No. 115/20, 42–54
ISSN 2657-6988 (online)
ISSN 2657-5841 (printed)
DOI: 10.26408/115.05

Submitted: 20.02.2020
Accepted: 15.07.2020
Published: 30.09.2020

OPTIMIZATION STRATEGIES OF CONTAINER TERMINALS

Alexander L. Kuznetsov[1]*, Alexander V. Kirichenko[2], Anton D. Semenov[3], Albert Z. Borevich[4]

[1,2,3] Admiral Makarov State University of Maritime and Inland Shipping, 5/7, Dvinskaya str, Saint Petersburg, 198035, Russia

[4] Peter the Great St. Petersburg Polytechnic University, Saint Petersburg, 195251, Polytechnicheskaya, 29, Russia

[1] ORCID 0000-0002-5936-9326, e-mail: thunder1950@yandex.ru

* Corresponding author

Abstract: One of the main tasks of operational management of modern container terminals is effective utilization of existing technological resources. The search for possible variants of terminal activity optimization requires constant analysis of transport process technology and development of technical and technological approaches to increase the effectiveness of resources utilization. One of the possible solutions provided by terminal operational systems is the application of different strategies of container stack organization and container selectivity. At the same time, the complicated character of input and output container flow makes the effectiveness of these strategies doubtful. The paper states that this particular manner of container service is different from traditional ones, i.e. FIFO and FILO. As output container flow is randomly distributed the manner of container service can be considered as First In/Random Out. The paper also considers different strategies that are applied in practice and analyzes its influence on the productivity of handling equipment. The results of these strategies simulation modeling are represented. The results prove that no strategy can provide a productivity with theoretical selectivity; any strategy provides slowed intensity of operations. At the same time, it is proved that the only optimization strategy which increases the productivity of handling equipment is the organization of terminal activity whereby containers can be selected from the stack not in requirement sequence, but in order of its position in a stack.

Keywords: container terminal, stacking strategy, selectivity, productivity.

1. INTRODUCTION

The concept of queues and stacks is widely used in different fundamental and applied scientific disciplines [Christopher 2016; Zomaya and Sakr 2017; Shortle et al. 2018]. If an arbitrary object (so called ‘server’) receives a request to perform an intrinsic operation (a so-called “job”) but is not able to do it immediately, the job is suspended. The next job arriving in the system joins the queue in line behind the previous one, etc. The set of the jobs waiting for servicing is called a “queue”.

When the possibility to serve a job appears, the one that came first is selected. This servicing pattern is called "first in – first out" discipline and this queue type is named FIFO, or just "queue" [Suchánek and Bucki 2017]. A classic example of such a queue is trucks arriving at a terminal and "physically" passing the gate one by one, i.e. "first in – first out".

Another possible discipline of jobs selection from the queue is "first in – last out. A job which has just arrived becomes "the top one in the pile", and when the possibility to serve a job appears, it is selected first. This queue type is called FILO, or "stack". A sample of this queue is the vertical pile of boxes in container yard: a box that came last will inevitably be picked first.

The main function of the container terminal as a logistic object is to receive containers arriving by certain transport mode, to store them for some time, and to dispatch them from the terminal by selected transport mode. The immediate acceptation of a container and passing it to outbound transport (so-called "direct handling") in most cases is impossible. Consequently, the containers representing single jobs for the terminal as the queueing system, form a queue waiting for servicing, namely the container stack.

Two ***fundamental ways*** to select jobs from the queue, i.e. FIFO and FILO, are universal for all scientific disciplines. In applied domains there could be more ***specific ways*** for this selection, e.g. the well known in logistics principle FELO, when goods with the period of validity close to expiry are selected in the first head (First Expire-First Out).

There is yet another specific principle of selection practiced on container terminals, which could be called FIRO (First In/Random Out): the containers arriving at the terminal are picked for dispatching in random order.

