Walkowiak Tomasz

Mazurkiewicz Jacek
Wrocław University of Technology, Poland

Critical situations and human resources in discrete transportation systems analysis

Keywords

discrete transportation system, system dependability, critical situations analysis, Monte-Carlo simulation

Abstract

The paper presents an approach to critical situations analysis related to discrete transportation systems (*DTS*). The critical situation is understood as sudden shortage of some system recourses resulting in the transportation system performance degradation. The analysis is realized based on availability calculation of *DTS*. The system is described by the formal model, which includes reliability and functional parameters of *DTS* as well as human resources - drivers and management system. The availability and average availability of the system, defined in a functional way, is discussed as a function of different essential parameters of *DTS*. The proposed analysis is based on the modelling and simulating of the system behaviour using Monte-Carlo approach. The paper presents some exemplar systems – based on real Polish Post *DTS* - modelling and results.

1. Introduction

Administration of a large transportation system is not a trivial task. The transportation systems are characterized by a very complex structure. The performance of the system can be impaired by various types of faults related to the vehicles, communication infrastructure or even by traffic congestion [1], [8], [13], [14]. It is hard for human (administrator, owner) to understand the system behaviour. To overcome this problem we propose a functional approach. The transportation system is analysed from the functional point of view, focusing on business service realized by a system [17], [20]. The analysis is following a classical [4]: modelling and simulation approach. It allows calculating different system measures which could be a base for decisions related to administration of the transportation systems. The metric are calculated using Monte Carlo techniques [7]. No restriction on the system structure and on a kind of distribution is the main advantage of the method. The proposed model allows forgetting about the classical reliability analysis based on Markov or Semi-Markov processes [2] - idealised and hard for reconciliation with practice.

The paper presents an analysis of transportation system of the Polish Post regional centre of mail distribution (described in section 2). Base on which we have developed the discrete transportation system model presented in section 3. The main service given by the post system is the delivery of mails. From the client point of view the quality of the system could be measured by the time of transporting the mail from the source to destination. Important elements of the system are human resources – vehicle drivers. Their model includes formal, law-related, aspects, like for example number of hours he or she can work daily.

We offer three approaches to system management: based on time-tables, heuristic and focused on soft-computing (section 4). In our opinion the heuristic one seems to be the most adequate to the level of detail discussed in the described work. The quality of the analysed system is measured by the availability defined as the ability to realize the transportation task within a required time interval (described in section 5). The post system is very hard to be analysed by a formal model since it does not lay in the Markov process framework. Therefore, we have used a computer simulation [3],

[4], [12]. Next (section 6), we give an example of using presented model and simulator for the analysis of the Polish Post regional centre in Wroclaw transportation focusing on the most probability sources of the potential critical situations.

2. Model based on the Polish post solution

The analysed transportation system is a simplified case of the Polish Post. The business service provided by the Polish Post is the delivery of mails. The system consists of a set of nodes placed in different geographical locations. Two kinds of nodes could be distinguished: central nodes (*CN*) and ordinary nodes (*ON*). There are bidirectional routes between nodes. Mails are distributed among ordinary nodes by trucks, whereas between central nodes by trucks, railway or by plain. The mail distribution could be understood by tracing the delivery of some mail from point *A* to point *B*. At first the mail is transported to the nearest ordinary node *A*. Different mails are collected in ordinary nodes, packed in larger units called containers and then transported by trucks scheduled according to some time-table to the nearest central node. In central node containers are repacked and delivered to appropriate (according to delivery address of each mail) central node. In the Polish Post there are 14 central nodes and more than 300 ordinary nodes. There are more than one million mails going through one central node within 24 hours. It gives a very large system to be modelled and simulated. Therefore, we have decided to model only a part of the Polish Post transportation system – one central node with a set of ordinary nodes.

Essential in any system modelling and simulation is to define the level of details of modelled system. It could be done if a kind of measures calculated by the simulator is known. Since the business service given by the post system is the delivery of mails on time. Therefore, we propose to calculate the time of transporting mails by the system. Since the number of mails presented in the modelled system is very large and all mails are transported in larger amounts - containers, we have decided to use containers as the smallest observable element of the system.

Therefore, the main observable value calculated by the simulator will be the transportation time of a container from the source to the destination node. The income of mails to the system, or rather containers of mails as it was discussed above, is modelled by a stochastic process. Each container has a source and destination address. The central node is the destination address for all containers generated in the ordinary nodes. Where containers addressed to any ordinary node are generated in the central node.

The generation of containers is described by some random process. In case of a central node, there are separate processes for each ordinary node. Whereas, for ordinary nodes there is one process, since commodities are transported from ordinary nodes to the central node or in the opposite direction. The containers are transported by vehicles, which require a driver to control it. Each vehicle has a given capacity – maximum number of containers it can haul. Central node is a base place for all vehicles. They start from the central node and the central node is the destination of their travel. The vehicle hauling a commodity is always fully loaded or taking the last part of the commodity if it is less than its capacity. Vehicles operate according to the time-table. The time-table consists of a set of routes (sequence of nodes starting and ending in the central node, times of leaving each node in the route and the recommended size of a vehicle). The number of used vehicles, number of drivers and the capacity of vehicles do not depend on temporary situation described by number of transportation tasks or by the task amount for example.

