8-2009

Evaluating Straddle Carrier Deployment Policies: A Simulation Study for the Piraeus Container Terminal

Eleni HADJICONSTANTINOU
Imperial College London

Nang Laik MA
Singapore Management University, nlma@smu.edu.sg

DOI: https://doi.org/10.1080/03088830903056991

Citation

HADJICONSTANTINOU, Eleni and MA, Nang Laik. Evaluating Straddle Carrier Deployment Policies: A Simulation Study for the Piraeus Container Terminal. (2009). *Maritime Policy and Management*. 36, (4), 353-366. Research Collection School Of Information Systems.
Available at: https://ink.library.smu.edu.sg/sis_research/760

Evaluating straddle carrier deployment policies: a simulation study for the Piraeus container terminal

ELENI HADJICONSTANTINOU*† and NANG LAIK MA‡

†Imperial College Business School, Imperial College London, London, UK

‡School of Information Systems, Singapore Management University, Singapore

Most container terminals in the world today are operating up to their capacities. In this paper, we have developed a decision support system to optimise yard operations by considering all container flows (import, export and transshipment) through the yard with the view to improving terminal performance and efficiency. In another paper, we proposed an optimization model that determines optimal container locations and straddle carrier (SC) movements with the objective of minimizing the overall storage and handling cost of containers. In this paper, a discrete event simulation tool for container terminal operations has been developed with three objectives: (i) to validate the operational plan resulting from the optimization model; (ii) to test the robustness of possible deployment policies for straddle carriers; and (iii) to analyse yard resource requirements for future terminal expansion. The model has been applied to the largest container terminal in Greece—the Port of Piraeus—and computational results are reported for the case study.

1. Introduction

Container terminal operations play a vital role in today's world economy. Some of the busiest terminals in the world today handle millions of twenty footer equivalent units (TEUs) of container throughput annually. Terminal operations can be broadly classified into three types: gate-, yard- and quay-side. As most of the container flows occur in the yard-side, yard operations are the bottleneck for the container terminal operations. The core functionality of the yard operations is *container assignment* which involves the assignment of locations to all the containers coming to the terminal. There are three kinds of containers flowing through the terminal:

- *Export (EX)* containers come to the terminal via the gate from the customers. Customers need to book in advance to deliver their EX containers, which normally can be stored in the container terminal free-of-charge for three to seven days before they will be loaded onto the vessel for the next destination. In this case, it is important for the yard planner to assign the containers evenly to different yard blocks so as to facilitate the loading process.

---

*To whom correspondence should be addressed. e-mail: e.hconstantinou@imperial.ac.uk

- *Import (IM)* containers are shipped by the vessel to the port where after being discharged from the vessel, they are either stored in the container storage yard for a period of time until they are collected by the external trucks (XTs) or directly picked up by the customers without storage. In the latter case, customers need to make advance booking arrangements with the terminal operators.
- *Transhipment (TS)* containers are discharged from the ship, transported by internal trucks (ITs) and stored in the yard before they are transshipped to another vessel.

Containers in the yard are stored according to a storage policy. The assignment of IM, EX and TS containers is a routine, daily operation decision problem for the yard planners. In today's market, finding 'good' locations for the containers so as to shorten the vessels' port stay and satisfy the service demand using minimal yard resources is enormously important in remaining competitive.

Straddle carriers (SCs) are used for both transporting and stacking containers in the yard because they are very flexible. They are usually man-driven and able to stack three or four levels high. Clearly, if congestion occurs at the yard-side then delays in vessel departure are possible as the trucks are unlikely to be at the quay-side in time to transport all the containers for loading. This means that, during the planning process, yard planners are required to take into consideration the availability of SC capacity to ensure that these machines are neither under-utilized nor overloaded. It is also important to estimate the optimal number of SCs needed to service all the containers in the yard during a planning time horizon. A tradeoff involves reducing the operational cost of SCs while at the same time trying to meet the required service level. The *container assignment* and *SC deployment* problems are closely inter-related and the quality of solution to both problems is fundamental to improve the efficiency of the overall terminal operation. This is a challenging and complex problem for which, due to large volume of container throughputs, the use of personal experience and basic rule of thumb cannot provide the optimal solution. On the other hand, the use of a decision support tool may considerably improve the decision-making process of yard planners in daily operations and hence contribute towards the competitiveness of the container terminal.

