Modeling Internet Host Reliability using Higher-Order Time Petri Net

Ali M. Meligy[1] Hani M. Ibrahim[2] Amal M. Aqlan[3]
Professor of Computer Science Lecturer of Computer Science PhD candidate
[1,2,3] Department of Mathematics, Faculty of Science, Menoufia University

ABSTRACT
The Higher-Order Petri Net is a new class of Petri Nets that exploit the properties of higher-order neural networks. Adding time to HOPN produces a new class called Higher-Order Time Petri Net, this is the subject of our study .In this paper, a method to model the internet host reliability with Higher-Order Time Petri Net is proposed. Analysis of HOTPN model is presented. A reachability graph is defined in a discrete way by using an enumeration procedure and the reachable states of the Time Petri Net and HOTPN. Finally, we compare between TPN and HOTPN by using the behavior properties and reachability graph.

Keywords:
Time Petri Net (TPN), Higher-Order Petri Net (HOPN), Reachability Graph, Internet Host Reliability.

1. INTRODUCTION
Petri nets have many applications in different issues in the different systems. These systems may be complex. It is for this reason that different Petri net types have been formally defined and created. Every one of them has its own particular specialized use which is very interesting to solve and address a particular problem. But having so many different types of Petri nets creates issues as how to choose the best Petri net type or class for a particular problem. Petri nets can be used to test and verify rigorously, system functionality and design implications at the very initial phases of analysis. They can be used to determine critical performance issues related to time and failure. Petri nets can be used to create different viewpoints of a system related to the conceptual, logical or physical design structures [1].

In some cases, the classical Petri nets cannot model the behavior of the system accurately. To solve this problem, researchers proposed a new class of Petri nets called Higher Order Petri Nets (*HOPN*) that are based on the properties of higher-order neural networks.

Time Petri Nets are derived from classical Petri nets. Additionally, each *transition* t is associated with a time interval $[a_t, b_t]$. Here, a_t and b_t are relative to the time, when t was enabled last [2]. Every possible situation in a given *TPN* can be described completely by a state $z = (m, h)$, consisting of a (*place*-) marking m and a *transition* marking h.

A reachability graph $RG(Z)$ for a *TPN* Z can be defined in such a way that its vertices are the reachable integer-states or the reachable essential-states, respectively. The edges are defined by the triples (z, t, z') and (z, τ, z'), $\tau \in N$, where $z \xrightarrow{t} z'$ and $z \xrightarrow{\tau} z'$, respectively. This graph is finite if and only if the set of the reachable markings of the net is finite. An enumeration procedure for computing the reachable graph of a given *TPN* can be constructed easily.

The most important behavioral properties of a *TPN* (and of a *PN* as well) are the *reachability*, the *boundedness* and the *liveness*. These properties are decidable for an arbitrary *classical PN*, but not for an arbitrary *TPN* in general. However, there are restricted classes of *TPN* for which the properties are decidable.

In paper [3] used method to model the internet host reliability with *Time Petri nets* While in this paper we will discuss modeling the internet host reliability with *HOTPN*.

This paper is organized as follows. The next section introduces some preliminary definitions and remarks. Next section, discusses definitions proposed of *Higher-Order Time Petri Net*, represents reachability graph, analyze of *HOTPN* and Comparison between two models (*TPN* and *HOTPN*) by behavior properties.

2. BASIC NOTATIONS AND DEFINITIONS
Definition 1 (*Higher-Order Petri Net*) An *HOPN* is formally defined in the same way as the *classical Petri nets*, *HOPN* = (P, T, F, W, M_0), where [4]

- $F \subseteq (P\times T) \cap (P_2\times T) \cap \ldots\ldots\ldots \cap (P_m\times T) \cap (T\times P)$ is a set of *arcs*.
- $W: F \to N$ is a weight function;
- $M_0: P \to N$ is the initial token distribution, called the *initial marking*.

The main difference is the definition of the set of arcs.

Definition 2 (*Firing Rule of HOPN*) A *transition t* is said to be enabled or firable if there exist at least one of its *k*th-order input arcs such that each of this arc's places have at least as many tokens as the weight of this *k*th-order arc. Such an arc is defined as an enabled arc.