It means that it is possible to realise the route by completely empty vehicle or the vehicle cannot load the available amount of commodity (the vehicle is too small). Time-table is a fixed element of the system in observable time horizon, but it is possible to use different time-tables for different seasons or months of the year. Summarising, the movement of the containers in the system, a container is generated with destination address in some of node (source) at some random time. Next, the container waits in the node for a vehicle to be transported to the destination node. Each day a given time-table is realised, it means that at a time given by the time table a vehicle, selected from vehicles available in the central node, and driver selected form available drivers starts from the central node.

A vehicle is loaded with containers addressed to each ordinary nodes added to a given route. This is done in a proportional way. When a vehicle approaches the ordinary node it is waiting in an input queue if there is any other vehicle being loaded/unload at the same time. There is only one handling point in each ordinary node. The time of loading/unloading vehicle is described by a random distribution. The containers addressed to given node are unloaded and empty space in the vehicle is filled by containers addressed to a central node. Next, the vehicle waits till the time of leaving the node (set in the time-table) is left and starts its journey to the next node. The operation is repeated in each node on the route and finally the vehicle is

approaching the central node when it is fully unloaded and after it is available for the next route.
The process of vehicle operation could be stopped at any moment due to a failure (described by a random process). After the failure, the vehicle waits for a maintenance crew (if there are no available due to repairing other vehicles), is being repaired (random time) and after it continues its journey. The vehicle hauling a commodity is always fully loaded or taking the last part of the commodity if it is less than its capacity.

3. Discrete transportation system formal model

3.1. Model overview

The described in the previous section regional part of the Polish Post transportation system with one central node and several ordinary nodes was a base for a definition of a formal model of a discrete transportation system (*DTS*).
Generally speaking users of the transportation system are generating tasks which are being realized by the system. The task to be realized requires some services presented in the system. A realization of the system service needs a defined set of technical resources. Moreover, the operating of vehicles transporting mails between system nodes is done according to some rules – some management system. Therefore, we can model discrete transportation system as a 5-tuple:

$$DTS = \langle Client, Driver, BS, TI, MS \rangle \quad (1)$$

Client – client model,
Driver – driver model,
BS – business service,
TI – technical infrastructure,
MS – management system.

3.2. Technical infrastructure

During modelling of technical infrastructure we have to take into consideration functional and reliability aspects of the post transportation system [6].

Therefore, the technical infrastructure of *DTS* could be described by three elements:

$$TI = \langle No, V, MM \rangle , \quad (2)$$

No – set of nodes,
V – set of vehicles,
MM – maintenance model.

Set of nodes (*No*) consists of single central node (*CN*), a given number of ordinary nodes (ON_i). The distance between each two nodes is defined by the function:

$$distance : No \times No \rightarrow R_+ . \quad (3)$$

Each node has one functional parameter the mean (modelled by normal distribution) time of loading a vehicle:

$$loading : No \rightarrow R_+ . \quad (4)$$

Moreover, the central node (*CN*) has additional functional parameter: number of service points (in each ordinary node there is only one service point):

$$servicepoints : CN \rightarrow N_+ . \quad (5)$$

Each vehicle is described by following functional and reliability parameters:

- mean speed of a journey
  $$meanspeed : V \rightarrow R_+ , \quad (6)$$
- capacity – number of containers which can be loaded
  $$capacity : V \rightarrow R_+ , \quad (7)$$
- mean time to failure
  $$MTTF : V \rightarrow R_+ , \quad (8)$$
  a time when failure occurs is given by exponential distribution with mean equal to a value of *MTTF* function,
- mean repair time
  $$MRT : V \rightarrow R_+ . \quad (9)$$

The traffic is modelled by a random value of vehicle speed and therefore the time of vehicle (*v*) going from one node (n_1) to the other (n_2) is given by a formula

$$time(v, n_1, n_2) = \frac{distance(n_1, n_2)}{Normal(meanspeed(v), 0.1 \cdot meanspeed(v))} , \quad (10)$$

where *Normal* denotes a random value with the truncated Gaussian distribution.
Maintenance model (*MM)* consists of a set of maintenance crews which are identical and indistinguishable. The crews are not combined to any node, are not combined to any route, they

operate in the system. Only by the number of the crews is known.
The time when a vehicle is repaired is equal to the time of waiting for a free maintenance crew (if all crews involved into maintenance procedures) and the time of a vehicle repair which is a random value with the Gaussian distribution ($Normal(MRT(v), 0.1 \cdot MRT(v))$.

3.3. Business service

Business service (*BS*) is a set of services based on business logic that can be loaded and repeatedly used for concrete business handling process.

Business service can be seen as a set of service components and tasks, which are used to provide service in accordance with business logic for this process. Therefore, *BS* is modelled a set of business service components (*sc*):

$$BS = \{sc_{1,} ..., sc_n\}, n \ = length(BS) > 0, \qquad (11)$$

the function *length*(*X*) denotes the size of any set or any sequence *X*. Each service component in DTS consists of a task of delivering a container from a source node to the destination one.

3.4. Client model

The service realised by the clients of the transportation system are sending mails from a source node to a destination one. Client model consist of a set of clients (*C*) [15].
Each client is allocated in one of nodes of the transportation system:

$$allocation : C \to No. \qquad (12)$$

A client allocated in an ordinary node is generating containers (since, we have decided to monitor containers not separate mails during simulation) according to the Poisson process with destination address set to ordinary nodes. In the central node, there is a set of clients, one for each ordinary node. Each client generates containers by a separate Poisson process and is described by intensity of container generation:

$$intensity : C \to R_+. \qquad (13)$$

The central node is the destination address for all containers generated in ordinary nodes.

