

## TIMED PETRI NETS AS A VERIFICATION TOOL

Miryam Barad

Department of Industrial Engineering, Faculty of Engineering,  
Tel Aviv University, Ramat Aviv, Tel Aviv 69978, ISRAEL

### ABSTRACT

This paper presents Timed Petri Nets (TPN) as an analytical approach for verification of computerized queueing network simulation models at *steady state*. It introduces a generic approach to decomposing a queueing network into separate and independent TPNs, each representing a resource of specific type, such as a processing resource or a transporting resource. A decomposed TPN model enables a quick calculation of the resource expected utilization at steady state thus providing a stick yard against which to verify its long run simulation based utilization estimate.

A factorial experimental design frame is applied to investigate the conditions and assumptions affecting the accuracy of the TPN decomposition, as compared to the simulation results. The relatively low differences obtained between the numerical results of the two methods provide evidence that this TPN technique can be used to verify simulation models. Eventual flaws in the computerized simulation model can be detected through observed discrepancies between the TPN based and the simulation results.

### 1 INTRODUCTION

An important and difficult task in any simulation project is the validation and verification of the simulation model. Validation is commonly defined as the process intended to substantiate that the conceptual simulation model, within its domain of applicability, is an accurate representation of the real world system it describes. The verification process seeks to determine that the computerized simulation model was built right and operates as intended. There is a rich literature on these topics [see e.g. Shannon (1981), Sargent (1991), Balci (1994)].

From an output analysis perspective, system simulation is typically classified into terminating or steady state. While terminating simulations aim to estimate system parameters for periods well defined in terms of starting and ending conditions, steady state simulations aim to estimate system parameters as limits obtained for

infinite simulation time. Before reaching *steady state* conditions when the system parameters attain stable values, the system behavior undergoes transition periods, that reflect the initial system operating conditions (see e.g. Kelton 1989). In the analysis of such systems an important objective is to avoid bias in parameter estimation, eventually introduced by data collected during transition periods. One among the many approaches to simulation verification is comparing simulation outputs with analytical results (Kleijnen 1995). In many simulation studies dealing with manufacturing systems or computers and communication systems, the reality modeled by the system simulation is represented by networks of queues. Hence, for simulation verification of such systems analytical approaches describing *queueing networks* may be appropriate.

This article presents timed Petri nets as an analytical approach for verification of computerized queueing network simulation models at steady state. Timed Petri Nets (TPN) were found useful for performance evaluation of systems in general and manufacturing systems in particular but have been criticized as becoming too large, cumbersome and time consuming when large numbers of products (entity types) and work stations (resource types) are involved.

To overcome drawbacks such as the above, Timed Petri Net models of some manufacturing queueing networks were decomposed (see Barad 1994 and Barad and Sinriech 1998). It has been proved that under reasonably general conditions, comprising a given deterministic routing of discrete units through a system and unlimited buffers, the decomposed Petri nets are able to assume valid modeling and performance evaluation roles. The TPN approach puts into evidence the flow realization along a variety of network paths and may consider different network structures. The emphasis here is on developing a generic approach to decomposing a queueing network into separate and independent TPNs, each representing a specific resource type. Such a decomposed TPN model of a resource enables a quick calculation of its expected utilization at steady state thus providing a yard

stick against which to verify the long run simulation based utilization estimate of the same resource.

The paper is organized as follows; first, TPN modeling and results obtained for two TPN decomposition cases are briefly summarized. Then, a generic decomposition method that builds on the analytical results of these cases, is presented. A factorial experimental design frame is applied next, to investigate the conditions and assumptions affecting the accuracy of the TPN decomposition, as compared to simulation results. At the end of the paper some conclusions are drawn.

## 2 TIMED PETRI NET MODELING

A Petri Net is a graph  $N=(T,P,A)$  where  $T$  is a set of transitions  $(T_1, T_2, \dots, T_n)$ , portrayed by bars, representing instantaneous or primitive events,  $P$  is a set of places  $(P_1, P_2, \dots, P_m)$ , portrayed by circles, representing conditions.  $I(T)$  denotes the set of all input places of transition  $T$ . Similarly,  $O(T)$  denotes the set of all output places of transition  $T$ .  $A$  is a set of directed arcs  $(T^*P) \cup (P^*T)$  derived through an incidence matrix  $C$ :

$$C_{ij} = \begin{cases} -1 & \text{for } P_i \in I(T_j) \\ +1 & \text{for } P_i \in O(T_j) \\ 0 & \text{Otherwise} \end{cases}$$

A 'marking'  $M$  of a Petri Net is a distribution of tokens (or markers) to the places of a Petri Net. A transition can be 'fired' (meaning the respective event occurs) if there is at least one marker in each of its input places (the necessary conditions for its occurrence are fulfilled). Following the firing of a transition the number of tokens in the corresponding input places is depleted by one.

