# Vehicle Dispatching at Seaport Container Terminals Using Evolutionary Algorithms

Jürgen Böse[1], Torsten Reiners[1], Dirk Steenken[2], and Stefan Voß[1]

[1] Institut für Wirtschaftswissenschaften
Technical University Braunschweig, D-38106 Braunschweig, Germany
E-mails: {j.boese, t.reiners, stefan.voss}@tu-bs.de

[2] Hamburger Hafen- und Lagerhaus-AG, DC/EDP Department Container
D-20457 Hamburg, Germany E-mail: steenken@hhla.de

## Abstract

*In the last four decades the container, as an essential part of a unit-load-concept, has scored a great success in international sea freight transportation. With increasing containerization the number of seaport container terminals and competition has increased considerably. One of the success factors of a terminal is related to the time in port for container vessels and the transshipment rates the ship operators have to pay.*

*In this paper we describe the main logistics processes in seaport container terminals and present methods for their optimization. We focus on the process of container transport by gantry cranes and straddle carriers between the container vessel and the container yard. The primary objective is the reduction of the time in port for the vessels by maximizing the productivity of the gantry cranes, or in other words, minimizing the delay times of container transports that causes the gantry cranes to stop.*

*We investigate different dispatching strategies for straddle carriers to gantry cranes and show the potential of evolutionary algorithms to improve the solutions.*

## 1 Introduction

Containers came into the market for international conveyance of sea freight in the early sixties. They continue to gain acceptance due to the fact that containers are the foundation for a unit-load-concept. Containers are relatively uniform boxes whose contents do not have to be unpacked at each point of transfer and have been designed for easy and fast handling of freight. Besides the advantages for the discharge and loading process, the standardization of metal boxes provides many advantages for the customers, as there are protections against weather and pilferage, and improved and simplified scheduling and controlling resulting in a profitable physical flow of cargo. Especially if two or more different means of transport are integrated into one single transport chain (e.g., the intercontinental container vessel operation or the land-based bi- or multimodal traffic) and a reloading of the containers has to be done, the precise information about their position, dimension, weight etc. helps in solving the existing dispatching and scheduling tasks [26, 7, 32].

First sea container service began 1961 with an international container service between the US East coast and points in the Caribbean, Central and South America. The breakthrough after a slow start was achieved with large investments in specially designed ships, adapted seaport terminals with suitable equipment, and availability (purchase or leasing) of containers. A large number of container transshipments[1] then led to economic efficiency and a rapidly growing market share.

Today over 60% of the world's deep-sea general cargo is transported in containers, whereas some routes, especially between economically strong and stable countries, are containerized up to 100% [26, 19]. An international containerization market analysis shows that in 1995 9.2 million TEU (Twenty foot Equivalent Units describing the length of a short container) were in circulation. The container fleet had almost doubled in ten years from a size of 4.9 million TEU

This research was supported by the German bmb+f (Bundesministerium für Bildung und Forschung).

[1] In this context, transshipment describes the transfer or change from one conveyance to another with a temporarily limited storage on the container yard.

in 1985. Figure 1 shows the growth rates for the ten largest seaport terminals in the world from 1993 to 1998 [14, 3, 4]. Due to the positive forecast for container freight transportation a similar development can be expected in the future.

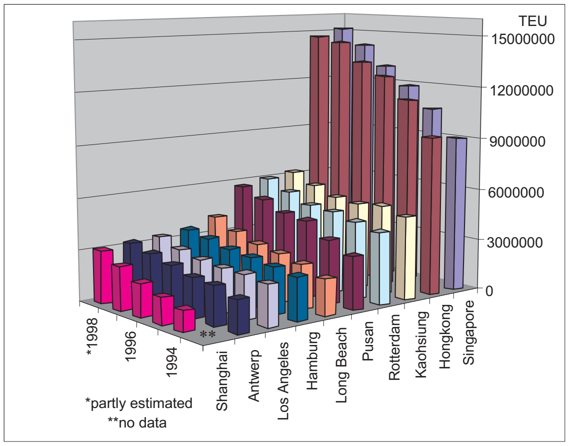


**Figure 1. Growth rates of the ten largest seaport terminals in the world from 1994 to 1998.**

The increasing number of container shipments causes higher demands on the seaport container terminals, container logistics, and management, as well as on technical equipment. Therefore, an increased competition between seaports, especially between geographically close ones, is a result of this development. The seaports mainly compete for ocean carrier patronage and short sea operators (feeders) as well as for the land-based truck and railroad services. The competitiveness of a container seaport is marked by different success factors, particularly the time in port for ships (transshipment time) combined with low rates for loading and discharging [26, 19]. Therefore, a crucial competitive advantage is the rapid turnover of the containers, which corresponds to a reduction of the time in port of the container ships, and of the costs of the transshipment process itself.

With regard to the described situation of world-wide container transport services this paper demonstrates an approach for reducing the time in port by optimizing the waterside transshipment process of a container port. Our aim is to increase the productivity of the gantry cranes used for container loading and discharge between ships and wharf, and of the vehicles engaged in container transports on the yard. Two approaches may be envisioned, as there are organizational changes as well as algorithmic improvements. While for the first case we focus on some ideas to modify the assignment of straddle carriers to bridges, in the second case modern heuristic search concepts are considered that make use of evolutionary and genetic algorithms. The model in this paper is based on the layout of the terminal Burchardkai in Hamburg, which is run by the Hamburger Hafen- und Lagerhaus-AG (HHLA).

The next section gives a general overview of the functionality of a container seaport terminal with a focus on physical container movements. Furthermore, we present a survey of current approaches for optimizing logistics processes in seaport container terminals. In Section 3 we describe a problem formulation that is later used for the application of a genetic algorithm (see Section 4). In Section 5 numerical results are presented. Next to the change of the strategy from static to semi-dynamic and dynamic, we explore the application of the genetic algorithm. In Section 6 conclusions from our approach are discussed with regard to the ongoing research project in the field of optimizing the time in port and the straddle carrier movements on the yard.

