A Generalized Stochastic Petri Net Approach for Modeling Activities of Human Operators in Intermodal Container Terminals

Original Citation:
A Generalized Stochastic Petri Net Approach for Modeling Activities of Human Operators in Intermodal Container Terminals / Maione, Guido; Mangini, Agostino Marcello; Ottomanelli, Michele. - In: IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. - ISSN 1545-5955. - STAMPA. - 13:4(2016), pp. 1504-1516. [10.1109/TASE.2016.2553439]

Availability:
This version is available at http://hdl.handle.net/11589/83811 since: 2022-06-07

Published version
DOI:10.1109/TASE.2016.2553439

Publisher:

A Generalized Stochastic Petri Net Approach for Modelling Activities of Human Operators in Intermodal Container Terminals

G. Maione, *Member, IEEE*, A.M. Mangini and M. Ottomanelli

***Abstract*— This paper proposes a Petri net representation of the activities performed by the key human operators for unloading/loading containers in an intermodal maritime container terminal (CT) with a low level of automation. These processes are the core of the export, import, and transshipment cycles executed in the terminal. The aim of the paper is to take into account both the human component and the material handling resources, e.g. cranes and transporters, by defining an accurate model which describes how to coordinate humans and use the system resources necessary for serving vessel or feeder ships. The developed Generalized Stochastic Petri net based model is of limited complexity and verifies important properties, as shown by a dedicated proof. The modular integrated model is tested and validated by simulation of typical and perturbed scenarios of the Taranto container terminal, a real terminal which is taken as a case-study for its complexity and similarity to CTs with multiple transport modes.**

***Note to Practitioners*— This paper proposes a new perspective for modeling, simulating, and controlling container terminals (CT). Usually, the participation of human operators to the unloading/loading processes is neglected. Moreover, few CTs are characterized by a low or partial level of automation. Then, the cooperation between humans and the material handling equipment is important for executing several key tasks with efficiency (with respect to the deadlines), safety and sustainability for humans. The complexity arising from a human-in-the-loop system demands for more accurate models, that can reduce prediction errors and avoid unnecessary extra-equipment costs. Therefore, we propose an approach to explicitly take into account the roles and main functions performed by a typical team of specialized operators. To verify the model accuracy and validate it, we simulate both standard and perturbed conditions of the Taranto CT. Given that operators are usually assigned well-established tasks, the approach can be useful: *a*) to improve the employment of humans and their supervisory activity; *b*) to identify critical or "faulty" situations that involve operators or are caused by their errors and anomalous behaviors, thereby to improve safety and protect the handling equipment (e.g., by identifying delays and message losses in operators' communication); *c*) to increase automation, by reassigning tasks from humans to automatic equipment.**

***Index Terms*— Discrete-event systems, intermodal container terminal, modeling, performance evaluation, simulation, Generalized Stochastic Petri nets.**

I. INTRODUCTION

REAL-TIME managing a Container Terminal (CT) is a complex activity, which aims at minimizing the generalized cost of port agency. This is a key problem for the competing Italian CTs because of the growing freight traffic flows in the Mediterranean area.

This paper defines an accurate model of a typical CT devoted to transshipment, export, and import activities. The aim is to provide a tool for easily and quickly representing, with a certain degree of abstraction, CT processes that are not fully automated, in which the human component is essential. Namely, most of the terminals are not automated and particular processes need human supervision and operations that cannot be executed by automatic equipment. Consequently, the CT efficiency also depends on human operators. Indeed, human activities are of fundamental importance to guarantee safety, efficiency and speed of operations. Their failure can significantly affect the terminal performance or safety of humans and handling equipment. Most of the times human activities are defined according to operational handbooks, and they are committed by terminal planners, sometimes commanded by using modern communication technologies. But they are not monitored and tracked as it is used for machines or handling equipment that are sometimes software controlled. Humans receive commands and communicate final results of their single operations or, most of the times, of the complete task they are assigned.

Then, modeling human activities can help the system managers to monitor and control the whole process, and to achieve better performance. Therefore, a Discrete-Event Systems (DESs) modeling framework can be used to simulate and evaluate the performance of the human activities in order to support the system managers in the planning and organization of the activities. In particular, the model simulations can measure the performance indices of the terminal, i.e., system throughput, resource utilization and can help detect the system bottlenecks. Moreover, a discrete event formalism describing the system dynamics may be useful for testing alternative system solutions and capturing emerging criticalities.

For this reasons, in this paper we propose a Generalized Stochastic Petri Net (GSPN) model that has a twofold utility: *i*)

G. Maione and A.M. Mangini are with the Department of Electrical and Information Engineering, Politecnico di Bari, Bari 70125, Italy (e-mail: guido.maione@poliba.it, agostinomarcello.mangini@poliba.it).

M. Ottomanelli is with DICATECh, Politecnico di Bari, Bari 70125, Italy (e-mail: michele.ottomanelli@poliba.it).

a graphical aspect for allowing a modular design of a single CT and the description of the supply system of intermodal freight transport networks, as part of a more aggregate freight traffic assignment model; *ii*) a well-developed mathematical theory for process analysis. From this view, literature is poor in freight transport supply simulation tools, even if operations in a CT represent most of the total transport cost from one origin to one destination.

The paper is organized as follows. Section II reviews the literature on intermodal CT models, while Section III overviews the typical services provided by a CT, the resources used and processes required. Section IV defines the PNs associated with all the considered human operators. Section V analyzes liveness and boundedness of the proposed PN model. Section VI gives some simulation results for validation and performance analysis. Section VII discusses some properties of the overall interconnected model.

II. LITERATURE REVIEW ON INTERMODAL CONTAINER TERMINALS

A. *Literature Review*

Actually, modelling CTs must describe a complex system of many operations that have to be coordinated so as to minimize the operational time and costs. Hence, technical literature presents many approaches for modelling a single operation, a sequence of operations as well as the operation of the entire CT. A detailed literature review is given in [1-3], whereas the modelling of complete CTs for management and control purposes is considered in [4-6]. The literature reviews highlight two main classes of modelling approaches, namely microscopic and macroscopic approaches [7].

Macroscopic modelling assumes continuous containers flow along the whole sequence of activities. This aggregate representation is suitable for supporting strategic decisions, relevant for designing the system. The most frequent problems in this class are berth planning, marshalling strategies, allocating space and system layout, handling equipment capacity and technologies. Application of linear and non-linear programming for improving the transfer efficiency in multimodal CT [8-10] and the model for the Salerno CT [11] are excellent examples of the macroscopic point of view. However, few papers model human operations because macroscopic approaches are generally addressed to determine the optimal manpower or scheduling policy [12-14]. To sum up, macroscopic methods generally use mathematical programming without explicitly taking into account the crucial role of the specific operations carried out by humans along the complete process at CT.

Microscopic modelling approaches are generally based on discrete-event formalism, and may include Petri nets (PNs) and/or object-oriented approaches as well as simulation based on queuing networks theory. Even if high computational effort may be required, microscopic simulation allows the explicit modeling of each activity within the CT as well as of the whole system. In this way, it is possible to achieve a detailed estimate of performance as consequence of different system designs and/or management scenarios. In particular, microsimulation models proved that advanced technologies and automation may speed up the CT operations [15].

Complete CT microsimulation using an object-oriented approach or a discrete-event formalism is proposed in [16] and in [17-18], respectively. Analogously, a proposal of analytical and discrete-event simulation for berth planning with queuing networks is introduced in [19-20]. Another approach to discrete-event microsimulation concerns PNs [21] that represent a well-established theoretical background for systems modelling. The basic capability of a PN is in the graphical representation and formal analysis of all processes in a discrete event system. PNs are suitable to describe precedence relations, synchronization, mutual exclusion, and many other forms of interaction between concurrent events. When compared with other formalisms, PNs show the possibility to derive formal properties, which allow safe and efficient operation. This can help the system designer in preventing blocking, deadlock, congestion, which can reduce the terminal performance.

The PN modelling and control techniques could also be significant for modelling human behaviors in industries or service companies. For example, in [22] the authors develop a fuzzy attributed PN to represent uncertainty in disassembly process due to a large amount of human intervention. Moreover, in [23] PNs are applied for modelling and controlling a distributed robotic system, in which both humans and computers command robots, and interactions are supervised to avoid collisions and deadlock. Humans are considered also in [24] when using PNs and Java for the supervisory control of human-in-the-loop systems. In [25] PNs are applied to supervise human operations, to prevent errors and to guarantee safety in semiconductor manufacturing.

Our use of PNs is further motivated by past successful applications to logistics [26] and to activities optimization in the CT [27-29]. On the other hand, PNs are also applied for scheduling transport vehicles in an automatic material-handling environment of a manufacturing system [30]. With reference to CTs, PNs describe automated handling and synchronization problems in using shared resources, and analyze faulty situations [31-33]. Furthermore, in [34] a stochastic PN describes containers transfer in a rail-road transshipment yard, and in [35] a timed-place PN models automated guided vehicles in CTs. Moreover, in a recent work, Dotoli *et al*. [5] propose a general modeling framework based on Timed Petri Nets (TPNs) for Intermodal Freight Transport Terminals (IFTTs) in order to test the as-is IFTT performance and evaluate alternative possible to-be improvements in order to identify and eliminate emerging criticalities in the terminal dynamics. Finally, the case of the Taranto CT (TCT) is considered in [36-37], and digraphs and PNs are used as a first attempt to define a precise model for the main processes in a CT.

B. *Paper contribution*

In the literature of the sector both microscopic and macroscopic simulation studies have paid less attention to the problem of operations executed by human workers. However, this is critical in not fully automated CTs, where many activities need to be carried out explicitly by human agents. This paper proposes a GSPN model for the microsimulation of the activities performed by the key human operators during processes for unloading/loading containers in an intermodal terminal. The aim is to take into account both the humans and the material handling resources, i.e. cranes and trailers, by defining a complete microsimulation model which describes how to use and coordinate the system resources. Diversely by the contributions of [12-14] that are particularly addressed on the human operation scheduling, our focus is on human participation and supervision of key activities for the terminal efficiency, then on improving the utilization of humans in a not fully automated CT. Therefore, the proposed representation can be used for simulation and monitoring of real processes in which humans take part, but also for operational control and prevention of particular dangerous events. To this aim, a complete detailed model is needed, in all the composing human activities, which are interconnected, even in complex ways for some details, and which affect one with another.