Definition 3 (*Time Petri Net*) The structure $Z = (P, T, F, V, M_0, I)$ is called a *Time Petri net* (*TPN*), if [2]

- $S(Z) := (P, T, F, V, M_0)$ is a Petri nets, i.e. P, T, F are finite sets with $P \cap T = \varphi$, $P \cup T \neq \varphi$, $F \subseteq (P \times T) \cup (T \times P)$, $V : F \to N^+$ (weight of the *arcs*), $M_0 : P \to N$ (*initial marking*).
- $I : T \to Q_0^+ \times (Q_0^+ \cup \{\infty\})$ and $I_1(t) \leq I_2(t)$ for each $t \in T$, where $I(t) = (I_1(t), I_2(t))$.

The Petri net $S(Z)$ is referred to as the skeleton of Z. I is the interval function of Z, $I_1(t)$ and $I_2(t)$ are the earliest firing time of t ($eft(t)$) and the latest firing time of t ($lft(t)$), respectively. A transition t is called immediate, if $eft(t) = lft(t) = 0$. A *TPN* is called strong, if there does not exist an infinite latest firing time in the net, i.e., $I : T \to Q_0^+ \times Q_0^+$. In a strong *TPN*, each transition is forced to fire within a finite time interval.

Definition 4 (*state*) Let $Z = (P, T, F, V, M_0, I)$ be a *TPN* and $h : T \to R_0^+ \cup \{\#\}$. $z = (m, h)$ is called a state in Z iff:

- M is a reachable marking in $S(Z)$,
- $\forall t\ ((t \in T \wedge t^- \leq m) \rightarrow h(t) \leq lft(t))$ and
- $\forall t\ ((t \in T \wedge t^- \not\leq m) \rightarrow h(t) = \#)$.

The state $z_0 := (m_0, h_0)$ with $h_0(t) = \begin{cases} 0 & iff \quad t^- \leq m_0 \\ \# & iff \quad t^- > m_0 \end{cases}$ is set as the initial state of the *TPN Z.*

The interpretation of the notion "*state*" is as follows: within the net, each transition t has a clock $h(t)$. If t is enabled at a marking M, its clock $h(t)$ shows the time elapsed since t became most recently enabled. If t is disabled at m, the clock is switched off (indicated by $h(t) = \#$). Thus, the vector h which is a vector of clocks is actually a transition marking and the already defined notion "marking" is in fact a place marking. In the following we call the places marking M a p-marking and the transitions marking h a t-marking. The state $z = (m, h)$ is called an integer state, if $h(t)$ is an integer for each enabled transition t in m.

Each transition $t \in T$ induces the marking t^- and t^+, defined as follows:

$$t^- = \begin{cases} W(p,t) & iff \ (p,t) \in F \\ 0 & iff \ (p,t) \notin F \end{cases}, \quad t^+ = \begin{cases} W(t,p) & iff \ (t,p) \in F \\ 0 & iff \ (t,p) \notin F \end{cases}$$

and $t^- \leq$ m (i.e. $t^-(p) \leq m(p)$ for every place $p \in P$).

The behavior of a *TPN* is defined by changing from one state into another by firing a transition or by time elapsing.

Definition 5 (*state changing*) Let $Z = (P, T, F, V, M_0, I)$ be a *TPN*, $\hat{t}$ be a transition in T and $z = (m, h)$, $z_0 = (m_0, h_0)$ be two states [2]. Then

- the transition is ***ready*** to fire in the state $z = (m, h)$, denoted by $z \xrightarrow{\hat{t}}$, iff

  i. $\hat{t}^- \leq m$ and

  ii. $eft(\hat{t}) \leq h(\hat{t})$.