## 2 Functioning of a Seaport Container Terminal and Existing Optimization Approaches

A seaport container terminal is a complex system consisting of a number of interconnected logistics processes. Figure 2 shows typical operation areas of a container terminal along with the flow of containers (symbolized by the arrows). Subsequently a description of the illustrated spheres of action as well as the existing interdependencies is given.

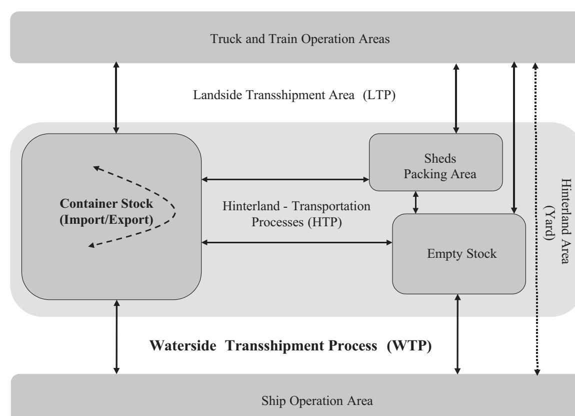


**Figure 2. Schematic visualization of logistics areas of a seaport container terminal.**

A container vessel enters the port and is assigned to an appropriate wharfage equipped with special container cranes (also called gantry cranes, see e.g., [28] for a detailed description). All containers assigned for transshipment are discharged or loaded by gantry cranes. They have to go through a limited storage buffer below the gantry crane being one of the main bottlenecks in the waterside transshipment process (see Section 3) due to the limited space. The container movements to and from the wharfage required for the loading and discharging process are carried out by mobile vehicles or crane installations (see [28, 26]). Depending on possible special requirements of the freight in the containers (reefer containers that need an electric connection, hazardous goods), they might be moved to special sites. The majority of the import containers are transported to positions in pre-reserved areas near the place where they will be transshipped next. Only a small number of them might be reloaded without intermediate steps on land-based means of transport. Generally, the container transshipment as well as the container movement to and from the wharf are associated with the waterside transshipment process (WTP), discussed in more detail in the next section.

Containers arriving by road or railway at the terminal are cleared within the provided truck and train operation areas. The containers are distributed to different stocks fulfilling the necessary requirements. The landside transshipment process (LTP) is carried out by specially constructed vehicles or gantry cranes (see [26, 20]). Furthermore, there are several internal moves between the different areas in the hinterland (container stock, packing area, and empty stock), see Figure 2 and [33]. These movements encompass the container transports from the empty stock to the packaging center, packed containers to the container stock as well as others. They are regarded as a part of the hinterland transportation processes (HTP). Figure 3 shows a simplified model of the terminal Burchardkai with container vessels, truck and train operations, and stock areas. This model is used for the computational experiments in this paper.

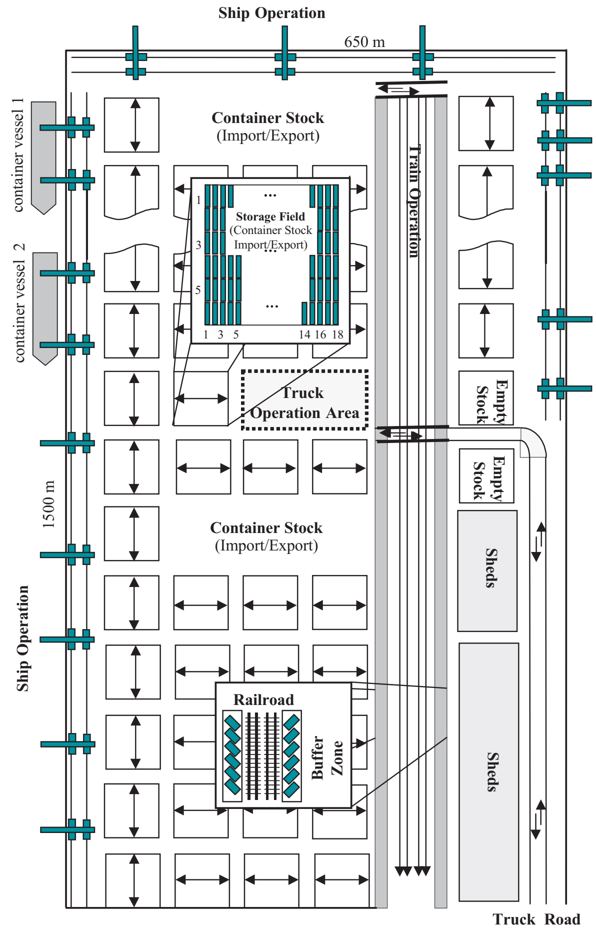


**Figure 3. Simplified model of the container terminal.**

To improve the different logistics processes of seaports (WTP, HTP, LTP), several approaches have been published in the past. They can be divided generally in two different approaches: simulation models and mathematical solution methods (heuristic and exact approaches).

Due to the multitude of stochastic influences (for instance ship delays, technical failure, human factors, etc.), numerous approaches use (discrete-event) simulation for recreating selected terminal processes or complete terminal installations. Capacity planning for technical resources and the spatial or functional design of terminals are investigated next to dispatching and scheduling strategies for (personal-)controlling the logistics processes. Simulation results provide valuable decision support information for terminal planners, operators, and managers (see, e.g., [37, 23, 27, 36, 11, 15, 25]). Analytical approaches that use modern queuing techniques instead of discrete-event simulation in order to evaluate the terminal allocation and layout planning problems can be found in, e.g., [21, 16].