To model time in TPN a positive number,  $s(P)$  or  $s(T)$ , is associated to a place  $P$  or to a transition  $T$ , describing the occurrence of non-primitive events (with time greater than zero). The approach that associates time with places has been developed by Sifakis (1980). The delay may be modeled either as a deterministic or as a stochastic variable. However, if a stochastic model is used, calculations will only consider the expected values.

Our approach uses Sifakis' version for timed Petri Nets. Since most of the variables involved are stochastic by nature, the actual values are represented by their respective expected values. The reciprocal of the renewal-interval mean serves to estimate the average input flow of parts or disturbances, as in Whitt 1983. This procedure can be considered adequate in the present

context, since we are concerned with estimating *expected utilization at steady state* (see also section 4) by their respective expected values. The reciprocal of the renewal-interval mean serves to estimate the average input flow of parts or disturbances, as in Whitt 1983. This procedure can be considered adequate in the present context, since we are concerned with estimating *expected utilization at steady state* (see also section 4).

The method is first illustrated in Fig. 1 that represents a resource processing two part types. The arrival of part type  $j, j=1,2$  is respectively through  $T_{j1}$  with input flow  $I_j$ .

Upon its arrival, a part of type  $j$  will eventually wait in place  $P_{j1}$  for the resource to be available, i.e. for a token to be in place  $P_0$ . When this happens, starting processing of the part is enabled (transition  $T_{j2}$  fires). During processing, (whose duration is  $t_{j3}$  time units) a token resides in  $P_{j3}$ .

At the end of processing, transition  $T_{j3}$  fires and the resource becomes available, as marked by the return of the token to  $P_0$ . As stated by Sifakis, given a Timed Petri Net by its incidence matrix  $C$ , the net functions at its steady-state rate for a given flow vector  $I^0$  iff  $I^0$  satisfies the equation:

$$CI^0 = 0 \quad (I^0 > 0) \tag{1}$$

and

$$J_s Q(0) = J_s Z (C^+) I^0 \tag{2}$$

where  $[J_s]_{s=1,2,\dots,k}$  is a generator of the set of solutions of (1),  $[J_s C = 0]$ ,  $Z$  is the delay matrix and  $Q(0)$  is a vector representing the initial marking of the net.

$Z$  is defined as a square diagonal matrix with elements  $Z_{ij} = z_i \quad i=1,2,\dots,m$  for  $i=j$  and zero otherwise.

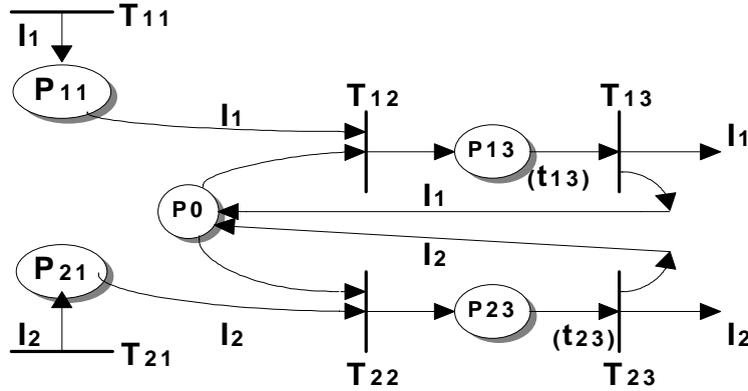
$(C^+)$  is obtained by replacing the '-1' elements in the incidence matrix  $C$  by zeros.

Intuitively, equations (1) and (2) can be respectively associated with conservation of flow and conservation of tokens.

