Reachability problem of state machines with batch processing arcs

Nami Mizuno[1], Atsushi Ohta[2] and Kohkichi Tsuji[3]
[1]DENSOTECHNO CO., Ltd.
Kariya, Aichi, Japan
[2,3]Department of Information Science and Technology, Aichi Prefectural University,
Nagakute, Aichi, Japan.
E-mail : [2]ohta@ist.aichi-pu.ac.jp

Abstract: Petri net is an effective model for concurrent systems. Petri net with batch processing arcs, batch Petri net for short, is one of the Turing machine equivalent extended classes. Number of tokens moved by a firing of transition through the batch processing arcs equals to the minimum number of tokens among its input places connected with batch processing arcs, while fixed number of tokens are moved through normal arcs. This paper studies reachability problem of batch state machines, a subclass of batch Petri net. Its computational complexity is shown to be NP-hard. Sufficient conditions for reachability are derived based on classification of transitions.

1. Introduction

Petri net [1] is an effective mathematical and graphical modeling tool for concurrent systems. It is a bipartite digraph with nodes called places and transitions. Typically, places represent conditions and resources, while transitions represent events. Places have nonnegative integer number of tokens, which represents truth of condition or number of resources. Distribution of tokens to the places is called marking, which is the state of Petri net.

Batch Petri net is an extended Petri net that has batch processing arcs, which is defined in [2], If a transition t has only one batch processing arc of unity weight from a place p, firing of t removes all tokens from p and adds the same number of tokens to its output places connected with batch processing arcs. On the other hand, if batch processing arcs of unity weights are connected from places $p_1, p_2, \ldots, p_n$ to a transition t, firing of t in the marking M removes and adds $\min_{k=1}^{n} M(p_k)$ tokens. Thus tokens moved by firing of a transition depends on the marking. Formal definition of batch Petri net is given in Section 2.

Batch Petri net can be used to model batch process in production system, buffer flushing and so on. And this is a subclass of extended Petri nets where arc weight is a general function of marking, which has many successful practical applications including bioinformatics [3], network file systems [4]. However, theoretical analysis, especially behavioral analysis of the general extended Petri net is very difficult.

We have shown the Turing machine equivalence of general class of batch Petri net [5]. This implies that most of significant analysis problems including reachability are undecidable. Thus successful analysis needs some restriction on the structure of Petri net as a graph. In [6], reachability problem of batch marked graphs is studied. In this paper we study reachability problem of batch state machines. We show sufficient conditions based on classification of transitions according to the sort of input/output arcs. Computational complexity is also considered.

2. Definitions and Notations

2.1 Batch Petri Nets

Definition 1: A Petri net with batch processing arcs (batch Petri net, for short) is a tuple $\Sigma = (P, T, F, B, W_F, W_B, \underline{n}, M_0) = (N, M_0)$, where P and T are disjoint finite sets of places and transitions, respectively. $F \subseteq (P \times T) \cup (T \times P)$ is the set of 'normal' arcs. $B \subseteq (P \times T) \cup (T \times P)$ is the set of batch processing arcs, $W_F : (P \times T) \cup (T \times P) \mapsto \{0, 1, 2, \ldots\}$ is a weight function of arcs, and $W_B : (P \times T) \cup (T \times P) \mapsto \{0, 1, 2, \ldots\}$ is a weight function of batch processing arcs. These weights satisfy the relations $\forall x, y \in P \cup T$, $W_F(x, y) = 0 \iff (x, y) \notin F$ and $W_B(x, y) = 0 \iff (x, y) \notin B$. In this paper, it is assumed that each arc has a unity weight. $\underline{n} : T \mapsto \{0, 1\}$ is the minimal firing velocity with respect to batch processing arcs. Marking M is a mapping from P to the set of nonnegative integers. A place $p \in P$ has $M(p)$ tokens in the marking M. M_0 is the initial marking. □