With regard to a suitable automation as well as usage of computerized decision support systems some of these processes were analyzed in the past resulting in several promising algorithmic approaches. For example different heuristic and exact solution methods (mathematical solution approaches) have been published for the resource allocation problem, whereas the dispatching problems in container terminals have not had an important relevance in the literature so far. The objectives are an intelligent assignment of technical equipment (e.g., gantry cranes and straddle carriers) to the different terminal areas (compare the model in Figure 3) as well as an efficient job assignment to the utilized resources (see, e.g., [22, 10, 33, 6, 30]). These algorithms are mostly combined with suitable simulation models to analyze the influences on the logistic processes. In this context, another research area concerns berth and yard planning such as an assignment of ships to wharf positions [17] and containers to yard locations. Due to the large inter-dependency, the berth and yard planning are frequently considered in a common optimization model [10, 5, 34].

# 3 Problem Formulation

The primary focus of this paper is the discussion of ideas to reduce the time in port of container vessels. As a primary objective this may aim at improving the productivity of the employed gantry cranes. In this regard, we concentrate on the contemplation of the WTP, in particular, on the container movements to and from the wharfage, performed by straddle carriers. Thereby, maximizing gantry crane productivity can be achieved by efficiently scheduling/dispatching the given straddle carriers.

The static strategy uses a specified number of straddle carriers which are statically assigned to one gantry crane during a shift. In several cases, this assignment strategy for watersided container transports is a bottleneck due to the small storage buffer below the gantry cranes. In this paper, we set the size of the storage buffer to two containers (based on the construction of gantry cranes [28]) and the number of straddle carriers to three. Considering the discharging process, a gantry crane has to interrupt its work if the assigned straddle carriers are not able to remove containers from the container buffer in a given time slot depending on the throughput of the gantry crane. Analogously, the loading process stops if there is no container available for the transshipment to the storage buffer such that the gantry crane can work uninterruptedly to achieve a high productivity.

In this paper, we assume a constant average time $\overline{t_g},\ g \in \{1,\ldots,G\}$ for each of the $G$ gantry cranes allowing to define a sequence of jobs $j_{g,i}$ with fixed transshipment times where $i \in \{1,\ldots,n_g\}$ with $n_g$ being the number of jobs in the sequence. A job $j$ denotes a container transport from one location to another one. During the discharging process of a container vessel, every $\overline{t_g}$ seconds, a new container is placed in the storage buffer below the gantry crane. This container is available for a straddle carrier and can be scheduled for carrier assignment (*birth time* of the container). Similarly, for the loading process the straddle carriers have to deliver a container at a certain point of time to the gantry crane. Here, the birth time of the delivering job is the time when the transport from the storage position in the hinterland to the wharf has to begin such that it is available for the gantry crane on time (this time is calculated using the average transshipment time of the gantry crane assuming a just-in-time delivery). Furthermore, the number of straddle carriers is limited to $C$ carriers due to the costs for staff, material, and operating as well as space on the terminal.

The time $t_{g,i}^{birth}$ for the delivery to or pick up from the gantry crane $g$ of the $i^{th}$ job (container) within a sequence is calculated by

$$t_{g,i}^{birth} = \begin{cases} t_g^{initial} + i \cdot \overline{t_g} & (1) \\ t_g^{initial} + (i-1) \cdot \overline{t_g} - t_{g,i}^{trans} & (2) \end{cases}$$

with

| | |
|---|---|
| $t_g^{initial}$ | time where gantry crane $g$ starts to operate in the shift |
| $\overline{t_g}$ | average transshipment time of gantry crane $g$ |
| $t_{g,i}^{trans}$ | transportation time used to execute job $i$ of gantry crane $g$ |

where (1) describes the discharging process of containers by the gantry crane $g$ and (2) the loading process.

This model assumes that the jobs can be delivered to or removed from the storage buffers just-in-time which is not the case for every sequence of jobs for a straddle carrier. Each straddle carrier $c$ with $c \in \{1,\ldots,C\}$ has a sequence of $v_c$ jobs assigned to it. The end of a job for a straddle carrier $t_{c,k}^{death}$ (death or completion time) depends on the finishing time of the preceding job $t_{c,k-1}^{death}$, the empty running time $t_{c,k}^{empty}$ to reach the pick up position of the container, the time for the transport $t_k^{trans}$ to the destination, and the waiting time $t_{c,k}^{wait}$ whenever the straddle carrier reaches the pick up position too early (before the birth time of the job). This death time is calculated by the recursive function

$$t_{c,k}^{death} = t_{c,k-1}^{death} + t_{c,k}^{empty} + t_k^{trans} + t_{c,k}^{wait} \qquad (3)$$

with $k = 1,\ldots,v_c$ being the number of the job in the sequence for a straddle carrier. Note, that $t_{c,k}^{wait} = 0$ if the straddle carrier arrives at the pick up position after the birth time of this job. Due to the fact that the straddle carriers do not have to be assigned to a special gantry crane, the job number $k$ is not equivalent to the job number $i$ in the sequences of the gantry crane. Therefore, a mapping function to assign the jobs to a straddle carrier is used:

$$\Gamma : j_{g,i} \rightarrow j_{c,k} \qquad (4)$$

with $j_{c,k}$ being the $k^{th}$ job in the sequence for straddle carrier $c$ and $j_{g,i}$ being the $i^{th}$ job in the sequence for gantry crane $g$.

The planning horizon for the calculation is one shift with the duration $T$. Therefore, the length of the sequence for a straddle carrier is limited:

$$\sum_{k=1}^{v_c} \left( t_{c,k}^{empty} + t_k^{trans} + t_{c,k}^{wait} \right) \leq T \quad \forall c = 1,\ldots,C \qquad (5)$$

Furthermore, a job $j_{g,i}$ has to be unique and can only be assigned to exactly one straddle carrier.

The objective is the minimization of all delays (a delay specifies the time period when a straddle carrier arrives at the pick up position after the birth time), because a delay of a certain size forces the gantry cranes to wait and, therefore, lowers their productivity. The objective function is as follows:

$$\text{Min} \sum_{g=1}^{G} \sum_{i=1}^{J_g} \text{Max} \left( \left( t_{c,k-1}^{death} + t_{c,k}^{empty} \right) - t_{g,i}^{birth}, 0 \right) \quad (6)$$

Due to the assignment of the jobs to different straddle carriers, a mapping of the job $j_{c,k}$ to the corresponding job $j_{g,i}$ of the gantry crane $g$ is done using the function $\Gamma$ in (4) ($c \to g, k \to i$). Furthermore, $J_g$ is the number of jobs that the gantry crane $g$ has to work through during the shift.