Applying equations (1) and (2) to calculate  $r$ , the TPN steady state utilization of the resource here, we obtain:

$$r = I_1 t_{13} + I_2 t_{23}$$

$T_{j1}$  - part  $j$  arrival,  $j=1,2$   
 $T_{j2}$  - start processing part  $j$   
 $T_{j3}$  - end processing part  $j$   
 $P_0$  - resource available  
 $P_{j1}$  - part  $j$  waiting to be processed  
 $P_{j2}$  - part  $j$  is being processed



	T <sub>11</sub>	T <sub>12</sub>	T <sub>13</sub>	T <sub>21</sub>	T <sub>22</sub>	T <sub>23</sub>
P <sub>11</sub>	1	-1				
P <sub>13</sub>		1	-1			
P <sub>21</sub>				1	-1	
P <sub>23</sub>					1	-1
P <sub>0</sub>		-1	1		-1	1

Figure 1: A TPN of a Processing Resource Visited by Two Part Types and its Incidence Matrix

### 2.1 Steady state resource utilization: two TPN decomposition cases

*Case 1:* a TPN model of an  $n$ -part type network with  $M$  processing stations is decomposed into  $M$  independent TPNs, each representing a specific processing station. This enables to deal separately with each station and thus to easily consider a variety of entering and exiting flows of parts as well as eventual flows of disturbances and/or set-ups. The main result of this decomposition as proved in Barad, is summarized as follows; the total expected steady-state utilization  $r_m$  at station  $m$ ,  $m=1,2,\dots,M$  is the sum of its expected utilization by each class of customers. These comprise the  $n$ -part types as determined by a given deterministic routing and eventual machine failures or other disturbances including set-ups.

$$r_m = \sum_{j=1}^n r^{(m)}_j + r^{(m)}_f \quad (3)$$

where  $r^{(m)}_j$  and  $r^{(m)}_f$  represent the respective contributions of  $I^0_j$ , the input flow of part types  $j$ , and that of  $I_f$ , the input flow of disturbances, to the station

utilization  $r_m$ ;  $r^{(m)}_j$  is defined for possible multiple visits of part type  $j$  to station  $m$  as:

$$r^{(m)}_j = \sum_{k=1}^{V_j} t^{(m)}_{j,k} R^{(m)}_{j,k} I^0_j \quad ,$$

$$\text{for } j=1,2,\dots,n, m=1,2,\dots,M \quad (4)$$

where  $t^{(m)}_{j,k}$  is the expected duration of operation  $k$  of part  $j$  at station  $m$  and  $R^{(m)}_{j,k}$  is the given routing of parts to machine  $m$ ;  $R^{(m)}_{j,k}$  may assume a “0” or “1” value.

A similar equation defines  $r^{(m)}_f$  as:

$$r^{(m)}_f = t^{(m)}_f * I_f \quad (5)$$

where  $t^{(m)}_f$  is the expected duration of the disturbances. Similarly to multiple part types, multiple disturbance types representing failures and/or set-ups can be considered. Provided the constraint  $r_m < 1$  is satisfied for each  $m=1,2,\dots,M$ , and all buffers are of unlimited size, eq. (3) is a valid expression of the TPN based expected steady state utilization  $r_m$ , of any processing station  $m$ ,  $m=1,2,\dots,M$  in the system.

Simulation models of similar networks of operations can be verified by comparing simulation based estimates of the processing station utilization at steady state, with their respective TPN based calculated utilization.

**Case 2:** the procedure is extended, to include Material Handling and Transportation Systems, here Automated Guided Vehicles (AGVs) in Segmented Flow Topology (SFT) Systems (see Barad and Sinriech, 1998). The term SFT denotes a network comprising one or more zones, each of which is separated into non-overlapping segments with each segment serviced by a single AGV. Transfer buffers are located at both ends of each segment and serve as interface devices between the segments. Under these conditions, the AGVs in the system are also modeled as decomposed, independent TPNs. In contrast to a workstation (or an Input/Output station), that is located at a fixed node and is visited there by a part  $j, j=1,2,\dots,n$ , to have an operation  $k, k=1,2,\dots,v_j$ , carried out by the workstation, an AGV has eventually to travel empty towards the place where the part requesting transport is located and only then start transporting the part. When this has been accomplished, the AGV is bound to travel empty again. Hence, the travel of an AGV consists of empty travel and loaded (effective) travel. The TPN modeling of an AGV also necessitates assigning a waiting location to an AGV, denoted here its staging place.