A detailed review of the literature—see [1–4]—suggests that there is very little research done towards addressing the complete container terminal operations problem in a unified way and container assignment problems have been explored by only a few researchers [e.g. 5–7]. The authors of [8, 9] examined the 'unproductive moves' in the container yard using empirical studies and the resulting impact of these moves on operations. Henesey [10] developed a multi-agent simulation model named SIMPORT to test different operational policies for a major container port in India. In this paper, we present a high-level decision support system to assist terminal managers in making informed decisions relating to the core yard operations problem of *container assignment* and *SC deployment* based on the use of a purpose-built optimization model [11].

The rest of the paper is organized as follows. The optimization model is briefly described in Section 2, summarizing key decision variables, constraints, and objective and solution methodology. Two SC deployment policies are examined providing both a lower and an upper bound on the number of SCs required in the terminal within a given planning time horizon. In Section 3, we propose a simulation model for container terminal operations which is used to validate/verify the operational plans resulting from the optimization model. Section 4 presents an application of the simulation model to the largest port in Greece, the Port of Piraeus; the process of generating input data for the model is discussed and preliminary computational results are reported. Finally, conclusions are given in Section 5.

2. An overview of the optimization model

This section provides an overview of an optimization model used to solve the problem of *container assignment* and *SC deployment* in a container terminal.

2.1. *Decision variables*

IM flow of containers: Vessel to customer

- # IM containers to be discharged from the vessel per time period
- # IM containers to be stored in a yard block
- # IM containers to be loaded directly to the XTs
- # SCs required during discharging of IM containers
- # SCs required to service the XTs during pick up of the IM containers.

EX flow of containers: Customer to vessel

- # EX containers to be loaded to the vessel per time period
- # EX containers to be stored in a yard block
- # EX containers to be loaded directly to the vessel (without storage) given that these containers arrived at the gate after the vessel has arrived
- # SCs required in the yard to unload the EX containers from the XTs
- # SCs required during loading of EX containers.

TS flow of containers: Vessel to vessel

- # TS containers to be discharged from the vessel per time period
- # TS containers to be stored in a yard block
- # TS containers to be loaded directly onto the loading vessel given that the latter has arrived before the discharging vessel
- # TS containers to be loaded to the vessel per time period
- # SCs required for the unloading of the TS containers
- # SCs required for the loading of the TS containers.

Mathematical model. We have formulated the container assignment and SC deployment problem described in this paper as an integer programming (IP) model. The objective of the model is to minimize the total operational cost (handling—SC deployment and storage) associated with all container flows (IM, EX, TS) over a given planning time horizon, while at the same time balancing the distribution of workload among different yard blocks. The model output has to satisfy a number of constraints, such as, supply and demand of containers between various Origin–Destination (O–D) pairs, SC working capacity, space vacancy and flow conservation at each yard block.

The above model has been implemented in C++ and a solution algorithm has been developed using the commercial code CPLEX 10.1. For details regarding the mathematical formulation and its computational implementation, see [11].

Straddle carrier deployment policies. Due to various types of uncertainties occurring in daily terminal operations, it would be very difficult for the yard manager to determine the exact number of SCs required during the actual loading/unloading of containers (especially for those containers coming via the gate). Although customers book in advance for the delivery/pick up of their containers, trucks can actually arrive at the gate anytime within a four-hour time window. In order to better plan SC requirements in the terminal yard, the model developed in this paper considers two SC deployment policies which enable the computation of both a lower and an upper bound on the actual number of SCs required during operations over a given planning horizon.

Policy I assumes that there is at most one SC deployed in a yard block during a working shift (eight hours) which is not shared with other blocks. This, referred to as the ‘‘No sharing of SCs’’ policy, is used to estimate the maximum number (upper bound) of SCs required servicing all the containers in the yard within the planning time horizon.

Policy II allows the ‘‘Sharing of SCs’’ and assumes that the SC can move around different yard blocks without restriction. For example, it would be possible to move an idle SC, originally assigned to yard block *j1*, to another block, say *j2*, where there are jobs waiting for service assuming that (i) not all SC capacity has been utilized and (ii) moving the SC among yard blocks is not prohibited due to space limit or one directional road. If any of these restrictions arise during the actual operations, we may require more SCs to service all the containers in the yard within the planning horizon. Hence, the deployment of this policy results in the computation of the minimum number (lower bound) of SCs actually required.

In actual operations, either of the two SC policies described above can be deployed depending on the workload in the yard. If a yard block is busy, then the SC will be serving all the containers in the same yard block throughout the planning period. It is worth noting that the SCs are bulky equipment which can not move around the yard as freely as it would be desirable. Therefore, even if the ‘‘Sharing of SCs’’ policy is adopted, the SCs can only move once or twice among different yard blocks within a shift.

For a practical application of the optimization model, we refer the reader to [12].