The model is simplified in a way that we ignore stacking of containers and the destination for discharged containers on the terminal, and the location of containers to be loaded is supposed to be known in advance and kept constant during the shift. Furthermore, we assume an unrestricted storage buffer below the gantry cranes in the simplified model.

Real world influences (e.g. different speed of straddle carriers, failure of the technical equipment, etc.) are ignored at first and have to be examined with a simulation model together with the effects resulting from the described bottleneck situations (storage buffer below the gantry cranes).

To avoid the bottlenecks, we have to compute (job) sequences for the used straddle carriers where the sum of delays for the container transports becomes minimal.

In this paper, we suggest different vehicle assignment strategies. The first approach suspends the static binding of carriers to gantry cranes using a dynamic strategy where a predetermined number of carriers perform container transports for several gantry cranes (straddle carrier pooling). Thus, we aim to increase the productivity of the used straddle carriers. Depending on the number of loading and discharging processes (structure of the WTP), the carriers can be used in a *double cycle* mode such that empty runnings are replaced by jobs for other gantry cranes. We consider two different cases of straddle carrier pooling:

- semi-dynamic assignment (a fixed number of straddle carriers is assigned to the gantry cranes of one vessel),
- dynamic assignment (a fixed number of straddle carriers can perform transports for all gantry cranes).

In order to obtain results that can be compared to the static assignment strategy we set the number of straddle carriers to $3 \cdot G$ with $G$ being the number of gantry cranes.

After a straddle carrier finishes its current job, the next job is allocated using the following rule: the jobs within a certain distance (shorter empty runnings) to the current straddle carrier position are examined and the one with the longest delay (or shortest waiting time if there is no one with delays) is assigned. Whenever there is no job available in this area, the job with the longest delay (or shortest waiting time, respectively) on the whole yard is selected. Thus, the delay time should be reduced by trying to keep the empty running time small and the waiting time low to be available for the next job as soon as possible.

This (simple) assignment rule can be used to achieve a straddle carrier pooling. Furthermore, in a second approach we investigate how a simple genetic algorithm can be combined with the assignment strategies described above (static, semi-dynamic, dynamic) to improve the quality of the sequences. In Section 5 experimental results are shown.

## 4 Evolutionary Algorithms

Evolution strategies, genetic algorithms, and evolutionary programming are three different mainstreams in the research field of evolutionary algorithms or evolution programs, respectively (see, e.g., [1, 24]). These approaches have in common that they represent algorithms based on principles from the natural evolution, but they differ in the underlying strategies.

Compared to classical methods, evolutionary algorithms have several properties that make them interesting for certain hard problems. These properties are simple evolutionary operators, simple adaptation to multi objective functions and parallel exploration of the solution space due to the existence of a population. A further advantage of the evolutionary algorithms (see [12]) is the generality for the application to optimization problems. Besides a common interface in the solution representation, the algorithm is independent from the problem specifics and "learns" the structure of the solution space during the optimization process.

Meta-heuristics like simulated annealing and tabu search are seen as keen competition, but Goldberg [12] notes that the evolutionary algorithms may be preferable in complex solution spaces due to their possible adaptation to the problem and the solution space. Evolutionary algorithms allow the efficient discovery of areas near (local) optima but are not well suited for fine-tuning structures which are close to optimality [13]. Therefore, the integration of local search strategies (hybrid methods) to improve individuals for the exploration of these local optima can be advantageous.

Without revealing more characteristics of the evolutionary strategies and evolution programming, we restrict ourselves to genetic algorithms in this paper. This decision is based on the problem (integer optimization problem) and the solution representation (permutation vector).

### 4.1 Basic Structure of Genetic Algorithm

This section describes the basic structure of genetic algorithms developed by [18] (see also [12, 1]). At first, the starting population (collection of individuals) has to be constructed using a preferably high variability for the genes (attributes or variables) of the chromosome (representation of an individual or a solution vector). The requested variability can be achieved by using non-deterministic methods (random generation of individuals, gradual random variations of a given individual) or, in cases where special knowledge of the problem is available, the usage of starting heuristics (e.g., cheapest insertion, next best for sequencing problems), and the random variation of these solutions.

Several operators are used to evolve the population (i.e. based on the fitness of the individuals measured by the objective function). These operators are selection, recombination, and mutation.

**Selection** The selection of individuals for the next population is based on the fitness using either a deterministic or random (to achieve a larger variability) selection. Individuals with a better fitness have a higher probability to be chosen for the next population or to produce an offspring. Various mechanisms may be used, like fitness-proportional selection, rank-based selection, or competition selection (see e.g, [12, 35, 1]). Further selection mechanisms that train the individuals using a local search strategy can be found in [2, 29].The selection can be seen as the intensification within the search space.

**Recombination** Recombination is an operator to associate the attributes of two or more individuals to combine them to one or more new individuals. The recombination facilitates the diversification of evolution by creating new individuals while keeping promising fragments of the chromosome. The most known recombination operator is the crossover, based on its analogy in biology. Two or more individuals are chosen with a certain probability to participate in a crossover. Randomly, one or more locations (crossover points) are selected where the individuals are split. These locations are the same on every individual. The resulting fragments are exchanged between the individuals and recombined to new ones (see e.g., [1, 12]). This universal operator is not problem specific and, therefore, the recombination can produce infeasible individuals (lethal mutation). A problem specific repair operator can be applied to the individuals emerging from the crossover to regain feasibility.

**Mutation** The variability of the population is increased by a mutation (variation) of the individuals that occurs with a certain probability. The variation causes a change of genes in the chromosome (e.g., swapping of genes, deletion of genes). If the individual is infeasible after the mutation, a repair process has to be applied to restore feasibility. The mutation is a mechanism to provide diversification but with less influence than the crossover.