The main result of the TPN decomposed model of an AGV consists in providing the proof and procedure for calculating the total expected steady-state utilization  $r_q$  of AGV  $q, q=1,2,\dots,Q$  as a sum of its expected empty travel utilization and its expected loaded travel utilization. As in case 1, eventual machine failures or other disturbances including set-ups can be considered. Equation (3) without disturbances as adapted to AGV  $q$  now becomes:

$$r_q = \sum_{j=1}^n r^1(q)_j + \sum_{j=1}^n r^2(q)_j \quad (6)$$

where  $r^1(q)_j$  and  $r^2(q)_j$  respectively represent the contributions of  $I^0_j$ , the input flow of part types  $j$ , to the *empty* travel utilization and the *loaded* travel utilization of AGV  $q$ . Given the transportation assignments of the AGVs in the system, any part type unit is defined as a composite of specific AGV trips. Hence, calculating  $r^2(q)_j$ , the AGV loaded travel utilization is equivalent to calculating  $r^{(m)}_j$  in case 1.

To estimate the expected *empty* travel utilization of an AGV, Barad and Sinriech considered:  $z_4$ -the expected empty travel per trip of the AGV from a drop-off station to a new pick-up station;  $z_5$ -the expected empty travel per trip from a drop-off station to staging and  $z_7$ -the expected empty travel per trip from staging to a new pick-up station (see Fig. 2).

Based on the above, they proved that the AGV empty travel utilization can be approximated by the following equation:

$$r^1(q)_j = t^1(q)_j \cdot I^0_j \cong [z_4 r^{2(q)} + (z_5 + z_7) (1 - r^{2(q)})] Nt_j^{(q)} \cdot I^0_j \quad (7)$$

where  $t^1(q)_j$  is the expected empty travel duration of AGV  $q$  per type  $j$  unit and  $Nt_j^{(q)}$  is the number of its trips per type  $j$  unit.

To estimate  $z_4, z_5$  and  $z_7$ , a conditional approach, formulated in equation (8) was adopted. Its essence is explained below.

$z_4$  is first conditionally estimated as the empty travel duration between a given last drop-off station, say  $D$ , and a given next pick-up station, say  $P$ , and is denoted  $z_4|D,P$ . Then, by averaging  $z_4|D,P$  over all possible next pick-up stations  $P, P=1,2,\dots,M_p$ , the condition 'given  $P$ ' is removed, yielding  $z_4|D$  the expected empty travel duration between  $D$  and any next pick-up station. By averaging  $z_4|D$  over all possible last drop-off stations  $D, D=1,2,\dots,M_D$ , the condition 'given  $D$ ' is also removed, yielding the unconditioned  $z_4$ .

These averages are calculated using *flow-weighted* means of the travel durations between station  $D$  and each of the pick-up stations in the segment. The rationale is that it is not known in advance which station, among all pick-up stations in the segment, will initiate the next call. Thus, the weight  $v_p$  that is assigned to pick-up station  $P$ , represents the *expected flow* requirements at station  $P$ , normalized with respect to the *expected flow* requirements at all pick-up stations ( $\sum v_p=1$ ). Similarly, the weight  $w_D$  that is assigned to drop-off station  $D$ , represents the *expected flow* requirements at station  $D$ , normalized with respect to the *expected flow* requirements at all drop-off stations ( $\sum w_D=1$ ).

The calculations of  $z_5$  and  $z_7$  are driven by a rationale similar to the above. First, the conditional value of  $z_5$  upon a given drop-off station  $D, z_5|D$ , and the conditional value of  $z_7$  upon a given pick-up station  $P, z_7|P$ , are calculated. Then, their respective unconditional values ( $z_5$  and  $z_7$ ) are obtained as *flow-weighted* means of the conditional values.

$$z_4 = \frac{M_D}{\sum_{D=1} z_4|D} = \frac{M_P}{\sum_{P=1} z_4|D,P};$$

$$z_5 = \frac{M_D}{\sum_{D=1} z_5|D}; \quad z_7 = \frac{M_P}{\sum_{P=1} z_7|P} \quad (8)$$

Equations (4) to (6) are valid iff:  $r_q < 1$  for each  $q$ ,  $q=1,2,\dots,Q$ ,  $r_m < 1$  for each  $m$ ,  $m=1,2,\dots,M$  and buffers are of unlimited size.

This decomposed TPN model of an AGV enables calculating its expected utilization at steady state (loaded travel and empty travel), against which to verify the long run simulation based estimate of an AGV in an SFT System.