3.5. Driver model

Each driver is describing by actual state of him/her (s_d)

- rest (not at work),
- unavailable (illness, vacation, etc.),
- available (at work – ready to start driving),
- break (during driving),
- driving.

The number of driver working hours is limited by the labour law. The regulation is rather complicated – depends on a lot of parameters and the number of drivers fixed to the truck for example, but for transportation system we analyse we can say that the daily limit for each driver equals to 8 or 9 hours and single driver operates with one truck. This assumption is legal in European Union.
So the problem of working hours (w_h) we can solve as follow:

- if w_h > *limit* then s_d = rest & w_h = 0,
- where *limit* = 8 hours or *limit* = 9 hours,
- *limit* – is the time period of a single change in work.

The single shift can be distinguished as: morning or afternoon.
So at 6am for each driver:

- if *shift* == morning & s_d == rest then s_d = available,

So at 1pm for each driver:

- if *shift* == afternoon & s_d == rest then
  s_d = available,

The next problem ought to be modelled is the driver's illness state. We propose the following approach:

- for every driver at 4am: if s_d == rest,
- if *rand()* < d_i then during *x* days (according to the given distribution) the driver is in s_d = unavailable,
- where d_i – driver's illness parameter.

Moreover we propose to categorise the driver's illnesses as follow:

- short sick: 1 to 3 days,
- typical illness: 7 to 10 days,
- long-term illness: 10 to 300 days.

We record the daily history of the driver's work. The algorithm to fix the driver to the vehicle is the last part of the driver model:

- if no driver – the vehicle does not start,
- driver can be chosen if: s_d = available
  and w_h + *estimated time of journey* < *limit* * 1.1,

- the driver is chosen randomly or by least value approach:
  abs(limit – w_h - *estimated time of journey*).

4. Management solutions

4.1. Legacy-type management system

The management system (*MS*) of the *DTS* controls the operation of vehicle. It consists of a sequence of routes:

$$MS = \langle r_1, r_2, ..., r_{nr} \rangle . \quad (14)$$

Each route is a sequence of nodes starting and ending in the central node, times of leaving each node in the route (t_i) and the recommended size of a vehicle (*size*):

$$r = \langle CN, t_0, n_1, t_1, ..., n_m, t_m, CN, size \rangle$$
$$v_i \in No - \{CN\} \quad 0 \le t_0 < t_1 < ... < t_m < 24h \quad (15)$$

The routes are defined for one day and are repeated each day. The management system selects vehicles to realise each route in random way, first of all vehicles (among vehicles available in central node) with capacity equal to recommended size are taken into consideration. If there is no such vehicle, vehicles with larger capacity are taken into consideration. If still there is no vehicle fulfilling requirements vehicle of smaller size is randomly selected. If there is no available vehicle a given route is not realised [5], [10], [11].

4.2. Heuristic management system

The decisions (send a truck to a given destination node) are taken in moments when a container arrives to the central node. The truck is send to a trip if:

- the number of containers waiting in for delivery in the central node of the same destination address as that just arrived is larger than a given number,
- there is at least one available vehicle,
- the simulated time is between 6 am and 22 pm minus the average time of going to and returning from the destination node.

The truck is send to a node defined by destination address of just arrived container. If there is more than one vehicle available in the central node, the vehicle with size that a fits the best to the number of available containers is selected, i.e. the largest vehicle that could be fully loaded. If there are several trucks with the same capacity available the selection is done randomly.
On the other hand we observe in the same way the vehicles available in the ordinary nodes. The only difference is the greater level of threshold to initialise the vehicle journey.
The restriction for the time of truck scheduling (the last point in the above algorithm) are set to model the fact that drivers are working on two 8 hours shifts.

4.3. Soft computing management system

The system consists of a multilayer perceptron to decide if and where to send trucks. The input to the neural network consists of:

$$in = \langle pkc_1, pkc_2, ..., pkc_{npk}, cnc_1, cnc_2, ..., cnc_{npk}, nfv \rangle \quad (16)$$

npk – number of ordinary nodes,
pkc_i – number of containers waiting for delivery in the central node with destination address set to *i*-th ordinary node,
nfv – number of free vehicles in the central node,

Each output of the network corresponds to each ordinary node:

$$nnout = \langle out_1, out_2, ..., out_{npk} \rangle , \quad (17)$$

The output of the network is interpreted as follows (for sigmoid function used in output layer):

$$j = \arg\max_{i=1...npk} \{out_i\} . \quad (18)$$

If out_j is greater than 0.5 send a vehicle to node *j* else do nothing. If there are more vehicles available in the central node, the largest vehicle that could be fully loaded is selected. If there are available several trucks with the same capacity selection is done randomly. The neural network decision (send a truck or not and where the truck should be sent) are taken in given moments in time. These moments are defined by following states of the system:

- the vehicle comes back to the central node and is ready for the next trip,
- if in central node there is at least one available vehicle and the number of containers of the same destination address is larger than the size of the smallest available vehicle.

