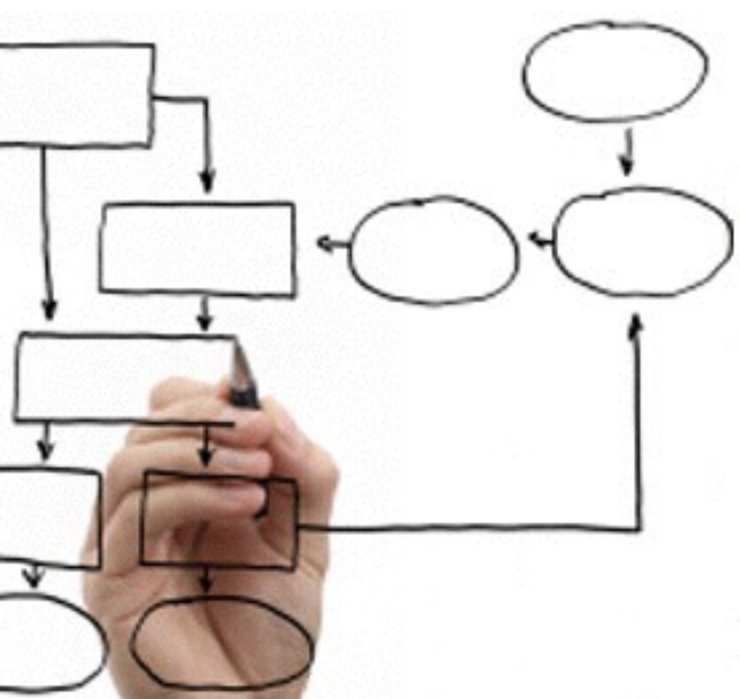
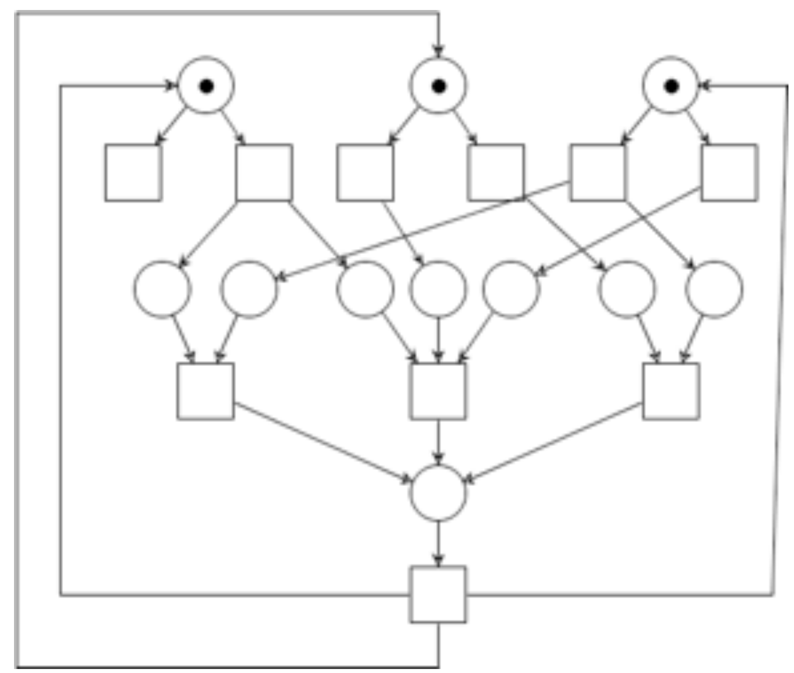
Methods for the specification and verification of business processes
MPB (6 cfu, 295AA)

Roberto Bruni
http://www.di.unipi.it/~bruni



18 - Free-choice nets

Object

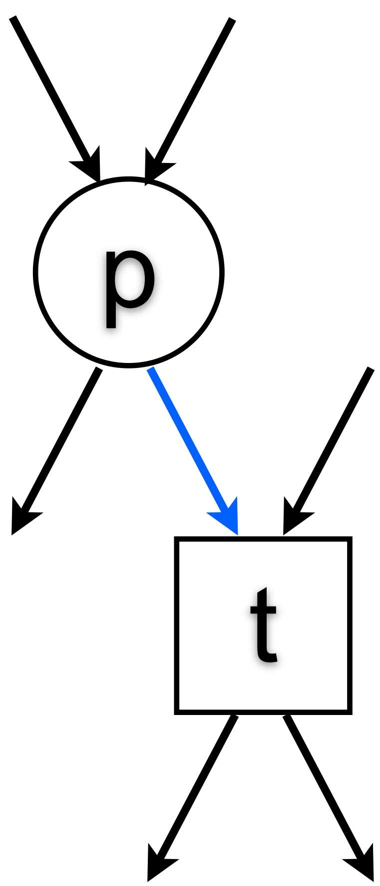


We study some “good” properties of free-choice nets

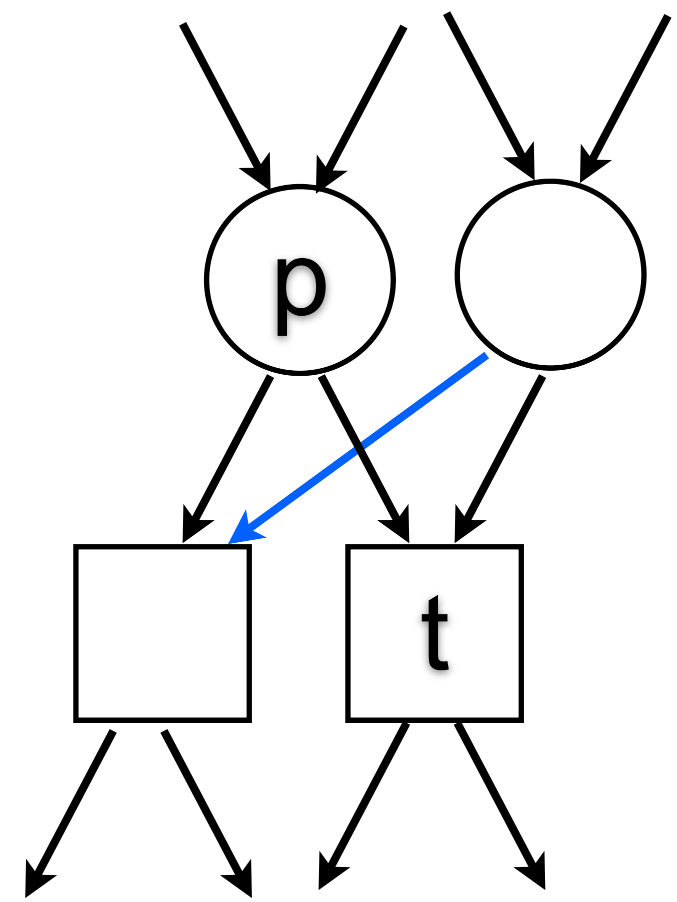
Free Choice Nets (book, optional reading)
https://www7.in.tum.de/~esparza/bookfc.html

Free-choice net

Definition: We recall that a net N is **free-choice** if whenever there is an arc (p,t), then there is an arc from any input place of t to any output transition of p



implies



Free-choice net: alternative definitions

Proposition: All the following definitions of free-choice net are equivalent.

1) A net (P, T, F) is free-choice if:
$\forall p \in P, \forall t \in T$, $(p, t) \in F$ implies $\bullet t \times p \bullet \subseteq F$.

2) A net (P, T, F) is free-choice if:
$\forall p, q \in P, \forall t, u \in T$, $\{(p, t), (q, t), (p, u)\} \subseteq F$ implies $(q, u) \in F$.

3) A net (P, T, F) is free-choice if:
$\forall p, q \in P$, either $p\bullet = q\bullet$ or $p \bullet \cap q\bullet = \emptyset$.

4) A net (P, T, F) is free-choice if:
$\forall t, u \in T$, either $\bullet t = \bullet u$ or $\bullet t \cap \bullet u = \emptyset$.

Free-choice net: my favourite definition

4) A net (P, T, F) is free-choice if:
$\forall t, u \in T$, either $\bullet t = \bullet u$ or $\bullet t \cap \bullet u = \emptyset$.

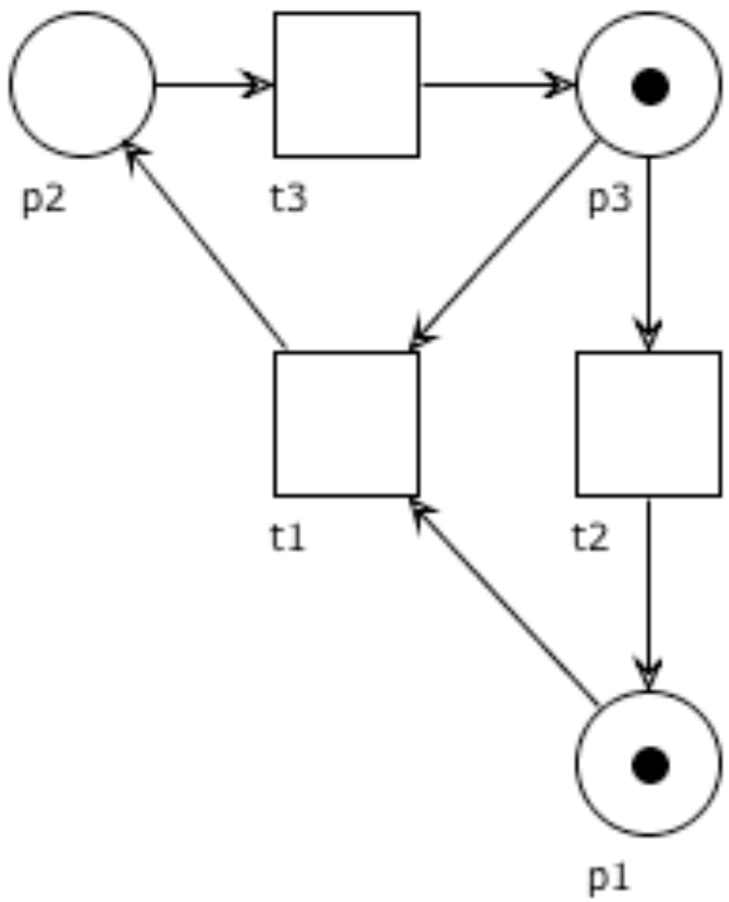
Free-choice system

Definition: A system (N,M_0) is **free-choice**
if N is free-choice

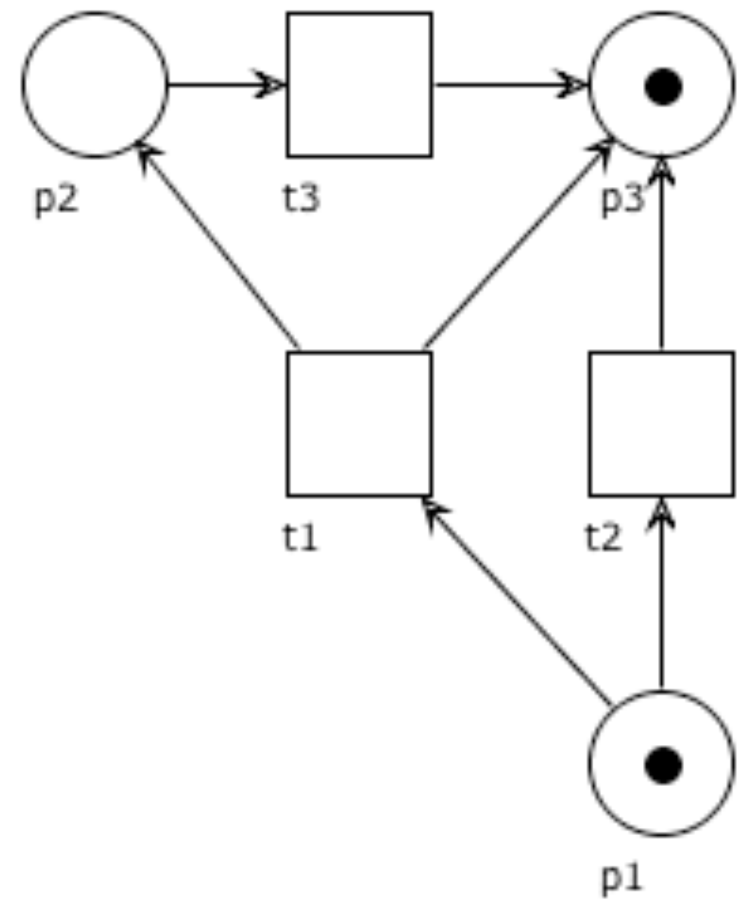
Example

$$\begin{aligned} \bullet t_1 &= \{ p_1, p_3 \} \\ \bullet t_2 &= \{ p_3 \} \\ \bullet t_1 &\neq \bullet t_2 \\ \bullet t_1 \cap \bullet t_2 &= \{ p_3 \} \neq \emptyset \end{aligned}$$

$$\begin{aligned} \bullet t_1 &= \bullet t_2 \\ \bullet t_1 \cap \bullet t_3 &= \emptyset \\ \bullet t_2 \cap \bullet t_3 &= \emptyset \end{aligned}$$



non free-choice



free-choice

Fundamental property of free-choice nets

Proposition: Let (P, T, F, M_0) be free-choice.
If $M \xrightarrow{t}$ and $t \in p\bullet$, then $M \xrightarrow{t'}$ for every $t' \in p\bullet$.