- the state $z = (m, h)$ is ***changed*** into the state $z' = (m', h')$ by *firing* the transition $\hat{t}$, denoted by $z \xrightarrow{\hat{t}} z'$, iff

  i. $\hat{t}$ is ready to fire in the state $z = (m, h)$

  ii. $m' = m + \Delta\hat{t}$ and

  iii. $$\forall t(t \in T \rightarrow h'(t) = \begin{cases} \# & \text{iff } t^- > m' \\ h(t) & iff \ t^- \leq m \wedge t^- \leq m' \wedge {}^\bullet t \cap {}^\bullet\hat{t} = \phi \\ 0 & otherwise \end{cases}$$

  Where

  ${}^\bullet t = \{p \mid p \in P \wedge (p,t) \in F$ *and* $\Delta\hat{t}$ *denotes* $\hat{t}^+ - \hat{t}^-$

- the state z = (m, h) is changed into the state z′ = (m′, h′) by the time elapsing $\tau \in R_0^+$, denoted by $z \xrightarrow{\tau} z'$, iff

  i. $m' = m$ and

  ii. $\forall t(t \in T \wedge h(t) \neq \# \rightarrow \mathrm{h(t)} + \tau \leq \mathrm{lft(t)}$ (i.e. the time elapsing τ is possible), and

  iii. $\forall t(t \in T \rightarrow h'(t) = \begin{cases} h(t) + \tau & iff \ t^- < m' \\ \# & \text{iff } t^- > m' \end{cases}$

Definition 6 (*Reachability Graph*) The graph $RG_z(z_0)$ is called a *reachability graph* of the *time Petri net Z* iff its nodes are the integer-states from $Z_z(z_0)$ and its arcs are defined by the triples (z, τ, z') and (z, t, z'), where $z \xrightarrow{\tau} z'$ *and* $z \xrightarrow{t} z'$, respectively [5].

Definition 7 (*Boundedness*) Let $Z = (P, T, F, V, M_0, I)$ be a *TPN*.

- A place $p \in P$ is called bounded (at z_0) iff there exists a natural number K with $m(p) \leq K$ for each marking $m \in Rz(z_0)$,
- The net Z is bounded (at z_0) iff all places p are bounded (at z_0).

Definition 8 (*Liveness*) Let $Z = (P, T, F, V, M_0, I)$ be a *TPN*, z a reachable state and $t \in T$ [2].

- t is called live in the state z iff:

  $\forall z'(z' \in RS_z(z) \rightarrow \exists z''(z'' \in RS_z(z) \wedge z'' \xrightarrow{t}))$ (i.e. t is ready to fire in z'')

- Z is live iff all transitions are live in z0.

The set of all reachable states in Z, starting at $z \neq z_0$,is denoted by $RS_z(z)$.

Definition 9 (*Reachability*) Let $Z = (P, T, F, V, M_0, I)$ be a *TPN*,

- The state $z = (m, h)$ is called *reachable* in Z (starting at z_0), if there exist states $z_1, z_1', ..., z_n, z_n'$, transitions $t_1, ..., t_n$ and times $\tau_i \in R_0^+$, $i = 0, 1, ..., n$ and it holds

$$z_0 \xrightarrow{\tau_0} z_1 \xrightarrow{t_1} z_1' \xrightarrow{\tau_1} z_2 \xrightarrow{t_2} z_2' \xrightarrow{\tau_2} \cdots z_n \xrightarrow{t_n} z_n' \xrightarrow{\tau_n} z$$

The sequence of transitions $\sigma = t_1 \ldots t_n$ can fire in Z starting at z_0, because there is a sequence $\sigma(\tau) = \tau_0 t_1 \tau_1 \cdots t_n \tau_n$. We call such a transition sequence σ a feasible one (or a firing sequence for short). The sequence $\sigma(\tau)$, which is a concrete execution of σ in Z, is called a (feasible) run of σ. It is clear that in a given *TPN* the state changes generally consist of alternating series of time elapsing and transition fires. Obviously, for a given run the transition sequence is well defined, and for a given transition sequence there are infinitely many runs in general.