It is assumed that if a transition t has no batch input arcs, then it has no output batch arcs (Assumption A). This is necessary to define firing velocity n in the following firing rule.
Definition 2: A transition t is enabled in a marking M if

$$\forall p \in P; \quad M(p) \geq W_F(p, t) + \underline{n}(t) W_B(p, t)$$

holds. This is denoted as $M[t\rangle$. An enabled transition may or may not fire. If an enabled transition t fires in a marking M, then marking changes to M' where

$$M'(p) = M(p) - W_F(p, t) + W_F(t, p) - nW_B(p, t) + nW_B(t, p) \quad (1)$$

$$n = \min_{W_B(p,t)>0} \left\lfloor \frac{M(p) - W_F(p, t)}{W_B(p, t)} \right\rfloor \quad (2)$$

$$(n = 1 \text{ if } t \text{ has no input batch processing arcs})$$

and this is denoted as $M[t^{[n]}\rangle M'$. n denotes the firing velocity. Note that if $\underline{n}(t) = 0$ then velocity n can be zero. □

Fig. 1 shows an example of batch Petri net. Double arrowed arcs such as (p_2, t) and (t, p_5) are batch processing arcs. Single firing of t changes marking from $M_1 = [1, 3, 2, 0, 0]$ to $M_2 = [0, 1, 0, 1, 2]$ ($M_1[t^{[2]}\rangle M_2$) and $M_3 = [2, 4, 5, 0, 0]$ to $M_4 = [1, 0, 1, 1, 4]$ ($M_3[t^{[4]}\rangle M_4$). If $\underline{n}(t) = 0$ then t can fire in M_4 and the resulting marking after firing t in velocity 0 is $M_5[0, 0, 1, 2, 4]$ ($M_4[t^{[0]}\rangle M_5$).

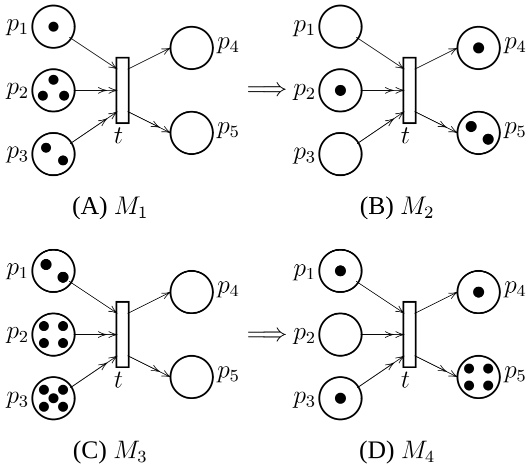

Figure 1. A batch Petri net

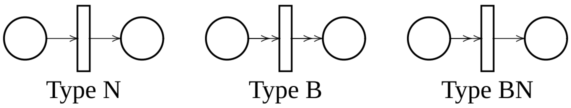

Figure 2. Types of transitions in batch state machines

Definition 3: A sequence $w = t_{i_1}^{[n_{i_1}]} t_{i_2}^{[n_{i_2}]} \cdots t_{i_q}^{[n_{i_q}]}$ is a firing sequence from M if $M = M_1[t_{i_1}^{[n_{i_1}]}\rangle M_2$, $M_2[t_{i_2}^{[n_{i_2}]}\rangle M_3, \ldots$, $M_q[t_{i_q}^{[n_{i_q}]}\rangle M_{q+1} = M'$ hold and this is denoted as $M[w\rangle M'$. $M[\varepsilon\rangle M$ holds for empty sequence ε. M' is said to be reachable from M if $M[w\rangle M'$ for some w. The reachability set from marking M is the set of markings that are reachable from M and this is denoted as $R(M)$. □

Let Σ be a batch Petri net. The Petri net Σ' obtained after replacing batch processing arcs by normal arcs is called the underlying net of Σ and denoted as $u(\Sigma)$.

2.2 Batch State Machines

A batch Petri net is a state machine with batch processing arcs (batch state machine, for short) if every transition has exactly one input arc and exactly one output arc and each arc has unity weight. These arcs may be either of normal arcs or batch processing arcs. Thus there are three types of transitions.

Definition 4: Transitions of a batch state machine are classified as follows (See Fig. 2).

- A transition t is of type N if both of its input and output arcs are normal ones.
- A transition t is of type B if both of its input and output arcs are batch processing ones.
- A transition t is of type BN if its input arc is a batch processing one and its output arc is a normal one.

Type BN transitions are further classified as follows.

- A type BN transition t is of type BN0 if minimal firing velocity $\underline{n}(t)$ is 0.
- A type BN transition t is of type BN1 if $\underline{n}(t)$ is 1.