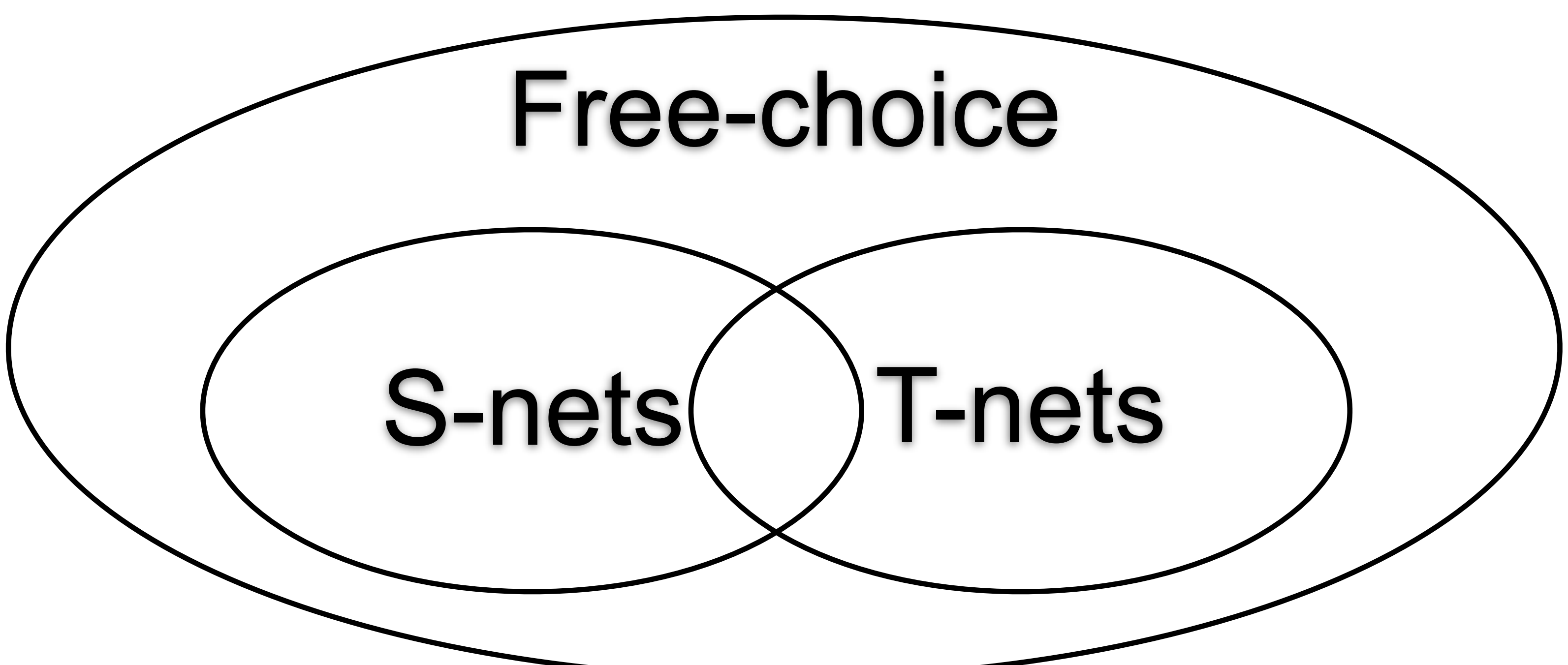
The proof is trivial, by definition of free-choice net

Exercises

Prove that every S-net is free-choice

Prove that every T-net is free-choice

Show a free-choice net that is neither an S-net nor a T-net



Free-choice N*

Proposition: A workflow net N is free-choice iff N* is free-choice

N and N* differ only for the reset transition, whose pre-set (o) is disjoint from the pre-set of any other transition

Rank Theorem (main result)

Theorem:

A free-choice system (P,T,F,M0) is live and bounded iff

1. it has at least one place and one transition
2. it is connected
3. M_0 marks every proper siphon
4. it has a positive S-invariant
5. it has a positive T-invariant
6. $\text{rank}(N) = |C_N| - 1$

(where C_N is the set of clusters)

Clusters

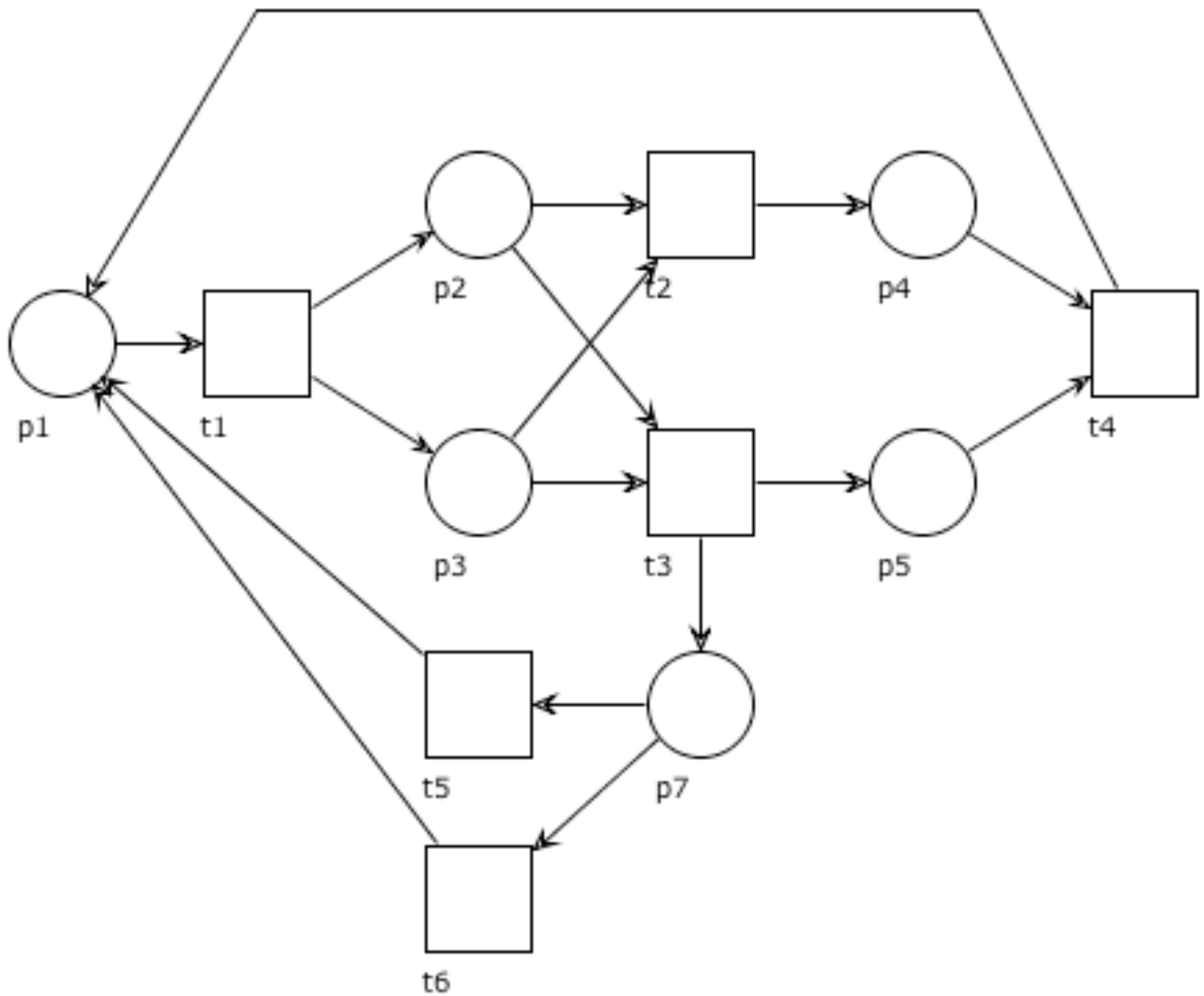
Cluster

Let x be the node of a net $N = (P, T, F)$ (not necessarily free-choice)

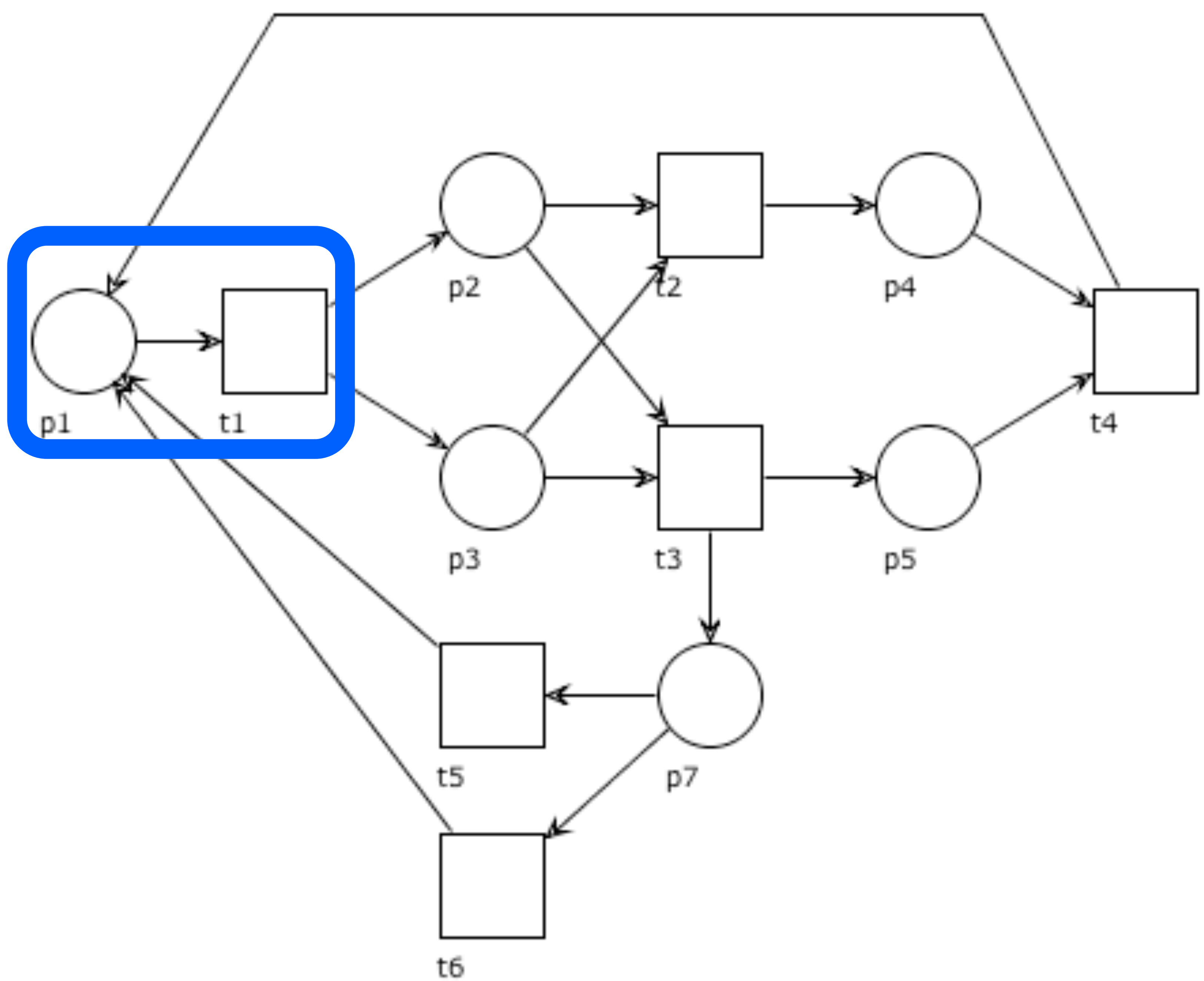
Definition:
The **cluster** of x, written $[x]$, is the least set s.t.

1. $x \in [x]$
2. if $p \in [x] \cap P$ then $p\bullet \subseteq [x]$ (if a place p is in the cluster, then all transitions in the post-set of p are in the cluster)
3. if $t \in [x] \cap T$ then $\bullet t \subseteq [x]$ (if a transition t is in the cluster, then all places in the pre-set of t are in the cluster)

