1.3 Analysis of Petri nets

In this section, we will study the methods determining which properties apply to a Petri net. The first method is graph-based and analyses a reachability tree of a Petri Net. The second method is slight modification of the first method that is based on analysis of so-called coverability tree. Finally, the third method does analysis of a Petri Net in terms of linear algebraic equations. This method uses the adjacency matrix, which describes how nodes in a Petri net are interconnected.

![An ordinary Petri Net](image.png)

Fig.1.3.1 An ordinary Petri Net.
1.3.1 Reachability and coverability trees

The reachability and coverability problems are very basic Petri net analysis problems and are useful for establishing properties such as boundedness and reversibility. These problems form the basis for graph-based analysis methods.

Fig.1.3.2 The first three levels of the reachability tree.

The reachability set of a Petri net is often drawn as a reachability tree, where the nodes of the tree represent the states of the Petri net. The goal of the reachability tree is to find all the markings, which can be reached starting from $M_0$. The coverability tree is derived from the reachability tree. At the expense of the loss of some information, the size of the coverability tree remains finite even if the size of the reachability tree is not
finite. Both trees are tools used to analyze the behavior of a Petri net.

The reachability tree can be generated starting with the initial marking of the Petri net and adding directly reachable markings as leaves. Next we proceed with these new marking and determine their directly reachable markings. These marking now become the new leaves of the already generated part of the reachability tree. If we reach a marking previously explored, we need not continue building the tree any further at that node.

![Reachability Tree Diagram](image)

**Fig.1.3.3 Reachability tree**

![Reachability Graph Diagram](image)

**Fig.1.3.4 Reachability graph.**

To illustrate these concepts, let us consider an ordinary Petri Net of Fig. 1.3.1. The initial marking of
this Pet Net is $M_0 = <2,1,0>$. $M_0$ enables transitions $t_1$ and $t_3$. Firing $t_1$ leads to marking $M_1^1 = <1,2,1>$ and if the transition $t_3$ is firing then the new marking will be $M_3^1 = <3,0,0>$; it is the first level of the reachability tree. This enables $t_1$, $t_2$, $t_3$. Firing these transitions leads respectively to marking $M_2^1 = <0,3,2>$, $M_2^2 = <2,1,0>$, $M_2^3 = <2,1,1>$. Marking $M_2^1$ only enables $t_1$. Firing $t_1$ leads to marking $M_3^1 = <2,1,1>$. This construction is the second level of the reachability tree and the third level can be construct in the same way. Note that the same marking may appear at various levels of the tree and it is easily understandable that a reachability tree may have an infinite number of levels.

The first three levels of the reachabilty tree of the Petri Net of Fig. 1.3.1 are represented in Fig. 1.3.2. The root of this tree is the initial marking.

Notice that a reachability tree can be transformed directly into graph by removing multiple nodes and connecting the nodes appropriately. Such a graph is called a reachability graph. The reachability graph of the reachability tree displayed in Fig.1.3.3 is shown in Fig.1.3.4.

Unfortunately, the method of creating the reachability tree by simulating the token game fails in the case of unbounded nets. In order to cope with infinite reachability sets a special symbol $\omega$ is introduced for unbounded nets to
represent the marking of an unbounded place of the Petri net. The symbol $\omega$ can be referred to as infinity. The arithmetic rules for $\omega$ are as follows:

\[ w + a = w , \]
\[ w - a = w , \]
\[ a < w , \]
\[ w \leq w . \]

where $a$ is an arbitrary integer.

![Petri net with initial setting <0,1,0,0>](image)

Fig.1.3.5 Petri net with initial setting $<0,1,0,0>$.

We distinguish between four types of vertices of a reachibility tree. A vertex is said to be boundary vertex if it requires further analysis. A vertex is the duplicated vertex if it matches with one of the already analyzed vertices. A vertex is the passive vertex (or terminal vertex) if all of its transitions are disable. Finally, a
vertex is the internal vertex if it is neither duplicated nor passive, and its status is known.

Consider the Petri Net shown in Fig. 1.3.5. This Petri Net is characterized by the initial marking $M_0 = \langle 0, 1, 0, 0 \rangle$. It can be easily seen that, initially there are two enable transitions. These are $t_1$ and $t_2$. The occurrence of $t_1$ causes the change the state of the Petri Net from $\langle 1000 \rangle$ to
<1100>. On the other hand, occurrence of \( t_2 \) changes the state of the Petri Net from <1000> to <0010>. If so, the Petri Net becomes dead which means all transitions at this stage are disable. This is the reason why vertex <0010> does not contain any child. Now suppose that \( t_1 \) fires first. If so, there are two more possibilities for transitions to fire. These possibilities are depicted by two distinct branches at vertex <1100>. The leftmost branch corresponds to the repeatedly occurrence of \( t_4 \). This increases the number of tokens in \( p_2 \) by 1. This vertex is substituted by <1w00> (Fig.1.3.7). On the other hand, processing the right branch increases by 1 the number of tokens in \( p_2 \) and \( p_4 \). Hence, corresponding vertex is substituted by <0w1w>. In Fig.1.3.7, markings <1000>, <1w00> and <0w1w> are boundary vertices, <0010> is terminal vertex, and <0w1w> and <1w00> are duplicated vertices.

If the \( w \) symbol is absent, the tree is also called a reachability tree. The coverability tree is finite and contains every reachable marking from \( M_0 \) that is either explicitly represented by a node, or a “covered” by a node through the \( w \) notation.
1.3.2 The method of matrix invariants

The method of matrix invariants is based on matrix representation of Petri Nets. The user constructs a set of equations that is proved to be satisfied for all reachable system markings. The equations are used to prove properties of the modeled system.

We represent input and output arcs of the Petri Net by their adjacency matrices $D^-$ and $D^+$. The matrices $D^-$ and $D^+$ represent the set of incoming arcs and the set of outgoing arcs of a Petri Net. The difference of the two matrices $D=D^+-D^-$ holds information about whole Petri Net. For instance, the matrices $D^+$, $D^-$ and $D$ for the Petri Net shown in Fig.1.3.5 are as follows:

$$D^-=\begin{bmatrix}0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0\end{bmatrix},\quad D^+=\begin{bmatrix}1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1\end{bmatrix},\quad D=\begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1\end{bmatrix}$$

A marking $M_1$ is reachable from initial setting $M_0$ if the system of linear algebraic equations $M_1=M_0+x\cdot D$ has a solution. The number of occurrences of the $i$th transition is determined by the value of the $i$th component of the vector $x$. If solution is unique then the transition $t_i$ must fire $x_i$ times. When there are multiple solutions of the
above system, the number of occurrences of \( t_i \) varies in the range of values of \( x_i \). If system has no solution then \( \mu' \) is not reachable from \( \mu \). For example, in the above considered example \( M_1 = \langle 1, 0, 1, 2 \rangle \) is reachable from \( M_0 = \langle 0, 1, 0, 0 \rangle \) only when \( x = (3, 1, 2) \). That is, if \( t_1 \), \( t_2 \) and \( t_3 \) fire 3, 1 and 2 times, respectively. Then the Petri Net will be set to \( \langle 1, 0, 1, 2 \rangle \). Although the Matrix Analysis Method allows to find the number of occurrences for each transition, it does not provide any information about order in which transitions occur.