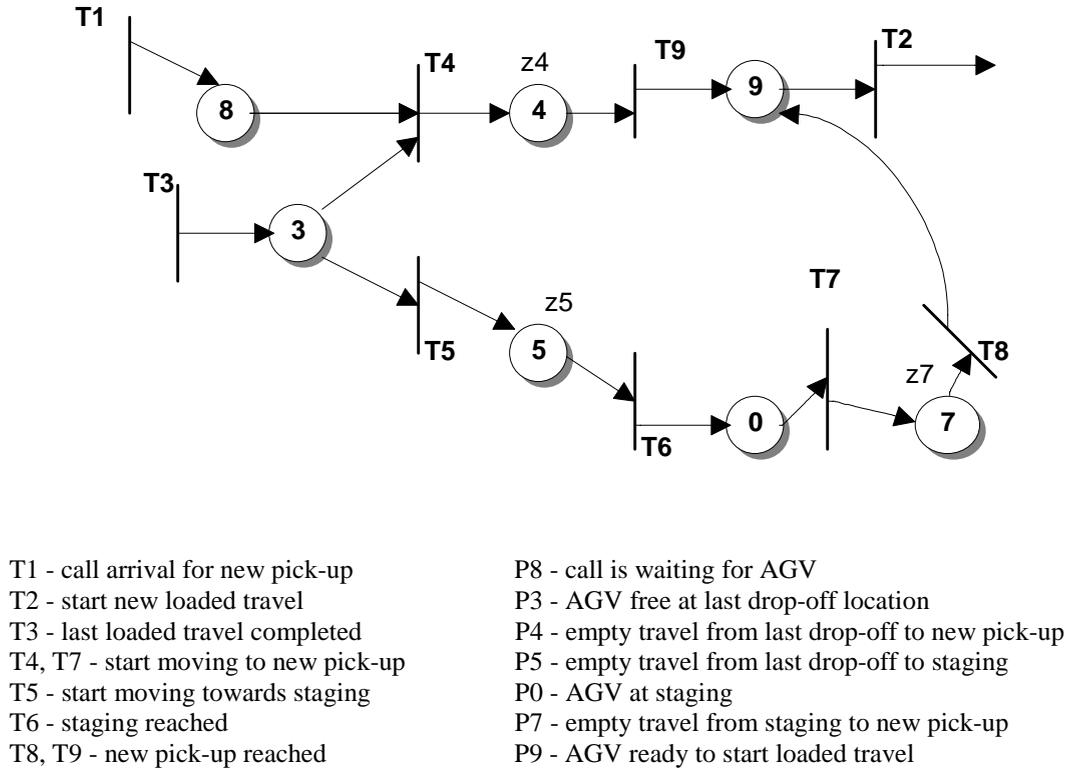


Figure 2: A partial TPN of an AGV Focusing on its Empty Travel

### 3 THE GENERIC DECOMPOSITION MODEL

To integrate the results of the two decomposition models into a generic approach we define the *elements* of the decomposed systems, the resource *utilization components* stemming from the network structure and discuss the modeling *assumptions*.

#### 3.1 The system elements: resources and entities

The resources here are of two types: fixed-location and moveable resources. Typically, modeling the TPN decomposition of a moveable resource is a more complex task than it is to model the decomposition of a fixed location resource. To verify a simulation based utilization estimate of a moveable resource, beside its loaded travel duration, a TPN based modeling of its *empty* travel duration is necessary. Common modeling features of the

resources here: both types serve multiple customers (part types) and are subjected to disturbances.

Entities are also of two types: physical (parts) and logical (failures and set-ups). A part type here is defined as a composite of specific operations to be performed by processing resources in the system in a predetermined order. As a disturbance may take hold of a resource regardless of its location, in principle modeling of disturbances here is the same for fixed location resources and for moveable resources. Two versions, A and B, of modeling failures (disturbances) were considered in Barad. According to version A, after its repair the resource becomes free. This implies that the part, whose processing (inspection or transfer) had been interrupted by the machine failure, has been discarded. In version B, instead of discarding the part whose processing was interrupted by the failure's arrival, after repair the resource resumes its interrupted activity. Since it is not possible to keep track of

the remaining processing time, a preemptive repeat assumption was applied there. Equation (5) here may be adapted to both versions:  $t_f^{(m)} = (t_f^{(m)} + t_d^{(m)})I_f$ , where  $t_d^{(m)}$  may represent some additional activity to be performed by the resource, as a result of the interruption such as repeating the interrupted processing activity. Assigning a zero value to  $t_d^{(m)}$  makes the equation compatible with version A.