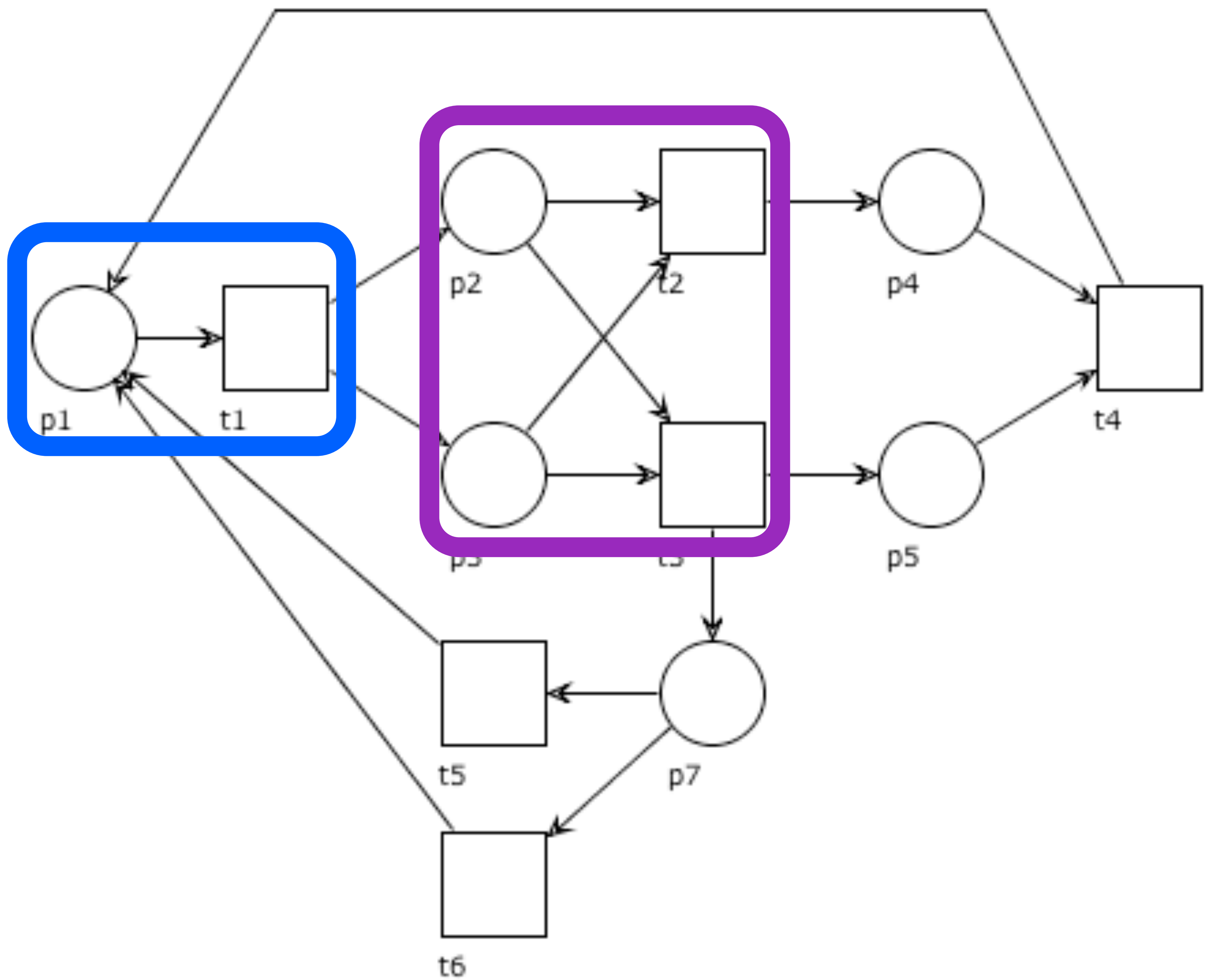
Cluster: example



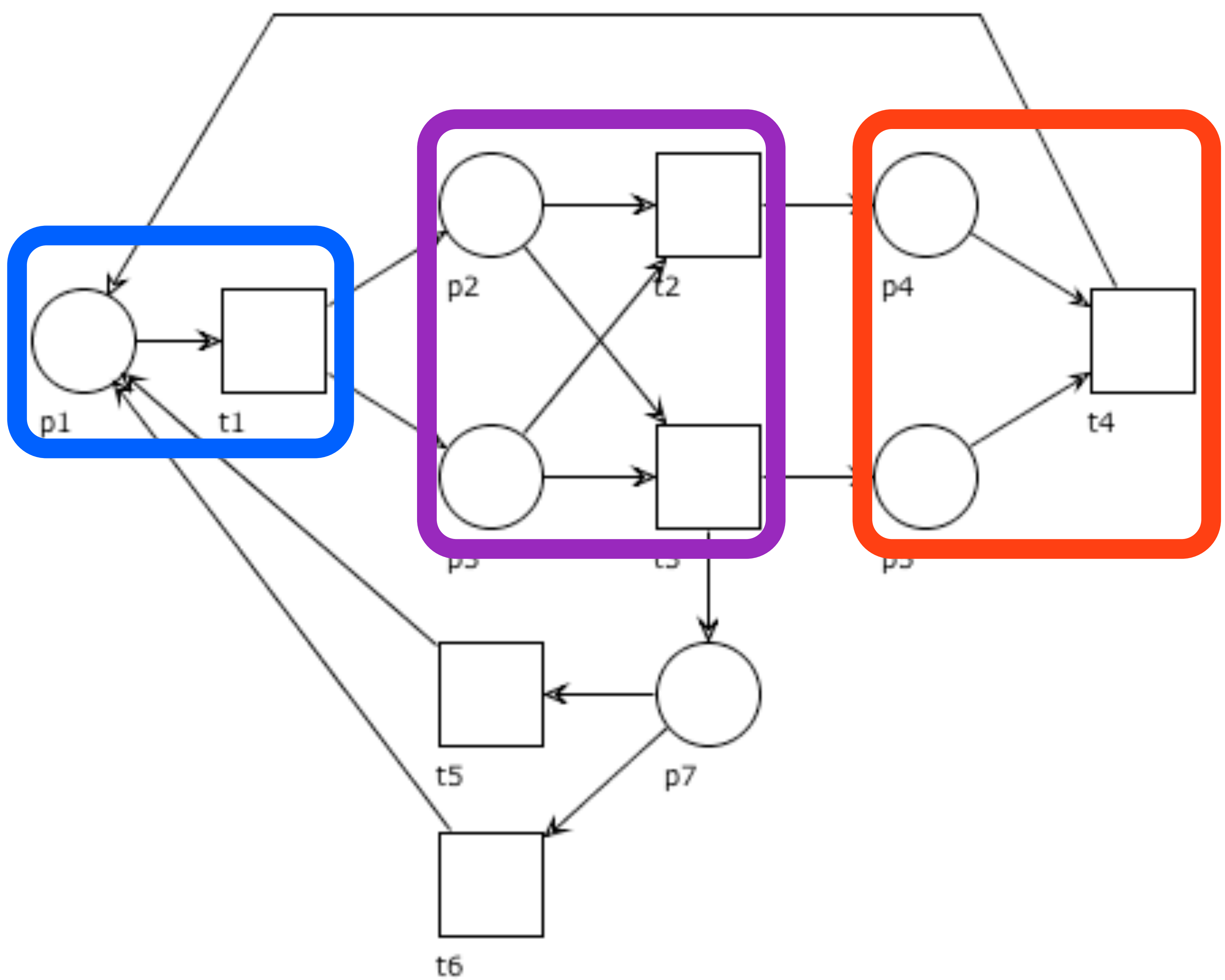
Cluster: example



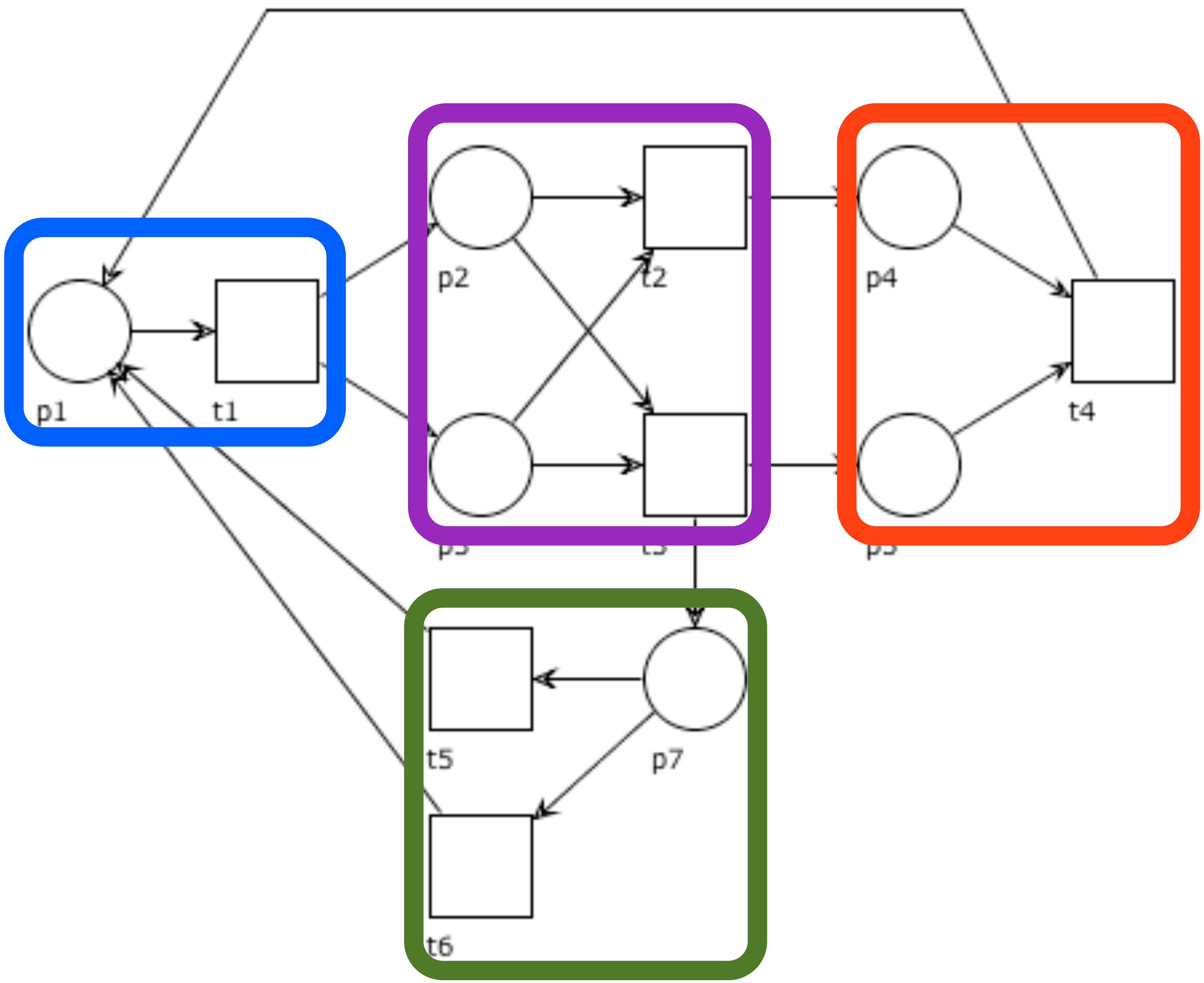
Cluster: example



Cluster: example



Cluster: example



Clusters partition

Lemma: The set $\{ [x] \mid x \in P \cup T \}$ is a partition of $P \cup T$

Take the reflexive, symmetric and transitive closure E of

$$F \cap (P \times T)$$

From the definition, it follows that

$$y \in [x] \qquad \text{iff} \qquad (x, y) \in E$$

Since E is an equivalence relation, its classes define a partition

Fundamental property of clusters in f.c. nets

Proposition:

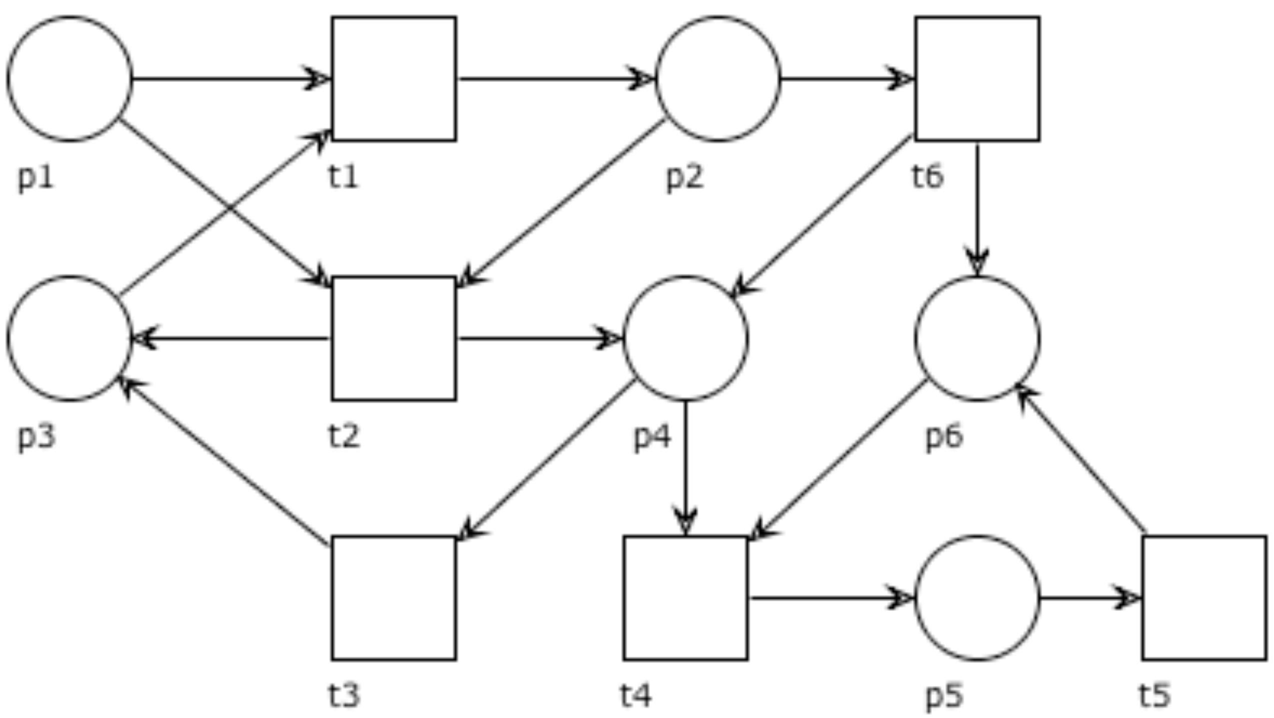
If $M \xrightarrow{t}$, then for any $t' \in [t]$ we have $M \xrightarrow{t'}$

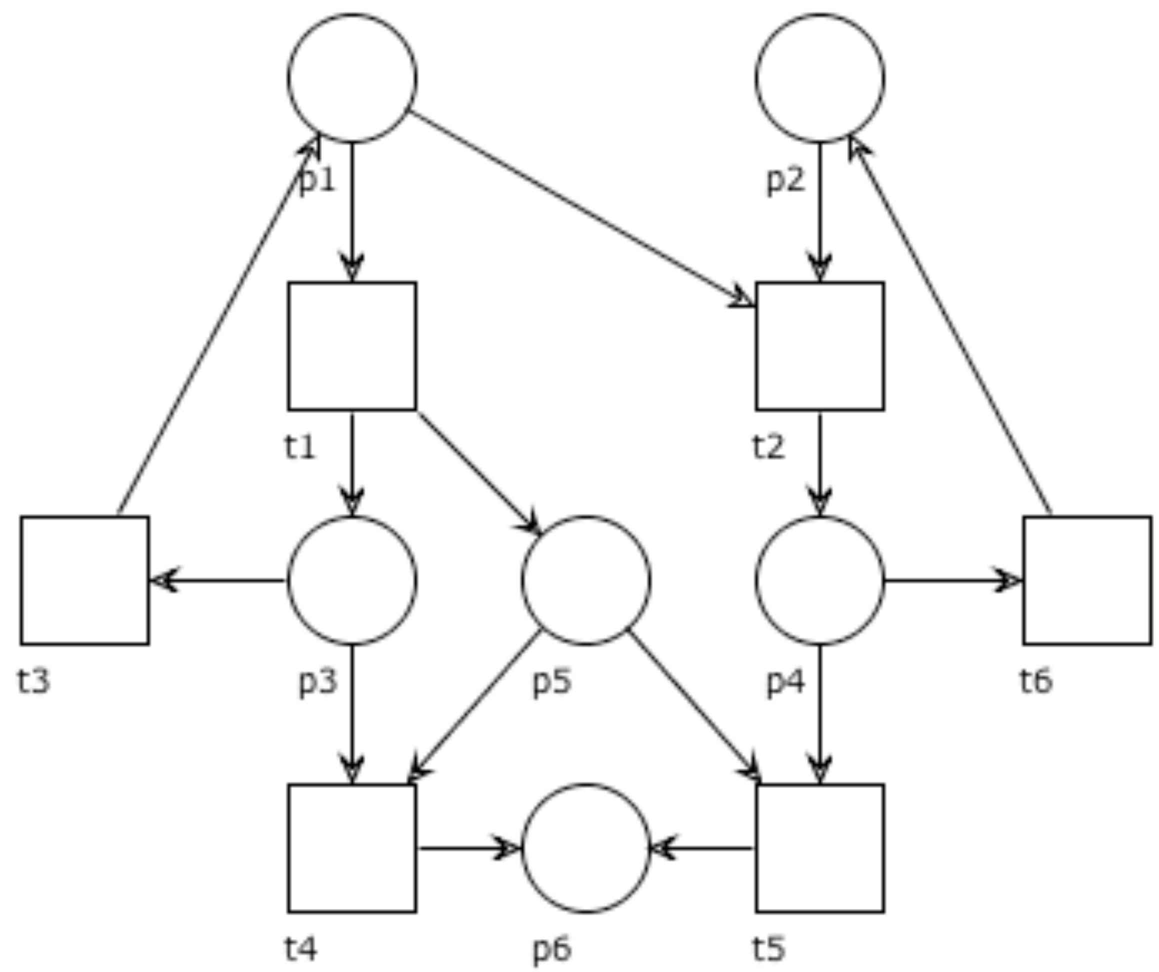
Immediate consequence of the fact that, for free-choice nets