At the same time, in majority of cases the "physical" organization of any specific queue servicing discipline could be reduced into a basic schema of FIFO and FILO: this could be a "queue" with an entrance for arriving jobs and an exit for departing ones, or a "stack" where a new job "pushes" the existing ones in the system to the bottom and takes the closest position to the entrance that likewiseserves as an exit (to select a certain job one needs to remove all the other jobs upon it). Selection of a random job from the middle of a queue or a stack can be implemented only by the repetition of classical FIFO or FILO operations. These operations are executed until there is a necessary job on the exit. At the same time, the removed jobs are not considered as serviced and must be placed back to the sequence of jobs waiting for service. In the case of stack, this operation would require another stack temporary storage of jobs. In the case of queue, there are two ways: (i) to create an additional queue or (ii) to loop an existing one (this operation means that not-suitable jobs are returned to the beginning of the queue).

Since containers arrive at a terminal in a random consequence, unknown and uncontrollable for a terminal operator, and are dispatched from the terminal in the same stochastic way, the procedure of their handling requires more complex research of FIRO discipline and methods applied in terminals for the reduction of this discipline's influence.

2. GENERAL DESCRIPTION OF THE PROBLEM

Containers arrive to a terminal in a random consequence $\alpha_1, \alpha_2, \dots, \alpha_i$. As containers cannot be directly loaded from one transport to another, the sequence forms a set of jobs waiting for service. Containers leave a terminal in another sequence $\beta_1, \beta_2, \dots, \beta_j$. This sequence is completely different from both the entering order $\alpha_1, \alpha_2, \dots, \alpha_i$ and the reversed order $\alpha_i, \dots, \alpha_2, \alpha_1$. As a result, the disciplines FIFO and FILO could not be applied to the terminal operations directly. To export a container β_j from a terminal it is necessary to select it from the set of boxes waiting for a transport to leave port.

Containers are stacked one upon another in order to conserve space in the terminal. This manner of storage forms a classical "stack": the last container that was put in it has to be selected first [Kuznetsov, Semenov and Levchenko 2019]. If containers are stacked in one tier the "stack" has a deepness equal to one: to take a container from the stack one needs to perform a single operation (commonly named "movement"). If containers are stacked one upon another in a stack with height H then its deepness is H. To take a container from the highest tier one needs to make a single movement, and two movements to take a container which is the second from the top. The lowest container in the stack requires H movements to get it. Therefore, if all the containers are randomly required, the average number of movements necessary to get all the containers can be calculated as:

$$N = \frac{H+1}{2} \tag{1}$$

Due to technical and operational restrictions it is impossible to place all the containers in a single "stack". Particularly, top containers would destroy the structure of the ones placed below and it also would be too difficult to get a container from this stack as the operation would require a large number of movements [Kirichenko et al. 2017]. These are the reasons why containers are stored in a set of stacks limited by their height. This set is called a container yard stack.

Assume that the indexes of containers in the sequence of their export is decreased with the height of storage, i.e. a container β_l with a lower index would be stored under container β_k with higher index ($l < k$). In this case, all the containers require one movement to get them from a stack. If these conditions are not fulfilled and there is an inversion in the consequence of container export, there occurs a need for additional movements (in order to gain access to the target container). As a result, to get a target container one needs to remove all the containers that are stored upon it. These "taken, but not serviced" jobs must be placed to another stack in container yard, as was pointed out earlier. It would be better if these removements bring the monotony of export consequence indexes. However, this rarely can be true: as it was pointed earlier the sequence $\beta_1, \beta_2, \dots, \beta_j$ has a stochastic nature and its further part is formed after this decision is made. Therefore, the strategy when containers that must be removed are placed in stacks with more space for storage seems to be the most rational decision.

Regarding everything said above, a conclusion can be drawn: in such difficult environment there could not be any optimization strategy that seeks to decrease the number of movements necessary to handle the stated cargoflow. The complexity of container handling is defined by the size of a stack. The place of a container in the stack could not be forecasted by the sequence of container export. Of course, there are optimization methods and algorithms that are effectively applied in today's practice. However, usually the goal of these methods is to decrease the handling time of a transport vehicle, not the number of movements. In its turn, additional movements increase the number of handling equipment units needed. In this case, the optimization criterion that would be considered farther is:

$$N \rightarrow min \tag{2}$$

where N – is the average number of movements per one container.

As a rule, when a container stack structure is organized and operations connected with it are managed, some methods and approaches are applied. These methods are usually stated in the operational personnel instructions and sometimes included in the terminal operating system (TOS). It cannot be doubted that for the personnel, these approaches become optimization strategies [Cordeau et al. 2015; Ji et al. 2015; Euchi et al. 2016].