□

Due to the Assumption A there is no transition with normal input arc and batch processing output arc. For a transition t of Type B, its firing with velocity zero does not change the marking thus it is assumed that $\underline{n}(t) = 1$.

Fig. 3 shows an example of batch state machine.

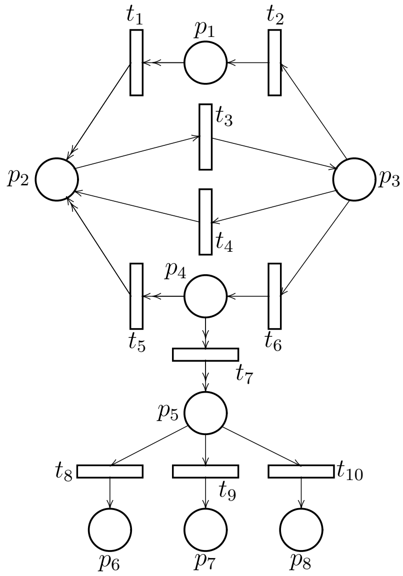

Figure 3. Batch Petri net model of water usage

Table 1. Meanings of the nodes of the Petri net of Figure 3

p_1	Water for cooking
p_2	Water tank
p_3	Water vessel
p_4	Bath tub
p_5	Waste water
p_6	Water for flowers
p_7	Water for washing clothes
p_8	Water for toilet
t_1	Return water to the tank
t_2	Reserve water for cooking
t_3	Pure water into the water vessel
t_4	Return water to the tank
t_5	Return water to the tank
t_6	Pour water into the bath tub
t_7	Take a bath
t_8	Use water for flowers
t_9	Use water for washing
t_{10}	Use water for toilet

3. Reachability of batch state machines

Given a batch state machine $\Sigma = (N, M_0)$ and a target marking M_d, reachability problem is to verify whether M_d is reachable from M_0.

3.1 Reachability of state machines without batch processing arcs

For state machine without batch processing arcs necessary and sufficient conditions for reachability have been studied [1].

Property 1: For strongly connected state machine, M_d is reachable from M_0 if and only if total token count of M_d equals to that of M_0. □

Property 2: For weakly connected state machine, M_d is reachable from M_0 if and only if $M_d = M_0 + Ax$ has a non-negative integer solution x, where A is the incidence matrix of the net. This can be verified in deterministic polynomial time. □

3.2 Reachability of batch state machines without type BN transitions

Firings of transitions of Type N and B do not change the total number of tokens. So the following result holds.

Theorem 1: Let $\Sigma = (N, M_0)$ be a strongly connected batch state machine and it has at least one Type N transition and no Type BN transitions. The target marking M_d is reachable from M_0 if and only if token number of M_0 equals to that of M_d. □

Proof: Necessity is straightforward from the fact that firings of type N and type B transition do not change total number of tokens.

For sufficiency, let t_0 be a type N transition, p' and p'' be the input and output place of t_0, respectively. Let P_1 be the set of places except p' such that every path from t_0 to any of them includes p' and $P_2 = P - (P_1 \cup \{p'\})$. Places of P_1 and P_2 are sorted in the descending order of distance from p'. Note that length of a directed path is defined as the number of its arcs and distance from one node x to another node y is defined as the minimum length of the directed path from x to y.

First fire transitions on the path from each place having token in M_0 to p' in order to move all tokens to p'. Then each place p_j of P_2 is given $M_d(p_j)$ tokens in the following way. (1) Fire t_0 appropriate times so that p' has $M_d(p_j)$ tokens. (2) Fire each transition on the shortest path from p' to p_j. Since places of P_2 is sorted in descending order of the distance from p', places on this path have no tokens. The transitions on the path fire in the order of appearance on it. Transition of type N fires $M_d(p_j)$ times and transition of type B fires once in velocity $M_d(p_j)$. (3) Tokens in p'' are moved to p' by firing transitions on the path from p'' to p'. Note that this path has no place of P_2.

Lastly each place p_k of P_1 is given $M_d(p_k)$ tokens by $M_d(p_k)$ firings of t_0 followed by firing transitions on the shortest path from p' to p_k. ■

Theorem 2: Let $\Sigma = (N, M_0)$ be a batch state machine without Type BN transitions. If the following two conditions hold, then the target marking M_d is reachable from M_0 if and only if M_d is reachable from M_0 in the underlying state machine $u(\Sigma)$.