3. Container terminal simulation

Computer simulation is used when no closed form analytical solution for the system of interest is possible; it is very popular in the business world today. We can find vast applications in engineering systems, logistics and transportation, manufacturing, military operations, business processes, etc. In an engineering system, simulation may be useful in simulating the turbulent flow of liquids, investigating the effect of earthquake on building structures or in the construction of a water dam. In logistics and transportation, simulation may be useful in decision making to schedule and route vehicles, evaluate new dispatching strategies to improve system performance or determine the number of vehicles to deploy based on forecasted demand. In business, we find applications of simulation in evaluating the impact of additional workload on container terminal operations, designing of a new airport or runways or evaluating the waiting time in queues at call centers. For more applications of simulation, please refer to [13–16].

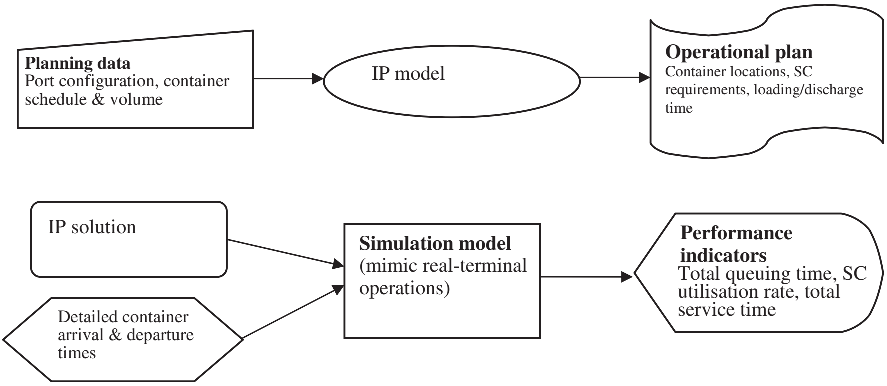

Figure 1. Relationship of optimization (IP) and simulation models.

Figure 1 illustrates graphically how the optimization model described in the previous section relates to the simulation model presented in this paper. In the latter, we are able to model all the actual activities taking place in a terminal, namely, the dynamic arrival of containers, queuing at the gate, yard or quay, travelling of trucks and routing of SCs, whereas, due to their complexity, the details of all these operations could not be represented in the IP model. The output from the optimization model (which determines optimal container locations and number of SCs deployed) is used as input to run the simulation model with a view to evaluating the operational performance of the resulting plan. Usually, a container terminal has a Service Level Agreement (SLA) with shipping lines and trucking companies so that vessels depart on-time and trucks coming to the terminal can leave the yard within 30 minutes. Using the simulation model, we can measure the terminal efficiency and performance using various indicators, such as, the average total time that a truck spends in the container terminal, average queuing time of the truck at the gate, yard or quay-side or average SC utilization rate.

3.1. *Conceptual model of container terminal operations*

We model the key terminal processes in detail, namely: (a) gate/vessel arrival process; (b) queues at gate/yard/quay; (c) container-location assignment; (d) travelling of trucks/SCs; and (e) assignment of jobs to SCs. It is assumed that scheduling of quay cranes (QCs) is already done and QCs are available for the loading and unloading of containers at a constant rate. Containers which are discharged from the ship (IM/TS) to store in the yard and EX containers coming to the terminal via the gate are categorized as *incoming* assigned type (to the terminals). On the other hand, (EX/TS) containers to be loaded on the ship and IM containers leaving the terminal are classified as *outgoing*. Table 1 summarizes the sequence of events for the three types of container flows (IM/EX/TS) and assigned types (*incoming*/*outgoing*).

3.2. *Development of container terminal simulator*

In this section we provide an outline of the main components that are essential to building the container terminal simulation model.

(a) *Simulation clock*
One of the most important building components is the simulation clock which gives the current value of the simulated time as opposed to the

Table 1. Sequence of events for different container-assigned types.

Container and assigned type.	Sequence of events
EX(incoming)/IM (outgoing)	1. XT arrives/queues at gate 2. XT leaves gate for yard (travelling time) 3. XT arrives/queues at yard 4. SC serves 5. XT leaves yard and exits
IM/TS (incoming)	1. IT arrives/queues at quay 2. QC serves 3. IT leaves quay for yard (travelling time) 4. IT arrives/queues at yard 5. SC serves 6. IT leaves yard and exits
TS/EX (outgoing)	1. IT arrives/queues at yard 2. SC serves 3. IT leaves yard for quay (travelling time) 4. IT arrives/queues at quay 5. QC serves 6. IT leaves quay and exits

real time. There is no relationship between the simulated time, real time and running time of the simulation. In our simulation model, the time unit is in minutes (min).