The neural network used in the management system requires a learning process that will set up the values of its weights. The most typical learning in the case of multilayer perceptron is the back propagation algorithm. However, it cannot be used here since it is impossible to state what should be the proper output values of the neural network. Since it is hard to reconcile what are the results of a single decision made by the management system. Important are results of the set of decisions. Since the business service realised by transportation system is to move commodities without delays, the neural network should take such decisions that allows reducing delays as much as possible. To train neural network to perform such task we propose to use genetic algorithm [18], [21]. Similar approach to training neural network is applied in case of computer games. The most important in case of genetic algorithm is a definition of the fitness function. To follow business service requirements of transportation system we propose following definition of the fitness function calculated for a given neural network after some time (T) (therefore after a set of decisions taken by neural network):

$$fitness(T) = \frac{N_{ontime}(0,T) + N_{ontimeinsystem}(T)}{N_{delivered}(0,T) + N_{insystem}(T)}. \qquad (19)$$

It is a ratio of on-time containers (delivered with 24h and being in the system but not longer then 24h) to all containers (that already delivered $N_{delivered}(0,T)$ and still being presented in the system $N_{insystem}(T)$).
The solution described above is very complicated and the first necessary step – learning phase – takes a lot of time and requires a lot of data to create the proper weights and other initial parameters [19].

5. Functional metrics of *DTS*

5.1. Overview

The formal model described previously was designed to allow developing a simulator which allows to observe the time of transporting each container. Based on these observations several metrics could be defined. As it was mentioned in the introduction we focus here on the service oriented approach [9], [19]. Therefore we propose that the availability to be a key parameter for the evaluation of the quality of the *DTS*. One can define the availability in different ways, but always the value of availability can be easy transformed into economic or functional parameters perfectly understood by owner of the system. The availability in mostly understood as a probability that a system is up. And is defined as a ratio of the expected value of the uptime of a system to the observation time. It is simple definition but requires defining what does it mean that transportation system is working. The similar metric is the acceptance ratio defined in information since as a number of accepted requests to the total number of requests.

5.2. Acceptance ratio

Let introduce the following notation:

- T – a time measured from the moment when the container was introduced to the system to the moment when the container was transferred to the destination (random value),
- T_g – a guaranteed time of delivery, if exceeded the container is delayed.

In [17] we have proposed performance metric – availability ratio. It is defined as a ratio of on-time containers (containers for which $T < T_g$) to all containers within a given time of observation $(0, \tau)$. Within the time period a given number of containers are delivered ($N_{delivered}(\tau)$), a part of them or all delivered on time ($N_{ontime}(\tau)$), but at the end of analysed period time there could be some containers not yet delivered (waiting in the source node or being transported) ($N_{insystem}(\tau)$) and all or part of them being not late yet ($N_{ontimeinsystem}(\tau)$). Taking into consideration introduced symbols the availability could be calculated as the expected value (Monte-Carlo approach) of ratio of on-time containers to all containers:

$$AR_\tau = E\left(\frac{N_{ontime}(\tau) + N_{ontimeinsystem}(\tau)}{N_{delivered}(\tau) + N_{insystem}(\tau)}\right). \qquad (20)$$

6. Case study

6.1. Availability tests

We proposed for the case study analysis an exemplar *DTS* based on Polish Post regional centre in Wroclaw. We have modelled a system consisting of one central node (Wroclaw regional centre) and twenty two ordinary nodes - cities where there are local post distribution points in Dolny Slask Province. The lengths of roads were set according to real road distances between cities used in the analyzed case study. The intensity of generation of containers for all destinations was set to 4.16 per hour in each direction giving in average 4400 containers to be transported each day. The vehicles speed was modelled by Gaussian distribution with 50km/h of mean value and 5 km/h of standard deviation. The average loading time was equal to 5 minutes. There were two types of vehicles: with capacity of 10 and 15 containers. The MTTF of each vehicle was set to 2000. The average repair time was set to 5h (Gaussian distribution). The containers were assumed delayed when they are delivered within a time longer than 24h. The system was simulated over 30 days period.

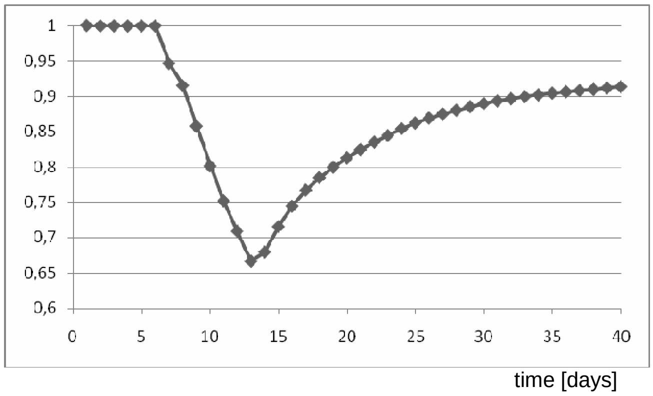
To transportation all generated in the system containers time-table was set-up. For a described example it was designed manually in a manner to allow to transportation all containers (with 10% overhead) and 15 minutes of break between each drive. It resulted in 180 drives each day with trucks of size 10 and 15. The next parameter to set-up was a number of trucks of a size 10 and 15. At first it was set to a minimum size that comes from the given time-table. In our case it was 13 trucks of size 10 and 21 of size 15. However, it was much too few resulting in dropping the availability to almost zero after a few days. After a set of experiments the optimal (giving availability more then 0.999 for the smallest number of vehicles) number was estimated as a 21 trucks of size 10 and 31 of size 15 when 20 service points was used in the central node.