(1) Every strongly connected component including two or more places has at least one Type N transition.

(2) For every strongly connected component consisting of a single place p, if p has a Type B output transition t, then there exists a Type N output transition t' such that there exists a directed path from t' to the output place of t.

Moreover, if the above conditions do not hold, then there exists a pair of markings M_0' and M_d' where M_d' is not reachable in N and yet M_d' is reachable in the underlying net. □

Proof: Let w be the firing sequence of the underlying net which drives M_0 to M_d. Let S be the set of places of the strongly connected component of N such that there exists no directed path to S from any other strongly connected components. If S has no token in M_0, then it has no token in M_d. Otherwise, there are two cases. (Case 1) S has two or more places. First of all, fire each transition in $S^\bullet - {}^\bullet S$ as many times as it appears in w. Then marking of S is made as indicated by M_d. These are possible from the result of the Theorem 1. (Case 2) S has only one place p_0. If p_0 has type B output transition t, then modify w by replacing every appearance of t with transitions on the path from the type N output transition t' to the output place of t. The resulting sequence w' is fireable in the underlying state machine and drives the marking to the target marking M_d. Fire t' as many times as it appears in w' and p_0 has $M_d(p_0)$ tokens.

In both cases, repeat the above on the state machine resulted by removing S, where the restriction of target marking M_d to $P - S$ is reachable in the underlying net.

Lastly, consider the case where the conditions (1) and/or (2) are not satisfied. If there exists a strongly connected component that has two or more places and no Type N transitions, it has a transition t and its input place p_1 and its output place p_2. Set M_0' and M_d' as

$$M_0'(p_1) = 2, \quad M_0'(p) = 0 \ (p \neq p_1)$$

and

$$M_d'(p_1) = M_d'(p_2) = 1, \quad M_d'(p) = 0 \ (p \neq p_1, p_2)$$

M_d' is reachable from M_0' in the underlying net Σ', however it is not true in Σ since all type B transition of the strongly connected component moves two tokens at once. On the other hand, if there exists a strongly connected component that consists of a single place p_1 and it has an output transition t and no Type N output transition from which there exists a path to the output place p_2 of t, similarly set the markings M_0' and M_d'. M_d' is not reachable from M_0' since firing of t removes all tokens from p. ■

3.3 Complexity of reachability problem of batch state machines

Since every batch state machine without type BN transitions is bounded, reachability problem is decidable. However, number partitioning problem, which is known to be NP-complete, can be reduced to reachability problem of batch state machines. Number partition problem is stated as follows.

Definition 5: Number partitioning problem

[Instance] Nonnegative integers $\{a_1, a_2, \ldots, a_n\}$.

[Question] Is there any partition of index $I_1, I_2 \in I \equiv \{1, 2, \ldots, n\}$ such that $I_1 \cup I_2 = I$, $I_1 \cap I_2 = \emptyset$ and $\sum_{k \in I_1} a_k = \sum_{k \in I_2} a_k$? □

Theorem 3: Reachability problem of batch state machines without Type BN transitions is NP-hard. □

Proof: Let $\{a_1, a_2, \ldots, a_n\}$ be an instance of number partitioning problem. Construct the batch state machine $\Sigma =$

$(P,\ T,\ F,\ B,\ M_0)$ as follows.

$$P = \{p_1,\ p_2,\ \ldots,\ p_n\} \cup \{q_1,\ q_2\}$$
$$T = \{t^1_1,\ t^1_2,\ \ldots,\ t^1_n\} \cup \{t^2_1,\ t^2_2,\ \ldots,\ t^2_n\}$$
$$F = \emptyset$$
$$B = \{(p_k,\ t^j_k) | j = 1,\ 2;\ k = 1,\ \ldots,\ n\}$$
$$\cup\ \{(t^j_k,\ q_j) | j = 1,\ 2;\ k = 1,\ \ldots,\ n\}$$
$$M_0(p_k) = a_k \quad (k = 1,\ \ldots,\ n)$$
$$M_0(q_j) = 0 \quad (j = 1,\ 2)$$

Set the target marking M_d as

$$M_d(p_k) = 0 \quad (k = 1,\ \ldots,\ n)$$
$$M_d(q_j) = \frac{1}{2}\sum_{k=1}^{n} a_k \quad (j = 1,\ 2)$$

M_d is reachable from M_0 if and only if there exists a partition of the numbers. ■

3.4 Reachability of batch state machines with type BN transitions

Type BN0 transition can fire even if its input place has no token. So the following theorems hold.