$$t, t' \in [x] \qquad \text{iff} \qquad \bullet\, t = \bullet t'$$

Exercise

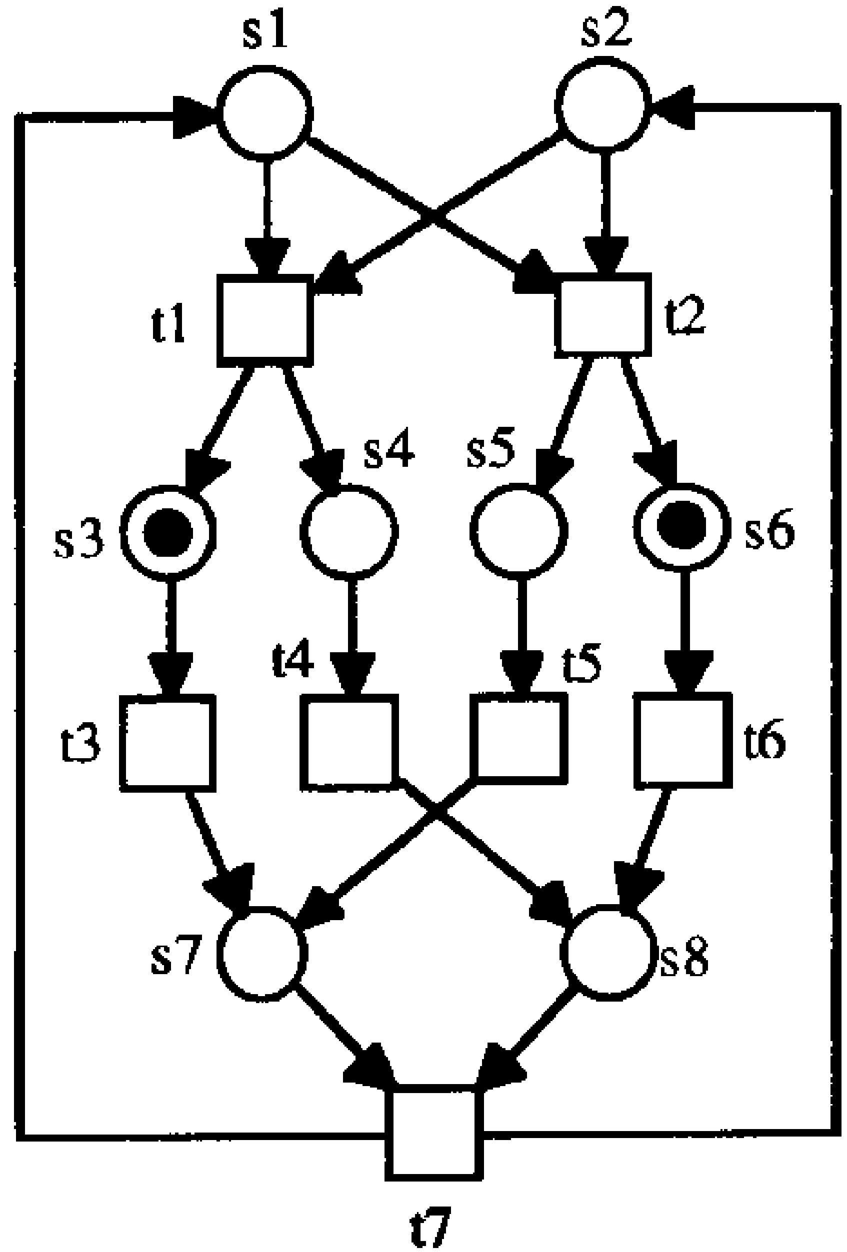
Draw all clusters in the nets below





Exercise

Draw all clusters in the free-choice net below



Stable markings

Stable set of markings

Definition: A set of markings $\mathbf{M}$ is called **stable** if

$$M \in \mathbf{M} \qquad \text{implies} \qquad [\, M \,\rangle \subseteq \mathbf{M}$$

(starting from any marking in the stable set **M**,
no marking outside **M** is reachable)

Question time

Given a net system:

Is the singleton set { 0 } a stable set?

Is the set of all markings a stable set?

Is the set of live markings a stable set?

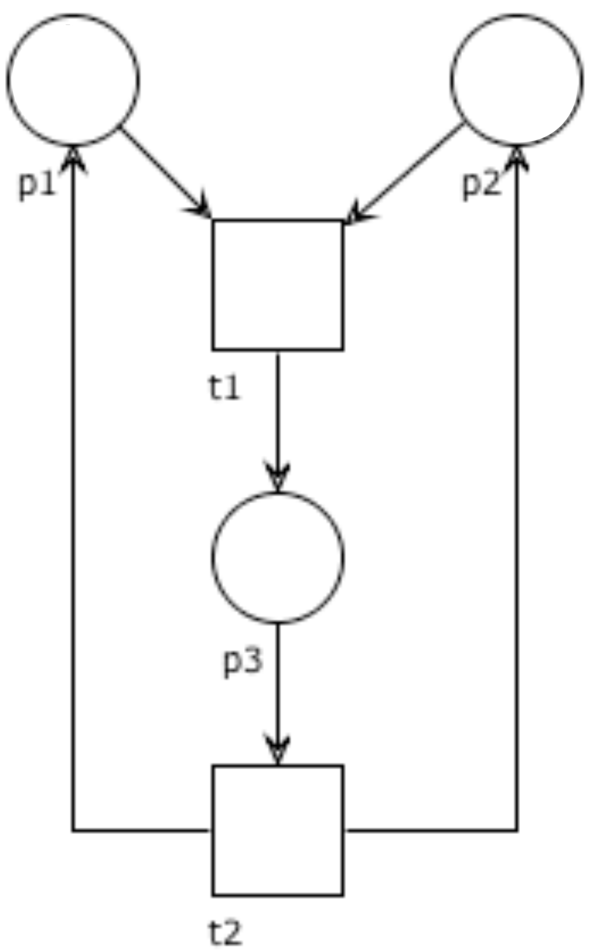
Is the set of deadlock markings a stable set?

Stability check

$$\mathbf{M} \text{ is stable iff}$$
$$\forall M, t, M'.\, (M \in \mathbf{M} \;\wedge\; M \xrightarrow{t} M' \text{ implies } M' \in \mathbf{M})$$

Example

Which of the following is a stable set of markings?



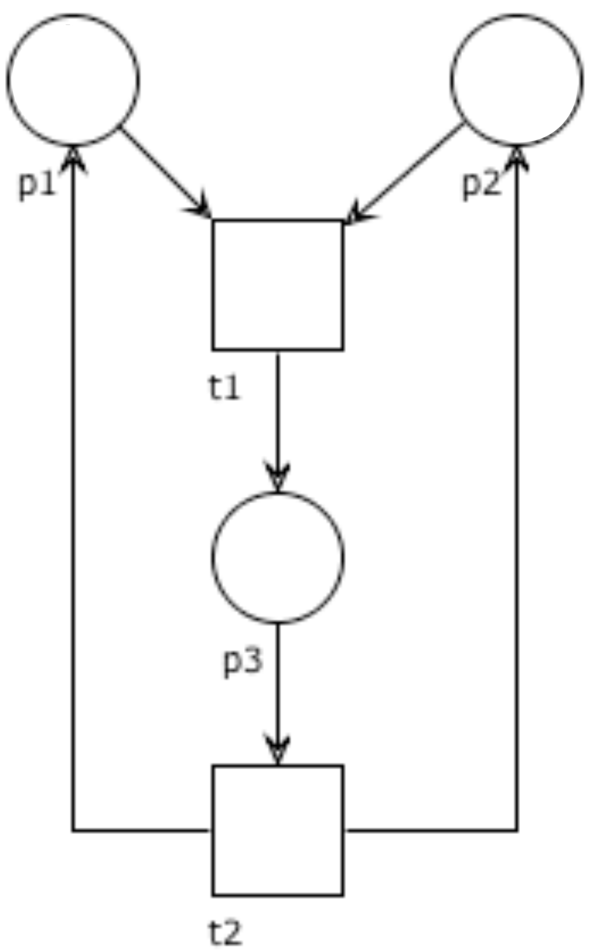
$\{ 2p_1+p_2 \}$

$\{ 2p_1+p_2 , p_1+2p_3 \}$

$\{ p_1 , p_2 \}$

Exercises

Which of the following is a stable set of markings?



$\{ p_1 , \; p_3 \}$

$\{ 2p_1+2p_2 , 2p_3 \}$

$\{ 2p_1+2p_2 , p_1+p_2+p_3 , 2p_3 \}$

$\{ p_1, 2p_1+2p_2 , p_1+p_2+p_3 , 2p_3 \}$

Exercises

Given a net system:

Is the set { M | M(P)=1 } a stable set?

Is the set of markings reachable from M_0 a stable set?

Is the set { M | M(P)<k } a stable set?

Exercises

Let **I** be an S-invariant

Is the set $\{ M \mid \mathbf{I} \cdot M = \mathbf{I} \cdot M_0 \}$ a stable set?

Is the set $\{ M \mid \mathbf{I} \cdot M \neq \mathbf{I} \cdot M_0 \}$ a stable set?

Is the set $\{ M \mid \mathbf{I} \cdot M = 1 \}$ a stable set?

Is the set $\{ M \mid \mathbf{I} \cdot M = 0 \}$ a stable set?

Exercises

Let **M** and **M'** be stable sets
Is their union a stable set?
Is their intersection a stable set?
Is their difference a stable set?

What is the least stable set that includes a marking M_0?

What is the largest stable set of a net?

Siphons

Proper siphon

Definition:

A set of places R is a **siphon** if $\bullet R \subseteq R \bullet$

It is a **proper siphon** if $R \neq \emptyset$

Siphons, intuitively

A set of places R is a siphon if

all transitions that can produce tokens in the places of R

require some place in R to be marked

Therefore:
if no token is present in R,
then no token will ever be produced in R

Siphon check

Let R be a set of places of a net

mark with √ all transitions that consume tokens from R

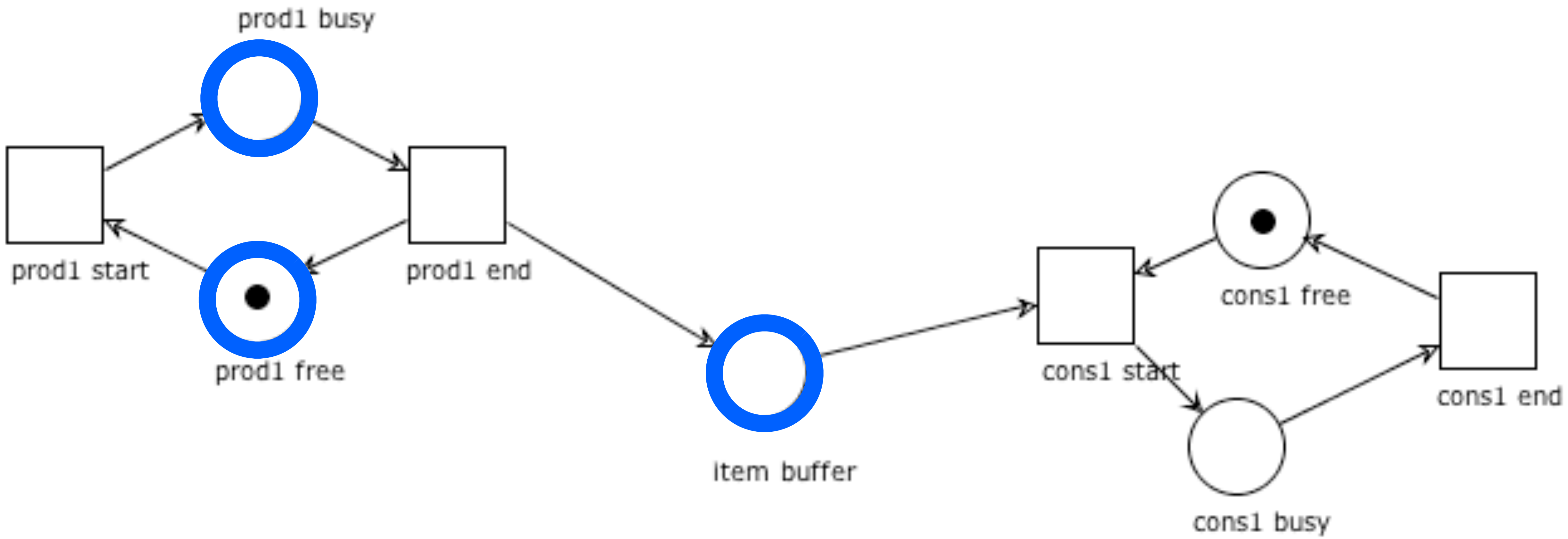
if there is a transition producing tokens in some place of R that is not marked by √, then R is not a siphon

Otherwise R is a siphon

$\bullet R \subseteq R \bullet$

Siphon check: example

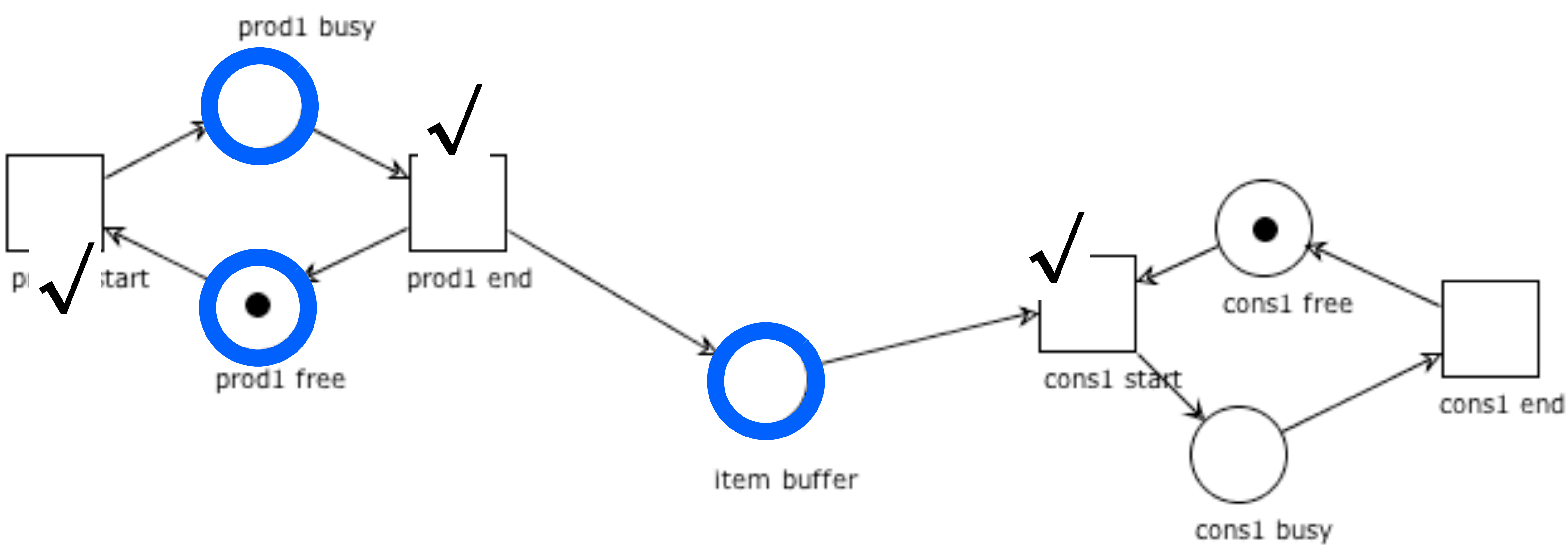
Is R = { prod1busy, prod1free, itembuffer} a siphon?