As it was mentioned in the introduction we want to analyze the system performance in case a critical situation. We have assumed that after 5 days of normal system performance a critical situation will occur – a given number of drivers will be not available due to some contagious diseases (flu for example). Therefore for next 7 days the system is operating with 16 vehicles less. After these seven days the drivers are coming back to work. The achieved results (for 1000 simulations) are presented in *Figure 1a*. As it could be expected the availability in day 6 is starting to drop down to 0,6 and when drivers come back (on day 13) is slowly increasing.

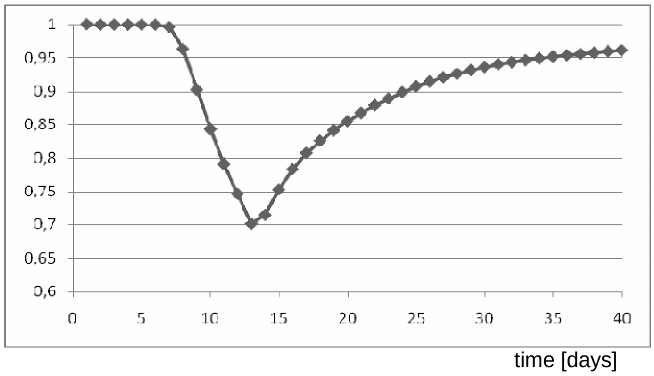
The examples show that the presented approach allows showing consequences of a defined critical situation to transportation systems performance. Now, the questions could be raised how to decrees the consequences of the critical situation? How to minimize the availability lowering? The easiest solution is to have more drivers so the some contagious diseases will not lower the number so much. However such solution enlarges operational costs very much. Therefore, we will take into consideration only the possible reconfigurations of the transportation system that will not change the system operational costs during normal operation. We have to suggestions:

- changing the time-table, but in the way that is acceptable but Work Law (i.e. the drivers could not work 24 hours per day) – results are presented in *Figure 1b*
- changing the containers loading algorithm during a critical situation (results are presented in *Figure 1c*)
    - load on vehicles not yet delayed containers before delayed ones,
    - in normal situation the containers are loaded in FIFO like mode.

(a)



(b)



(c)

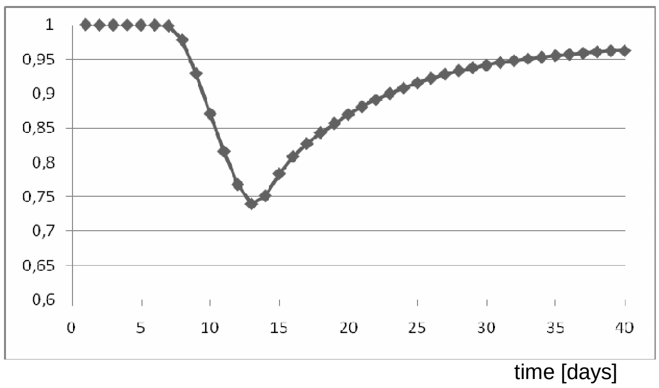


Figure 1. Availability for the exemplar *DTS* in case of critical situation (from day 5 to day 12):
(a) reference system,
(b) system with optimized timetable,
(c) system with changed loading algorithm.

6.2. Management algorithms tests

All three presented management algorithms give acceptance ratio almost equal 1 for presented system. To compare these algorithms more deeply we propose to analyse the transportation system performance in case of some critical situations [16]. Let's assume that for some days the system is working at 50%. The tie-up of the system could be caused for example by a drivers' strike or some contagious diseases resulting in situation that only 50% of vehicles are in operation. After a given number of days the system is again fully working. The achieved results (acceptance ratio calculated according to (20)) for 4 and 14 days tie-up are presented in *Figure 2* and *Figure 3.* As it could be expected the acceptance ratio in day 10 (when critical situation starts) is starting to drop down and when drivers come back is increasing. However the system with heuristic management as well as with soft computing one is coming back to normal operation much faster than legacy one. The soft computing one is slightly outperforming the heuristic one.

Next, we have analysed the ability of each management system to return the *DTS* to normal performance after critical situation of different duration. Results are presented in Figure 5. As it could be seen soft computing and heuristic algorithm give almost identical results, which significantly out performance the legacy management system.

6.3. Human resource tuning

We tried to realise the transportation tasks defined above using: 41, 44, 46 and 48 trucks. The number of drivers to operate the vehicle fleet we changed from 75 to 100. As we can see in *Figure 4*. 75 drivers are absolutely inefficient for the analysed system – the acceptance ratio is almost zero and it does not depend on the number of used trucks.

The number of vehicles equal to 44, 46, 48 can be noticed as correct to make the *DTS* operative. The acceptance ratio grows up quickly and reaches the value of 0.9 – when we can say that system works "safety-far" from the border of inacceptable state. It is interesting that we need (at the level of acceptance ratio equal to 0.9) only 82 drivers to operate with 48 trucks, 85 drivers if we have 46 trucks and 90 drivers for 41 trucks.