3. STACKING STRATEGIES

At the same time, the TOS is the set of software tools that implements the methods and strategies of container stacking created by men. Let's now discuss the frequently used decisions that are usually referred to as optimization.

Pre-stacking. The preliminary placement of containers, that forms an export party in a special stack behind the transport vehicle handling operations area, is called pre-stacking. Theoretically, a certain loading plan of a transport vehicle allows to form such structure of a stack that would require only one movement per container to load a container onto a vehicle. In this case, the vehicle handling procedure is optimal as all movements are productive. Such technology provides the minimum time of vehicle handling operation.

With regard to the stated terminology and designation, pre-stacking supposes that the export sequence $\beta_k, \beta_{k+1}, \dots, \beta_{k+K}$ that matches the loading plan of a vehicle (for example, ship, barge, freight train) with the capacity $K > 0$ is formed in any event.

It is obvious that the forming of a queue in the direct sequence $\beta_k, \beta_{k+1}, \dots, \beta_{k+K}$ or stack in the reverse $\beta_{k+K}, \dots, \beta_{k+1}, \beta_k$ allows implementation of handling operations of transport vehicles without any additional movement: any given container would require only one movement which, in this case, is productive.

At the same time, the process of $\beta_k, \beta_{k+1}, \dots, \beta_{k+K}$ or $\beta_{k+K}, \dots, \beta_{k+1}, \beta_k$ sequence forming remains a random one. Therefore, all the reasonings stated above are true for this strategy. In other words, preliminary stack formation would require the same number of movements which would be needed to load a vehicle in a direct way, without it. At the same time, replacement of containers from the preliminary set of boxes (a queue or a stack) would require K more additional movements.

Consequently, from the technological point of view pre-stacking does not decrease the number of movements needed to implement a handling operation; it increases it. The optimizational value in this case is not the number of moves of container, but the time of vehicle loading operation. Only K productive movements (to handle containers from a premilirary stack to a transport vehicle) must be done during that time, while all other movements can (from an operational point of view) be executed in a more suitable time. Actually, pre-stacking is an effective technology minimizing the time of loading operations, but it is too sensitive to the changes in loading plan as additional unproductive movements in a preliminary stack could increase their total value to the limits where the effectiveness of pre-stacking becomes illusory.

Post-stacking. The symmetric decision is post-stacking, when containers are unloaded from a vehicle to a special temporary stack located behind the handling operations area and, after they have been unloaded, all the containers are moved from the temporary stack to the container yard. This strategy mostly reduces the necessary number of equipment units providing horizontal transportation and synchronization of unloading operation as placement of containers in the stack does not require additional movement to get them from the stack (so-called selectivity). The further selection of containers from that stack requires one more additional movement.

Distribution of sections by transport modes. This strategy could be considered as a "weak variant" of pre-stacking. It assumes that a certain portion of the containers that have arrived at the terminal is placed in a block located close to the tracks servicing the mode of transport that would be used for dispatching containers.

This strategy is supposed to be most efficient for rail operations, when ship arriving intervals are close to those of the train. Even in this case the train layout and cargo plan are known only after the allocation of containers in the block's stack, which causes the container shuffling.

Eventually, it makes no difference from the point of view of operation laboriousness, where these extra moves are made in the dedicated stack or in any other one.

Distribution by clients. If the client's structure is homogeneous, there is no difference for operational planning whether the stacks are organized by the clients or not. Possibly, a block of unified color containers looks more attractive from an esthetic point of view and is more controllable by the client, providing a visible and compact block. In general, it offers no operational advantage for the terminal operator, while causing a potential inefficiency of the container yard space utilization and the increase of total transportation distance of cargo handling equipment.

4. EXTREME CASE

The most efficient way of partial (local) optimization of this kind is an extreme case whereby the selection of containers included in the day task is governed not by the succession of vehicles, but by the minimal laboriousness of the selection. In other words, a vehicle arrived to pick a container from the day task is loaded with the container which is the closest to the upper surface of the stack, i.e. the container with the maximal selectivity. Still, its selection could alter the structure of the stacks, blocking another container from the day-task.