Definition 10 (*An Enumeration Procedure of TPN*) Starting at the initial state all integer-state successors of a reached integer-state can be derived in a successive way (i.e. breadth-first search) [6]:

Basis: $z_0 \in RG(Z)$,
Step: Let z be in $RG(Z)$ already.
1. For i=1 to n do
   If $z \xrightarrow{t_1} z'$ possible in Z (*cf. def 5*) then $z' \in RG(Z)$
   end
2. If $z \xrightarrow{t_1} z'$ possible in Z (*cf. def 5*) then $z' \in RG(Z)$

3. THE PROPOSED HIGHER-ORDER TIME PETRI NET

We now integrate the concept of the network with the concepts of Net to obtain new definitions and suitable for the proposed class.

3.1 The Proposed Definitions of HOTPN

Definition 11 (*Higher-Order Time Petri Net*) The structure *HOZ* = (*P*, *T*, *F*, *V*, M_0, *I*) is called a *Higher-Order Time Petri net* (*HOTPN*), if

- *S*(*HOZ*) := (*P*, *T*, *F*, *V*, M_0) is a Higher-Order Petri nets,
- $I : T \rightarrow Q_0^+ \times (Q_0^+ \cup \{\infty\})$ and $I_1(t) \leq I_2(t)$ for each $t \in T$, where $I(t) = (I_1(t), I_2(t))$.

Definition 12 (*Firing rule of HOTPN*) A transition *t* is said to be *enabled* or *firable* if

- There exist at least one of its *k*th-order input arcs such that each of this arc's places have at least as many tokens as the weight of this *k*th-order arc.
- the state *z* = (*m*, *h*) is ***changed*** into the state *z*′ = (*m*′, *h*′) by *firing* the transition $\hat{t}$, denoted by $z \xrightarrow{\hat{t}} z'$, iff

  i. $\hat{t}$ is ready to fire in the state *z* = (*m*, *h*)

  ii. $m' = m + \Delta\hat{t}$ and

  iii.

$$\forall t(t \in T \rightarrow h'(t) = \begin{cases} \# \quad \text{iff } t^- > m' \\ h(t) \quad \text{iff } t^- \leq m \wedge t^- \leq m' \wedge {}^\bullet t \cap {}^\bullet\hat{t} = \phi \\ 0 \quad \text{otherwise} \end{cases}$$

Definition 13 (*Reachability Graph*) The graph $RGHO_z(z_0)$ is called a *reachability graph* of the Higher-Order Time Petri net *HOZ* iff its nodes are the states from $HOZ_z(z_0)$ and its arcs are defined by the triples (*z*, *τ*, *z*′) *resp*. (*z*, *t*, *z*′), where $z \xrightarrow{\tau} z'$ *resp.* $z \xrightarrow{t} z'$.

Definition 14 (*Boundedness*) Let *HOZ* = (*P*, *T*, *F*, *V*, M_0, *I*) be a *HOTPN*.

- A place $p \in P$ is called bounded (at z_0) iff there exists a natural number *K* with $m(p) \leq K$ for each marking $m \in Rz(z_0)$,
- The net *HOZ* is bounded (at z_0) iff all places *p* are bounded (at z_0).

Definition 15 (*Liveness*) Let *HOZ* = (*P*, *T*, *F*, *V*, M_0, *I*) be a *HOTPN*, *z* a reachable state and $t \in T$

- *t* is called ***live*** in the state *z* iff:

  $\forall z'(z' \in RS_z(z) \rightarrow \exists z''(z'' \in RS_z(z) \wedge z'' \xrightarrow{t}))$ (i.e. *t* is ready to fire in z'')

- *HOZ* is *live* iff all transitions are *live* in z_0.