Lemma 1: Let $\Sigma = (N,\ M_0)$ be a strongly connected batch state machine that has at least one transition of Type BN0. Any marking $M_d \neq 0$ is reachable from M_0 such that at least one input place p_0 of a Type BN0 transition has no token in M_d. □

Proof: Let t_0 be the Type BN0 output transition of p_0. Let P_1 be the set of places except p_0 such that every path from t_0 to any of them includes p_0 and $P_2 = P - (\{p_0\} \cup P_1)$. First move all tokens to p_0. Then add appropriate number of tokens to each place p of P_1 in the descending order of distance from t_0 one token by one. This is done by repeating the firing sequence of transitions on the path from t_0 to p $M_d(p)$ times. Lastly add appropriate number of tokens to each place of P_2 in the similar way. ■

Theorem 4: Let $\Sigma = (N,\ M_0)$ be a strongly connected batch state machine that has at least one transition of Type N and at least one transition of Type BN0. If $M_0 = 0$ then any marking is reachable from M_0. Otherwise any nonzero marking is reachable from M_0. □

Proof: Let t_1 be a Type N transition, t_0 be a Type BN0 transition, p_1 be the input place of t_1, p_0 be the input place of t_0, and π be the directed path from t_1 to p_0. For the given target marking M_d, set the marking M'_d be the marking as

$$M'_d(p) = \begin{cases} M_d(p) & p \in P - (\pi \cup \{p_1\}) \\ \sum_{q\in\pi} & p = p_1 \\ 0 & p \in \pi \end{cases}$$

M'_d is reachable from M_0 from the lemma 1. Then fire the transitions on π and move tokens one by one to reach M_d. ■

Theorem 5: Let $\Sigma = (N,\ M_0)$ be a batch state machine. If the following two conditions hold, then the target marking M_d is reachable from M_0 if M_d is reachable from M_0 in the underlying state machine. (1) Every strongly connected component including two or more places has at least one transition of Type N. Moreover, if it has any Type BN1 transition, it has also at least one Type BN0 transition. (2) For every strongly connected component consisting of a single place p, if p has an output transition t of Type B or Type BN, then there exists a Type N output transition t' from which there exists a directed path to the output place of t. □

Proof: Similar to the proof of theorem 2. ■

4. Conclusion

Some sufficient conditions for reachability of batch state machines are obtained based on classification of transitions by the sorts of connected arcs. For the batch state machine without type BN transitions, it is shown that if the structural condition does not hold, there exists a pair of markings M_0 and M_d where M_d is not reachable in spite that it is reachable in the underlying net. Computational complexity of the problem is shown to be NP-hard, while the reachability problem of state machine without batch processing arcs is solvable in deterministic polynomial time. We have an expectation, although we have not found the proved, that reachability of batch state machines without Type BN transitions is NP-complete.

References

[1] T. Murata, "Petri nets: properties, analysis and applications," Proc. IEEE. vol.77, no.4, pp.541–580, (1989).

[2] "Renew(The Reference Net Workshop)," http://www.renew.de/

[3] H. Matsuno, "Genomic Object Net: On-going report on biopathway modeling and simulation," Proc. 1st International Symposium on BioSystems, pp.5–22, (2002).

[4] S. Yamaguchi, K. Wasaki and Y. Shitama, "A Java/PNML verification framework for concurrent system design based on an extended Petri net," IEICE Technical Report, CST2006-9, (2006).

[5] A. Ohta, C. Kato and K. Tsuji, "Analysis of Petri Nets with Batch Processing Arcs," Proc. SICE Annual Conference, pp.517–520 (2004)

[6] N. Mizuno, A. Ohta and K. Tsuji, "Reachability Problem of Marked Graphs
with Batch Processing Arcs," Proc. The 33rd Annual Conference of the IEEE Industrial Electronics Society, pp.70-75 (2007)