(b) *Events list*
A discrete-event simulation models a sequence of events for each entity in the system; we need to maintain the events list to keep track of the system behaviour. An event consists of the event time (when it happens), the event performance (what happens when an event takes place) and a possible end time (when it finishes). All event sequences are stored in the event−s list and are always sorted in ascending order of event start time. Since during the course of simulation, all new events are inserted into the events list, it is important to maintain a manageable size of this list otherwise it will grow exponentially. In our simulation program, we have deleted all the past events from the events list to improve the performance of it. Thus the events list only includes those events which are to take place in the future (*future events list*). It is important to keep in mind that before we delete an event, we need to store the event information in the system for statistical purposes.

(c) *Random number generator*
The simulation program needs to generate random variables to capture the stochastic nature of the real system. This can be achieved by using *pseudorandom number generators (PRNG)*. We use the Linear Congruential method, proposed by Lehmer [17] to produce a long sequence of integers between 0 and $m-1$ depending on the initial see X_0; the resulting random numbers then follow the recursive relationship $X_{i+1} = (aX_i + c) \bmod m$, $i = 0, 1, 2, \ldots$ where a is a constant multiplier, c is the increment value and m is the modulus.

Using the random numbers, we can generate different arrival patterns which may follow an input distribution like Poisson process with inter-arrival

Table 2. Description of performance indicators from simulation model.

Performance indicator	Type of container	Formula
Avg. total time spent in the system	EX (Incoming)/IM (outgoing)	Exit time (yard)—arrival time (gate)
	IM/TS (incoming)	Exit time (yard)—arrival time (quay)
	EX/TS (outgoing)	Exit time (quay)—arrival time (yard)
Avg.queuing time (gate)		Exit time (gate)—arrival time (gate)
Avg.queuing time (yard)		Time when served by SC—arrival time (yard)
Avg. queuing time (quay)		Time when served by QC—arrival time (quay)
SC utilization		Total job done/SC capacity

times $1/\lambda$, an exponential service time for SCs and time wasted on the road due to traffic congestion.

(d) *Initial and terminating conditions*
The initial condition of our simulation program is to start at time 0. Prior to initiating the simulation process, a list of random arrival times (in minutes) is generated for each container (at gate or quay) based on given data relating to vessel arrival times and customers' booking times. The stopping criteria of the simulation depend on the simulation designer or users; it can stop after certain time or when the system has reached a steady state. In our simulation program, the simulation stops when no more events exist in the *future events list*.

(e) *Collection of statistics*
The simulation program keeps track of all the events which have occurred during the simulation process; in the end we keep record of the interesting results, such as, the total time or average time spent by each truck in the terminal, average queuing time at the gate, yard or quay side, SC utilisation rate. Table 2 explains the details of each performance indicator.

We use the object-oriented (OO) paradigm to develop our port simulation program. C# is one of the most popular OO programming languages which can be used to model many real-world applications, as opposed to the customised simulation packages which are quite rigid. We also use Microsoft Access Database as a storage media to store the data related to simulation events. At the start of the simulation, the simulation clock is set to 0 but it will then advance to the first event time in the list. The execution of the current event will trigger the generation of a subsequent event depending on the current event and the assigned type of the given container. As soon as a new event is inserted into the event list, the latter is sorted again by ascending order of time. The current event which has been already processed is deleted from the event list and the simulation clock will then advance to the next event in the list. The same process is repeated until the future event list becomes empty and all events are processed. It is worth noting that queues may occur at the gate, yard and the quay which are being handled using the first-in-first-out (FIFO) service principle. Once the first element in the queue is processed, it will be deleted from the queue and the next available element in the queue is served. Figure 2 shows the main structure of the simulation model.