III. CONTAINER TERMINALS: SERVICE CYCLES, TYPICAL PROCESSES AND RESOURCES

A CT is an interconnection node or hub in which different transport modes intersect. The connection between the different modes using road networks, railways, and sea networks, gives rise to *intermodality*. Goods are usually transported and delivered in TEUs (Twenty Equivalent Units). A TEU is a container 20 feet long, 8 feet wide, and 8 feet high. Containers arrive to or depart from a CT by ships, trucks or trains. Vessel ships travel on long distances, coming from (or going to) far-away ports. Feeder ships travel on short distances, coming from (or going to) close ports. We may distinguish between: *i*) an *import cycle*, when TEUs arrive on a vessel/feeder ship and depart by trains/trucks, corresponding to a transition from sea to railway/road modes; *ii*) an *export cycle*, when TEUs arrive by trains/trucks and depart on a vessel ship, for a transition from railway/road to sea mode; *iii*) a *transshipment cycle*, when TEUs arrive and depart by ship, without mode change: in this case, they are moved from vessel to feeder ships for close destinations or from feeder to vessel ships for far destinations.

To execute these cycles, the companies managing CTs provide several services with the following processes:

- *loading* (i.e. embarking TEUs from internal trailers to vessel/feeder ships) and *unloading* (i.e. removing TEUs from vessel/feeder ships to internal trailers);
- *quay-yard transporting* (i.e. moving TEUs from quayside (ships) to yard, and backwards);
- *train-loading and train-unloading* (i.e. putting/removing TEUs on/from trains);
- *railway-yard transporting* (i.e. moving TEUs from railway area (trains) to yard, and backwards) and *gate-yard transporting* (i.e. moving TEUs from road-gate to yard, and backwards);
- *truck-loading and truck-unloading* (i.e. putting/removing TEUs on/from trucks);
- *stacking* TEUs in yard blocks and *marshalling*, i.e. redistributing TEUs between blocks.

Several resources and equipment are available to execute the above processes. Basically, the dedicated or shared resources used to handle and transport TEUs are:

- different types of cranes and storage and retrieval systems: quay cranes (**QC**s for brevity), yard cranes (**YC**s for brevity), railway cranes, jolly mobile cranes; reach stackers (**RS**s for brevity); side loaders (**SL**s for brevity); automatic stacking cranes and other storage and retrieval systems;
- transport vehicles: trailers (**TR**s for brevity), multi-trailers, automatically guided vehicles, straddle carriers;
- space resources: slots in yard blocks, where TEUs are distributed and stacked; other infrastructures like quays and berths, lanes for internal transport, gates, railway tracks;
- skilled human operators for executing the processes.

All processes and resources must be planned, scheduled, monitored, and controlled so that the terminal provides services in time with short delays in transport and delivery.

A. *Import, export and transshipment cycles*

Each import, export, and transshipment cycle is a sequence of macro-operations that terminal resources execute on containers. In an import cycle using road network, **QC**s unload TEUs from a berthed ship to **TR**s, which transfer cargo from quay to yard blocks. Here, **YC**s pick up TEUs from **TR**s and stack them in assigned positions. TEUs stay there for a certain delay time while waiting for their destination; sometimes, they are relocated by **YC**s in a more proper position, according to a marshalling procedure, which may use **TR**s and cranes to move TEUs between blocks. The process planning must avoid accumulation of too many TEUs in a specific yard area, to prevent a large amount of work for the **YC**s and congestion of **TR**s.

After the marshalling procedure, **RS**s load TEUs from yard blocks to trucks, which exit from the main CT gate. Similarly, in an import cycle using the railway system, **YC**s pick up TEUs from blocks and load them on **TR**s moving from yard to the railway connection, where special railway cranes pick up TEUs to put them on departing trains.

In an export cycle using road network, the sequence goes in the opposite direction: trucks enter the gate, and directly deliver TEUs to yard blocks. Then, TEUs are stacked by **RS**s. After a marshalling or delay time, **TR**s move TEUs to the quay section where the ship is berthed. Here, **QC**s load them on vessel ships. Similarly, in a railway export cycle, TEUs are moved from trains to vessel ships.

In a transshipment cycle, when a vessel ship arrives, TEUs are unloaded to **TR**s by QCs, transferred to yard blocks by **TR**s, picked up and stacked by **YC**s. After a possible marshalling and/or a delay in their position, TEUs are picked up and transferred to the quay area, where they are loaded on a feeder ship. The opposite process may occur when a feeder ship arrives and a vessel ship departs. In TCT, about 90% of the total traffic is currently of transshipment type.

In all import, export, and transshipment cycles humans are required and play an important role in guaranteeing fast operations and reducing problems caused by perturbations.

B. Human resources and their tasks

Several specialized operators work to fulfill different tasks for moving TEUs in the terminal environment, according to pre-specified rules and methods. Apart office operators working in planning activities, we focus on the critical operators working in the quay or yard area. Operators communicate by means of Personal Digital Assistants (PDAs). The current location of TEUs is stored in central databases, and moving and stacking strategies are managed by a central supervisory system.

In particular, the activities executed for unloading/loading TEUs from/on the ships are fundamental for the terminal efficiency, i.e. for guaranteeing fast service and reducing delays. We refer to the TCT case in Taranto, where, for each active **QC**, a team of 25 people is involved to handle containers: 1 foreman, 2 checkers, 2 quay crane operators, 4 lashers, 5 trailer drivers, 8 yard crane operators, 1 yard checker operator, 1 reach stacker driver, 1 side loader driver.

The *foreman* (***Fo*** for brevity) is the head of the team, who receives instructions and messages from the ship planner (***SP*** for brevity), to start embarking or landing of coded TEUs. He sends the TEU identification code to checkers and lashers. Moreover, he compiles reports to the planning team, for example when a damage occurs to an unloaded TEU.

Checkers monitor the process: the first one, ***Ch1***, works inside the kiosk of the **QC**, retrieves (or stores) and verifies information from database, and communicates the position of the TEU to be unloaded/loaded to the crane operator; the second checker, ***Ch2***, is located below the crane, verifies external conditions and type of TEUs, and makes TEUs to be embarked ready to be fixed or sets them free from fixtures.

The first quay crane operator, ***Qo1***, operates the crane to pick up the TEU from a **TR** (or a ship cell) and put it in a ship cell (or a **TR**), during the embarking (or landing) cycle; the other, ***Qo2***, works on board to help his colleague during maneuvers and eventually substitutes him.

A lasher (***La*** for brevity) blocks/unblocks TEUs on/from the ship by using special fixtures. He receives commands from ***Fo*** to fix (or free) TEUs to (or from) their blocked positions in the ship hold or cover.

A trailer driver (***Td*** for brevity) delivers TEUs from/to the yard blocks to/from the quay crane. His tasks are established by the yard planner (***YP*** for brevity). He always waits a start-signal from ***Ch2***, after receiving/delivering the TEU from/to the **QC** and before leaving for the yard.

A yard crane operator (***Yo*** for brevity) picks up TEUs from trailers or trucks and deposit them in a block, or picks up from the block and loads them on transport vehicles. He is involved in marshalling activities.

The yard checker operator (***Yc***) manages a parking area for trailers: he opens it and registers the trailers going in or going out, then he closes the parking. Finally, besides verifying the efficiency of **TR**s, **QC**s and **YC**s, he schedules rosters between operators and vacation.

IV. PETRI NET MODELLING OF HUMAN OPERATIONS

As previously stated, CTs have several resources, basically material handling equipment (i.e. different kinds of cranes and transporters) and human resources. The critical resources for terminal efficiency, as shown by common experience, are cranes and trailers, which are all driven by human operators. The processes executed in a terminal are determined by unloading/loading/transshipment cycles, as described in Section III. Then, PNs can be used to represent both the processes and the resources required to execute them. The literature distinguishes (as remarked in [38], for automated manufacturing systems) between process-oriented models [39-41], and resource-oriented models [42-43]. The first approach is more widely applied because of its clarity and simplicity. It requires modeling the sequence of operations in each process (by the so-called operation places), the resources available (by the so-called resource places), and the resource requirements for each operation (by links between places). The drawback is the increase of size and complexity of the models. The second approach is based on modeling the resources, then defining the subnets associated with the considered processes, and finally merging the subnets. That means introducing a unique place for each resource, with no operation place, which gives rise to a more compact representation for control purpose. It is known that modeling can be done by following a top-down design methodology, i.e. a stepwise refinement or a bottom-up design methodology, i.e. a modular composition [44].

The approach herein used to describe human intervention in unloading/loading processes can be compared to that for industrial processes, decomposable into parallel subprocesses, each modelled by a PN [45]. The PN modelling all the human activities in the hub can be defined by integrating different modules, each associated with a specialized operator. A module can be easily defined by the standard construction rules of PNs. Here, we developed the modules for all cited operators, which can be easily connected together by input or output messages exchanged between operators. Finally, PNs show many advantages [44]: modeling is easy, the controller can be directly generated from the graphical representation, validation and performance analysis is possible by mathematically based computer tests, the simulation is driven from the model, the status information from distribution of tokens allows real-time monitoring.

A. Notations and assumptions

A marked Petri net is a bipartite graph formalized by a five-tuple $PN=(P, T, A, W, M_0)$, in which nodes correspond to *places* in set P and *transitions* in set T, respectively. Directed edges (or arcs) are elements in set A and link places and transitions together [21].

Each place, graphically depicted as a circle, represents a condition. Usually, places denote the execution of activities or the availability of resources, where each resource has a finite capacity, i.e. a finite number of units. A *token* (black dot) marking a place indicates the truth of the condition associated to that place, or an available resource unit. Transitions, shown as bars, represent events changing the state of the modelled system. Input arcs are oriented from places to transitions and output arcs from transitions to places. All arcs are drawn as arrows and represent how tokens flow through the net. The arc *weight* corresponds to the number of tokens flowing through the arc. Namely, each arc is labelled by a weight function $W:A \to \{1,2,\ldots\}$, specifying how many tokens flow through the arc. Unity weights are omitted.