The search for candidates for selection should be repeated every time after the loaded vehicle leaves the terminal. Obviously, this hypothetic and highly idealized selection procedure is the best from the stack structure (i.e. terminal operator) point of view. Any other strategy would only asymptotically approach the ideal variant, so the latter could be treated as a top rating for any realistic strategies claiming to have some optimal features.

Let us address once again the single act of container selection from the stack. Let us assume that we have E containers allocated over the area which consists of w ground slots. Respectively, these containers would form a block with the height of:

$$H = \frac{E}{w} \tag{3}$$

If a target container is located in the top tier, its selection requires only one move; if it is in the tier next to the top two moves will be needed (removal of the blocking container and selection of the target one) while the container in the bottom tier will need H moves. If the probability of the target container allocation is equal for all tiers, its value is

$$p = \frac{1}{H} \tag{4}$$

Accordingly, the mathematical expectation of the number of moves is:

$$\begin{aligned} M[N] &= 1 \cdot p + 2 \cdot p + \ldots + H \cdot p = \\ &= \frac{1}{H} \cdot (1 + 2 + \cdots + H) = \\ &= \frac{1}{H} \cdot \frac{(H+1) \cdot H}{2} = \frac{H+1}{2} \end{aligned} \tag{5}$$

This expression could be gained easier, observing that the top container is selected in one move, the bottom one – in H moves, so the average number of moves (with equal probability of allocation in tiers) is $\frac{H+1}{2}$ moves. Still, the former inference is important for description of the solution for the next task – assessment of the efficiency of the multiple selection of containers.

Let us suppose that there are still E containers stored over the area of w ground slots in the block $H = \frac{E}{w}$ high. Let us also assume that there is a set of $K < w$ containers and we have to select any one of them which is above all in the block. This selection would require one move if there is at least one target container from the set of K. Two moves would be needed when there is no target container in the top tier and at least one target container is in the tier to top. Three moves appear when two upper tiers have no target container and the third one from top has at least one target container from the set of K. Maximal number of moves H is needed when all K containers are in the bottom tier.

What is the mathematical expectation of the required number of moves for selection? What is the distribution of movements by the set of containers?

This task could be solved by the methods of classical theory of probability. Let us regard the stack with the volume of $E = w \cdot H$ containers (Fig. 1).

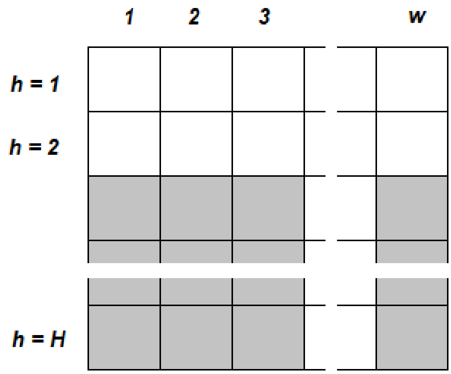

Fig. 1. Container stack parameters

Let us find the probability of an event A_h, which is that in h top tiers of the stack w raw wide there is not a single target container from the set K. This event could be implemented by C_{E-K}^{wh} differeтt variants [Gnedenko 2018]. On the other hand, there are C_E^{wh} different variants to fill this upper segment, so the probability of target containers' absence in h top tiers is:

$$P(A_h) = \frac{C_{E-K}^{wh}}{C_E^{wh}} \tag{6}$$

The probability of the opposite event $\overline{A_h}$, i.e. the presence of the target container(s) in h top tiers is:

$$P_h = P(\overline{A_h}) = 1 - P(A_h) = 1 - \frac{C_{E-K}^{wh}}{C_E^{wh}} \tag{7}$$

Respectively, P_1 is the probability to find at least one target container in the top tier $h = 1$, or $p_1 = P_1$. The value P_2 is the probability to find at least one target container in the top two tiers $h = 1$ and $h = 2$. Thus, the probability to find at least one target container in the tier $h = 2$ is $p_2 = P_2 - P_1$.

5. RESULTS

In general, the probability to find at least one target container in the tier h is given by a recursive expression $p_h = P_h - P_{h-1}$. The family of events p_h, $h = \overline{1, H}$ constitute the required probabilities to find a target container from the set of K in the tier h (calculated from top to bottom).