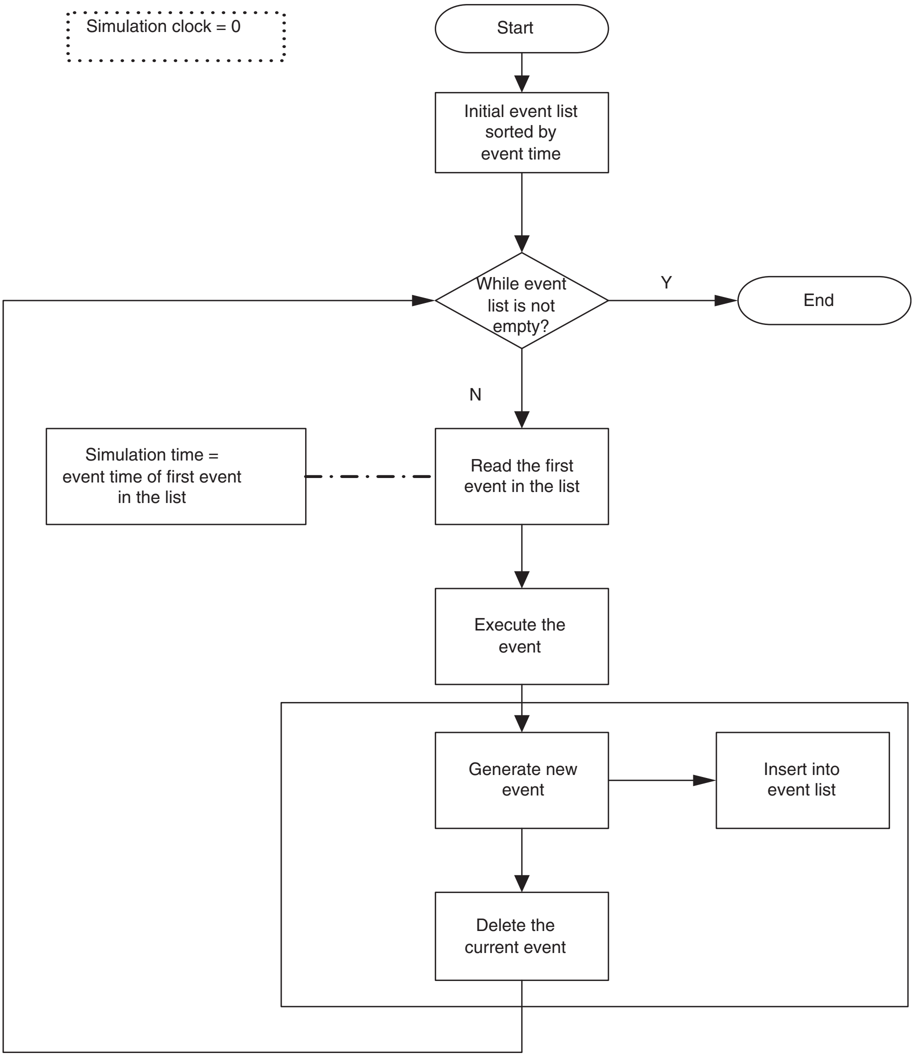

Figure 2. Flow chart for container terminal simulator.

In the next section, we present a practical application of the simulation model proposed in this paper in the context of planning container terminal operations at the Port of Piraeus. Computational results are reported.

4. Application of the simulation model—Port of Piraeus

Port of Piraeus is a major container port in Eastern Mediterranean, playing a leading role as a transhipment centre with an annual throughput of 1.4 million TEUs.

4.1. *Input data*

The input planning requirements to run the model for the Port of Piraeus were mainly generated using publicly available data collected from the relevant website

Table 3. Port configuration data.

User-defined input parameters	Value
Planning time horizon	24 hours
% yard blocks reserved for IM containers	20%
% yard blocks reserved for EX containers	20%
% yard blocks reserved for TS containers	60%
Number of yard blocks	65
Number of slots per block	20
Number of rows per slot	7
Stacking height	4
SC working rate (containers/hour)	10
Total number of vessels calling at the port	5
Total number of containers (all types) to be loaded/discharged on/from the vessels	2682
Total number of (IM/EX)containers transported via the gate	1956
Total number of container moves in the yard	4638
Number of O–D pairs for container flows (all types)	50
Speed of truck (km/h)	15 m/h
Speed of SC (km/h)	10 m/h
Service time of SC	Exponential mean 6 minutes
Average service time at gate	Average 2 minutes
Average service time of QC	Average 2 minutes

and the personal experience of the authors. Table 3 summarizes the yard configuration data used by the model. The proportions of IM, EX and TS containers for the given port are 2:2:6 and the same ratios are used to generate container types carried by a vessel. For example, a vessel carrying a total of 1000 containers for the Port of Piraeus will discharge 200 IM and 600 TS containers; on the other hand, 200 EX containers will be loaded on a vessel before departure.

During a 24-hour planning horizon, five vessels on average are expected to call at Piraeus carrying a total of 2682 IM and TS containers, whereas within the same period, a total of 1956 IM and EX containers are transported via the gate resulting in a total of 4638 container moves; the latter figure is equivalent to an annual throughput of 1.7 million TEUs. Using the information described above, we randomly generated 50 O–D pairs of containers such that the source containers arrive at the terminal before the departure/pick up time at destination.