Given a PN, for each place $p \in P$ the following sets of transitions may be considered: $°p$ represents the set of the input transitions to place p and is named pre-set of p; while $p°$ represents the set of the output transitions from p and is named post-set of p. Analogously, for each transition $t \in T$ the following sets of places may be considered: $°t$ denotes the set of the input places to transition t and is named pre-set of t; while $t°$ characterizes the set of the output places from t and is named post-set of t.

Input places to a transition t ($p \in °t$) are associated to the pre-conditions for that event to occur, or to the resources necessary for a certain task. Output places from t ($p \in t°$) are associated to the post-conditions consequent to the event, or the resources released after the task. Pre-conditions enable t, post-conditions are verified after the occurrence of t.

The distribution of tokens in the net defines the system state, and transitions trigger the state changes. The state is associated with a marking vector $M:P \to \{0,1,2,\ldots\}$, where $M(p_i)$ gives the current number of tokens in a marked place $p_i \in P$. Vector M_0: $P \to \{0,1,2,\ldots\}$ specifies the initial state (marking), and $R[M_0]$ indicates all the states reachable from M_0. The states change according to two basic rules:

a) *Enabling rule*: a transition t is enabled if each of its input places p is marked with at least $w(p,t)$ tokens, where $w(p,t)$ is the weight of arc from p to t.

b) *Firing rule*: a firing of an enabled transition t consumes $w(p,t)$ tokens from each input place p and produces $w(t,p)$ tokens for each output place p, where $w(t,p)$ is the weight of arc from t to p. By assumption, only one transition fires at a time.

In addition, to allow the representation of a decision, choice or conflict, a place may be connected to more than one output transitions, which, in this way, represent nondeterministic events, each associated with a probability or possibility of occurrence. Transitions representing choices are identified by subscripts with letters (e.g. t_{3A} and t_{3B} in Fig. 1). Conflicts are solved by external decision supports to select the transition to be fired. The external decision may use intrinsic information obtained from PN analysis or may be based on other parameters, such as priority or probability of transitions. Selection by means of probabilities defines random switches (see [46] and references therein). In this paper we define probabilities from statistical observed occurrence of the related events (e.g. in case of malfunctions to cranes). Therefore, each conflicting transition is associated to a probability r_i, which labels the transition, such that $\sum r_i = 1$.

We consider the operations cyclically executed by human operators in the same way as production steps sequentially executed by workstations in manufacturing systems. Then, human activities are specified by a sequence of interleaved transitions and places. In this sense, each place is related to a logical condition of the operator, or to the availability of a resource. Moreover, each timed transition corresponds to a step that needs time to be executed in the operator cycle. Each immediate transition occurs instantaneously, and it is associated to a logical state change with zero firing time.

This common interpretation allows us to build up a GSPN [46], which is useful to evaluate the performance of the modelled processes and which has been already used in logistic applications [26-27]. This class of PNs is defined as GSPN = (P, $T_I \cup T_E$, A, W, M_0, FIR), which comes from PN when T is partitioned in a set T_I of immediate transitions (sharp bars) and a set T_E of timed exponential transitions (thick bars). Moreover, $FIR:R[M_0] \times T_E \to \Re^+$ is a *firing function* for exponential transitions [47]. The firing time of a transition $t \in T_E$ is a continuous random variable with exponential distribution. Then, for each $t \in T_E$ in each $M \in R[M_0]$, the real number $FIR(M,t)$ establishes the rate of the firing of t (each $t \in T_I$ has zero firing time in all markings). The mean firing time of an exponential transition represents the average duration of the considered process.

Adding time to ordinary Petri nets in the firing of transitions allows us to measure performance. In particular, if the exponential distribution is used, the resulting stochastic Petri net can be transformed into an equivalent Markov chain but the net allows a more compact representation of a complex system. On one side, it is known that Markov chains typically allow computation of performance indices, like throughput, average resource utilization, probability that a resource is blocked or starved, expected time to complete a process [21, 44]. On the other side, GSPNs are one of the most used tools to simulate and evaluate performance of systems with concurrent and synchronized processes (e.g. flexible manufacturing systems, distributed computing systems, local area networks, communication protocols, etc.). In our case, the available data from TCT transport processes are enough to justify the exponential assumption. In particular, the performance can be measured by the throughput of some transitions showing how many TEUs are served in the time unit, how many TEUs flow through the terminal, how many are stored, how much time is needed to unload/load a ship, etc.. Moreover, performance of human operators can be also measured by recording time they spend to complete the processes. In the next subsections, a different GSPN module will be defined for describing the activities and checks made by every operator.

B. *Petri net of the foreman, **Fo***

Fig. 1 depicts the foreman initial state, before starting his operation in an unloading/loading task. In the sequel, we describe how the flow of tokens in the PN of Fig. 1 models his activities. Places named p_{QC}, p_{Ch}, p_{Q1}, and p_0 are duplicated for sake of clarity, but the two copies should be considered one time only.

When ***Fo*** receives information from ***SP*** about the TEUs to be loaded/unloaded, he is ready to start and sends messages to ***Ch1***, ***Ch2***, and ***Qo1*** (t_1). Information includes the ship arrival time, the ship mooring location, the unloading/loading plan, the working schedule, the **QC** to manage. No other foreman will have this crane assigned for contemporaneous unloading/loading operations. When all information is available (p_1), and if the **QC** is available and ***Ch1***, ***Ch2***, and ***Qo1*** are ready, ***Fo*** supervises the setting of the **QC** in the proper position (t_2), with the help of checkers (arc from place p_{Ch} to t_2 has a weight of 2). Now, he is ready to verify the functionality of the **QC**: if he detects a normal condition ($t_{3\mathrm{A}}$), he can proceed further, otherwise ($t_{3\mathrm{B}}$) he begins waiting for the restoration (t_4) until the crane is repaired.

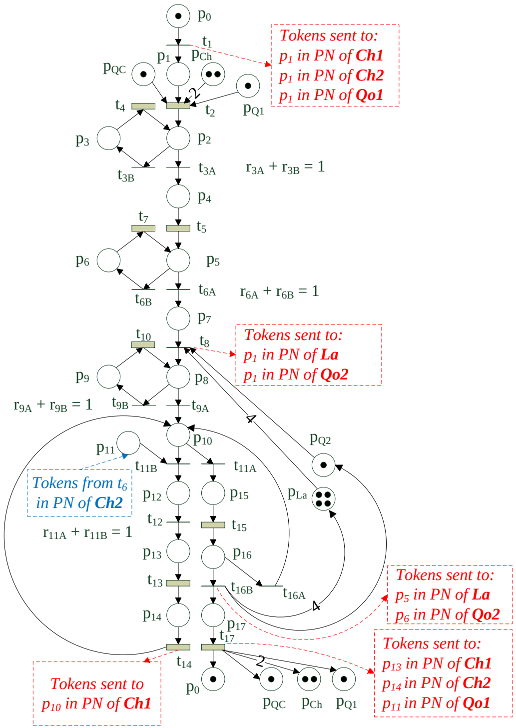


Fig. 1. GSPN of the foreman, ***Fo***.

Then, ***Fo*** recovers all necessary tools to bring them close to the **QC** (t_5). Tools are necessary for freeing (fixing) the unloaded (loaded) TEUs. After that, he is ready for a sanitary test which is sometimes performed on sampled TEUs. If the check is positive ($t_{6\mathrm{A}}$), he can proceed with operations on board, otherwise some problems are detected ($t_{6\mathrm{B}}$) and he must wait a solution (t_7) until normal conditions are restored.

If ***Qo2*** and four lashers are available (arc from p_{La} to t_8 has a weight of 4), ***Fo*** commands them to go on board (t_8). Then, he may check if the bay plan by ***SP*** matches the real distribution of TEUs: if everything is ok ($t_{9\mathrm{A}}$), he may start the supervision; if some problem is identified ($t_{9\mathrm{B}}$), then he redistributes tasks to the cranes (t_{10}). After the unloading/loading supervision is started, an ordinary working condition may be verified ($t_{11\mathrm{A}}$) or a damage, fault or accident may be identified ($t_{11\mathrm{B}}$) if signals are received from ***Ch2*** (token in p_{11}). In the first case, ***Fo*** waits for the duration of the unloading/loading process (t_{15}), after which he may continue his activity ($t_{16\mathrm{A}}$) or terminate it and command ($t_{16\mathrm{B}}$) ***Qo2*** and the four lashers to get off (arc from $t_{16\mathrm{B}}$ to p_{La} has a weight of 4). At the end of the supervision, he follows the parking of the **QC** and sends stop-signals (t_{17}) to ***Qo1*** and checkers ***Ch1*** and ***Ch2*** (arc from t_{17} to p_{Ch} of weight 2). All operators are available again and the crane will be ready for new operations (arc from t_{17} to p_{QC}).

After a fault detection, ***Fo*** sends a message to ***SP*** or to first-aid or safety team (t_{12}), and he goes to the point in which the problem occurred (t_{13}). Here, he collects data, compiles a report to ***SP***, and warns ***Ch1*** to set the TEU aside (t_{14}).

Note that, to analyze terminal efficiency, the cumulative number of times t_{14} and $t_{16\mathrm{A}}$ are executed gives the number of handled TEUs, while the time elapsed between the start of t_1 and the completion of t_{17} gives the time spent by the foreman to serve a ship. Therefore, the developed model can be useful to measure a system performance (throughput of TEUs) and an index of the operator's efficiency.

C. Petri net of the first checker, Ch1

The GSPN module of Fig. 2 depicts the on-board activities normally executed by ***Ch1***. Even if places and transitions have the same consecutive enumeration as in Fig. 1, they refer to different conditions and events. Places p_{QC}, p_{Q1}, p_{TR}, p_{Td}, p_0, and p_2 are replicated.

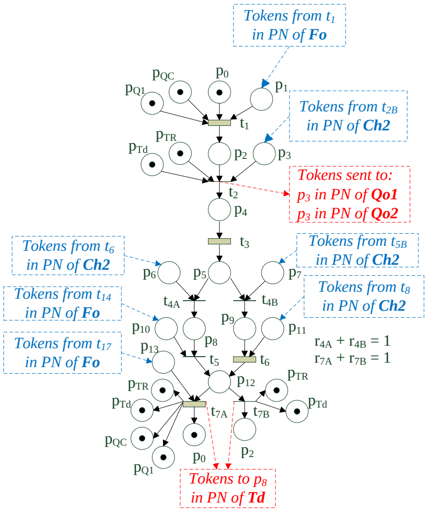


Fig. 2. GSPN of the 1st checker, ***Ch1***.