If we require the acceptance ration at the level of 1 – 85 drivers are necessary for 48 trucks, 90 drivers for 46 vehicles and 95 drivers for fleet of 44 trucks.

It is easy to notice that 41 trucks are not acceptable size of vehicle fleet to make the example *DTS* operative. There is no chance to substitute the shortage of trucks by the number of drivers. The acceptance ratio cannot reach the value of 0.6 even if we use 200 drivers.

The example shows how we can tune the size of *DTS* if we know the possible tasks definitions. The *DTS* owner's decisions ought to be taken in multidimensional environment. Our approach to *DTS* modelling and simulation can make the decision easier – we can observe immediately the results of possible solutions. We final decision ought to be the best – because it always generates financial consequences.

7. Conclusion

We have presented an approach to analyse management algorithms of discrete transportation system (*DTS*) by modelling and simulation. The *DTS* model was simulated using developed by the authors' simulation software. The simulator was implemented using the Scalable Simulation Framework (*SSF*) [19]. It allows performing reliability and functional analysis of the *DTS*, for example:

- determine what will cause a "local" change in the system,
- make experiments in case of increasing number of containers per day incoming to system,
- identify weak point of the system by comparing few its configuration,
- better understand how the system behaves.

We have compared three different management algorithms: legacy one (based on real time-table management in Post system), heuristic and soft computing one. The presented results show that soft computing management system allows the *DTS* to return to normal operation after some critical situation in the shortest time. The results for heuristic management system are very close to soft computing approach, whereas, the legacy system is performing much worse. Since the soft computing algorithm requires a time consuming learning procedure we propose to use in practice much simpler heuristic management system. The performance of heuristic solution (measured as the time of returning *DTS* to normal work) is only slightly worse than available by soft computing approach. The analysis of the system performance, like acceptance ratio and the time of returning to required value acceptance ratio, in a case of critical situations (shortage of vehicles or drivers) were given.

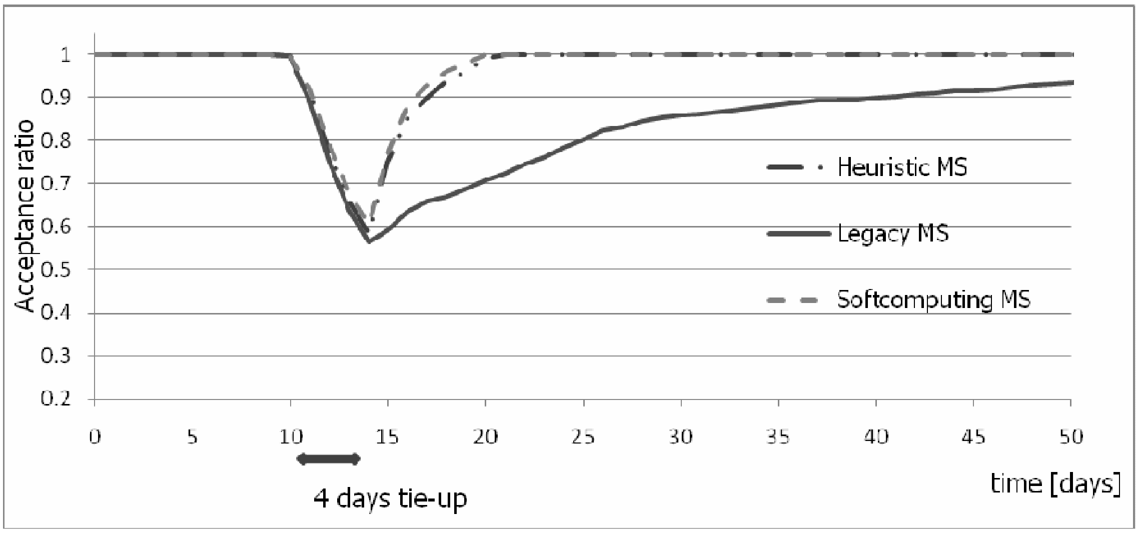


Figure 2. Acceptance ratio for 4 days tie-up vehicles for legacy, heuristic and soft computing management system

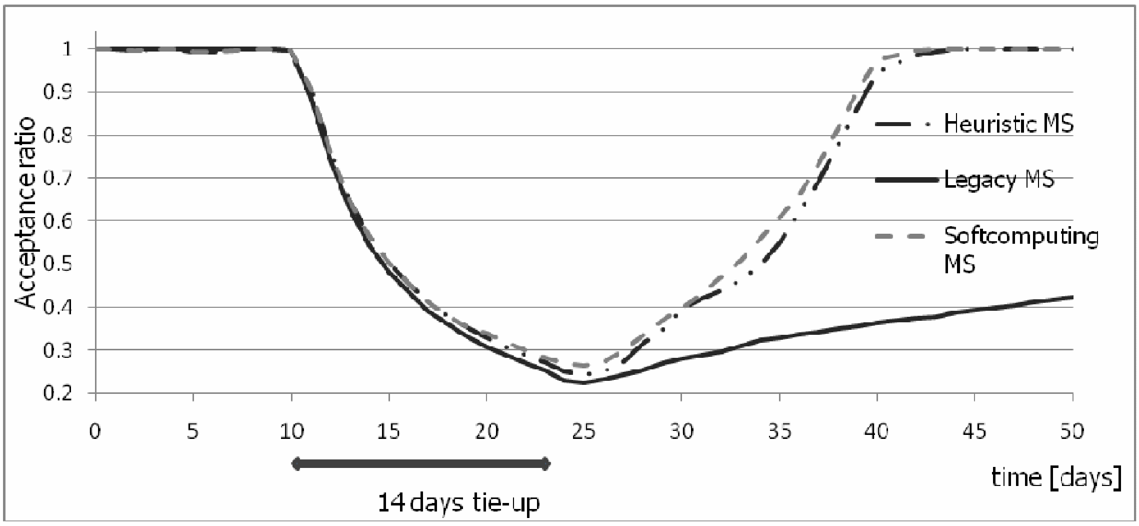


Figure 3. Acceptance ratio for 14 days tie-up vehicles for legacy, heuristic and soft computing management system