4.2. *Optimal operational plan (results from optimization model)*

This section reports the preliminary findings of applying the optimization model to the Port of Piraeus container terminal. Consider a typical 24-hour planning horizon which is further divided into three working shifts: t1 (between times 1–8 hours), t2 (9–16 hours) and t3 (17–24 hours). Using data over the 24-hour time period, as described in Section 4.1, and given the set of container flows between all generated O–D pairs, we run the optimization model (section 0). CPLEX finds the optimal solution within seconds: optimal yard locations for all types of containers flowing through the yard and the minimum and maximum number of SCs that will be required during each shift of the 24-hours planning horizon, using the two SC deployment policies described in Section 0. Table 4 shows that, within the same planning period, a total of 24 more SCs are required under the ‘‘No sharing of SCs’’

Table 4. Optimal SC requirements over a 24-hour period.

Optimal solution	CPLEX solver		# of SCs required per time period			
SC deployment policy	Status	Run time (seconds)	Shift 1(1–8)	Shift 2(9–16)	Shift 3(17–24)	Total (1–24)
"No sharing" policy	Optimal	15 sec	26	28	39	84
"Sharing" policy	Optimal	9 sec	18	21	21	60

Table 5. Simulation output using Policies I & II.

Simulation output (minutes)	Policy I: "No sharing of SC"	Policy II: "Sharing of SCs"
Avg. total time spent in system	21.85	58.86
Avg. SC utilization rate	69.0%	96.63%
Avg. waiting time at gate	0.39	0.68
Avg. waiting time at yard	10.3	46.0
Avg. waiting time at quay	0.0	0.08

policy compared with the "Sharing of SCs" policy; in fact, the number of SCs proposed by our model falls within the range of (18, 39) during any one shift. It is interesting to note that the Port of Piraeus posses about 64 SCs and, in actual terminal operations, either of the two SCs policies described above can be deployed depending on the workload in the yard. If a yard block is busy, then the SC will be serving all the containers in the same yard block throughout the planning period. It is worth noting that the SCs are bulky equipment which can not move around the yard as freely as it would be desirable. Therefore, even if the "*sharing of SCs policy*" is adopted, the SCs can only move once or twice among different yard blocks within a shift.

4.3. *Performance indicators (results from simulation model)*

It is well-known that, the use of analytical or optimization models to capture the complexity and dynamic nature of real operations has obvious limitations. We used the container terminal simulator described in Section 0 in order to validate the "goodness" of the operational plan produced by the optimization model described above. Using the port configuration data (Table 3), the optimal number of SCs (Table 4) and the optimal container locations determined by the optimization model (Section 2), we run the simulation model in order to computationally evaluate SC deployment policies I and II. As it can be seen from the results shown in Table 5, the average SC utilization rates for the two policies are 69% and 96.62%, respectively. The SCs deployed under the "Sharing of SCs" policy are almost performing up to their capacity and the average waiting time in the system is 46 minutes; in fact, this is not a desirable SC performance during operations.

Table 6. **Simulation output using Policy III.**

Simulation output (minutes)	Policy III: "Sharing of SC among two blocks"
Avg. total time spent in system	26.77
Avg. SC utilization rate	82.82%
Avg. waiting time at gate	0.68
Avg. waiting time at yard	10.3
Avg. waiting time at quay	0.02
Total SCs required	70

From the simulation results, it is clear that, using Policy II for SC deployment, a lower system performance was achieved as the SC has wasted significant time travelling among different yard blocks without carrying any containers. This suggests that it is worth considering an alternative SC deployment policy whereby the SC movement is limited to two yard blocks only so that empty travelling time is minimized. We refer to this policy as "Sharing of SCs among two blocks" – Policy III.

Under Policy III, if the number of jobs in a yard block is high, then the SC will stay in the same yard block throughout the shift. However, if there is still some SC capacity left, then the SC can move to another yard block to service the containers. If no SC is available or the capacity is full, then a new SC will have to be deployed. Using simulation, we find that, under Policy III, the total number of SCs required is 70 (that is, 14 SCs less than the corresponding number under Policy I of "No Sharing"). The corresponding values of performance indicators for Policy III are given in Table 6 showing that this is a "good" policy to be used in practice.

In order to collect sample performance indicators mimicking system dynamics at the container terminal, we run 50 iterations. Each independent simulation run uses a different initial random seed and statistical sample is collected. By doing so, we are reasonably confident that, from the collected statistical sample, we produce a reliable and valid assessment of the "goodness" of our operational plan. These guarantees are useful in an attempt to persuade terminal managers of the robustness and applicability of the proposed optimal policies under different operational scenarios. Let μ be the average total time spent by each truck in the terminal.