The foreman sends a signal to start unloading or loading TEUs. Then, ***Ch1*** is ready to check data of the handled TEU (p_1). If the **QC** is available and operator ***Qo1*** is ready, ***Ch1*** goes up into the crane and checks data (t_1). If a **TR** is available and the driver ***Td*** is ready, and if a start signal is received from ***Ch2*** (token in p_3) when the **TR** is in position, then ***Ch1*** signals the TEU location to ***Qo1*** and warns ***Qo2*** (t_2). Then, ***Qo1*** knows the ship location from which the TEU is picked up (unloading) or where it is stowed (loading). At this point, ***Ch1*** waits for the completion of the process (t_3).

After the supervision is finished, if a damage alarm about an unloaded TEU is received from ***Ch2*** (token in p_6), ***Ch1*** acknowledges it ($t_{4\mathrm{A}}$). When ***Fo*** completes his report and commands to transfer the damaged TEU (token in p_{10}), ***Ch1*** signals to ***YP*** the need to transfer it in a different area (t_5). In this case, ***Ch1*** does not insert data in the information database.

After the supervision is finished, if ***Ch1*** receives an ok-signal from ***Ch2*** (token in p_7), he acknowledges that everything is ok

(t_{4B}). When ***Ch2*** sends the seal data (token in p_{11}), then ***Ch1*** inserts this information in the database (t_6). When ***Ch1*** is ready to start the **TR** (token in p_{12}), if ***Fo*** signals the end of all operations (token in p_{13}), ***Ch1*** acknowledges it and gives authorization to ***Td*** to start and may get off (t_{7A}). Otherwise, ***Ch1*** simply gives authorization to ***Td*** to start and prepares for the next task (t_{7B}).

The time elapsed between the start of t_1 and the completion of t_{7A} gives the time spent by ***Ch1*** to complete all the assigned operations, as an index of the operator's efficiency.

*D. Petri net of the first checker, **Ch2***

Fig. 3 depicts the GSPN module for the activities executed by the second checker ***Ch2***, below the **QC**. Places p_{Q1}, p_{TR}, p_{Td}, p_0, and p_1 are duplicated.

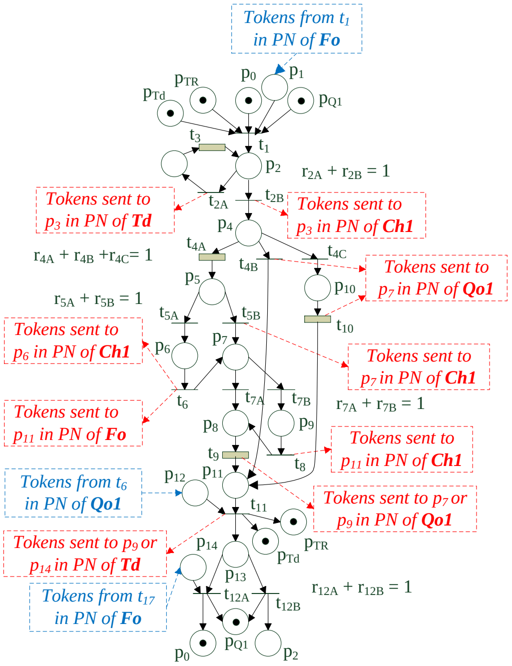


Fig. 3. GSPN of the 2nd checker, ***Ch2***.

Ch2 receives a signal from ***Fo*** to start checking operations executed on trailers. He is active both when TEUs are unloaded on trailers directed to the yard, and when TEUs are loaded from trailers on ships. If a **TR** is available, and if the trailer driver ***Td*** and the first crane operator ***Qo1*** are ready, ***Ch2*** moves to the area where the trailer stops (t_1). If ***Ch2*** verifies that the **TR** is not in a correct position below the quay crane (t_{2A}), then he blocks ***Qo1*** and gives directions to ***Td*** to adjust the position (t_3). Blocking of ***Qo1*** is achieved when transition t_2 in the PN of ***Qo1*** is disabled by marking place p_4 (see Fig. 4); this place is emptied when transition t_3 is executed in the PN of ***Ch2***, so that t_2 is again enabled in the PN of ***Qo1***. If the **TR** is correctly positioned, ***Ch2*** informs ***Ch1*** (t_{2B}). Then, he waits for the unloading to be completed (t_{4A}). Or he makes sure that automatic loading is possible (blocking tools are not necessary) and starts ***Qo1*** to pick up the TEU (t_{4B}). Or he may verify the need of blocks when manual loading is required (t_{4C}); in this case, he first fixes the blocks and then starts ***Qo1*** (t_{10}), who may in turn start picking up the TEU.

After unloading is finished, if a damage is detected on the TEU (t_{5A}), ***Ch2*** sends a signal to ***Fo*** and ***Ch1*** (t_6), otherwise only ***Ch1*** is warned (t_{5B}). After, an empty TEU can be recognized (t_{7A}) or a full TEU is identified (t_{7B}), in which case ***Ch2*** reads and communicates the seal to ***Ch1*** (t_8). Then, ***Ch2*** removes the blocks and warns ***Qo1*** (t_9), who can leave the unloaded TEU or, if it is damaged, pick up and put it in a safe position. If ***Qo1*** alerts the finish of the unloading/loading (token in p_{12}), ***Ch2*** starts the ***Td*** (t_{11}) to the yard for stacking the TEU or for picking up another TEU to be loaded. Finally, ***Ch2*** checks if the scheduled operations are over (t_{12A}) or not (t_{12B}). The first condition occurs when ***Fo*** stops the procedure (token in p_{14}).

The time elapsed between the start of t_1 and the completion of t_{12A} gives the time spent by ***Ch2*** to complete all the assigned operations, as an index of the operator's efficiency.

*E. Petri net of the first quay crane operator, **Qo1***

Fig. 4 depicts the GSPN for the activities executed by the first crane operator ***Qo1*** (p_{QC}, p_{Q2}, p_{C2}, p_0, and p_1 are duplicated).

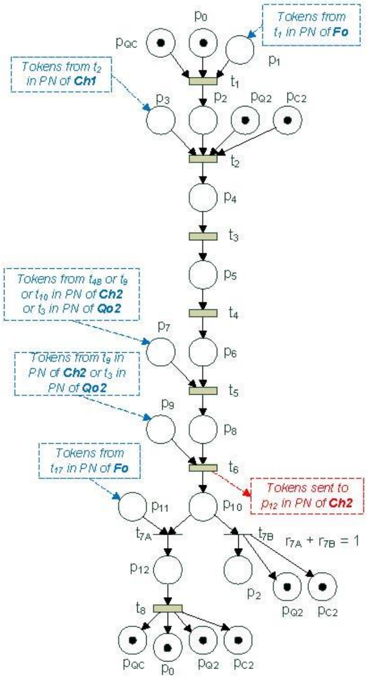


Fig. 4. GSPN of the 1st crane operator, ***Qo1***.

If ***Fo*** starts unloading/loading operations (token in p_1), and if the **QC** is available, then ***Qo1*** can get into the crane (t_1). Then, ***Qo1*** can move the crane (t_2) to the ship bay where TEUs are stored (before unloading) or to the trailers bringing the TEUs (before loading), if all the following conditions are met: the second crane operator, ***Qo2***, and the second checker ***Ch2*** are ready; ***Ch1*** gives information on TEUs to be unloaded/loaded from a ship bay/trailer to a trailer/ship bay (token in p_3).

Then, ***Qo1*** opens and sets the so-called crane spreader on the TEU (t_3) and hooks it up (t_4). If the second crane operator ***Qo2*** (***Ch2***), who is located on the ship (below the crane), gives the ok for the unloading (loading) operation, i.e. a token marks p_7, then ***Qo1*** picks up and handles the TEU (t_5) from the ship bay to the trailer (from the trailer to the bay). When ***Ch2*** (***Qo2***) gives the ok-signal to unhook the unloaded (or loaded) container, i.e. a token is in p_9, ***Qo1*** may drop it and alert ***Ch2*** when the task is finished (t_6). Finally, ***Qo1*** verifies if handling operations are over (t_{7A}) or not (t_{7B}). If ***Fo*** stops the activity (token in p_{11}), then ***Qo1*** can raise and park the crane in a fixed rest position and gets off (t_8).

The time elapsed between the start of t_1 and the completion of t_8 gives the time spent by ***Qo1*** to complete all the assigned operations, as an index of the operator's efficiency.

*F. Petri net of the first quay crane operator, **Qo2***

Fig. 5 (a) depicts the GSPN for the activities executed by the second crane operator ***Qo2*** (p_0 and p_1 are duplicated).

Qo2 helps ***Qo1*** to execute safely all operations on board. To this aim, he knows the ship unloading/loading plan. He cooperates with ***Qo1*** and ***Ch1*** for the task of handling containers, which is physically executed by ***Qo1***. His main function is giving a signal to ***Qo1*** when the TEU is properly hooked, for unloading it from the ship, or when the TEU is in the correct location, for loading it into the ship.

At the beginning, he receives a signal from ***Fo***, so he may go on board (t_1). Then, if ***Ch1*** gives the start command (token in p_3), ***Qo2*** begins coordinating the crane moves for the unloading/loading operation (t_2). When the crane is in correct position, to pick up or drop the TEU, ***Qo2*** sends a message to ***Qo1*** (t_3). Finally, he may check that the operations are finished (t_{4A}) or that they need to proceed (t_{4B}). In the first case, he gets off leaving the ship (t_5).

The time elapsed between the start of t_1 and the completion of t_5 gives the time spent by ***Qo2*** to complete all the assigned operations, as an index of the operator's efficiency.