$$\bullet R \subseteq R \bullet$$

Siphon check: example

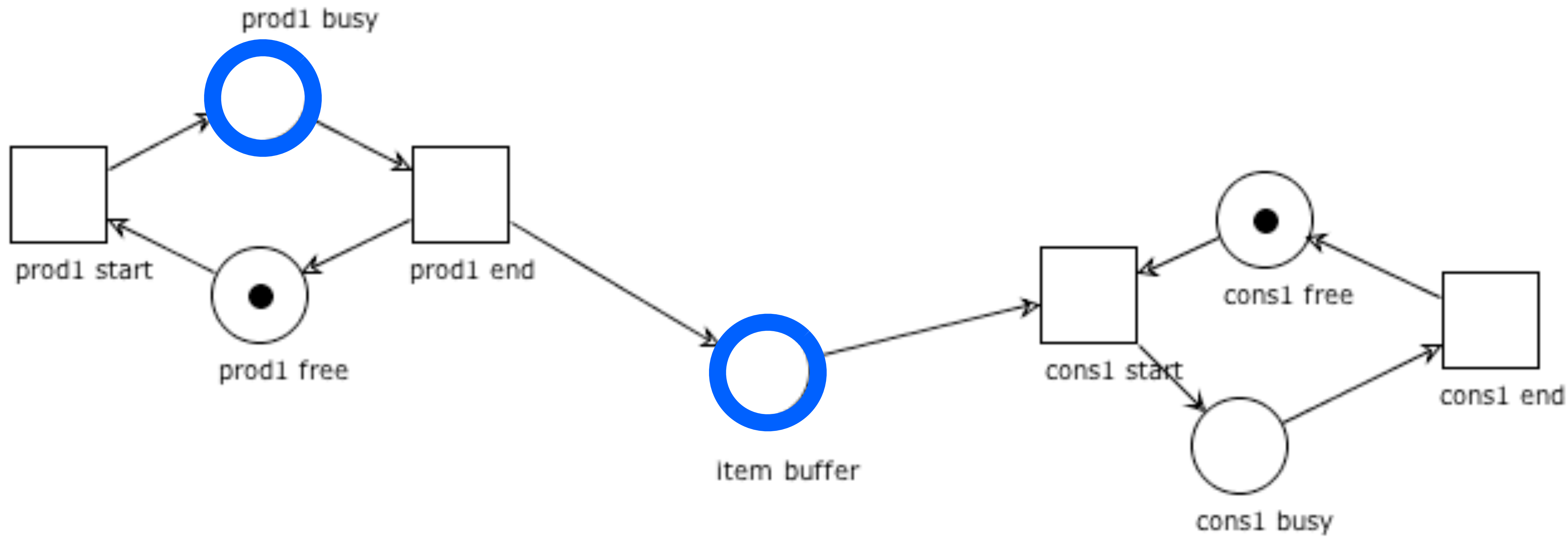
Is R = { prod1busy, prod1free, itembuffer} a siphon?



$\bullet R \subseteq R \bullet$

Siphon check: example

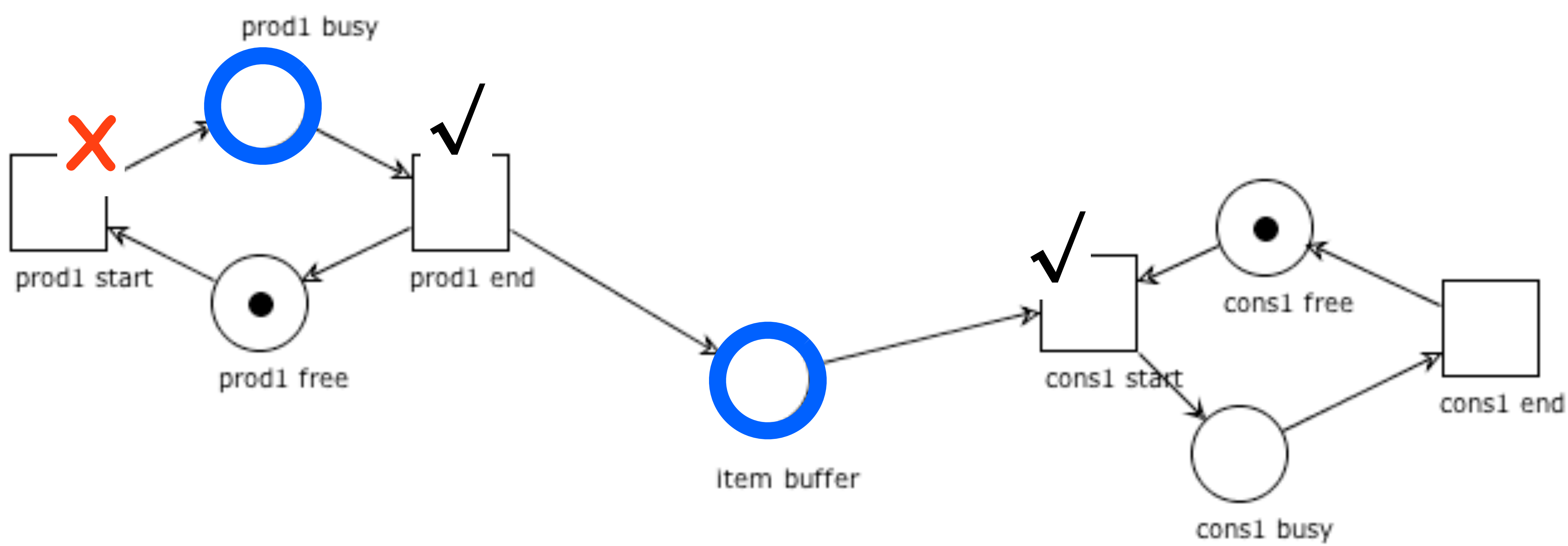
Is R = { prod1busy, itembuffer} a siphon?



$\bullet R \subseteq R \bullet$

Siphon check: example

Is R = { prod1busy, itembuffer} a siphon?



Fundamental property of siphons

Proposition: Unmarked siphons remain unmarked

Take a siphon R.

We just need to prove that the set of markings
M = { M | M(R)=0 }
is stable, which is immediate by definition of siphon

Consequence of the fundamental property

Corollary:
If a siphon R is marked at some reachable marking M, then it was initially marked at M_0

By hypothesis: $M(R)>0$

By contradiction: assume $M_0(R)=0$
Then by the fundamental property of siphons: $M(R)=0$
which is absurd

Siphons and liveness

Prop.: Live systems have no unmarked proper siphons (We show that every proper siphon R of a live system is initially marked)

Take $p \in R$ and let $t \in \bullet p \cup p\bullet$

Since the system is live, then there are $M, M' \in [\, M_0 \,\rangle$ such that

$$M \xrightarrow{t} M'$$

Therefore p is marked at either M or M'
Therefore R is marked at either M or M'
Therefore R was initially marked (at M_0)

Siphons and deadlock

Proposition:
Deadlocked systems have an unmarked proper siphon

Let M be a deadlocked marking

Let $R = \{ p \mid M(p) = 0 \}$

Since M is deadlock: $R\bullet = T$

Therefore $\bullet R \subseteq T = R\bullet$ and R is a siphon.
Since T cannot be empty, R is proper

A key observation

If we can guarantee that

all proper siphons are marked
at **every** reachable marking,

then the system is deadlock free

Exercise

Prove that the union of siphons is a siphon

Traps

Proper trap

Definition:

A set of places R is a **trap** if $\bullet R \supseteq R\bullet$

It is a **proper trap** if $R \neq \emptyset$

Traps, intuitively

A set of places R is a trap if

all transitions that can consume tokens from R

produce some token in some place of R

Therefore:
if some token is present in R,
then it is never possible for R to become empty

Trap check

Let R be a set of places of a net

mark with √ all transitions that produce tokens in R

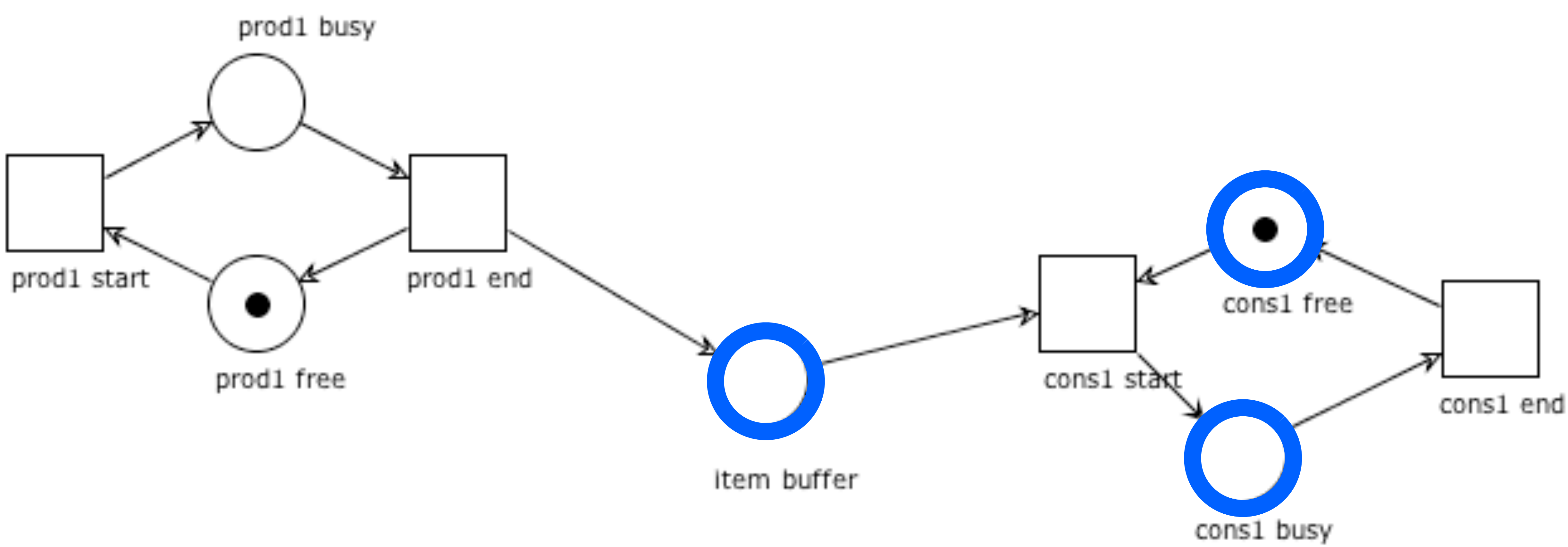
if there is a transition consuming tokens from some place in R that is not marked by √, then R is not a trap

Otherwise R is a trap

$\bullet R \supseteq R \bullet$

Trap check: example

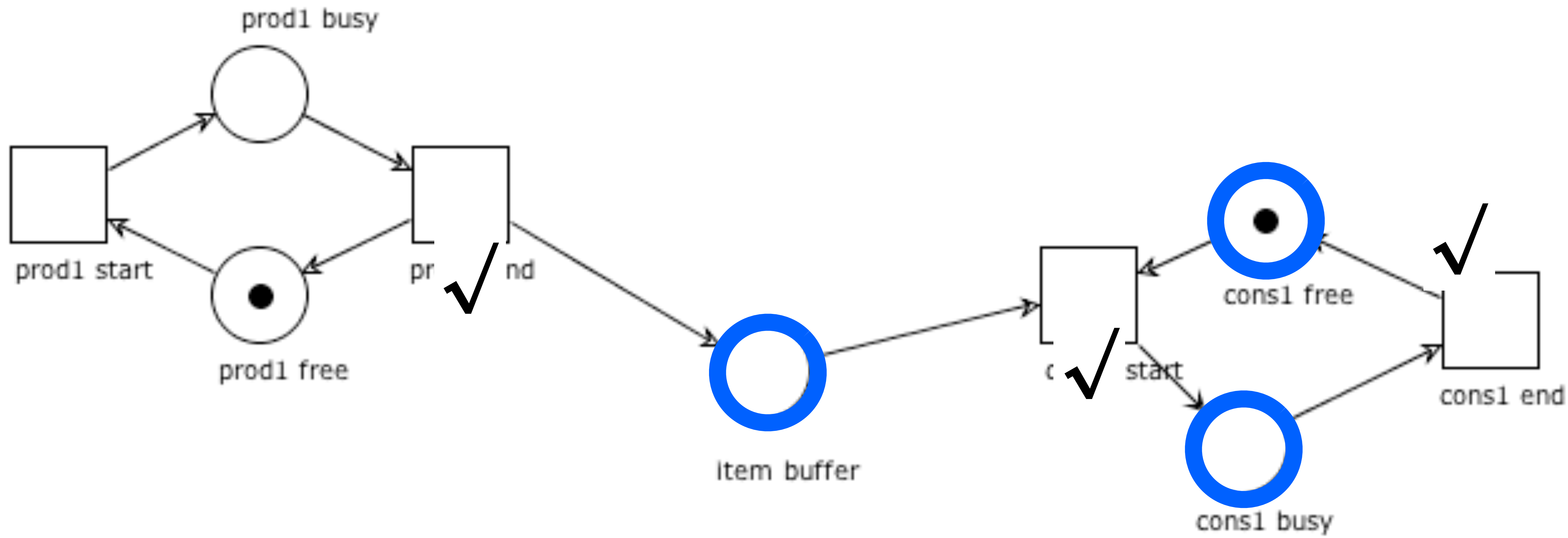
Is R = { itembuffer, cons1busy, cons1free} a trap?