## 4.2 Application

Each iteration of the genetic algorithm is a transition from one population to the next where the following operators are gradually applied: recombination, mutation, and selection. The recombination operator creates a new population containing the offspring of the parent population. After the mutation operator is applied to the new population, the selection operator creates the evolved population by selecting individuals from the offspring or parent population depending on the replacement strategy. Our implementation of the genetic algorithm, and therewith the operators as well, uses the framework HOTFRAME by [9].

**Representation of Individuals** The container movements (jobs) on the terminal are coded in a permutation vector (individual) with a job being a gene. The size of the population is a parameter of the algorithm and depends on various factors (e.g., the structure of the problem) and cannot be determined beforehand in general.

The objective function (6) (see Section 3) is used for the fitness of the individuals. An individual $s = (1, \ldots, n)$ is a permutation vector containing the jobs $s_i$ with $n$ being the number of jobs at all gantry cranes (see Figure 5). The solution vector implies the number of straddle carriers as well as the sequences for each of them, $c_k$ with $k \in \{1, \ldots, C\}$. The pseudo-code in Figure 4 describes the calculation of the objective function value for a shift of length $T$.

```
s                 (containing a permutation of the solution)
i ← 1             (order in the solution sequence)
T ← 14400         (length of a 4 hour shift in seconds)
C ← 1             (number of straddle carrier)
T_C ← 0           (time used by the straddle carrier N_c)
D_C ← 0           (delay time of the straddle carrier N_c)
W_C ← 0           (waiting time of the straddle carrier N_c)
while i ≤ n do    (Go through the solution vector)
  (Duration, Delay, Wait) = getTime (s_i)
  T_C ← T_C + Duration
  D_C ← D_C + Delay
  W_C ← W_C + Wait
  if (T_C ≥ T) ∨ (s_i ≤ s_{i-1} executed by the same straddle carrier) then
    C ← C + 1
    T_C ← 0; D_C ← 0; W_C ← 0
  else
    i ← i + 1
  endif
end
```

**Figure 4. Objective function calculation.**

The function getTime $(s_i)$ calculates the time that a straddle carrier needs to fulfill the job $s_i$. It is composed of the time to reach the starting position (pick up), the time for the transport of the container, and the waiting time that can occur if the straddle carrier arrives before the expected time. Note, that the number of straddle carriers is increased whenever the sequence of jobs for the straddle carrier exceeds $T$ or a job $s_i$ is scheduled before its predecessor $s_{i-1}$.

**Recombination** In this paper, a one-point crossover operator is investigated. A repair function guarantees that new individuals are feasible, i.e., the uniqueness of a job (gene) in the solution is kept. Examples for repair operators on a permutation vector are discussed in, e.g., [31, 8].

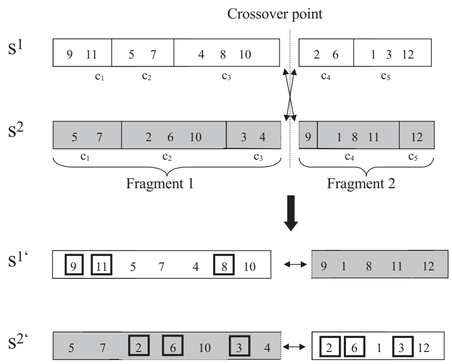


**Figure 5. Crossover**

As described in Section 3, the solution vector is divided into $C$ segments of which each represents a certain number of jobs to fill a shift of $T$ seconds for one straddle carrier. The location $l$ as the crossover point within one of the two individuals, $s^1$ and $s^2$, is selected randomly from a uniform distribution of the $C-1$ possibilities in a way that the sequences are not destroyed. The recombination is done as shown in Figure 5. Note, that the crossover is allowed to split segments in the second individual $s^2$. The duplicate jobs in the offspring can be replaced either in the fragment $f_1 = \{s_1^1, s_2^1, \ldots, s_{l-1}^1\}$ or in the fragment $f_2 = \{s_l^1, s_{l+1}^1, \ldots, s_n^1\}$ by jobs that are missing in this individual. The repair mechanism selects one of the fragments with a probability of 50%. The idea behind this implementation of the crossover is the heredity of good sequences for straddle carriers to the next population. Figure 6 shows the pseudo-code for the case where the fragment $f_2$ is kept constant and exchanges in fragment $f_1$ are done. The results after the application of the repair function are $s^{1*} = \{\mathbf{2}, \mathbf{3}, 5, 7\|4, \mathbf{6}, 10\|9\|1, 8, 11\|12\}$ and $s^{2*} = \{5, 7\|\mathbf{8}, \mathbf{9}, 10, \mathbf{11}\|4\|2, 6\|1, 3, 12\}$ with the replaced jobs marked boldfaced.

$s^{1*}$ *(individual containing the two fragments $f_1$ and $f_2$)*
$i \leftarrow 1$ *(iterator to traverse $s^{1'}$)*
**while** $i \leq l$ **do**
  **if** $s_i^{1*} \in f_2$ **then**
    $s_i^{1*} \leftarrow \mathrm{argmin}_{x \in [1,\ldots,n]}[(x \notin f_1) \wedge (x \notin f_2)]$
  **endif**
  $i \leftarrow i + 1$
**end**

**Figure 6. Repair function**

**Mutation** For each individual in the offspring population exists the possibility to experience a mutation of its solution vector $s$. With a probability of $p_m = \frac{1}{n}$, the contents of two randomly selected locations $k$ and $l$ with $k, l \in [1, \ldots, n]$ in $s$ are exchanged such that $s_k \leftarrow s_l$ and $s_l \leftarrow s_k$. The mutated individual replaces the original offspring.

**Selection** The creation of the next population is based on the selection of individuals from the current population and the replacement strategy. The framework HOTFRAME supports several strategies but preliminary experiments showed that the *ranking selection* in combination with *last replacement* results in the best overall solutions.