Table 1 gives an example of calculation of the probabilities p_h for the block of $E = 150$ containers w wide with different size of the target set K.

Table 1. Probability to find a target container in the tier h

	The number of target containers, K				
Tier, h	1	2	3	4	5
1	0,17	0,31	0,42	0,52	0,60
2	0,17	0,25	0,28	0,28	0,27
3	0,17	0,19	0,17	0,13	0,10
4	0,17	0,14	0,09	0,05	0,03
5	0,17	0,08	0,03	0,01	0,00
6	0,17	0,03	0,00	0,00	0,00
M[N]	3,50	2,52	2,03	1,75	1,56

Figure 2 shows the same results in graphic form.

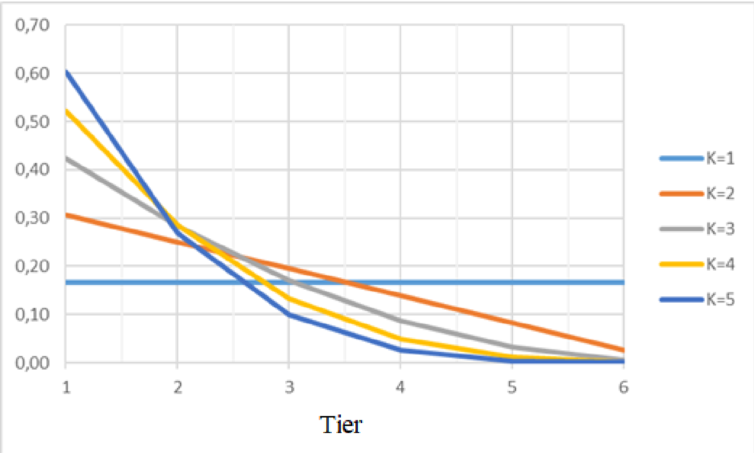

Fig. 2. Probability to find a target container in tier h

The mathematical expectation of the number of moves for every value of the size of the set K is given by the expression:

$$M[N] = 1 \cdot p + 2 \cdot p + \ldots H \cdot p \tag{8}$$

The correspondent values are represented in the last string in Table 1. The case $K = 1$ represents the single selection of containers, and the results coincide with the ones derived above. The required number of moves decrease significantly with the increase of the number of target containers (Fig. 3).

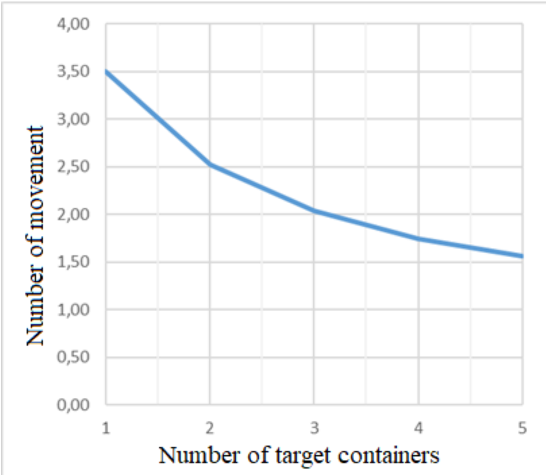

Fig. 3. Number of moves as a function of the target set size

The results show that the multiple selection procedure is the only efficient practice for container terminal operations. If a client agrees to use this discipline (called "multiple visits"), then the complexity of selection could be diminished dramatically. Usually this is the case in operations serving port satellite terminals, large retail companies, car manufacturers, etc.

6. DISCUSSION

The simulation modeling of terminal activity based on the strategies described above gives a complete picture. Figures 4–5 represent the result of two extreme strategies modeling: with optimization (selection of the containers placed on the higher positions) and without one (selection of containers in the consequence of their requirements). The calculated selectivity was increased by one for every container: this movement represents the operation of box placement to a stack.

The results prove that any methods of optimization constructed into TOS should not be overestimated: they could only make the selectivity closer to the combinatorial one. Regarding everything indicated earlier, it can be assumed that under different conditions of terminal activity, with application of various optimization strategies the commercial productivity would vary within the area represented in Figure 6.