*G. Petri net model of the lasher, **La***

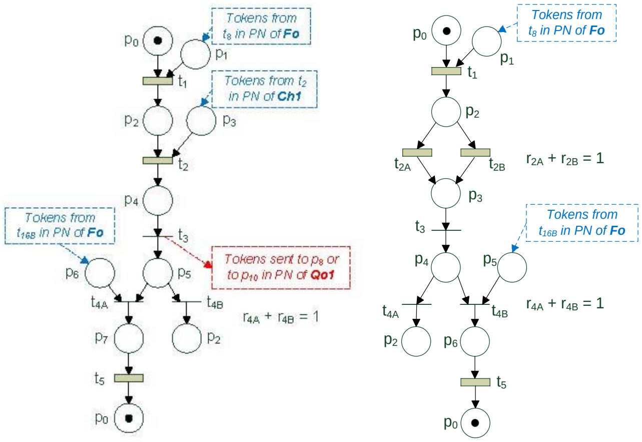
Fig. 5(b) shows the model of all the four lashers (p_0 and p_2 duplicated). When a lasher ***La*** receives a command from ***Fo***, he goes on board to the ship bay which was indicated by ***Fo*** (t_1). In case the bay is located in the ship hold, ***La*** operates to unblock the TEU to be unloaded (t_{2A}); in case the bay is in the ship cover, the lasher operates to unblock the TEU if it is unloaded, or to block it when it is loaded (t_{2B}). Then, he communicates the end of operation to ***Fo*** (t_3). If ***La*** verifies that unloading/loading is not completed (t_{4A}), he keeps going. Otherwise, if he detects that the process is completed (t_{4B}), upon receiving a signal from ***Fo*** (token in p_5), he leaves the ship (t_5).

The time elapsed between the start of t_1 and the completion of t_5 gives the time spent by ***La*** to complete all the assigned operations, as an index of the operator's efficiency.

*H. Petri net of the trailer driver, **Td***

The trailer driver ***Td*** cooperates to both unloading and loading processes (see Fig. 6 (a) with p_{TR}, p_0, and p_5 duplicated).

In unloading TEUs from a ship, he first travels to the quay with no cargo on the **TR**; he arrives below a **QC** and adjusts the trailer position with the help of ***Ch2***. Then, he waits for the TEUs coming from the ship through the **QC** and checks the weight. Being ready, he waits that ***Ch2*** reads the seal and gives him authorization to leave for the yard. Then, he transports cargo from quay to a yard block. Here, he arrives below a **YC** and waits that this crane picks up the TEUs. Finally, he waits the authorization from ***Yc*** to leave, and goes back to the quay or to the parking area.



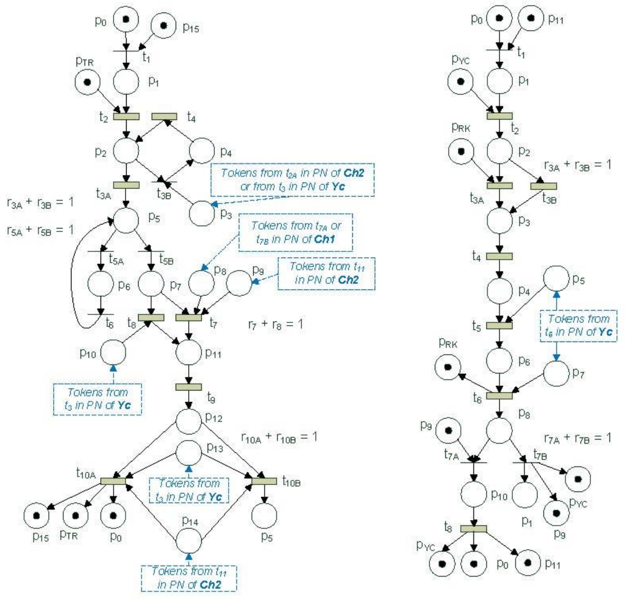
(a) (b)

Fig. 5. GSPN of the: (a) 2nd crane operator, ***Qo2***, (b) the lasher, ***La***.

In loading TEUs onto a ship, ***Td*** operates in the opposite way. He travels to the yard block with the **TR** empty, he arrives below a **YC** and sets the trailer position with the help of ***Yc***. He waits for the TEUs coming from the yard block through the **YC** and checks the weight. After receiving authorization from ***Yc***, he transports cargo from yard to the quay. Here, he arrives below a **QC** and waits for crane pick up. Finally, he waits for the authorization from ***Ch2*** to leave, and goes back to the yard or to the parking area. So, after receiving transport commands from ***YP*** (t_1), if the **TR** is available, ***Td*** drives it (t_2) to the side ***YP*** commanded: to the quay if taking unloaded TEUs is required, to the yard if picking TEUs to be loaded is planned. When the **TR** is below the **QC** or **YC**, ***Td*** may receive a signal (t_{3B}) from ***Ch2*** or ***Yc*** indicating an incorrect position (token in p_3), in which case he starts a maneuver to set the **TR** (t_4). When the position is correct, he waits (t_{3A}) for receiving the TEU coming from the ship or from the yard block.

When he has the TEU on, ***Td*** verifies if data on his PDA match the load: if a mismatch is visualized (t_{5A}), he notifies ***YP*** (t_6); otherwise, if no anomaly is detected (t_{5B}), he may proceed. In case of unloading, upon receiving the authorization from ***Ch1*** (token in p_8) and from ***Ch2*** (token in p_9), he transports from the quay to the yard block (t_7); in case of loading, upon receiving the start input from ***Yc*** (token in p_{10}), he transports from the yard block to the quay (t_8). After being arrived to destination, he waits for the hooking and picking up by the **YC** or **QC** (t_9). Finally, after receiving a start input from ***Ch2*** (token in p_{14}) or ***Yc*** (token in p_{13}), he may verify the end of activity, then go and park the **TR** (t_{10A}), or drive back to the quay or yard starting place to continue activities with another task (t_{10B}).

The time elapsed between the start of t_1 and the completion of t_{10A} gives the time spent by ***Td*** to complete all the assigned operations, as an index of the operator's efficiency.



(a) (b)

Fig. 6. GSPN of the: (a) trailer driver, ***Td***, (b) the yard crane operator, ***Yo***.

*I. Petri net model of the yard crane operator, **Yo***

The yard crane operator ***Yo*** (see Fig. 6 (b), with p_{YC}, p_{RK}, p_0, and p_1 duplicated) works in loading processes to move TEUs from yard blocks onto trailers or trucks. Trailers go to berthed ships when TEUs are exported or transshipped, trucks exit the terminal gate when an import cycle is served. ***Yo*** works also to move unloaded TEUs from trailers coming from a ship to a yard block, or from entering trucks in an export cycle. He also works in marshalling to change the position of TEUs in a block.

Yo starts his activity when he receives (loading/unloading/ marshalling) commands from ***YP*** (t_1). If the **YC** is available, he moves the crane (t_2) to the column of the block where the TEU is located before being loaded or repositioned; or to the column where the unloaded TEU is going to be stored. Note that in both cases a trailer or a truck is ready close to the column to receive or leave the TEU.

If the trailer/truck is available, the operator shifts the crane to pick up or put the TEU in its dedicated slot (t_{3A}). Otherwise, he may operate a crane translation (t_{3B}) in a marshalling procedure. Then, he may hook the container (t_4). Consequently, if the yard checker operator ***Yc*** gives the ok (token in p_5), he picks up the TEU from the slot or from the trailer/truck and puts it on the trailer or in the slot (t_5). If ***Yc*** allows it (token in p_7), ***Yo*** unhooks the TEU for loading/unloading/marshalling (t_6). At this point, ***Yo*** is ready for the next handling or for repositioning the TEU. Then, if ***Yo*** receives a signal for the end of unloading/loading/marshalling processes from ***YP*** (token in p_9), he detects it (t_{7A}) and parks the **YC** (t_8); otherwise, he detects the processes are not completed (t_{7B}).

The time elapsed between the start of t_1 and the completion of t_8 gives the time spent by ***Yo*** to complete all the assigned operations, as an index of the operator's efficiency.

*J. Petri net model of the yard checker operator, **Yc***

After receiving indications from ***YP*** (t_1), this operator may acknowledge a command to manage **TR**s and follow operations with **YC**s (t_{2A}), or to check equipment (t_{2B}), or to organize rosters, vacation, and leaves of operators (t_{2C}). To manage **TR**s, he opens or closes the parking area and assigns or withdraws the **TR**s by registering the entrance and the exit from the area (t_3). During this phase, he checks the **TR**s before allowing them to leave the yard for the quay area or for the parking zone. He may also receive a signal for a **YC** or **QC** (t_{4A}), or a signal for a **RS** or **SL** (t_{4B}), or a signal for terminal spaces (t_{4C}); consequently, he checks operation of the crane (t_6), or of the reach stacker or side loader (t_7), or the equipment in the signalled area (t_8). The managing of human operators is concentrated in t_5. Finally, he may continue (t_{9A}) or stop (t_{9B}) his activities. The model is depicted in Fig. 7 (p_0 and p_1 are duplicated).

The time elapsed between the start of t_1 and the completion of t_{9A} gives the time spent by ***Yo*** to complete all the assigned operations, as an index of the operator's efficiency.

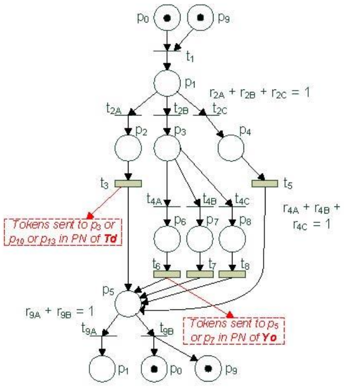


Fig. 7. GSPN of the yard checker, ***Yc***.

V. ANALYSIS OF LIVENESS AND BOUNDEDNESS

In this section, we show how the interconnection of the PN modules, which were defined for the different human operators, constitutes a live and bounded PN.

It is known that liveness and deadlock-free operation are related, so that simulation of a live PN or use of the model for control purpose never terminates into a state in which an indefinite wait (deadlock) is observed. A specific transition in a PN is said to be *live* if it can fire at least once in some firing sequence, for every marking M reachable from M_0 [21]. Then, a PN is live if every transition in the net is live. Moreover, the net is bounded if the number of tokens in each place does not exceed a finite number, for any reachable marking. These two properties are structurally guaranteed when they are not depending on the marking. Liveness is guaranteed by absence of empty siphons in Petri nets [21]. A siphon is a set S of places, such that all input transitions to S are also output transitions from S, i.e. $°p \subseteq p°$ for each place $p \in S$ (for a trap, the opposite is true). Then, tokens tend to exit from siphons when the net dynamically evolves by firing transitions: when all places in S are unmarked, S is empty, and emptiness is preserved in all future reachable markings. This defines a deadlock condition.