Definition 16 (*Reachability*) Let *HOZ* = (*P*, *T*, *F*, *V*, M_0, *I*) be a *HOTPN*,

- The state *z* = (*m*, *h*) is called *reachable* in *HOZ* (starting at z_0), if there exist states z_1, z_1', ..., z_n, z_n', transitions t_1, ..., t_n and times $\tau_i \in R_0^+$, *i* = 0, 1, ..., *n* and it holds

$$z_0 \xrightarrow{\tau_0} z_1 \xrightarrow{t_1} z_1' \xrightarrow{\tau_1} z_2 \xrightarrow{t_2} z_2' \xrightarrow{\tau_2} \cdots z_n \xrightarrow{t_n} z_n' \xrightarrow{\tau_n} z$$

3.3 The Modeling of HOTPN

In this section, we build a model for internet host reliability using Higher-Order Time Petri net from Time Petri Net. In Fig.2 (cf. [3]) is a extension of Fig.1 by the introduction of second-order arc of transition t_1. We note the most important differences between the two figures, as follows

- The ***static behaviour*** of model can be represented using ***places***, ***transition*** and ***arcs***.
  a. An increase in the number of ***places*** (p_{11}), then become the number 11 instead of 10.
  b. The same number of ***transitions***.
  c. The emergence of a ***second-order arc*** of p_1 and p_{10} to t_1. Also the emergence of a ***second-order*** from p_9 and p_{11} to t_0.
- The ***dynamic behaviour*** of model can be represented using ***token*** in the ***state*** of the model.
  a. The number of ***token*** in ***initial state*** is $M_0 = (1,0,0,0,0,0,0,0,0,0)_{1\times10}$ in *TPN*. While in *HOTPN* is $M_0 = (1,0,0,0,0,0,0,0,0,0,0)_{1\times11}$.

In Fig. 1, we note the presence of a single period of time on the transition t_1, while the rest transitions (t_2 to t_9 and t_0) do not contain the times. Thus, we suppose that part of the net that contains places p_1, p_2, p_{10} and p_{11} and transition t_1) is a subnet which denoted HOZ_1. While denote to the other part HOZ_2.

In the subnet HOZ_1, in the initial state, the transition t_1 is enabled, thus, z_0 can change into another state only as time elapses. For example, the change of $z_0 \xrightarrow{0.1*i} z_{1i}$ is feasible, where z_1 is given by

$$z_{1i} = (m_{1i}, h_{1i}) = \left(\begin{pmatrix}1 & 0 & 0 & \cdots & 0\end{pmatrix}_{1\times11}, \begin{pmatrix} \# \\ 0.1*i \\ \# \\ \vdots \\ \# \end{pmatrix}_{9\times1} \right)$$

and $m_{1i} = m_0, \quad i = 1,2,\cdots,60, \quad \tau_0 = 0.1*i$

In z_{1i} the transition t_1 can fire, yielding state z_2 where

$$z_2 = (m_2, h_2) = \left(\begin{pmatrix}0 & 1 & 0 & \cdots & 0\end{pmatrix}_{1\times11}, \begin{pmatrix} \# \\ \# \\ 0 \\ \vdots \\ \# \end{pmatrix}_{9\times1} \right)$$

Thus, the sequence $z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2$ is executable in HOZ_1. It present in every sequences firing in net *HOZ*.

3.3 Analysis of HOTPN

To analyze the behavior of Model using the reachability graph and it apply behavioral properties.

3.3.1 The Sequences of Transitions:

a. The *state space* of Time Petri net (*Z*) is the set of all reachable states of the net is (σ_1, σ_2, σ_3, σ_4), where

   $\sigma_1 = \{t_1, t_2, t_3, t_4, t_5, t_6, t_8, t_0\}$,

   $\sigma_2 = \{t_1, t_2, t_4, t_3, t_5, t_6, t_8, t_0\}$,

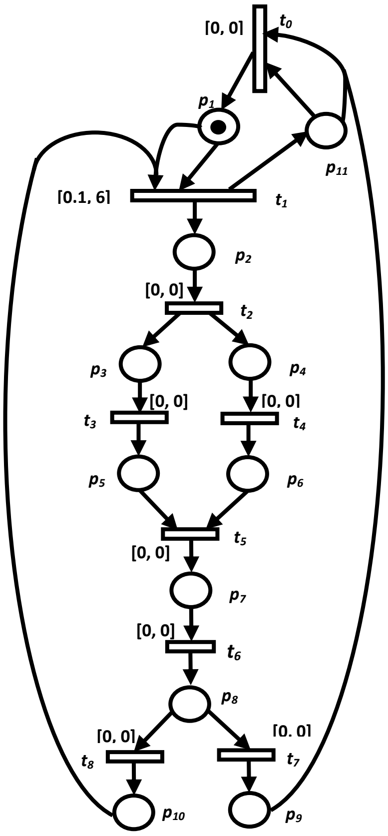

Fig. 1: Modeling Internet Host Reliability using HOTPN.