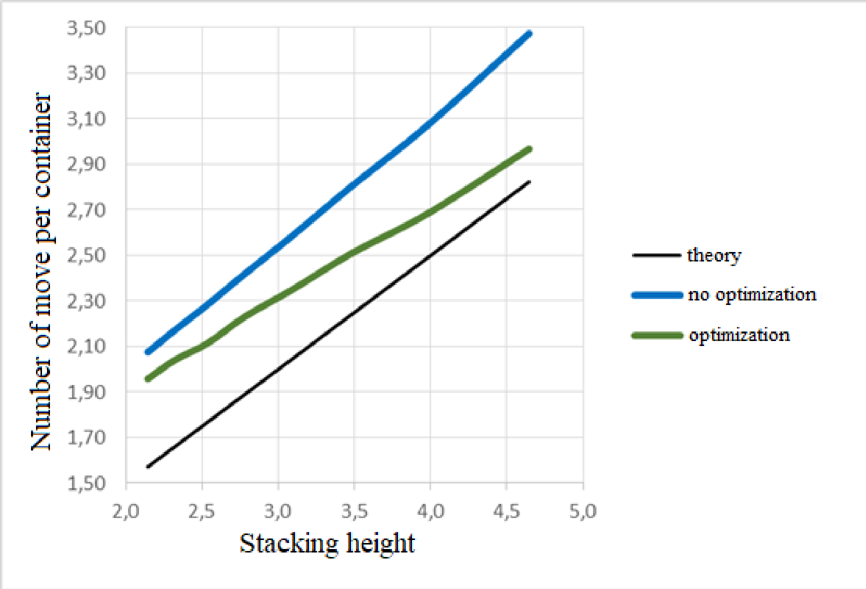

Fig. 4. Comparison of number of movements in different strategies

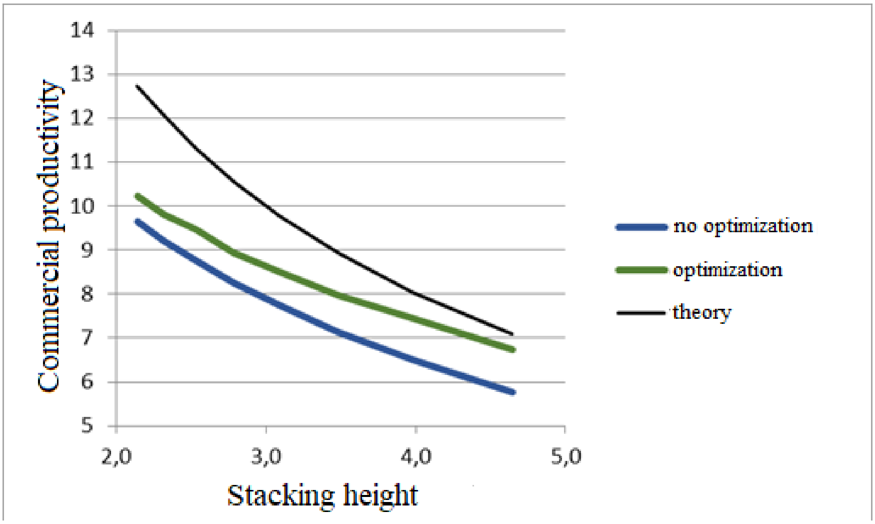

Fig. 5. Comparison of commercial productivity in different strategies

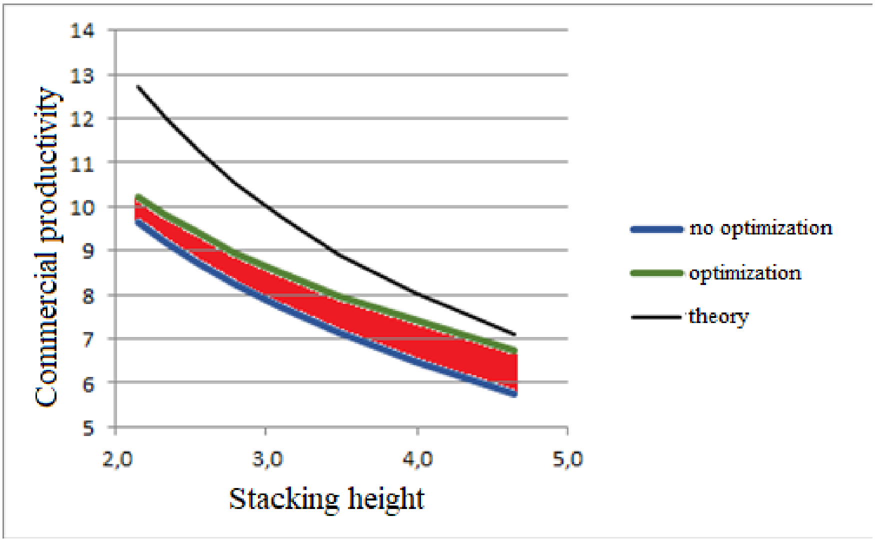

Fig. 6. Area of commercial productivity

Now let us apply the results to the rough estimation of a terminal's number of equipment units. If a ship with a 1500 container capacity (about 3000 TEU) calls at a terminal every three days, unloading and loading $1500 + 1500 = 3000$ containers, then the annual terminal cargoflow is 365 000 containers (about 700 000 TEU). Consequently, the average number of containers passing through a container yard is 1000 boxes per day or about $\frac{1000}{20} = 50$ containers per hour.

If the terminal uses RTG cranes with a productivity of 20 movements per hour and the stacking height is 4 tiers, then the commercial productivity of each machine is 4 movements per hour. Therefore, to handle the stated cargoflow, a terminal needs 13 cranes.

In this way, the results of the analytical calculations, which were based on the selectivity values defined by simulation modeling, correspond with the statistics data of the container terminal.

7. CONCLUSIONS

1. Containers arriving and dispatching from a terminal do not correspond to any known logistics disciplines of queue servicing (FIFO, FILO, FELO).
2. The sequence of containers export from a terminal has a stochastic nature and, therefore, the discipline of container handling can be named First In/Random Out (FIRO).

3. Optimizational strategies of container selection, offered by modern terminal operating systems, allow the reduction of vehicle handling time, at the same time increasing the number of movements and the number of necessary equipment units.