$\bullet R \supseteq R \bullet$

Trap check: example

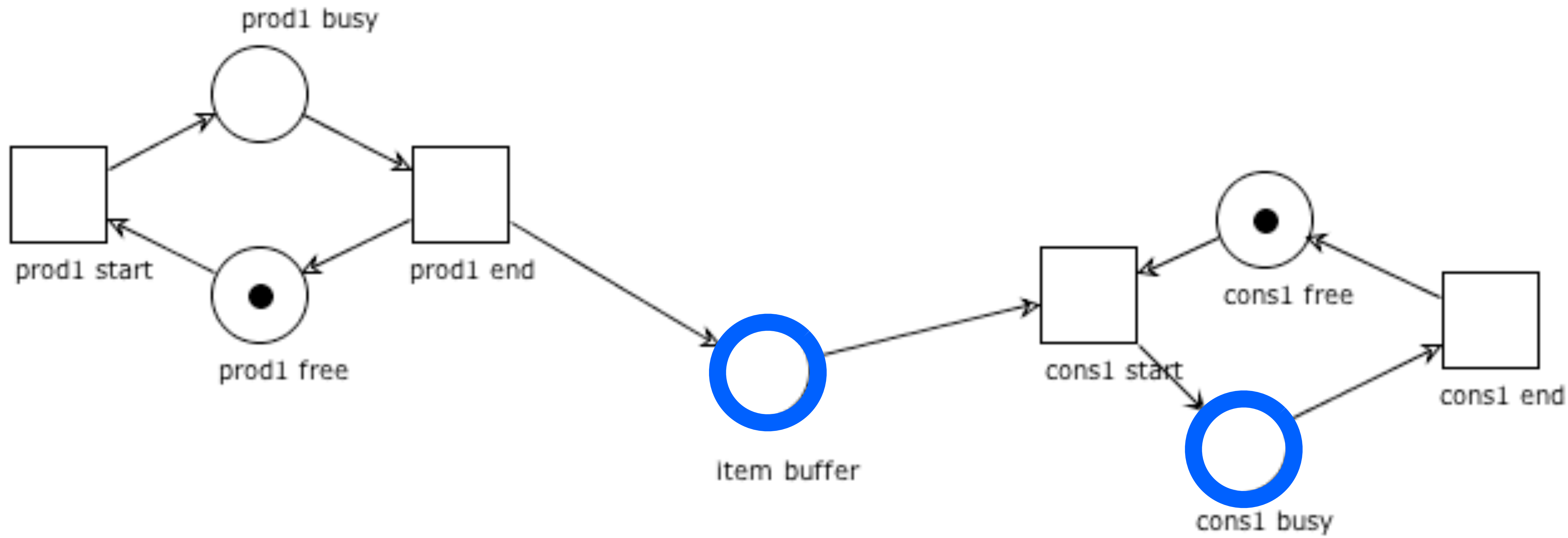
Is R = { itembuffer, cons1busy, cons1free} a trap?



$$\bullet R \supseteq R \bullet$$

Trap check: example

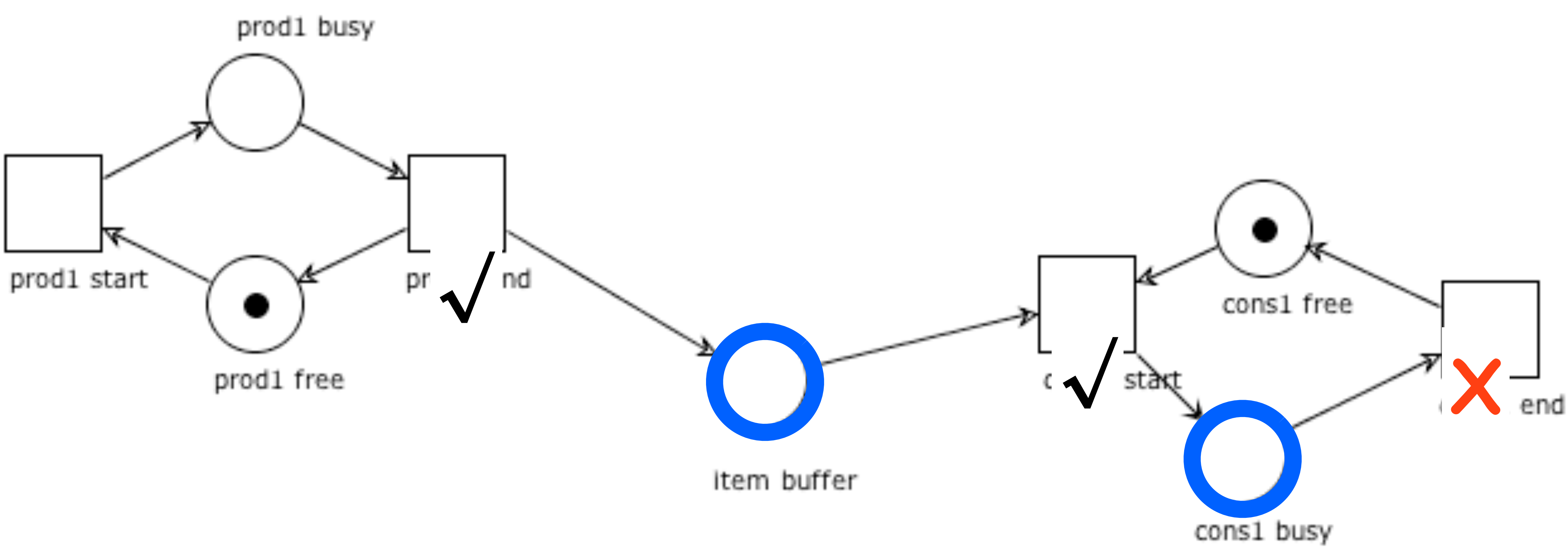
Is R = { itembuffer, cons1busy} a trap?



$\bullet R \supseteq R \bullet$

Trap check: example

Is R = { itembuffer, cons1busy} a trap?



Fundamental property of traps

Proposition: Marked traps remain marked

Take a trap R.

We just need to prove that the set of markings
M = { M | M(R)>0 }
is stable, which is immediate by definition of trap

Consequence of the fundamental property

Corollary:
If a trap R is unmarked at some reachable marking M, then it was initially unmarked at M_0

By hypothesis: $M(R)=0$

By contradiction: assume $M_0(R)>0$

Then by the fundamental property of traps: $M(R)>0$ which is absurd

Exercise

Prove that the union of traps is a trap

Putting pieces together

unmarked siphons stay unmarked
(marked siphons can become unmarked)

if a siphon is marked at M, it was marked at M_0

if all proper siphons always stay marked => deadlock-free

Putting pieces together

if all proper siphons always stay marked => deadlock-free

marked traps stay marked
(unmarked traps can become marked)

if a trap is unmarked at M, it was unmarked at M_0

if a siphon contains a marked trap, it stays marked

if all siphons contain marked traps, they stay marked
=> deadlock-free

A sufficient condition for deadlock-freedom

Proposition:
If every proper siphon of a system includes an initially marked trap, then the system is deadlock-free

We show that if the system is not deadlock free, then there is a siphon that does not include any marked trap.

Assume some reachable M is dead.
Let R be the set of unmarked places at M.
Then, we have seen that R is a proper siphon.
Since M(R)=0, then R includes no trap marked at M.
Therefore, R includes no trap marked at M_0

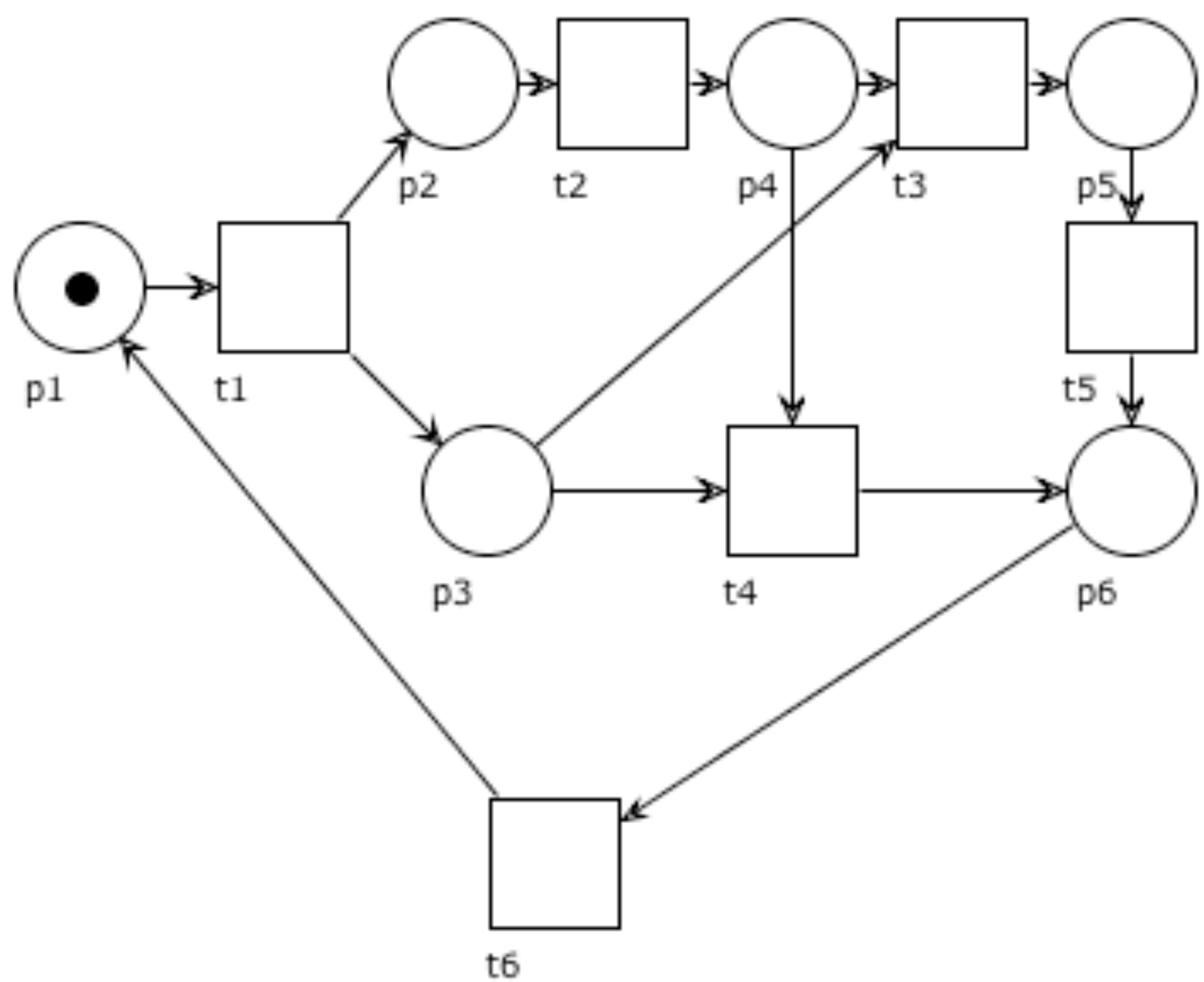
Note

It is easy to observe that every siphon includes a (possibly empty) unique maximal trap with respect to set inclusion

Moreover, a siphon includes a marked trap
iff
its maximal trap is marked

Exercise

Find all siphons and traps in the net below



Live and dead places (recall)

Place liveness

Definition: Let (P, T, F, M_0) be a net system.

A place $p \in P$ is **live** if $\forall M \in [\, M_0 \,\rangle .\, \exists M' \in [\, M \,\rangle .\, M'(p) > 0$

A place p is live
if every time it becomes unmarked
there is still the possibility to be marked in the future
(or if it is always marked)

Definition:
A net system (P, T, F, M_0) is **place-live** if every place $p \in P$ is live

liveness implies place-liveness

Dead nodes

Definition: Let (P, T, F) be a net system.

A transition $t \in T$ is **dead** at M if $\forall M' \in [\, M \,\rangle .\, M' \stackrel{t}{\not\longrightarrow}$

A place $p \in P$ is **dead** at M if $\forall M' \in [\, M \,\rangle .\, M'(p) = 0$

Some obvious facts

If a system is not live, it has a transition dead at some reachable marking

If a system is not place-live, it has a place dead at some reachable marking

If a place / transition is dead at M, then it remains dead at any marking reachable from M
(the set of dead nodes can only increase during a run)

Every transition in the pre- or post-set of a dead place is also dead

An obvious facts in free-choice nets

In a free-choice net:

if an output transition t of a place p is dead at M

then any output transition t' of p is dead at M

(because t and t' must have the same pre-set)

Dead t, dead p

Lemma: If the transition t is dead at M in a free-choice net, then there is a non-live place p in the pre-set of t (i.e., p is dead at some marking reachable from M)

By contraposition, we prove: if all input places of t are live then t is not dead

Let $\bullet t = [t] \cap P = \{p_1, ..., p_n\}$

Since all places $p_1, ..., p_n$ are live at M, there exists

$M \xrightarrow{\sigma_1} M_1 \xrightarrow{\sigma_2} ... \xrightarrow{\sigma_n} M_n$

such that $M_i(p_i) > 0$ for all i

If the sequence contains $u \in [t]$ then t is not dead at M

If no transition in $[t]$ appears in the sequence, then no token in $\bullet t$ is consumed

Hence $M_n(p_i) > 0$ for all i, and $M_n \xrightarrow{t}$ and t is not dead at M

Place-liveness implies liveness in f.c. nets

Proposition: If a free-choice system is place-live, then it is live

If a free-choice system is not live then there is a transition t dead at some reachable marking M

But then some input place of t must be dead at M, so the system is not place-live

Consequence in f.c. nets: place-liveness = liveness

If a free-choice system is place-live, then it is live

In any system, liveness implies place-liveness

Therefore:

A free-choice system is live iff it is place-live

Non-liveness and unmarked siphons

Lemma: Every non-live free-choice system has a proper siphon R and a reachable marking M such that M(R)=0

By non-liveness: the system is not place-live,
i.e., some p is dead at some L

Take $M \in [\, L \,\rangle$ such that every place not dead at M
is not dead at any marking of $[\, M \,\rangle$
i.e. all markings in $[\, M \,\rangle$ have the same set R of dead places
(dead places remain dead)

Next we prove that R is a proper siphon and $M(R) = 0$

Non-liveness and unmarked siphons

Lemma: Every non-live free-choice system has a proper siphon R and a reachable marking M such that M(R)=0

1. R is a siphon

   - any $t \in \bullet R$ is dead at M
     (if not any $q \in t \bullet \cap R$ would not be dead)
   - every t dead at M has an input place in R
     (t has some input place dead at some marking reachable from M)

2. R is proper

   p is dead at L, hence it is dead at M, hence $p \in R$, hence $R \neq \emptyset$

3. $M(R) = 0$ because it contains dead places

Commoner's theorem

Commoner's theorem

Theorem:
A free-choice system is live

iff

every proper siphon includes an initially marked trap

(we show just the "if" direction, which is simpler)

Commoner's theorem: "if" direction

(Non-live free-choice implies that
a proper siphon exists whose traps are all unmarked)

We know that a non-live free-choice system contains a proper siphon R such that M(R)=0

So every trap included in R is unmarked at M

Since marked traps remain marked,
every trap included in R must have been initially unmarked

Complexity of the non-liveness problem in free-choice systems

A non-deterministic algorithm for non-liveness

1. guess a set of places R

2. check if R is a siphon ($\bullet R \subseteq R\bullet$)
(polynomial time)

3. if R is a siphon, compute the maximal trap $Q \subseteq R$

4. if $M_0(Q)=0$, then answer “non-live”
(polynomial time)

A polynomial algorithm for maximal trap in a siphon

$\bullet R \subseteq R\bullet$ $\bullet Q \supseteq Q\bullet$

3. if R is a siphon, compute the maximal trap Q ⊆ R

Input: A net $N = (P, T, F)$ and $R \subseteq P$
Output: $Q \subseteq R$

$Q := R$
while $(\exists p \in Q,\ \exists t \in p\bullet,\ t \notin \bullet Q)$
$\quad Q := Q \setminus \{p\}$
return Q

Main consequence

The non-liveness problem for free-choice systems is in NP

Is the same problem in P?