$$\sigma_3 = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7\},$$
$$\sigma_4 = \{t_1, t_2, t_4, t_3, t_5, t_6, t_7\}$$

More detail

$\sigma_1(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{Z_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_9 \xrightarrow{t_0} M_0}^{Z_2}$$

$\sigma_2(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{Z_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_4} M_4 \xrightarrow{t_3} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_9 \xrightarrow{t_0} M_0}^{Z_2}$$

$\sigma_3(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{Z_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_7} M_8}^{Z_2}$$

$\sigma_4(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{Z_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_4} M_4 \xrightarrow{t_3} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_7} M_8}^{Z_2}$$

b. The *state space* of an Higher-Order Time Petri net (*HOTPN*) is the set of all reachable states of the net is (σ_1, σ_2, σ_3, σ_4), where

$$\sigma_1 = \{t_1, t_2, t_3, t_4, t_5, t_6, t_8, t_0, t_1\},$$
$$\sigma_2 = \{t_1, t_2, t_4, t_3, t_5, t_6, t_8, t_0, t_1\},$$
$$\sigma_3 = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_0\},$$
$$\sigma_4 = \{t_1, t_2, t_4, t_3, t_5, t_6, t_7, t_0\}$$

More detail

$\sigma_1(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{HOZ_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_9 \xrightarrow{t_0} M_{10} \xrightarrow{t_1} M_1}^{HOZ_2}$$

$\sigma_2(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{HOZ_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_4} M_4 \xrightarrow{t_3} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_9 \xrightarrow{t_0} M_{10} \xrightarrow{t_1} M_1}^{HOZ_2}$$

$\sigma_3(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{HOZ_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_7} M_8 \xrightarrow{t_0} M_0}^{HOZ_2}$$

$\sigma_4(t) =$

$$\overbrace{z_0 \xrightarrow{0.1*i} z_{1i} \xrightarrow{t_1} z_2}^{HOZ_1}$$
$$\overbrace{\xrightarrow{t_2} M_2 \xrightarrow{t_4} M_4 \xrightarrow{t_3} M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_7} M_8 \xrightarrow{t_o} M_0}^{HOZ_2}$$

The enumeration procedure for computing the reachable graph of a given *TPN* defined above can be performed using the definition 5. The reachability graph of the *TPN* is shown in Fig.2.

3.3.2 The behaviour properties of the models

Depending on definitions of *TPN* (and *HOTPN*) properties and Fig. 2 (and Fig.3), we can deduce following table.

Table 1: comparison of the behavioral properties between the TPN and HOTPN

properties	**TPN model**	**HOTPN model**
Boundedness	All sequences	All sequences
Liveness	σ_1 and σ_2 are live, σ_3 and σ_4 are not live	All sequences
reachability	All sequences	All sequences

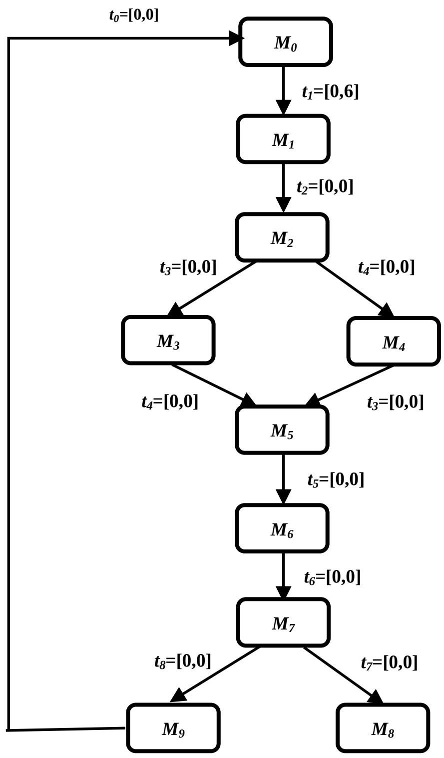

Fig. 2: The reachability graph of the TPN model in ref. [3].