The approach to ease here the analysis of the properties of the developed PN model is independent of the considered system and based on formal theoretical results and on PN reduction techniques, which transform the refined model (at a detailed low level) to a more abstract representation (at a high level). In particular, the *proof* of the following theorem is reported in Appendix:

Theorem. The net obtained by connecting the modules of terminal operators is live and bounded.

VI. MODEL VALIDATION AND PERFORMANCE EVALUATION

The GSPN model developed in previous sections was validated by extensive simulation with Matlab. The complete model can be derived by considering all the modules that were previously defined to represent the relevant human activities and by connecting them.

We remark that the TCT terminal has capacity to handle over 2M TEUs per year. The traffic lead to a steady throughput growth from 150.000 TEUs in 2002 to about 900.000 TEUs in 2006, but more recent years had experienced decrease in traffic due to several reasons (lack of investments, changes in international routes, etc.). However, the system under investigation is complex and requires optimization of processes, in case a better situation and its strategic location in the Mediterranean sea bring it back to normal operation.

Then, the data used to set up and run the simulation experiments are taken from observations recorded in 2010 when the TCT still had a good productivity (details are omitted for sake of space). For example, the usual freight of a vessel ship served by TCT amounts to 2000 TEUs, then we assume to load/unload 2000 TEUs. The number of simultaneously operating quay cranes assigned to a ship is 4. The transition firing times are taken from the average values that can be found in the literature for typical tasks, or from technical characteristics of the employed handling equipment (e.g. 2 minutes required for loading/unloading a TEU to/from a ship, 20 minutes required for transport and way back between quay and yard side). Conflicts and choices between mutual exclusive transitions are solved by a Matlab script that randomly selects the fired transitions. The simulation tests allowed us not only to verify the model accuracy, its correct dynamics and its formal properties, but also to check the consistency with trends observed in the terminal. Moreover, the analysis aims to identify critical parts of the handling processes, the resources that can be possibly augmented or improved in efficiency, and to provide indications on the strategic human resources.

A standard scenario and two perturbed operating scenarios were considered. The standard operation (“Scenario 1”) is with one berthed ship and without any problem, fault or malfunction to hardware and human resources. Perturbations may usually come from unavailability or faults to physical resources, unavailability of human operators in the team, sanitary processes, accidents, etc. Then, the first perturbed scenario (“Scenario 2”) considers a 10% probability of faults/malfunctions and a 10% probability of transient problems. In other words, in the GSPN models of the human operators the conflicts are solved setting a 10% probability to the firing of the transitions that represent the occurrence of faults/malfunctions or transient problems. This condition puts the terminal under stress. The second severe perturbed scenario (“Scenario 3”) considers 40% probabilities for the two mentioned kinds of perturbations, then it seriously affects performance and also creates hard and hazardous conditions for the people involved.

To analyze performance, we preliminary observe that a common indicator of terminal efficiency and speed of the unloading/loading processes is the number of TEUs handled in a roster, i.e. the *TEUs per crane hour* (TPCH for brevity):

$$\text{TPCH} = h_t / (N_C\, h_C) \qquad (1)$$

where h_t is the total number of handled TEUs, N_C is the number of available cranes, h_C is the number of working hours of cranes. In this study, $h_t = 2000$ and $N_C = 4$.

Another popular performance index is the *Ship Turn-around Time* (STT), i.e. the average time needed to unload and load TEUs from a single berthed ship. Minimizing STT is obviously very important for the terminal management. An empiric relation to calculate the minimum STT value is:

$$\text{STT}_{min} = h_t / (\text{TPCH}\ N_C) \qquad (2)$$

which is a rewriting of (1) but assumes a constant handling performance of cranes. Note that (2) is only a preliminary indicator for the terminal productivity, because it does not take into account the effects of internal transfers and the nonlinear dependence of TPCH on N_C, due to the interaction between cranes. Moreover, N_C could be shortened due to limitations in handling capacity in the quay space. Finally, note that the global ship service time depends on the time required for docking (about 2 hours), the time spent in loading/unloading processes (about 18 hours), and the time required to cast off (about 45 minutes). The focus is on the second component, which is the only one that can be controlled. Given these considerations, performance is evaluated by several indexes:

- the number of loaded/downloaded TEUs;
- the ship service time for loading/unloading;
- the number of TEUs that are transported by trailers;
- the percentage utilization of the quay cranes;
- the percentage utilization of the yard cranes.

If the first two indexes are clearly related to the classical aggregate TPCH and STT indicators, the last three provide an indication of the efficiency of the main and strategic resources. Namely, it is intuitive to understand the importance of quay cranes, but the efficiency of internal transport and of yard operation is also important (e.g. to avoid costs related to renting of external auxiliary equipment).

Several simulation runs were performed. The results from the most representative runs are indicated in the following tables. Table I shows the total number of loaded/unloaded TEUs in 5 hours and the average per hour. The last row indicates the average computed over all simulation runs. Table II shows the ship service time (in minutes), its average over the runs and the average hours required (see very last row). Table III indicates the total number and the average of TEUs that are subject to internal transports. Again the last row indicates the average over all the runs. Tables IV and V report the utilization of cranes.

The results in the Scenario 1 of Table I are in accordance with the real average performance of quay cranes that was reported by the TCT management for standard conditions. Namely, the range $27 \leq \text{TPCH} \leq 33$ was indicated, with a typical value TPCH = 32, and a maximum $\text{TPCH}_{max} = 36$ (see Table VII). Then, the results confirm the model prediction capability.

TABLE I
LOADED/UNLOADED TEUS IN 5 HOURS

Run	Scenario 1 Standard cond.		Scenario 2 10% pert.		Scenario 3 40% pert.	
	Value	Average	Value	Average	Value	Average
1	159	31.8	124	24.8	100	20.0
2	166	33.2	126	25.2	80	16.0
3	185	37.0	145	29.0	55	11.0
4	154	30.8	98	19.6	111	22.2
5	165	33.0	157	31.4	127	25.4
6	137	27.4	103	20.6	107	21.4
7	181	36.2	97	19.4	97	19.4
8	161	32.2	98	19.6	78	15.6
9	149	29.8	130	26.0	79	15.8
10	162	32.4	133	26.6	80	16.0
Average	*161.9*	*32.38*	*121.1*	*24.22*	*91.4*	*18.28*

In the Scenarios 2 and 3, the TPCH performance is reduced by 25% and by 44%, respectively, which confirms the sensitivity of the terminal to perturbations.

TABLE II
SHIP SERVICE TIME (IN MINUTES)

Run	Scenario 1 Standard cond.	Scenario 2 10% pert.	Scenario 3 40% pert.
1	943.40	1209.68	1500.00
2	903.61	1190.48	1875.00
3	810.81	1034.48	2727.27
4	974.03	1530.61	1351.35
5	909.09	955.41	1181.10
6	1094.89	1456.31	1401.87
7	828.73	1546.39	1546.39
8	931.68	1530.61	1923.08
9	1006.71	1153.85	1898.73
10	925.93	1127.82	1875.00
Average	932.89	1273.56	1727.98
Aver. hours	*15.55*	*21.23*	*28.80*

TABLE III
HANDLED TEUS IN INTERNAL TRANSPORT MOVEMENTS

Run	Scenario 1 Standard cond.		Scenario 2 10% pert.		Scenario 3 40% pert.	
	Value	Average	Value	Average	Value	Average
1	138	27.6	94	18.8	42	8.4
2	139	27.8	100	20.0	47	9.4
3	135	27.0	102	20.4	57	11.4
4	128	25.6	103	20.6	52	10.4
5	137	27.4	89	17.8	47	9.4
6	150	30.0	96	19.2	41	8.2
7	148	29.6	117	23.4	55	11.0
8	136	27.2	101	20.2	30	6.0
9	143	28.6	99	19.8	47	9.4
10	135	27.0	102	20.4	46	9.2
Average	*138.9*	*27.78*	*100.3*	*20.06*	*46.4*	*9.28*

The ship service time in Scenario 1 in Table II confirm the goodness of the proposed model. The average ship service time is 15.55 hours in standard condition, very closely to the real value given by the TCT management equal to 15.60 hours.

The values achieved in Scenarios 2 and 3 indicate that service time increases by 37% and 85%, respectively, with consequent economic losses for the ship ownership and the TCT company.

The results in Table III clearly indicate how perturbations reduce and slow down the internal transports of TEUs that are of primary importance for fast processes.

TABLE IV
QUAY CRANES UTILIZATION (IN MINUTES AND IN % OF TIME IN 5 HOURS)

Run	Scenario 1 Standard cond.	Scenario 2 10% pert.	Scenario 3 40% pert.
1	297.55	256.70	176.55
2	295.41	268.44	138.87
3	296.25	242.35	112.69
4	294.29	199.28	211.65
5	293.48	262.65	234.87
6	294.75	219.11	203.25
7	298.46	197.45	196.19
8	297.13	206.95	154.53
9	299.31	274.20	146.83
10	296.32	246.36	176.56
Average	296.30	237.35	175.20
Percentage	*98.77*	*79.12*	*58.40*

TABLE V
YARD CRANES UTILIZATION (IN MINUTES AND IN % OF TIME IN 5 HOURS)

Run	Scenario 1 Standard cond.	Scenario 2 10% pert.	Scenario 3 40% pert.
1	251.40	260.49	279.84
2	234.77	258.02	278.27
3	231.05	245.75	249.73
4	250.08	254.98	272.43
5	241.51	255.32	266.42
6	231.91	254.94	270.59
7	239.00	242.93	270.38
8	233.32	261.93	251.25
9	241.38	258.03	277.81
10	247.76	266.02	253.52
Average	240.22	255.84	267.02
Percentage	*80.07*	*85.28*	*89.01*

Table IV shows the high utilization of quay cranes in standard conditions (Scenario 1), which is very important to reduce as much as possible the ship service time and how long the ship stays in the terminal. Namely, the usual terminal policy is to concentrate efforts in quay side efficiency. Then, a high, almost uninterrupted, quay crane utilization allows the terminal company to offer a competitive cost service and attract freight transport. At the same time, the maritime company can guarantee fast transport. Obviously, the intensive utilization may cause problems, faults, accidents, stress of humans, etc. The analysis of perturbed scenarios can show how performance changes. Namely, the utilization of quay cranes and yard cranes is greatly changed in the perturbed scenarios, as it can be verified from Tables IV and V. If quay cranes are much less utilized, the yard cranes are more intensively used because trailers tend to concentrate in the yard area and because yard cranes get involved in many operations to redistribute TEUs in the yard stacking areas.