Since we have a large sample size, $m = 50$, we can use the Central Limit Theorem and Z-table to compute $100(1-\alpha)$ % confidence interval (CI) for μ:

$$\hat{\mu} \pm z_{\alpha/2} * \hat{s}, \text{ where } \hat{\mu}, \hat{s} \text{ are sample mean and standard deviation,}$$
$$\text{for 95\% CI, } z_{\alpha/2} = 1.96$$

Table 7 gives an illustrative example of the statistical results obtained from the simulation model. It is shown with 95% confidence that, the two policies of "No sharing of SCs" and "Sharing of SCs among two blocks" are feasible to use in practice with an average total time spent by a truck in the system being in the range of (18.11, 24.42) and (26.55, 35.33), respectively. However, the "Sharing of SCs" policy is found to be less useful in the real-operations as the SC has wasted much time in travelling with an average total time spent between 49.06 and 64.73 minutes. This outcome may have a negative impact on customer service level performance, suggesting that this policy may be impractical.

Table 7. Sample of statistics from multiple simulation runs.

Avg. total time spent by truck in the system (minutes)	Policy I	Policy III	Policy II
Sample mean	21.26	30.94	56.90
Sample standard deviation	1.62	2.24	4.0
95% Confidence interval	(18.11, 24.42)	(26.55, 35.33)	(49.06, 64.73)

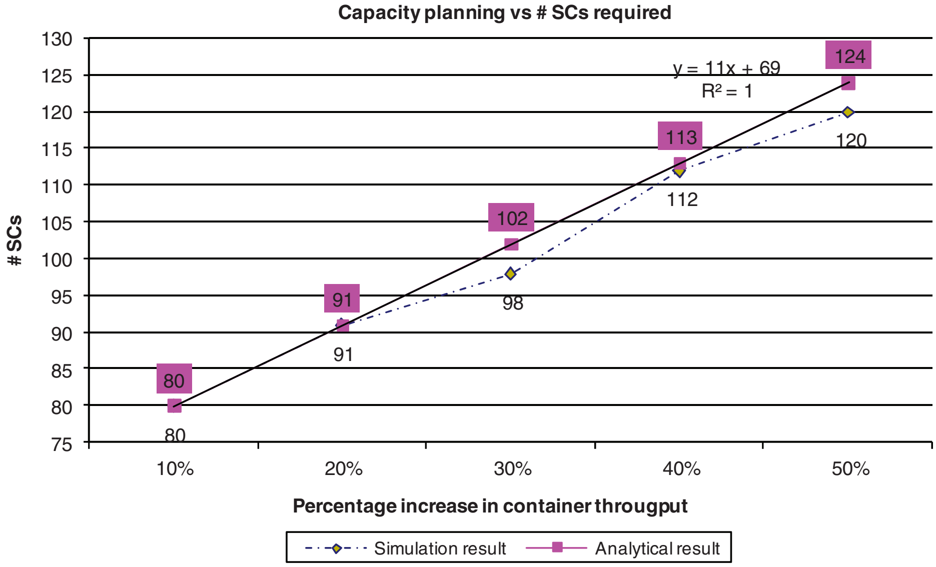

Figure 3. (%) increase in container throughput and number of SCs required.

4.4. *Capacity planning using simulation*

The world container volume has been increasing gradually and it is estimated that an average increase of 10% in annual container throughput is expected for most of the container terminals in the world. Capacity planning is normally carried out at strategic level; to this end, we have used the proposed simulation model to predict the number of SCs required during terminal operations based on forecasted future demand. Anticipating an increase of 10%, 20%, 30% and 50% in the volume of throughput, we run the simulation mimicking the actual terminal operations in the risk-free environment. We have implemented the "Sharing of SCs among two blocks" deployment policy for this study. The simulation model is used to analyse and predict the system behaviour and results are reported in Figure 3 and Table 8. From the diagram and using statistical analysis, we can conclude that there is a positive correlation (i.e. correlation coefficient of 0.996) between volume increase and number of required SCs. Using the simulated data available, we are interested to find an analytical formula that can express the relationship between container volume and SCs requirements. Let x and y denote the percentage increase (%) in volume and the number of SCs required, respectively. From Figure 3, we assume that x and y have a linear relationship $y = mx + b$ where m is the gradient and

Table 8. System performance for capacity planning (in minutes).