Ranking selection uses the absolute fitness values to compute a rank for each individual in the population. This rank determines the probability to be drawn as a source for the recombination operator to create new offspring for the next population. Furthermore, a given pressure value specifies the closeness of the probabilities, i.e., a low pressure would use the same probability for all individuals independent of their fitness. With an increasing pressure the chance to select an individual with a bad fitness value is decreasing.

Last replacement generates the next population by refreshing the worst parents with a specified number of the best offsprings if and only if these individuals (offspring) have a better fitness than the worst parents. The number of individuals determines the speed of improving the population and the ratio of convergence.

## 5 Experiments and Computational Results

This paper aspires to demonstrate the changes that are related to the application of the semi-dynamic and dynamic straddle carrier assignment strategies compared to the static strategy with a fixed assignment of straddle carrier to gantry cranes. Furthermore, the potential of a genetic algorithm for the considered allocation problem is illustrated. In this section we present numerical results which should be regarded as preliminary due to the fact that we ignore stochastic influences in our experiments. A final evaluation has to be done in conjunction with a simulation model which is part of ongoing research.

Based on the operations at container terminals realistic instances have been generated that are based on the terminal layout and that are close to the real-world transshipment processes. Figure 7 shows the location of four container vessels we use for our experiments. Each vessel is served by two gantry cranes that are either loading or discharging the predetermined job sequences. For each of these sequences, 80% of the jobs have their birth- or death-location within the large marked area and 20% in an area close to the corresponding gantry cranes. The jobs are randomly generated

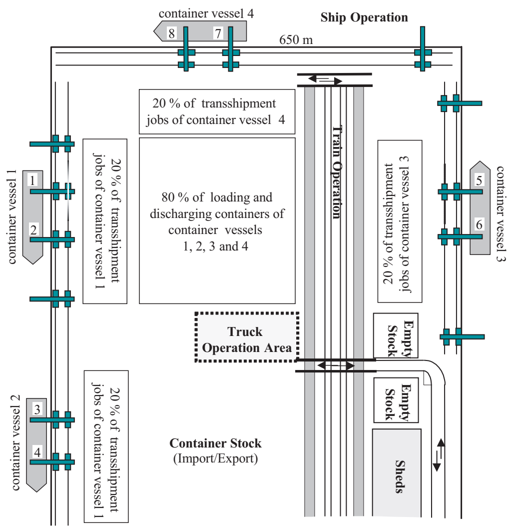


**Figure 7. Extract of the container terminal showing the berth of four container vessels.**

assuming a uniform distribution. Therefore, our experimental results are the average of ten runs using different seed values for the generation of the sequences.

We assume an average transshipment time for the gantry cranes of $\overline{t_g} = 90$ seconds, $g \in \{1, \ldots, G\}$ which results from preliminary experiments. In cases where we assumed a time over 90 seconds, the jobs had no delays during the shift such that the gantry cranes work with maximum productivity. The number of containers (or jobs) that have to be transported during a shift is set to $n = 125$, thus the workshift is of the length $T = \overline{t_g} \cdot n = 187.5$ minutes.

Different experiments show that a set of six different constellations can be used to show the possibilities for improvements of a strategy change as well as the effects of using genetic algorithms. The following list describes the constellations and their properties (see also Figure 7). We set the number of straddle carriers to three times the number of gantry cranes for all constellations.

**V3** Same assumptions as **V2** but for gantry crane 5 (loading) and gantry crane 6 (discharging).

**V3D** Similar to **V3** but both gantry cranes (5/6) are discharging containers from vessel 3 to analyze whether a semi-dynamic strategy improves the transshipment process in a constellation where both gantry cranes are in the same mode and the return cycle can not be used to transport a container for a different gantry crane.

**V2** The distance between the container vessel 2 and the main part of the container location is assumed to be long such that there is a higher probability for delays, offering a high potential for an optimizing procedure. Gantry crane 3 is loading while gantry crane 4 is discharging containers.

**V23** This constellation is used to analyze the interaction of two vessels between which the straddle carrier can transfer the containers. Here, on both vessels (2 and 3) are working two gantry cranes, one is loading while the other is discharging, and the vessels are located far away from the main part of the container locations.

**V14** Same as **V23** for the vessels 1 and 4 which are located closer to the main part of the container locations such that the transportation process involves less potential for optimization.

**V1234** On each vessel two gantry cranes are working, one is loading while the other is discharging. This constellation demonstrates the change to a full dynamic strategy without any binding of straddle carriers to gantry cranes.

The first experiment is the change of the strategy from static straddle carrier assignment to the semi-dynamic and dynamic strategies without using an optimization method. The results are shown in the first and second line of each case in Table 1. The columns show the time (**Time**) when the transport of the last container for the specific case is finished, the accumulated delay times (**Delay**) for all straddle carriers, the number of critical delays (**# Critical Delays**) that are larger than three times of the average transshipment time of the corresponding gantry crane, the length of all empty runnings (**Empty**) as well as the total distance of all straddle carriers (**Distance**) given in kilometers.