### 3.2 Network structure and utilization components of a decomposed TPN

In each of the two cases above it has been proved that the utilization of a decomposed TPN can be calculated by summing up the respective contributions of the various utilization components representing mutually exclusive states of the modeled resource. This is illustrated in Figures 1 and 2.

In Fig.1 describing a processing resource there are two utilization components: the two part types. It has been proved that any number of part types can be considered and that processing of a part can be further decomposed into the various visits of each part type. Disturbances can be modeled as additional classes of components, altogether contributing to the total utilization of the decomposed TPN as formulated in eq. (1).

The utilization of any moveable resource is modeled by separating it into loaded and empty utilization components. As seen in Fig.2, the empty travel contribution to the AGV utilization is further split into two components. One component represents the empty AGV while traveling towards a next pick-up station ( $z_4$ ). The other component represents the empty AGV while traveling towards staging ( $z_5$ ) and from there eventually to a next drop-off station ( $z_7$ ). These components respectively express the flow of part calls that find the AGV at staging (available) and those that find it while transporting another part (not available). Both, are solely relevant to the *empty* travel utilization of an AGV and as such are used in eq. (7).

### 3.3 TPN decomposition assumptions

The main assumption here is *free flow* of all entities in the system. To achieve this condition the total utilization of each resource should be less than 1 and unlimited buffer space should be provided. To estimate the empty travel utilization of an AGV, a general principle of free flow preservation expressed here by the TPN theories is applied, namely, that in the long run the flow at a drop-off processing station equals the flow exiting it.

When some resources in a queueing network are operated at persistently high utilization levels (even though below 1) typically, flows tend to be temporarily obstructed. Under these conditions very long simulation runs may be needed to achieve steady state and differences between

simulation based and TPN based results may occur. Typically, in this situation the simulation might underestimate the utilization level of some resources and the more valid estimate will be the TPN based estimate.

The closeness of results between the TPN based and the simulation based estimates may be also affected by other operating conditions such as the statistical distributions, and eventually by the queue discipline.

## 4 THE EXPERIMENTAL DESIGN

To methodically examine the accuracy of the proposed simulation verification method, a multi-factor simulation experiment of the system that was described in Barad and Sinriech has been carried out.

The effects of four factors were investigated:

1. The planned utilization of the resources (two levels).
2. The inter-arrival statistical distributions with four levels (Exponential, Erlang, Uniform and Normal); the coefficient of variance of these distributions varied from 0.07 to 1.
3. The generating seeds of Siman software (three levels). As we deemed that some bias might be introduced by the seeds used to generate the input values of the statistical distributions, we randomly selected three specific seeds, namely, #1, #2 and #5, from among Siman's ten regular seeds. The seeds played the role of blocks, i.e. represented homogeneous experimental environment.
4. The queue discipline with two levels (FCFS, SPT). This factor was only investigated with respect to the processing resources.

The dependent variable expressing the experimental results has been defined as the difference between the steady state simulation based estimate and its counterpart TPN based utilization estimate, normalized with respect to the TPN based estimate.

We separately analyzed the results of three resources in the system: two moveable resources (Automated Guided Vehicles) -AGV1 and AGV2- and one of the processing resources -station 4- that was randomly selected out of the six such resources in the system. Under the operating conditions there, the TPN based utilization levels were: AGV1 - 0.500, AGV2 - 0.650 and Station 4 - 0.750. In our planned experiment these represented the low utilization levels. To avoid reaching operating conditions impeding free flow, we limited the utilization level of all resources in the experiment to 0.9. Thus, the high utilization level of the most heavily loaded resource in the system, here station 4, became 0.9. To achieve that we decreased the inter-arrival

mean by a factor of 1.2. This way the high level TPN based utilization of the three investigated resources respectively became 0.6, 0.78 and 0.9.

We used Siman V and the ARENA framework to analyze the results. The run lengths were 5760 minutes and the ARENA output processor served to identify the steady state period and discard the transient periods' results. The total number of runs for AGV1 and AGV2 were 24, which allowed to perform an analysis of variance that took into account the main effects of the first three factors and their two factor interactions. The total number of runs for the processing station was 48, allowing to test the main effects of the four factors (including the queue discipline) and their two factor interactions.