Volume increase (%)	# SCs required	Total Avg. time	Queuing time at gate	Queuing time at quay	Queuing time at yard	SC utilisation (%)
10%	80	28.9	0.67	0.001	17.4	79.60
20%	91	30.33	0.65	0	18.84	76.50
30%	98	29.87	0.65	0.001	18.45	76.90
40%	112	29.56	0.64	0	18.67	72.50
50%	120	31.85	0.62	0.005	20.54	72.50

b is the y-intercept. We enter the data and plot the graph in Excel, we obtain a relationship given by: $y = 11x + 69$. Using this formula, we can then easily compute the number of SCs required analytically for a given percentage increase in volume. We also find that the average relative error (%) between the simulated and analytical result is less than 2% providing us with a ‘‘good’’ estimate of what is thought to be a complex and sophisticated process.

5. Conclusions

A detailed literature review of container terminal operations has revealed that most of the researchers in the field only focus on a particular area of the decision-making process resulting in the lack of an integrated approach to improve terminal performance.

In this paper, we have proposed a simulation model for complete terminal operations; the application of this model in the practical context of real operations at the Port of Piraeus container terminal has demonstrated that it can be a valuable tool for yard managers in making informed decisions and better utilizing expensive terminal resources such as SCs. The simulation model was used to validate a yard operational plan of assigning containers and deploying SCs resulting from an optimization model. Furthermore, the ‘‘robustness of the plan’’ was evaluating using simulation and the performance of alternative SC deployment policies was evaluated.

References

1. Ma, N. L. and Hadjiconstantinou, E., 2008, Decision problems in container yard operations: a review. Imperial College Business School, Imperial College, London, Working paper.
2. Stahlbock, R. and Voß, S., 2008, Operation research at container terminals – a literature update. *OR Spectrum*, **30**(1), 1–52.
3. Steenken, D., Voß, S. and Stahlbock, R., 2004, Container terminal operation and operations research – a classification and literature review. *OR Spectrum*, **26**(1), 5–49.
4. Vis, I. F. A. and de Koster, R., 2003, Transshipment of containers at a container terminal: an overview. *European Journal of Operational Research*, **147**(1), 1–16.
5. Kim, K. H. and Park, K. T., 2003, A note on a dynamic space-allocation method for outbound containers. *European Journal of Operational Research*, **148**(1), 92–101.
6. Kim, K. H., Park, Y. M. and Kwang-Ryul Ryu, F., 2000, Deriving decision rules to locate export containers in container yards. *European Journal of Operational Research*, **124**(1), 89–101.
7. Preston, P. and Kozan, E., 2001, An approach to determine storage locations of containers at seaport terminals. *Computers & Operations Research*, **28**(10), 983–995.

8. Chen, T., 1999, Yard operations in the container terminal – a study in the 'unproductive moves'. *Maritime Policy & Management*, **26**(1), 27–38.
9. Chen, T., Lin, K. and Juang, Y.-C., 2000, Empirical studies on yard operations Part 2: quantifying unproductive moves undertaken in quay transfer operations. *Maritime Policy & Management*, **27**(2), 191–207.
10. Henesey, L. 2005, A simulation model for analyzing terminal management operations, Proceedings of the 4th International Conference on Computer Applications and Information Technology in the Maritime Industries, Hamburg, Germany.
11. Ma, N. and Hadjiconstantinou, E., 2007, Container assignment and yard crane deployment in a container terminal: a new mathematical formulation. Imperial College Business School, Imperial College, London, Working paper.
12. Laik, N. and Hadjiconstantinou, E., 2008, Container assignment and yard crane deployment in a container terminal: a case study. *Maritime Economics & Logistics*, **10**, 90–107.
13. Hartmann, S., 2004, Generating scenarios for simulation and optimization of container terminal logistics. *OR Spectrum*, **26**(2), 171–192.
14. Lee, T.-W., Park, N.-K. and Lee, D.-W., 2003, A simulation study for the logistics planning of a container terminal in view of SCM. *Maritime Policy & Management*, **30**(3), 243–254.
15. Nevins, M. R., Macal, C. M., Love, R. J. and Bragen, M. J., 1998, Simulation, animation and visualization of seaport operations. *Simulation*, **71**(2), 96–106.
16. Ottjes, J. A., Veeke, H. P. M., Duinkerken, M. B., Rijsenbrij, J. C. and Lodewijks, G., 2006, Simulation of a multiterminal system for container handling. *OR Spectrum*, **28**(4), 447–468.
17. Banks, J., Carson, J. S. and Nelson, B. L., 2005, *Discrete-event system simulation*, 4th ed. (Princeton: NJ: Prentice Hall), pp. 249–269.