| **Case** | **Strategy** | **Time** (min) | **Delay** min | **# Critical Delays** | **Empty** (km) | **Distance** (km) |
|---|---|---|---|---|---|---|
| **V3** | S | 348.75 | 29743.01 | 248.05 | 183.49 | 190.11 |
| | SD | 304.97 | 17173.82 | 242.80 | 112.84 | 190.18 |
| | SDGA | 260.45 | 11472.87 | 237.60 | 46.51 | 190.18 |
| **V3D** | S | 356.26 | 29721.03 | 245.80 | 202.30 | 175.19 |
| | SD | 353.60 | 24300.21 | 244.00 | 210.29 | 175.31 |
| | SDGA | 337.16 | 23615.67 | 243.00 | 203.23 | 175.31 |
| **V2** | S | 274.43 | 11081.79 | 237.60 | 109.60 | 109.57 |
| | SD | 242.18 | 7816.87 | 235.00 | 74.64 | 109.57 |
| | SDGA | 218.22 | 4299.89 | 219.80 | 37.02 | 109.57 |
| **V23** | S | 348.75 | 40722.91 | 484.05 | 293.98 | 300.52 |
| | SD | 301.64 | 30411.55 | 482.25 | 284.51 | 298.21 |
| | SDGA | 242.98 | 17456.68 | 467.50 | 114.92 | 298.21 |
| **V14** | S | 237.35 | 11480.65 | 466.60 | 166.49 | 163.43 |
| | SD | 228.60 | 10656.42 | 459.60 | 173.61 | 163.43 |
| | SDGA | 197.35 | 3366.70 | 373.40 | 81.26 | 163.43 |
| **V1234** | S | 348.75 | 52305.45 | 952.25 | 459.58 | 463.11 |
| | D | 284.63 | 45507.08 | 944.75 | 562.43 | 462.91 |
| | DGA | 250.06 | 27828.51 | 918.25 | 307.80 | 462.91 |

S=static; SD=semi-dynamic; SDGA=semi-dynamic with GA; D=dynamic; DGA dynamic with GA

**Table 1. Numerical results**

The time that is needed for the whole transshipment process shows a significant reduction by changing the strategy to the pooling of straddle carriers depending on the constellation. There is a correlation between the distance of the vessels and the main container storage area in a way that an increasing distance effects a larger reduction (**V23** with 14% compared to 4% for **V14**). A reduction of only 1% was achieved for constellation **V3D** where all transports are from the gantry cranes to the yard, a reduction of approximately 13% for the constellations **V3** and **V2**, and of 18% for **V1234** with the highest potential of reassigning a straddle carrier. The empty runnings for the constellations **V3** and **V2** are reduced between 30% and 40%. As soon as more than one ship was involved or the gantry cranes were operating in the same mode, the total empty running varied only by up to 4% in both directions. With respect to the fact that delay times increase the risk for congestions (reduction of crane productivity), delay times have to be avoided. In this conjunction, pooling strategies show a better performance. While the number of delays larger than twice the average transshipment time stays nearly the same, the accumulated delay times of all straddle carriers could be reduced between 7% and 42% whereas the largest potential of reduction can be found for the constellations where the distances between the gantry cranes and container positions in the yard is high (**V3**, **V2**, and **V23**). Note that the number of delays can only show a higher risk for a congestion. In that respect, further analysis has to be done using a simulation model.

In consideration of these results, we have to conclude that a (simple) change of the strategies allows a better and more efficient (less empty runnings and total distance) straddle carrier assignment. We use the word “simple” due to the fact that the computational effort to compute the sequences for the straddle carriers is the same as for the static strategy but allows a better utilization of them.

In order to further enhancement we apply a genetic algorithm to find better straddle carrier assignments than in the previous runs. Our first approach, starting from a population with random individuals, was not successful due to general problems guiding the algorithm with penalty costs. Therefore, we generate the starting population by mutating the solutions from the previous approaches. We use 4000 iterations with a population size of 100. These values result from preliminary experiments performed to find good settings. The computation time for the genetic algorithms is ignored in this paper because we are mainly interested in the potential of these algorithms to improve given solutions. An extensive analysis of the behavior (computational time, parameters) has to be done when the experiments are validated in conjunction with the simulation model.

The runs with the static strategy show almost no reductions (i.e. we achieved the best reduction for **V2** with 6.6% and **V14** with 3.1%, but for the other constellations only an improvement below 0.6%) but the application of the genetic algorithm on the semi-dynamic results is promising. The results are shown in the third line of each case in Table 1. The time is reduced by approximately 12% (except **V3D** with only 4% due to the same mode of the gantry cranes). A larger reduction is possible for the empty runnings (with an average of 43% with a maximum of 58% for constellation **V23**), and the delays with an average of 40% and a maximum of 64% (**V14**). Considering the delay times the change of the strategy improved the transshipment process but did not cause a decrease of the number of delays. The genetic algorithm is able to reduce this number up to 18% (**V14**). Therefore, the risk of congestion can be reduced but, as before, has to be verified on a simulation model. This demonstrates explicitly the need for optimization and the benefits using genetic algorithms. Finally, the constellation **V3D** is without any relevant improvements using genetic algorithms. Therefore, one of the first requirements for the transshipment process has to be the usage of different modes for the gantry cranes that work on the same container vessel.

## 6 Conclusions

Reducing the time in port for container vessels has become one of the key elements for operators of the main container terminals worldwide. Following this theme, we have discussed one of the most crucial problems arising within the development of intercontinental container transport, i.e., improving the productivity of the gantry cranes in order to gain a reduction of the time in port.

Preliminary results show that such problems may be treated mainly in two ways. First, a careful reorganization or reengineering of the processes and strategies on the terminal may result in considerable improvements, and second, algorithmic enhancements for certain combinatorial optimization problems may help to support the planer and allow for further improvements. In this paper we have proposed ideas in both ways. First, a dynamic strategy may help to improve the use of gantry cranes and straddle carriers and their interplay when loading and unloading container vessels. That is, we have examined the effects on the berth time derived from the reorganization of the transshipment process. Second, simple evolutionary algorithms may successfully be applied to respective real-world problems. Within the scope of several experiments, we have investigated the potential of a genetic algorithm for a minimization of the container vessel processing time.

The results show a good performance of the used genetic algorithm combined with the reorganization. It has been shown that in some of the considered scenarios reasonable improvements for the gantry crane productivity are obtainable. While major improvements are due to the reorganization even simple genetic algorithms are able to gain additional improvements.

First of all, further research concentrates on developing a simulation model to verify and validate the current results in environments taking into account additional stochastic influences. With respect to the algorithms considered further meta-heuristics including hybrid methods shall be tested.

## References


[1] T. Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, New York, 1996.

[2] J. Baldwin. A new Factor in Evolution. *American Naturalist*, 30:441–451, 536–553, 1896.