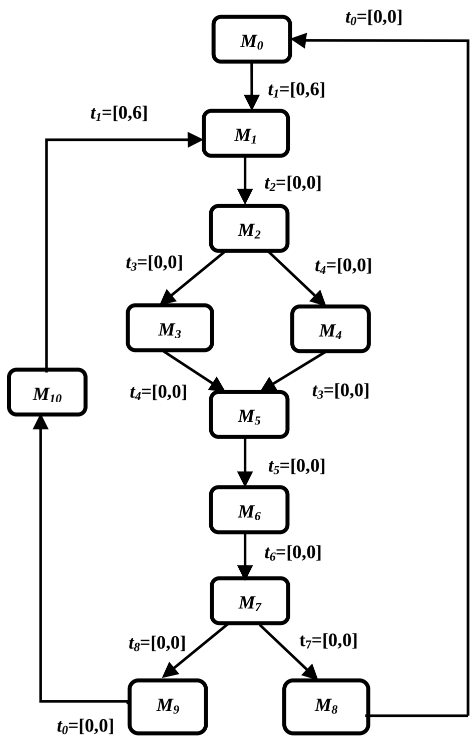

Fig. 3: The reachability graph of the HOTPN model in Fig.1

From Table 1 we note that:

- *TPN* model is reachability, Boundedness and not liveness.
- *HOTPN* model is reachability, Boundedness and liveness.

Table 4 summarizes the place set of the reachability graph. The header represents places and the left-most column represents the place set having marking. The reachability analysis was very useful to validate state transitions of the places.

Table 2: Place set of the reachability graph of the HOTPN model in Fig. 3

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}
M_0	1	0	0	0	0	0	0	0	0	0	0
M_1	0	1	0	0	0	0	0	0	0	0	1
M_2	0	0	1	1	0	0	0	0	0	0	1
M_3	0	0	0	1	1	0	0	0	0	0	1
M_4	0	0	1	0	0	1	0	0	0	0	1
M_5	0	0	0	0	1	1	0	0	0	0	1
M_6	0	0	0	0	0	0	1	0	0	0	1
M_7	0	0	0	0	0	0	0	1	0	0	1
M_8	0	0	0	0	0	0	0	0	1	0	1
M_9	0	0	0	0	0	0	0	0	0	1	1
M_{10}	1	0	0	0	0	0	0	0	0	1	0

4. CONCLUSIONS

This paper discusses internet host reliability modeling using Higher-Order Time Petri Net. Thus, an enumeration procedure can compute a reachability graph for a given *HOTPN*. The behaviour of the net can be studied by graph and properties. Thus, the *HOTPN* solves live problem in *TPN* model. Therefore, the *HOTPN* is live when its skeleton is live.

5. REFERENCES

[1] Staines, A., 2010 Supporting Requirements Engineering with Different Petri Net Classes" International journal of computers, vol.4 (4).

[2] Bachmann, J. and Popova-Zeugmann, L., 2010 Time-independent Liveness in Time Petri Nets. IOS Press, Fundamental Informaticaevo, vol.101, pp.1–17.

[3] Meligy, A., Ibrahim, H., and Aqlan, A., 2012 Internet Host Reliability Modeling with Time Petri Nets. International Journal of Computer Applications, vol.47, no.16.

[4] Chow, T., Li, J-Y. , 1997 Higher-Order Petri Net Models Based on Artificial Neural Networks. Artificial Intelligence, V. 92, pp. 289-300.

[5] Popova, L., 1991 On Time Petri Nets. J. Inform. Process. Cybern. EIK 27(1991)4, pp. 227–244.

[6] Popova-Zeugmann, L., 2007 Time Petri Nets State Space Reduction Using Dynamic Programming. Journal of Control and Cybernetics, 35(3):721–748.