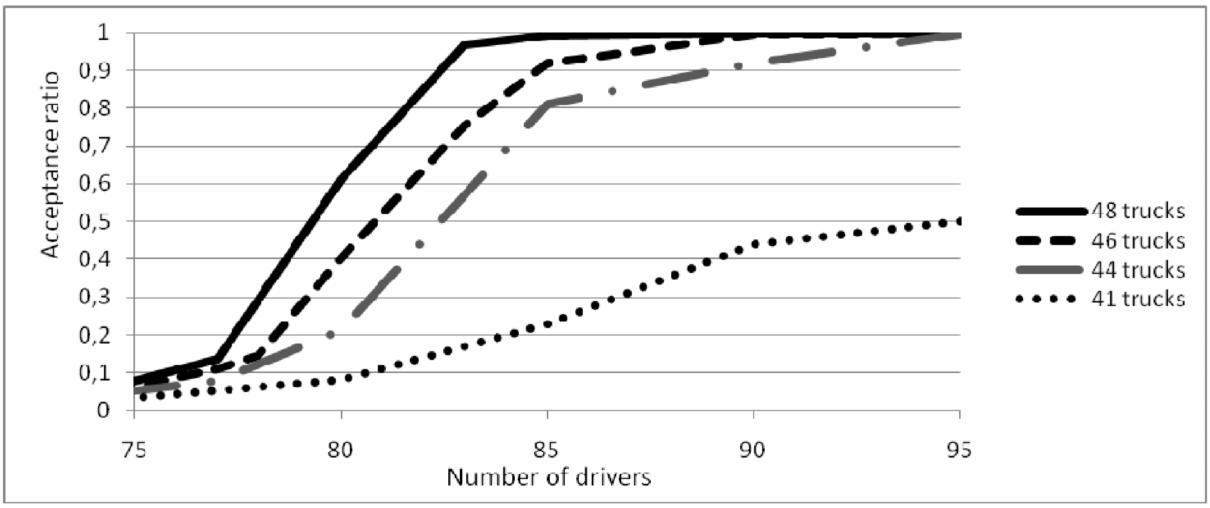


Figure 4. Acceptance ratio in a function of number of drivers for fixed number of vehicles

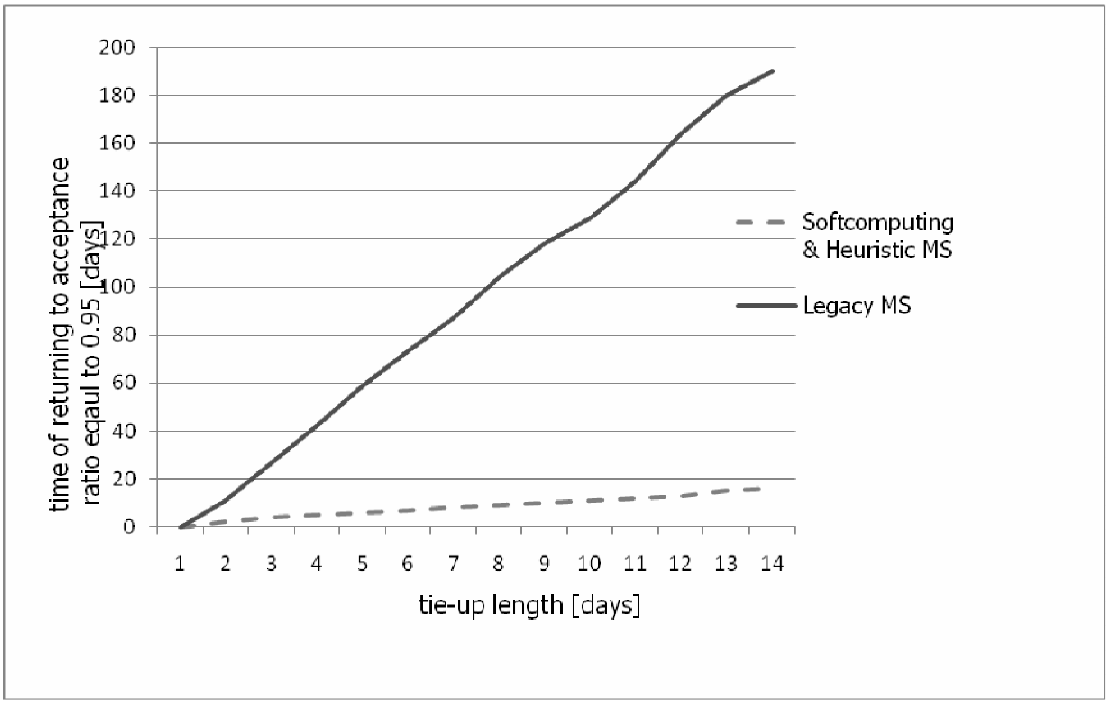


Figure 5. Time of returning *DTS* performance (measured by acceptance ratio) to 0.95 of normal performance after a critical situation (tie-up) ocuurance for legacy, heuristic and soft computing management system

The achieved simulation results look promising. We hope that it could be implemented in real transportation system.