To further validate the model, it is necessary determining how closely the simulation model represents the actual system and it is here achieved by the procedure proposed in [48] by applying the well-known *single mean test*. Table VI shows and compares the performance indices TPCH and STT obtained by the simulation in standard conditions with the relative half width of the confidence interval and the corresponding values computed by historical data in 2010 (see Table VII) from TCT. No data is available for operation of the terminal in perturbed conditions. Hence, denoting by PI the generic performance index provided by the simulation, by RPI the corresponding performance index obtained by real data and by ρ the relative half width of the confidence interval, Table VI shows that for each considered performance index it holds:

$$PI - \rho \leq RPI \leq PI + \rho \tag{3}$$

Applying the single mean test, the results prove that the simulation closely represents the actual system.

TABLE VI
COMPARISON BETWEEN SIMULATION RESULTS AND REAL DATA

Performance Index	Simulation value in Scenario 1	ρ	Real data from TCT
TPCH (TEU/hour)	32.38	1.66	32.23
STT (hour)	15.55	0.80	15.60

TABLE VII
REAL DATA FROM TCT (COMPUTED IN 2010)

Months	TPCH (TEU/hour)	STT (hour)
Jan-10	27.84	17.96
Feb-10	28.27	17.69
Mar-10	33.87	14.76
Apr-10	30.13	16.59
May-10	31.92	15.66
Jun-10	36.11	13.85
Jul-10	35.35	14.14
Aug-10	31.76	15.75
Sep-10	32.75	15.26
Oct-10	31.95	15.65
Nov-10	33.04	15.13
Dec-10	33.83	14.78
[*min-max*]	[*27.84-36.11*]	[*13.85-17.96*]
Average	*32.23*	*15.60*

To synthesize, we remark that effective and efficient operation is achieved only in standard conditions and with human operators at the maximum of their efficiency. This requires proper training but also prevention and reduction of delays and problems that can occur, which is not always possible. The negative effects that cause performance reduction can be only mitigated by investments on available quayside space and equipment, such as advanced cranes that could perform more operations in a faster way, or on more trailers that could expedite internal movements. However, the human component can be only partially replaced by automatic equipment and plays a crucial role in several tasks.

Finally, the GSPN-based approach can be further improved to a complete terminal simulator and used in conjunction with a sensor and monitoring system to track, predict and correct processes in real terminal operation. To this aim, it is interesting to note that each module in the GSPN model provides useful information (e.g. the estimated completion times of operations) that can be used to check correctness of human behaviors and resource operations. On the other hand, PDAs allow human operators to send and receive signals to a control center, where the simulator should be located. In this way, all the real operations can be followed on visualization screens. This allows monitoring and comparing the real operations to the state transitions of the model, then to verify what is expected, or to detect human errors, incorrect operations or problems in messages exchange.

VII. CONCLUSION

Modularity is the fundamental characteristic of the developed model. Namely, all the defined GSPN modules are connected, to give a complete integrated model of the coordinated operations in a well-organized team. The modularity of the whole net guarantees efficiency, because units with limited complexity describe each operator, with a low number of transitions, places and arcs. By integrating the model of the behavior of each human operator and the model of the sequence of operations to be executed by each resource (crane, trailer, etc.), a complete and accurate description of the complex processes in the terminal is achieved. This formalization can be translated to a discrete event simulation environment, useful to monitor and test the system.

The developed GSPN model can be profitably used to test future scenarios or control policies, which could be different from the ones currently employed in TCT, taken as reference.

APPENDIX

Theorem. The net obtained by connecting the modules of terminal operators is live and bounded.

Proof. If we consider isolated PN modules by neglecting the connections between them, we may classify each module as a *free-choice net* [21]. A free-choice net is characterized by having every arc from a place either as a unique outgoing arc from that place or as a unique incoming arc to a transition. Then, the following result is true (see Theorem 12 in [21]): a free-choice net is structurally live if and only if every siphon has a trap. But all the modules contain no empty siphon. This can be easily verified by formal reduction techniques (i.e. fusion of series places, fusion of series transitions, fusion of parallel places, fusion of parallel transitions, elimination of self-loop places, and elimination of self-loop transitions) applied to simplify each of the modules. Therefore, all the unconnected modules are structurally live free-choice nets.

Now, let's consider what happens when connections are established. All connections between a generic module N and other modules can be classified in two types as in Fig. 8, where the internal structure of N is not shown: *a*) arcs link transitions (e.g. t^*) in N to places in other modules $N' \neq N$, or *b*) one or more enabling places (e.g. p^*) receive tokens from other modules and arcs link them to transitions in N.

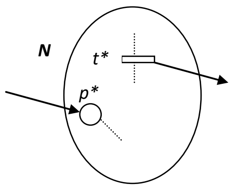


Fig. 8. Connections of a PN module.

Then, if we consider the live modules in isolation, each subset Q of places taken from one of these modules is not a siphon. This means that, if $p \in Q$ then not all input transitions in $^\circ p$ are, at the same time, output transitions in p°. In other words, there exists at least one input transition which does not belong to the set of output transitions: $\exists\, t \in {}^\circ p'$, such that $t \notin p'^\circ$, for some $p' \in Q$. Next, we will prove absence of siphons also after connections are made. The proof is by contradiction, by considering two cases associated to the two possible connection types.

First, consider the case in which a siphon is created due to a connection established in the first way, i.e. through an arc outgoing from t^*. Then, connection should make t^* become an output transition of a place $p \notin N$, i.e. $t^* \in p^\circ$, while t^* is an input transition for the same place, i.e. $t^* \in {}^\circ p$, before the connection is established. But this is impossible due to the direction of the connecting arc, because t^* can only become an input transition to some place external to N, and because t^* is not an input transition to this place before the connection.

The second case is when a link between two modules N and N' uses an enabling place p^* for N, and the associated linking arcs (t_1, p^*) and (p^*,t_2), with $t_1 \in {}^\circ p^*$, $t_1 \in N'$, and $t_2 \in p^{*\circ}$, $t_2 \in N$ (Fig. 9).

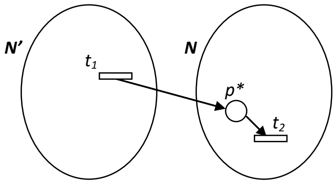


Fig. 9. Second type of connection.

A siphon S could be created in two possible ways:

b1) *Not inclusion*: adding the arcs creates S, which does not include p^*. Then, S exists because an input transition becomes also an output transition, such that $^\circ p \subseteq p^\circ$ for each place $p \in S$; this transition could only be t_2, internal to N, or t_1, external to N;

b2) *Inclusion*: adding p^* creates S, which includes p^*, other places in N, and places external to N, namely located in N'.

In case *b1*, if $t_2 \in N$ is becoming an output transition, this occurs only for p^*, because t_2 cannot be the output transition for any place external to N. But p^* does not belong to S, then we obtain a contradiction. If $t_1 \in N'$ is becoming an output transition, this occurs for some place p'. But p' cannot belong to N' and to every other module distinct from N, otherwise t_1 would be an output transition for p' even before the connection. Moreover, p' cannot belong to N, otherwise there should be an arc (p',t_1) outgoing from a place in N, which is not a possible connection. Then, we obtain another contradiction.

In case *b2)*, by assumption, a siphon $S = S_N \cup S_{N'}$ includes p^* and places from N, i.e. the set S_N, and places from N', i.e. $S_{N'}$. Consider all places in $S_{N'}$: there exists at least one transition $t' \in N'$ which is in the input set but not in the output set for these places, before the connection by p^*, otherwise we should have a siphon with places only from N', which is live when isolated (see Fig. 10).

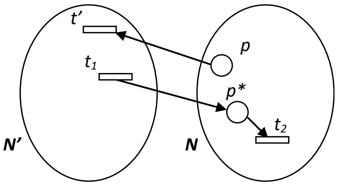


Fig. 10. Siphon with p^* is impossible.

Then, t' should be an output transition for some place $p \in N$ and we have an arc (p, t'), with $p \in N$ and $t' \in N'$. Note that this case is not an existing connection for most modules of the proposed model. However, even if this arc (p, t') exists, consider all places in S_N. Before considering the links through t_1 and t', no siphon exists, then there should be at least one transition, different from t_1, behaving as input but not as output for the places in S_N. This transition should become an output transition after connection: the only one is t', but t' is not an input for any place in N, before connection. Then, we obtain a contradiction.

Boundedness before the modules are connected can be also proven. Consider whichever module and call it N. The resource places enable the transitions representing activities of the associated operator. Therefore, the number of tokens flowing in the net is structurally limited by the capacity of the resource places, which is finite. Then, all places in the module are marked by a limited number of tokens, no matter how many times the transitions are fired.

When N is connected to other modules, boundedness might only be lost because of the enabling places through which it is connected (see p^* in Fig. 9). But, even the tokens marking p^* are finite, because they come from a module in which the token flow is limited. As a consequence, it doesn't matter the number of times the transitions linked to p^* (t_2 in Fig. 9) are fired, the token count in N keeps limited. □

REFERENCES

[1] D. Steenken, S. Voss, and R. Stahlbock, "Container terminal operation and operations research - a classification and literature review," OR Spectrum, vol. 26, no. 1, pp. 3-49, Jan. 2004.

[2] I. F. A. Vis and R. De Koster, "Transshipment of containers at a container terminal: An overview," European Journal of Operational Research, vol. 147, no. 1, pp. 1-16, May 2003.

[3] A. Perego, S. Perotti, and R. Mangiaracina, "ICT for logistics and freight transportation: A literature review and research agenda," Int. J. Phys. Distrib. Logist. Manag., vol. 4, no. 5, pp. 457–483, 2011.

[4] L. M. Gambardella, M. Mastrolilli, A. E. Rizzoli, and M. Zaffalon, "An optimization methodology for intermodal terminal management," J. Intelligent Manufacturing, vol. 12, no. 5–6, pp. 521–534, 2001.