The corresponding deterministic algorithm cannot make the guess in step 1

It has to explore all possible subsets of places
$2^{|P|}$ cases!

NP-completeness

We next sketch the proof of the reduction to non-liveness in a free-choice net of the CNF-SAT problem

(Satisfiability problem for propositional formulas in conjunctive normal form)

CNF-SAT formulas

Variables: $x_1, x_2, ..., x_n$

Literals: $x_1, \bar{x}_1, x_2, \bar{x}_2, ..., x_n, \bar{x}_n$

Clause: disjunction of literals

Formula: conjunction of clauses

Example: $\phi = (x_1 \vee \bar{x_3}) \wedge (x_1 \vee \bar{x}_2 \vee x_3) \wedge (x_2 \vee \bar{x}_3)$

Is there an assignment of boolean values to the variables such that $\phi = true$?

The free-choice net of a formula

The idea is
to construct a free-choice system (P,T,F,M_0)
and show that

the formula is satisfiable
iff
(P,T,F,M_0) is not live

CNF-SAT formulas

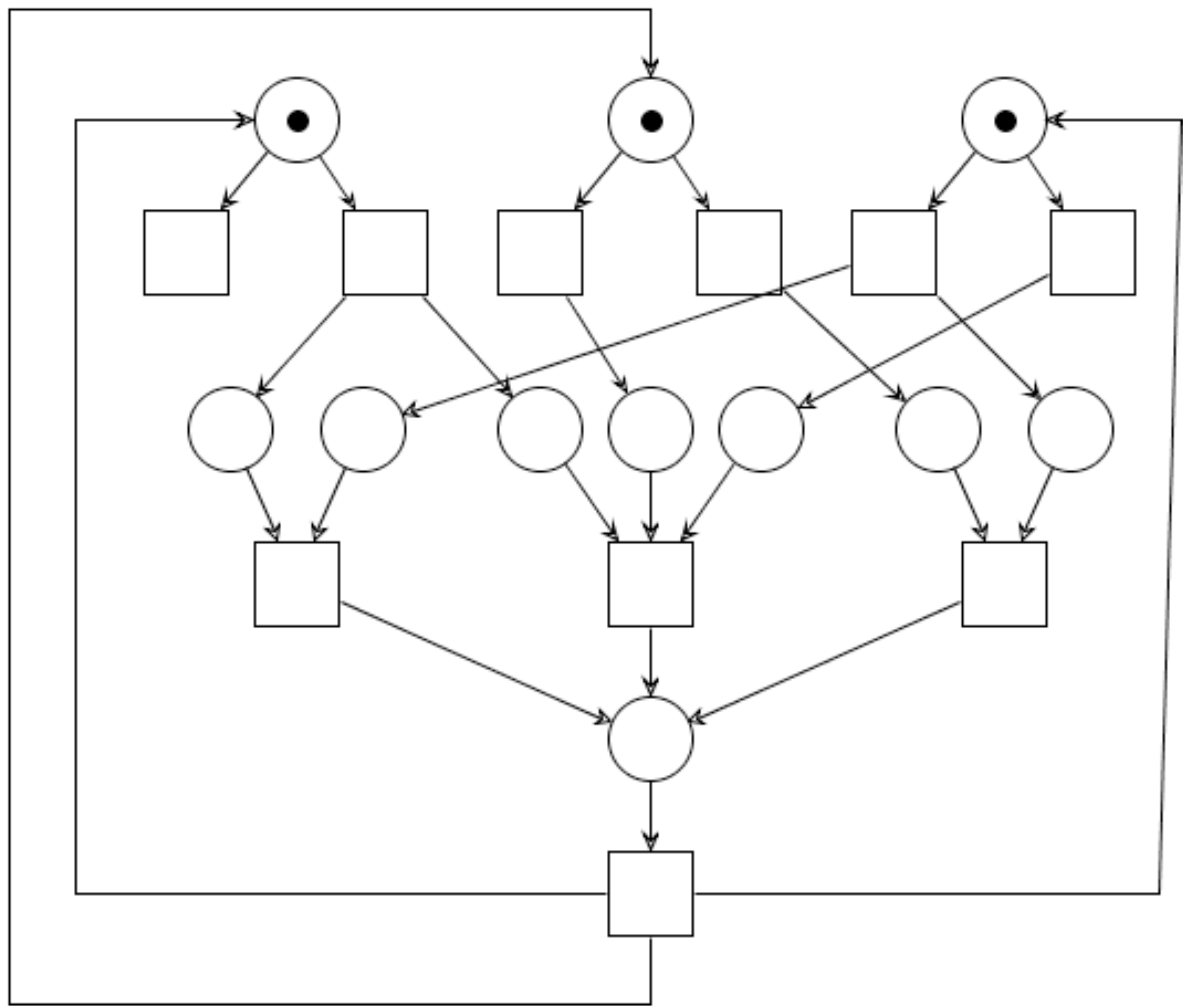
Is there an assignment of boolean values to the variables such that $\phi = true$?

Is there an assignment of boolean values to the variables such that $\neg\phi = false$?

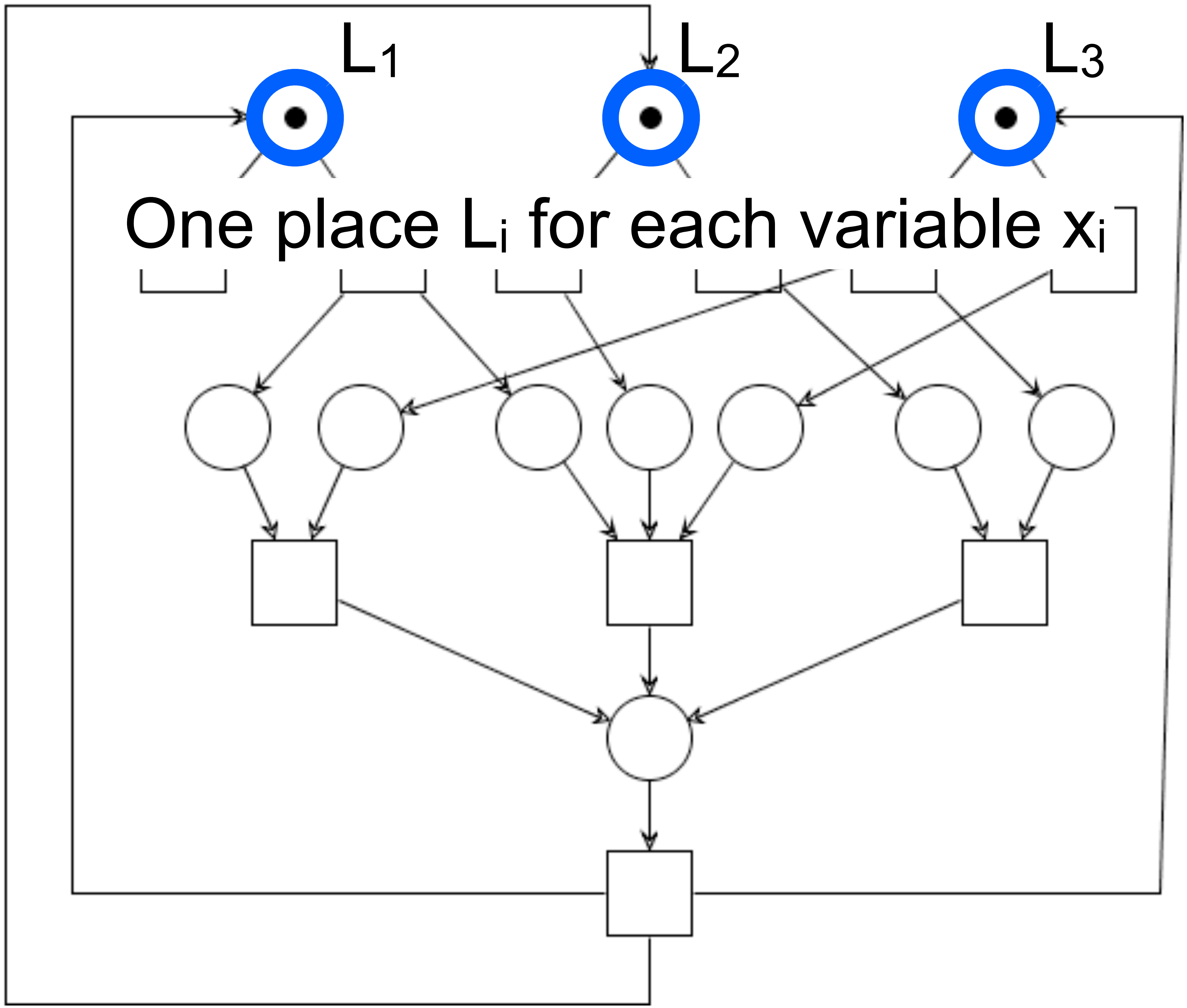
$$\phi = (x_1 \vee \overline{x}_3) \wedge (x_1 \vee \overline{x}_2 \vee x_3) \wedge (x_2 \vee \overline{x}_3)$$

$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$

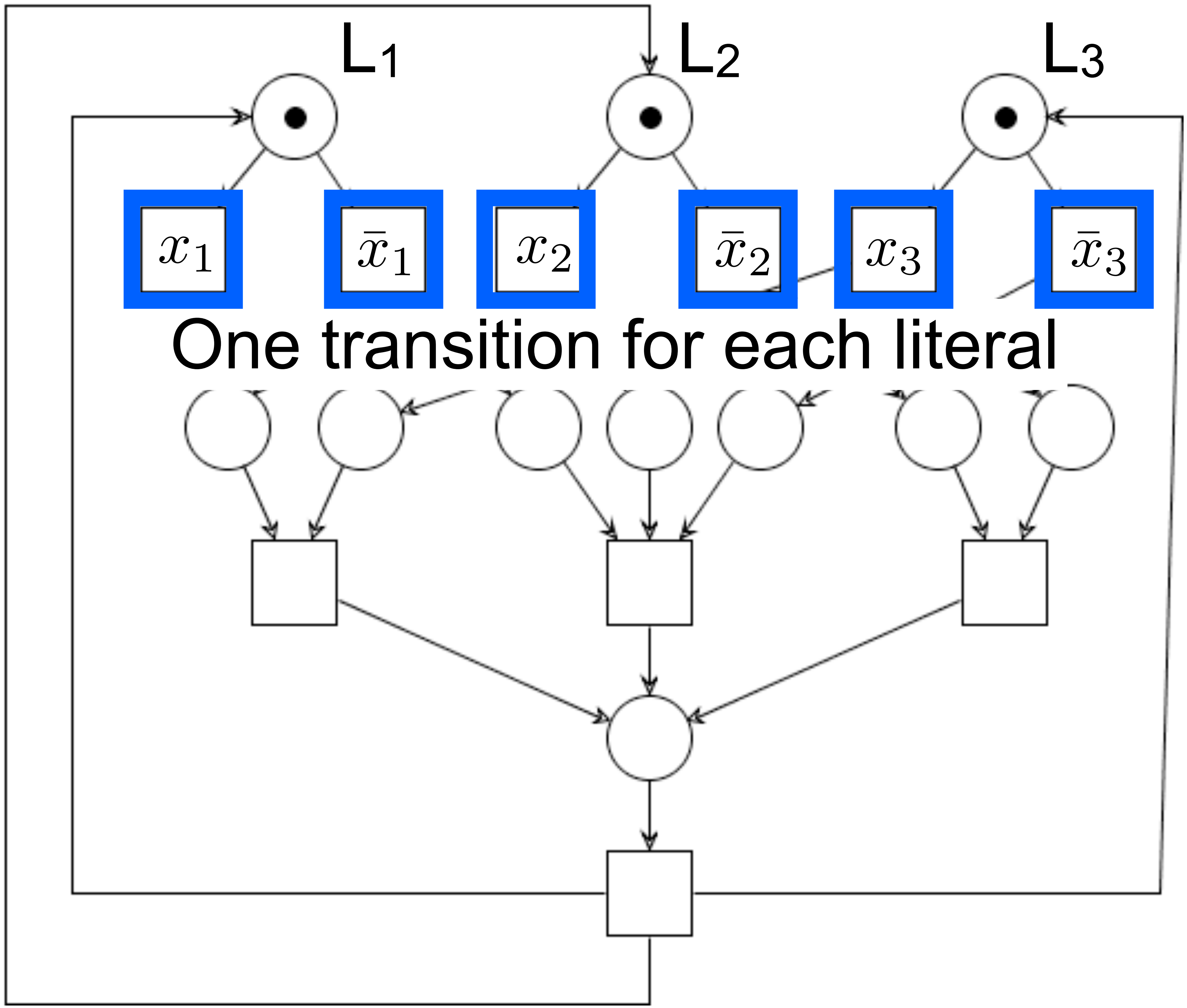
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



