Just-in-Time Truck Routing Problem. *Eur. J. Oper. Res.* **2023**, *306*, 707–741. [CrossRef]
15. Malhotra, S.; Khandelwal, M. Solving XpressBees Logistics Problem by Using Exact and Heuristic Method. *LOGI Sci. J. Transp. Logist.* **2022**, *13*, 37–48. [CrossRef]

16. Nucamendi, S.; Cardona-Valdes, Y.; Angel-Bello Acosta, F. Minimizing Customers' Waiting Time in a Vehicle Routing Problem with Unit Demands. *J. Comput. Syst. Sci. Int.* **2015**, *54*, 866–881. [CrossRef]
17. Dedović, U.; Gušavac, B.A. Optimal Vehicle Routing in Consumer Goods Distribution: A GNU Linear Programming Kit-Based Analysis. *Acadlore Trans. Appl. Math. Stat.* **2023**, *1*, 87–95. [CrossRef]
18. Jin, X.; Kim, K.H. Collaborative Inter-Terminal Transportation of Containers. *Ind. Eng. Manag. Syst.* **2018**, *17*, 407–416. [CrossRef]
19. Cao, P.; Zheng, Y.; Yuen, K.F.; Ji, Y. Inter-Terminal Transportation for an Offshore Port Integrating an Inland Container Depot. *Transp. Res. Part E Logist. Transp. Rev.* **2023**, *178*, 103282. [CrossRef]
20. Adi, T.N.; Bae, H.; Iskandar, Y.A. Interterminal Truck Routing Optimization Using Cooperative Multiagent Deep Reinforcement Learning. *Processes* **2021**, *9*, 1728. [CrossRef]
21. Suarez, J.; Millan, C.; Millan, E. Improved Modified Simulated Annealing Algorithm for Global Optimization. *Contemp. Eng. Sci.* **2018**, *11*, 4789–4795. [CrossRef]
22. Gonzalez-Ayala, P.; Alejo-Reyes, A.; Cuevas, E.; Mendoza, A. A Modified Simulated Annealing (MSA) Algorithm to Solve the Supplier Selection and Order Quantity Allocation Problem with Non-Linear Freight Rates. *Axioms* **2023**, *12*, 459. [CrossRef]
23. Alnowibet, K.A.; Mahdi, S.; El-Alem, M.; Abdelawwad, M.; Mohamed, A.W. Guided Hybrid Modified Simulated Annealing Algorithm for Solving Constrained Global Optimization Problems. *Mathematics* **2022**, *10*, 1312. [CrossRef]
24. Burke, E.K.; Kendall, G. *Search Methodologies*; Springer: New York, NY, USA, 2014; ISBN 978-1-4614-6939-1.
25. Park, N.-K.; Lee, J.-H. The Evaluation of Backhaul Transport with ITT Platform: The Case of Busan New Port. *J. Fish. Mar. Sci. Educ.* **2017**, *29*, 354–364. [CrossRef]
26. Arık, O.A.; Schutten, M.; Topan, E. Weighted Earliness/Tardiness Parallel Machine Scheduling Problem with a Common Due Date. *Expert Syst. Appl.* **2022**, *187*, 115916. [CrossRef]
27. Misztal, W. The Impact of Perturbation Mechanisms on the Operation of the Swap Heuristic. *Arch. Automot. Eng.* **2019**, *86*, 27–39. [CrossRef]
28. Tang, Y.; Agrawal, S.; Faenza, Y. Reinforcement Learning for Integer Programming: Learning to Cut. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 9367–9376.
29. Qi, M.; Wang, M.; Shen, Z.-J. Smart Feasibility Pump: Reinforcement Learning for (Mixed) Integer Programming. *arXiv* **2021**, arXiv:2102.09663.