4. The only possible optimizational strategy that could save the necessary number of equipment units is the way of commercial activity when-containers should be selected not consequent to arrival requirements, but in the sequence of their positions within a stack.

REFERENCES

Christopher, M., 2016, *Logistics & Supply Chain Management*, Pearson UK.

Cordeau, J.F., Legato, P., Mazza, R.M., Trumfio, R., 2015, *Simulation-Based Optimization for Housekeeping in a Container Transshipment Terminal*, Computers & Operations Research, vol. 53, pp. 81–95.

Euchi J., Moussi, R., Ndiaye, F., Yassine, A., 2016, *Ant Colony Optimization for Solving the Container Stacking Problem: Case of Le Havre (France) Seaport Terminal*, International Journal of Applied Logistics (IJAL), vol. 6 no. 2, pp. 81–101.

Gnedenko, B.V., 2018, *Theory of Probability*, Gnedenko B.V., Routledge.

Ji, M., Guo, W., Zhu, H., Yang, Y., 2015, *Optimization of Loading Sequence and Rehandling Strategy for Multi-quay Crane Operations in Container Terminals*, Transportation Research, Part E: Logistics and Transportation Review, vol. 80, pp. 1–19.

Kirichenko, A.V., Kuznetsov, A.L., Davydenko, A.A., Latukhov, S.V., Nikitin, V.A., 2017, *Morskaya Konteinernaya Transportno-Tekhnologicheskaya Sistema: monografiya*, Kirichenko A.V. (ed.), SPb.: Izd-vo MANEB.

Kuznetsov, A.L., Semenov, A.D., Levchenko, V.P., 2019, *The Influence of Container Handling Equipment Technical Limitations on the Operations Productivity*, Vestnik Gosudarstvennogo Universiteta Morskogo i Rechnogo Flota imeni Admirala S.O. Makarova, 11.3, St. Petersburg, pp. 417–429.

Shortle, J.F., Thompson, J.M., Gross, D., Harris, C.M., 2018, *Fundamentals of Queueing Theory*, John Wiley & Sons.

Suchánek, P., Bucki, R., 2017, *Modelling of the Logistic Supplier-Consumer Behavior*, KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications, Springer, Cham, pp. 213–222.

Zomaya, A.Y., Sakr, S. (ed.), 2017, *Handbook of Big Data Technologies*, Springer, Berlin.