[5] M. Dotoli,, N. Epicoco, M. Falagario, and G. Cavone, "A Timed Petri Nets Model for Performance Evaluation of Intermodal Freight Transport Terminals", IEEE Transactions On Automation Science and Engineering, 2015, DOI 10.1109/TASE.2015.2404438.

[6] A. Di Febbraro, N. Sacco, "Sensitivity analysis of the performances of seaport container terminals", in the Proceedings of 2013 International Conference on Control, Decision and Information Technologies, CoDIT, pp. 691-696, 2013.

[7] G. E. Cantarella, A. Cartenì, and S. de Luca, "A comparison of macroscopic and microscopic approaches for simulating container terminal operations," in Proc. of EWGT2006 Int. Joint Conferences, Bari, Italy, 27-29, pp. 556-558, Sept. 2006.

[8] E. Kozan, "Optimising container transfers at multimodal terminals," Mathematical and Computer Modelling, vol. 31, no. 10-12, pp. 235-243, 2000.

[9] M. Dotoli, N. Epicoco, M. Falagario, C. Seatzu, and B. Turchiano, "An optimization technique for intermodal rail-road freight transport

terminals," in Proc. IEEE Int. Conf. Robot. Autom., Hong Kong, China, 2014.

[10] A. Alessandri, C. Cervellera, M. Cuneo, M. Gaggero, and G. Soncin, "Management of logistics operations in intermodal terminals by using dynamic modelling and nonlinear programming,"Marit. Econ. Logist., vol. 11, no. 1, pp. 58–76, 2009.

[11] S. de Luca, G. E. Cantarella, and A. Cartenì, "A macroscopic model of a container terminal based on diachronic networks," Schedule-Based Modeling of Transportation Networks Operations Research/Computer Science Interfaces Series Volume 46, pp 1-26, 2009.

[12] S. Hartmann, "A general framework for scheduling equipment and manpower at container terminals," OR Spectrum, vol. 26, no. 1, pp. 51-74, Jan. 2004.

[13] P. Legato and M. F. Monaco, "Human resources management at a marine container terminal," European Jour. Operational Research, vol. 156, no. 3, pp. 769-781, 1 Aug. 2004.

[14] K. H. Kim, K. W. Kim, H. Hwang, and C. S. Ko, "Operator-scheduling using a constraint satisfaction technique in port container terminals," Computers & Industrial Engineering, vol. 46, no. 2, pp. 373-381, Apr. 2004.

[15] C.-I. Liu, H. Jula, and P. A. Ioannou, "Design, Simulation, and Evaluation of Automated Container Terminals," IEEE Trans. Intellig. Transp. Systems, vol. 3, no. 1, pp. 12-26, Mar. 2002.

[16] M. Bielli, A. Boulmakoul, and M. Rida, "Object oriented model for container terminal distributed simulation," European Journal of Operational Research, vol. 175, no. 3, pp. 1731-1751, Dec. 2006.

[17] A. A. Shabayek and W. W. Yeung, "A simulation model for the Kwai Chung container terminals in Hong Kong," European Journal of Operational Research, vol. 140, no. 1, pp. 1-11, July 2002.

[18] F. Parola and A. Sciomachen, "Intermodal container flows in a port system network: Analysis of possible growths via simulation models," International Journal of Production Economics, vol. 97, no. 1, pp. 75-88, July 2005.

[19] P. Legato and R. M. Mazza, "Berth planning and resources optimisation at a container terminal via discrete event simulation," European Journal of Operational Research, vol. 133, no. 3, pp. 537-547, Sept. 2001.

[20] P. Canonaco, P. Legato, R. M. Mazza, and R. Musmanno, "A queuing network model for the management of berth crane operations," Comput. Oper. Res., vol. 35, no. 8, pp. 2432–2446, 2008.

[21] T. Murata, "Petri nets: properties, analysis, and applications," Proceedings of the IEEE, vol. 77, no. 4, pp. 541-580, Apr. 1989.

[22] Y. Tang, M.-C. Zhou, and M. Gao, "Fuzzy-Petri-net-based disassembly planning considering human factors," IEEE Trans. Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 36, no. 4, pp. 718-726, July 2006.

[23] J.-S. Lee, M.-C. Zhou, and P.-L. Hsu, "An Application of Petri Nets to Supervisory Control for Human-Computer Interactive Systems," IEEE Trans. Ind. Electronics, vol. 52, no. 5, pp. 1220-1226, Oct. 2005.

[24] J.-S. Lee and P.-L. Hsu, "Remote supervisory control of the human-in-the-loop system by using Petri nets and Java," IEEE Trans. Ind. Electronics, vol. 50, no. 3, pp. 431-439, June 2003.

[25] J.-S. Lee, M.-C. Zhou, and P.-L. Hsu, "A petri-net approach to modular supervision with conflict resolution for semiconductor manufacturing systems," IEEE Trans. Automation Science and Engineering, vol. 4, no. 4, pp. 584-588, 2007.

[26] G. Alves, P. Maciel, and R. Lima, "A GSPN based approach to evaluate outbound logistics," in Proc. 2007 IEEE Int. Conf. Systems, Man and Cybernetics, Montreal, Quebec, Canada, 7-10, pp. 708-713, Oct. 2007.

[27] H. Zhang and Z. Jiang, "Modeling and Analysis of Container Terminal Logistics System by Extended Generalized Stochastic Petri Nets," in Proc. IEEE Int. Conf. Service Operations and Logistics and Informatics, Shanghai, P. R. China, 21-23, pp. 310-315, June 2006.

[28] M. Dotoli, M. P. Fanti, A. M. Mangini, G. Stecco, and W. Ukovich, "The impact of ICT on intermodal transportation systems: A modelling approach by Petri nets," Contr. Eng. Pract., vol. 18, no. 8, pp. 893–903, 2010.

[29] A. Di Febbraro, G. Porta, and N. Sacco, "A Petri net modelling approach of intermodal terminals based on Metrocargo system," in Proc. 9th Intell. Transp. Syst. Conf., Toronto, ON, Canada, pp. 1442–1447, 2006.

[30] H. Chen, L. Amodeo, C. Feng, and K. Labadi, "Modeling and performance evaluation of supply chains using batch deterministic and stochastic Petri nets," IEEE Trans. Autom. Sci. Eng., vol. 2, no. 2, pp. 132–144, Apr. 2005.

[31] C. Degano and A. Di Febbraro, "Modelling Automated Material Handling in Intermodal Terminals," in Proc. 2001 IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM'01), Como, Italy, 8-12, vol. 2, pp. 1023-1028, July 2001.

[32] C. Degano and A. Di Febbraro, "On using Petri nets to detect and recover from faulty behaviours in transportation facilities," in Proc. 2002 IEEE Int. Conf. Sys., Man and Cybern., Yasmine-Hammamet, Tunisia, 6-9, pp. 31-36, Oct. 2002.

[33] C. A. Silva, C. G. Soares, and J. P. Signoret, "Intermodal terminal cargo handling simulation using Petri nets with predicates," J. Eng. Marit. Env., 2014.

[34] M. Fischer and P. Kemper, "Modeling and Analysis of a Freight Terminal with Stochastic Petri Nets," in Proc. of 9th IFAC Int. Symp. Control in Transp. Systems, Braunschweig, Germany, 13-15, vol. 2, pp. 195-200, June 2000.

[35] C. I. Liu and P. A. Ioannou, "Petri Net Modeling and Analysis of Automated Container Terminal Using Automated Guided Vehicle Systems," Transportation Research Record, no. 1782, pp. 73-83, 2002.

[36] G. Maione and M. Ottomanelli, "A preliminary Petri net model of the transshipment processes in the Taranto container terminal," in Proc. 10th IEEE Int. Conf. Emerging Technologies and Factory Automation, Catania, Italy, 19-22, vol. I, pp. 165-171, Sept. 2005.

[37] G. Maione and M. Ottomanelli, "A Petri net model for simulation of container terminal operations," in Proc. 10th Meeting of the EURO Working Group on Transportation, Poznan, Poland, 13-16, pp. 373-378, Sept. 2005.

[38] N. Q. Wu, M. C. Zhou, and G. Hu, "On the Petri net modeling of automated manufacturing systems," in Proc. of 2007 IEEE Int. Conf. on Networking, Sensing and Control, London, UK, 15-17, pp. 228-233, Apr. 2007.

[39] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," IEEE Transactions on Systems, Man, & Cybernetics, Part A, vol. 34, no. 1, pp. 38-51, 2004.

[40] M. Zhou and F. DiCesare, Petri net synthesis for discrete event control of manufacturing systems, Kluwer Academic Publishers, Boston, 1993.

[41] M. C. Zhou and K. Venkatesh, Modeling, simulation and control of flexible manufacturing systems. A Petri net approach, World Scientific Publishing Co., Singapore, 1999.

[42] N. Q. Wu and M. C. Zhou, "Avoiding deadlock and reducing starvation and blocking in automated manufacturing systems," IEEE Transactions on Robotics and Automation, vol. 17, no.5, pp. 657-668, 2001.

[43] N. Q. Wu and M. C. Zhou, "Resource-Oriented Petri Nets for Deadlock Avoidance in Automated Manufacturing," in Proc. of the 2000 Int. Conf. on Robotics and Automation, San Francisco, CA, USA, 24-28, pp. 3377-3382, Apr. 2000.

[44] F. DiCesare and A. A. Desrorchers, Modeling, Control, and Performance analysis of automated manufacturing systems using Petri nets, in C. T. Leondes (Ed.), Control and Dynamic Systems, Academic Press, vol. 47, pp. 121-172, 1991.

[45] L. Gniewek and J. Kluska, "Hardware Implementation of Fuzzy Petri Net as a Controller," IEEE Trans. Systems, Man, and Cybernetics - Part B: Cybernetics, vol. 34, no. 3, pp. 1315-1324, June 2004.

[46] G. Chiola, M. Ajmone Marsan, G. Balbo, and G. Conte, "Generalized stochastic Petri nets: a definition at the net level and its implications," IEEE Trans. Softw. Eng., vol. 19, no. 2, pp. 89-107, Feb. 1993.

[47] N. Viswanadham and Y. Narahari, "Stochastic Petri net models for performance evaluation of automated manufacturing systems," Information & Decision Technologies, vol. 14, pp. 125-142, 1988.

[48] A.L. Law, Simulation Modeling & Analysis. New York, NY, USA: McGraw-Hill, 2007.