Following are the separate results regarding the three resources;

1. The accuracy in estimating the utilization of AGV1, the least loaded resource, was not significantly affected by any of the investigated factors. It is to be recalled that the low and high utilization levels of AGV1 were respectively 0.5 and 0.6. The overall accuracy was 0.2%.
2. A different result was obtained for AGV2 that was operated at a low utilization level of 0.65 and at a high utilization level of 0.78. Here the increase in the utilization level significantly increased the difference (decreased the accuracy) between the simulation and the TPN results: from 1.6%, at 0.65 utilization to 2.9% at 0.78 utilization. It seems interesting to note that the accuracy was also significantly affected by the different seeds (#1, #2 and #5) as well as by the interaction between the seeds and the distribution type.
3. Increasing the utilization level of the processing resource (station 4) from 0.75 to 0.9, did not significantly affect the accuracy of the TPN method. Presumably, for the system here, level 0.75 is in the same class as level 0.9. Here too the accuracy of the results was significantly affected by the seeds and also by their interaction with the distribution type. It is to be noted that the length of the transient periods as identified through the ARENA output processor, were different at lower and higher traffic. At lower traffic the transient period durations were about 400 minutes while at higher traffic they varied between 700 minutes (for Uniform and Normal distributions whose respective Coefficient of Variance was 0.25 and 0.07) and over 1000 minutes for the Exponential and the Erlang distribution (with 0.5 Coefficient of Variance).

## 5 CONCLUSIONS

The accuracy obtained for the currently simulated system varied from 0.2% at lower traffic, to about 2.9% at higher traffic. Lower traffic may be roughly defined here as a utilization level below 0.6, while higher traffic, that still flows freely, as a utilization level above 0.75 but not exceeding 0.9.

Additional experimentation with different resource types under various operating conditions is necessary. However, the relatively low differences obtained here between the numerical results of the two methods provide evidence that the TPN technique can be used to verify simulation models. Eventual flaws in the computerized simulation model can be detected through observed discrepancies between the TPN based and the simulation results.

An important advantage of the proposed method is its easiness of use. Also, the method imposes no restrictions on the size of the network to be decomposed nor on the statistical distributions selected to model the stochastic variables in the simulated system.

## REFERENCES

- Balci, O. 1994. Validation, verification and testing techniques throughout the life cycle of a simulation study. *Annals of Operations Research* 53: 121-173.
- Barad, M. 1994. Decomposing timed Petri nets of open queueing networks. *Journal of the Operational Research Society* 45(12): 1385-1397.
- Barad, M. and D. Sinriech. 1998. A Petri net model for the operational design and analysis of segmented flow topology (SFT) AGV systems. *International Journal of Production Research* 36(12): 1401-1425.
- Kelton, W. D. 1989. Random Initialization Methods in Simulation. *IIE Transactions* 21(4): 355-367.
- Kleijnen, J. P. C. 1995. Verification and validation of simulation models. *European Journal of Operational Research* 82: 145-162.
- Sargent, R. G. 1991. Simulation model verification and validation. *Proceedings of the 1991 Winter Simulation Conference*
- Shannon, R. E. 1981. Tests for the Verification and Validation of Computer Simulation Models. *Proceedings of the 1981 Winter Simulation Conference*: 573-577.
- Sifakis, J. 1980. Performance evaluation of systems using nets. *Net Theory Applications*, New York: Springer.
- Whitt, W. 1983. The queueing network analyzer. *The Bell System Technical Journal*, 62(9): 779-2815.

## **AUTHOR BIOGRAPHY**

**MIRYAM BARAD** is an Associate Professor of Industrial Engineering and Management in the department of Industrial Engineering, Tel Aviv University. She holds a B.Sc. in Chemical Engineering, an M.Sc. and a D.Sc. in Industrial and Management Engineering from the Technion IIT Haifa, Israel. Her research and professional interests focus on Quality strategies and design as well as on modeling and implementation of Flexibility in manufacturing systems, emphasizing the use of Design of Experiments and Simulation Techniques. Recently, Dr. Barad contributed several entries on Quality to International Encyclopedias: 'Quality Assurance - Industrial Analysis' to the Int. Encyclopedia of Analytical Science (Academic Press, 1995) and 'Total Quality Management' to the Int. Encyclopedia of Business and Management (Thomson Business Press, 1996 and 1997).