[3] J. Boyes, editor. *Containerisation International Yearbook*, chapter Ranking of Containerports of the World. National Magazin Co., London, 1994.

[4] J. Boyes, editor. *Containerisation International Yearbook*, chapter Ranking of Containerports of the World. National Magazin Co., London, 1997.

[5] A. Bruzzone and R. Signorile. Simulation and Genetic Algorithms for Ship Planning and Shipyard Layout. *Simulation*, 71:74–83, 1998.

[6] C. Daganzo. The Crane Scheduling Problem. *Transportation Research B*, 23B:159–175, 1989.

[7] S. Eickemeier. *Kombinierter Ladungsverkehr: Produktionsorientierte Strategiekonzepte für die Deutsche Bahn AG*. Peter Lang, Frankfurt, 1997.

[8] L. Eshelman and J. Schaffer. Crossover's Niche. In S. Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 9–14. Morgan Kaufmann, San Mateo, 1993.

[9] A. Fink and S. Voß. Generic Metaheuristics Application to Industrial Engineering Problems. *Computers & Industrial Engineering*, 1999. To appear.

[10] L. Gambardella, A. Rizzoli, and M. Zaffalon. Simulation and Planning of an Intermodal Container Terminal. *Simulation*, 71:107–116, 1998.

[11] R. Gibson, B. Carpenter, and S. Seeburger. A flexible Port Traffic Planning Model. In *Proceedings of the 1992 Winter Simulation Conference*, pages 1296–1306, 1992.

[12] D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, 1989.

[13] J. Grefenstette. Incorporating Problem Knowledge into Genetic Algorithms. In L. Davis, editor, *Genetic Algorithms and Simulated Annealing*, pages 42–59. Morgan Kaufmann, Los Altos, 1987.

[14] Hamburger Hafen- und Lagerhaus-AG. The 20 Leading Containerports. `http://www.hhla.de/grten.htm`, May 1999. Date indicates last check of address.

[15] Y. Hayuth, M. Pollatschek, and Y. Roll. Building a Port Simulator. *Simulation*, 63:179–189, 1994.

[16] K. v. Hee. Decision Support System for Container Terminal Planning. *European Journal of Operational Research*, 34:262–272, 1988.

[17] L. Hoffarth and S. Voß. Liegeplatzdisposition auf einem Containerterminal – Ansätze zur Entwicklung eines entscheidungsunterstützenden Systems. In H. Dyckhoff, U. Derigs, M. Salomon, and H. Tijms, editors, *Operations Research Proceedings 1993*, pages 89–95, 1994.

[18] J. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.

[19] L. Hulten. *Container Logistics and its Management*. PhD thesis, Chalmers University of Technology: Department of Transportation and Logistics, 1997.

[20] J. Koch. *Die Entwicklung des Kombinierten Verkehrs*. Gabler, Deutscher Universitäts-Verlag, Wiesbaden, 1997.

[21] E. Kozan. Comparison of Analytical and Simulation Planning Models of Seaport Container Terminals. *Transportation Planning and Technology*, 20:235–248, 1997.

[22] E. Kozan and P. Preston. Genetic Algorithms to Schedule Container transfers at Multimodal Terminals. *International Transactions in Operational Research*, 6:311–329, 1999.

[23] Y. Merkuryev, J. Tolujew, E. Blümel, L. Novitsky, E. Ginters, E. Viktorova, G. Merkuryeva, and J. Pronins. A Modelling and Simulation Methodology for Managing the Riga Harbour Container Terminal. *Simulation*, 71:84–95, 1998.

[24] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, Berlin, third edition, 1997.

[25] R. Mosca, P. Giribone, and A. Bruzzone. Simulation and Automatic Parking in a Training System for Container Terminal Yard Management. In *Proceedings of ITEC*, pages 65–72, 1994.

[26] G. Muller. *Intermodal Freight Transportation*. The Eno Transportation Foundation, Lansdowne and the Intermodal Association of North America, Greenbelt, third edition, 1995.

[27] M. Nevins, C. Macal, R. Love, and M. Bragen. Simulation, Animation and Visualization of Seaport Operations. *Simulation*, 71:96–106, 1998.

[28] Noell Stahl- und Maschinenbau GmbH. Crane Construction. `http://www.noell.de/cranes/index.htm`, May 1999. Date indicates last check of address.

[29] M. Norman and P. Moscato. A Competitive and Cooperative Approach to Complex Combinatorial Search. Technical Report C3P Report 790, Caltech Concurrent Computation Program, California Institute of Technology, 1989.

[30] R. Peterkofsky and C. Daganzo. A Branch and Bound Solution Method for the Crane Scheduling Problem. *Transportation Research B*, 24B:159–172, 1990.

[31] I. Rechenberg. *Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Dissertation, Technische Universität Berlin, 1970.

[32] D. Steenken. Latest Operational Experience with Satellite Positioning Systems. `http://www.hhla.de/DC/iir_lec.htm`, May 1999. Date indicates last check of address.

[33] D. Steenken, A. Henning, S. Freigang, and S. Voß. Routing of Straddle Carriers at a Container Terminal with the Special Aspect of Internal Moves. *OR Spektrum*, 15:167–172, 1993.

[34] M. Taleb-Ibrahimi, B. Castilho, and C. Daganzo. Storage Space vs Handling Work in Container Terminals. *Transportation Research B*, 27B:13–32, 1991.

[35] D. Thierens. Selection Schemes, Elitist Recombination, and Selection Intensity. In T. Bäck, editor, *Proceedings of the 7th International Conference on Genetic Algorithms*, pages 152–159. Morgan Kaufmann, San Mateo, 1997.

[36] G. Thiers and G. Janssens. A Port Simulation Model as a Permanent Decision Instrument. *Simulation*, 71:117–125, 1998.

[37] J. Tolujev, Y. Merkuryev, E. Blümel, and M. Nikitins. Port Terminal Simulation: State of the Art - A Survey. Technical report, Riga: AMCAIRiga Technical University, 1996.