Acknowledgment

Work reported in this paper was sponsored by a grant
No. N N509 496238, (years: 2010-2013) from the Polish Ministry of Science and Higher Education.

References

[1] Barcelo, J., Codina, E., Casas, J., Ferrer, J.L. & Garcia, D. (2005). Microscopic Traffic Simulation: a Tool for the Design, Analysis And Evaluation Of Intelligent Transportation Systems. *Journal of Intelligent and Robotic Systems: Theory and Applications*, Vol. 41, 173-203.

[2] Barlow, R. & Proschan, F. (1996). *Mathematical Theory of Reliability*. Philadelphia: Society for Industrial and Applied Mathematics.

[3] Ben-Akiva, M., Cuneo, D., Hasan, M., Jha, M. & Yang, Q. (2003). Evaluation of Freeway Control Using a Microscopic Simulation Laboratory. *Transportation Research*, Part C (Emerging Technologies), Vol. 11C, 29-50.

[4] Birta, L. & Arbez, G. (2007). Modelling and Simulation: Exploring Dynamic System Behaviour. London: Springer.

[5] Burt, C.N. & Caccetta, L. (2007). Match Factor for Heterogeneous Truck and Loader Fleets. *International Journal of Mining, Reclamation and Environment*, Vol. 21, 262-270.

[6] Duinkerken, M.B., Dekker, R., Kurstjens, S.T.G.L., Ottjes, J.A., & Dellaert, N.P. (2006). Comparing Transportation Systems for Inter-Terminal Transportation at the Maasvlakte Container Terminals. *OR Spectrum*, Vol. 28, 469-493.

[7] Fishman, G. (1996). Monte Carlo: Concepts, Algorithms, and Applications. Springer-Verlag.

[8] Gartner, N., Messer, C.J. & Rathi, A.K. (1998). *Traffic Flow Theory and Characteristics*. In: T.R. Board (Ed.). Texas: University of Texas at Austin.

[9] Gold, N., Knight, C., Mohan, A. & Munro, M. (2004). Understanding service-oriented software. *IEEE Software*, Vol. 21, 71-77.

[10] Ioannou, P.A. (2008). *Intelligent Freight Transportation.* Carolina: Taylor and Francis Group.

[11] Krzyzanowska, J. (2007). The Impact of Mixed Fleet Hauling on Mining Operations at Venetia Mine. *Journal of The South African Institute of Mining and Metallurgy*, Vol. 107, 215-224.

[12] Liu, H., Chu, L. & Recker, W. (2004). Performance Evaluation of ITS Strategies Using Microscopic Simulation. *Proc. of the 7th*

International IEEE Conference on Intelligent Transportation Systems, 255-270.

[13] Sanso, B. & Milot, L. (1999). Performability of a Congested Urban-Transportation Network when Accident Information is Available. *Transportation Science*, Vol. 33, No 1, 10-21.

[14] Taylor, M.A.P., Woolley, J.E. & Zito, R. (2000). Integration of the Global Positioning System and Geographical Information Systems for Traffic Congestion Studies. *Transportation Research*, Part C (Emerging Technologies), Vol. 8C, 257-285.

[15] Vis, I.F.A. (2006). Survey of Research in the Design and Control of Automated Guided Vehicle Systems. *European Journal of Operational Research*, Vol. 170, 677-709.

[16] Walkowiak, T. & Mazurkiewicz, J. (2009). Analysis of Critical Situations in Discrete Transportation Systems. *Proc. of International Conference on Dependability of Computer Systems*, Brunow, Poland, Los Alamitos: IEEE Computer Society Press, 364-371.

[17] Walkowiak, T. & Mazurkiewicz, J. (2008). Availability of Discrete Transportation System Simulated by SSF Tool. *Proc. of International Conference on Dependability of Computer Systems*, Szklarska Poreba, Poland, Los Alamitos: IEEE Computer Society Press, 430-437.

[18] Walkowiak, T. & Mazurkiewicz, J. (2008). Functional Availability Analysis of Discrete Transportation System Realized by SSF Simulator. Computational Science – ICCS 2008, 8th International Conference, Krakow, Poland. Springer-Verlag, LNCS 5101, Part I, 671-678.

[19] Walkowiak, T. & Mazurkiewicz, J. (2010). Algorithmic Approach to Vehicle Dispatching in Discrete Transportation Systems. Technical approach to dependability. Wroclaw, 173-188.

[20] Walkowiak, T. & Mazurkiewicz, J. (2010). Functional Availability Analysis of Discrete Transportation System Simulated by SSF Tool. *International Journal of Critical Computer-Based Systems*, Vol. 1, No 1-3, 255-266.

[21] Walkowiak, T. & Mazurkiewicz, J. (2010). *Soft Computing Approach to Discrete Transportation System Management*. Springer-Verlag, LNAI 6114, 675-682.