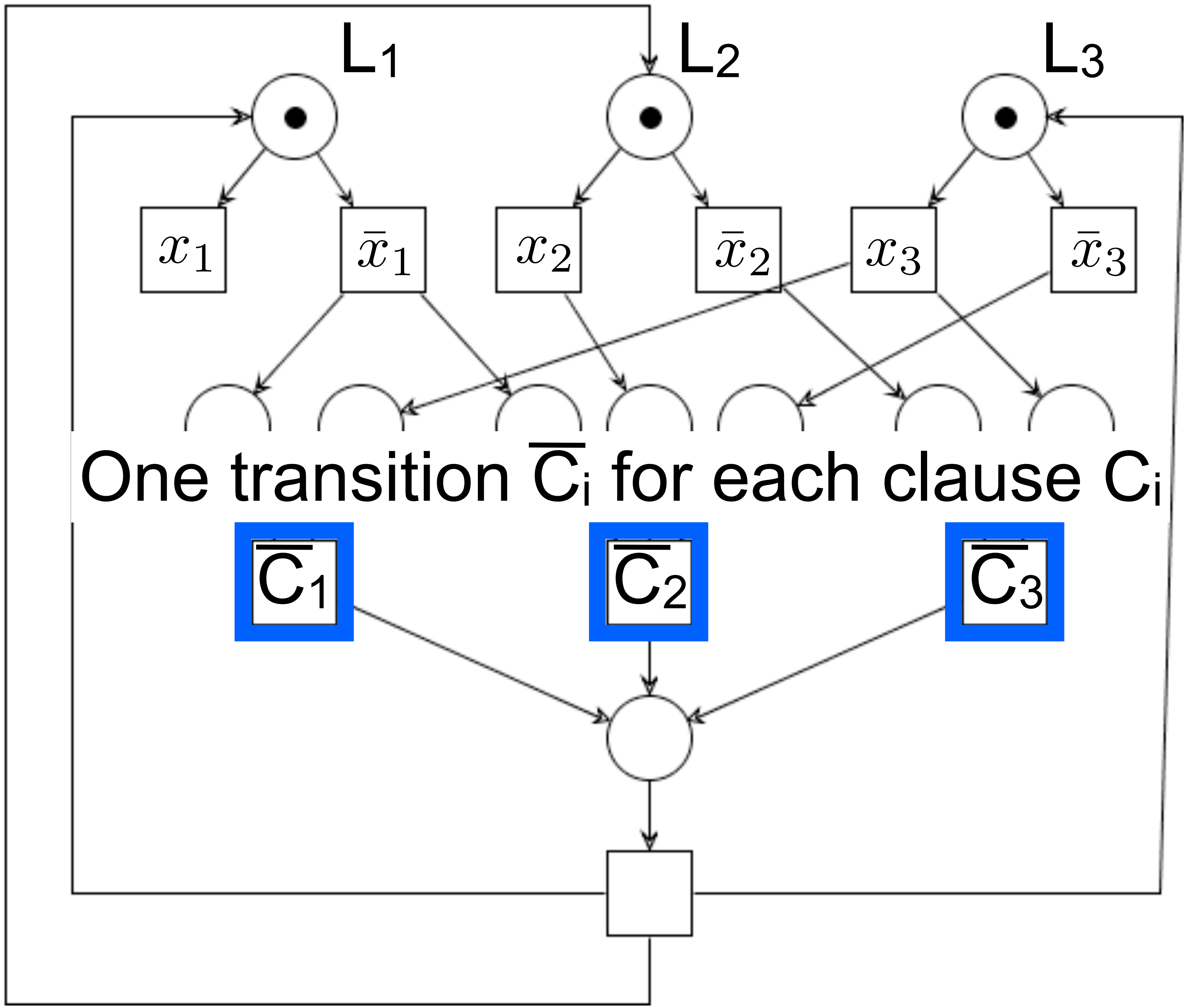
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



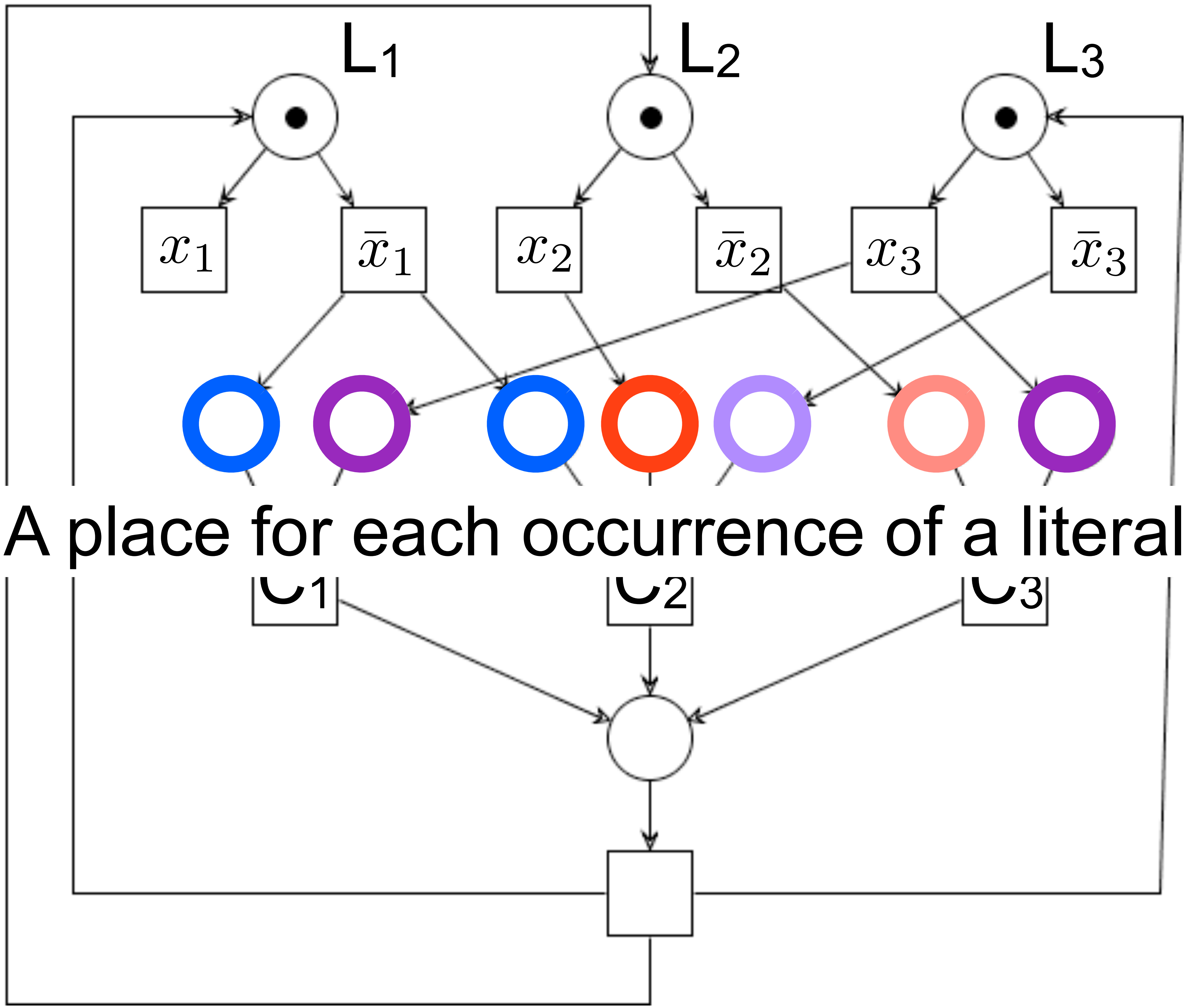
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$

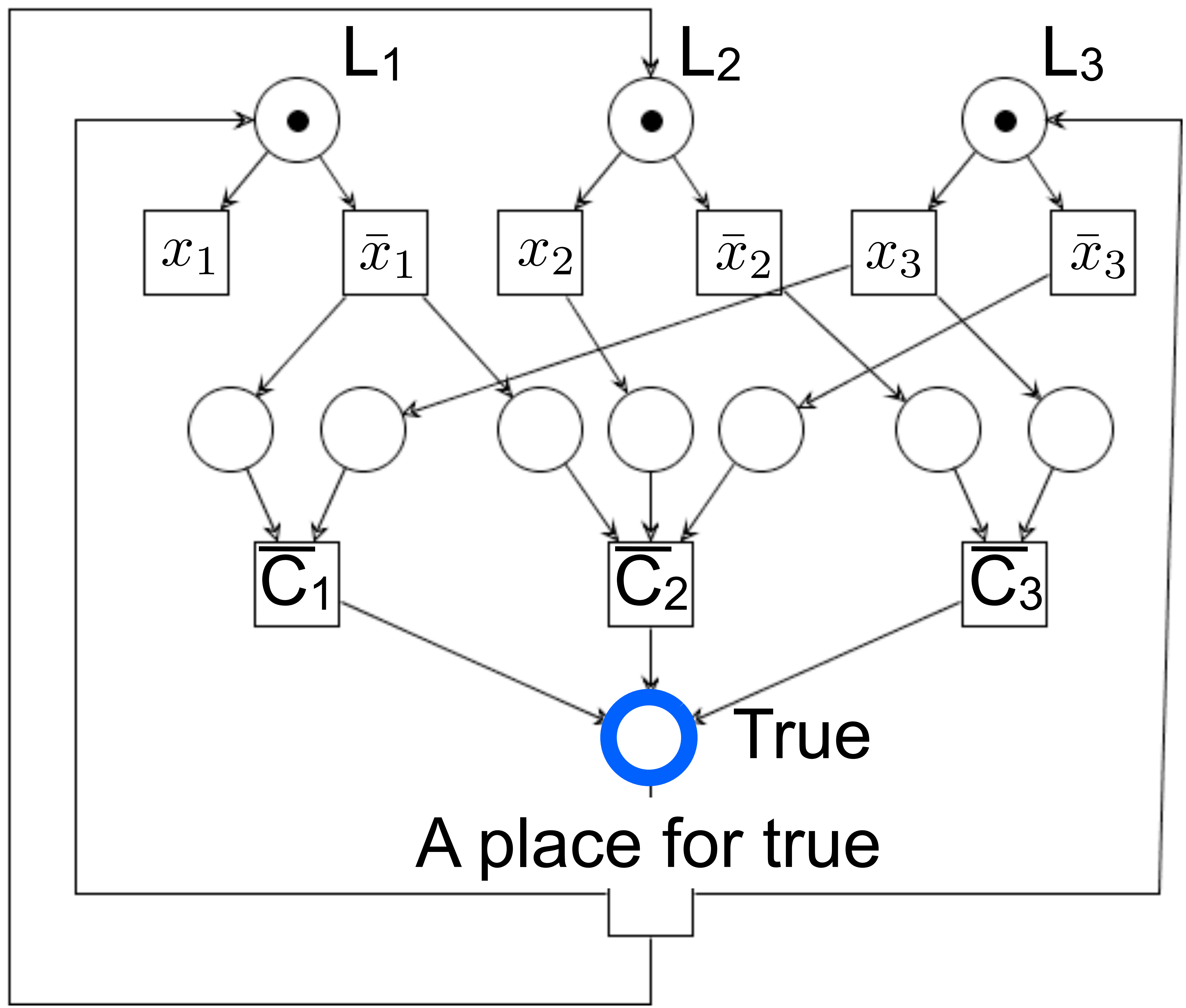


$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$

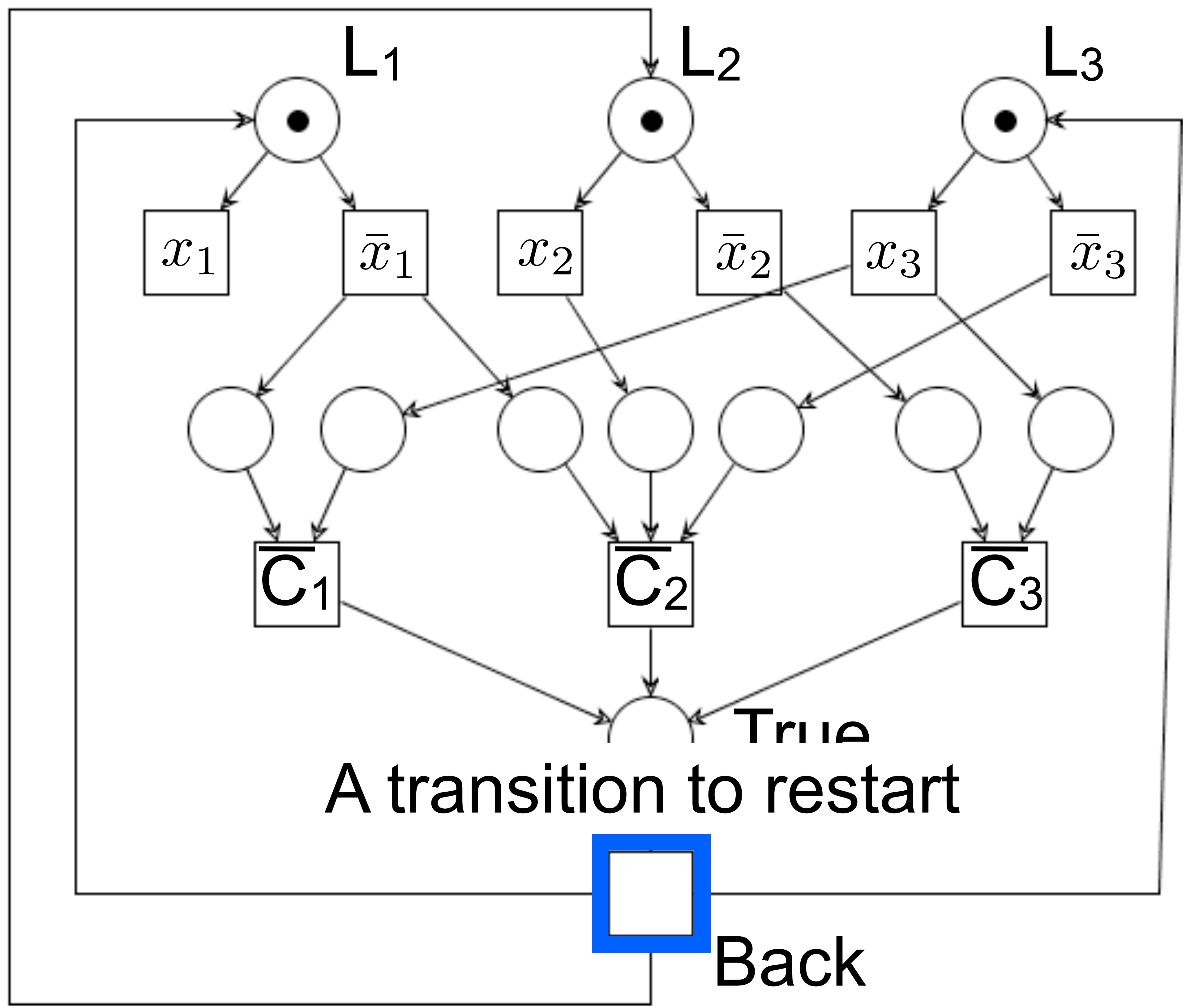


A place for each occurrence of a literal

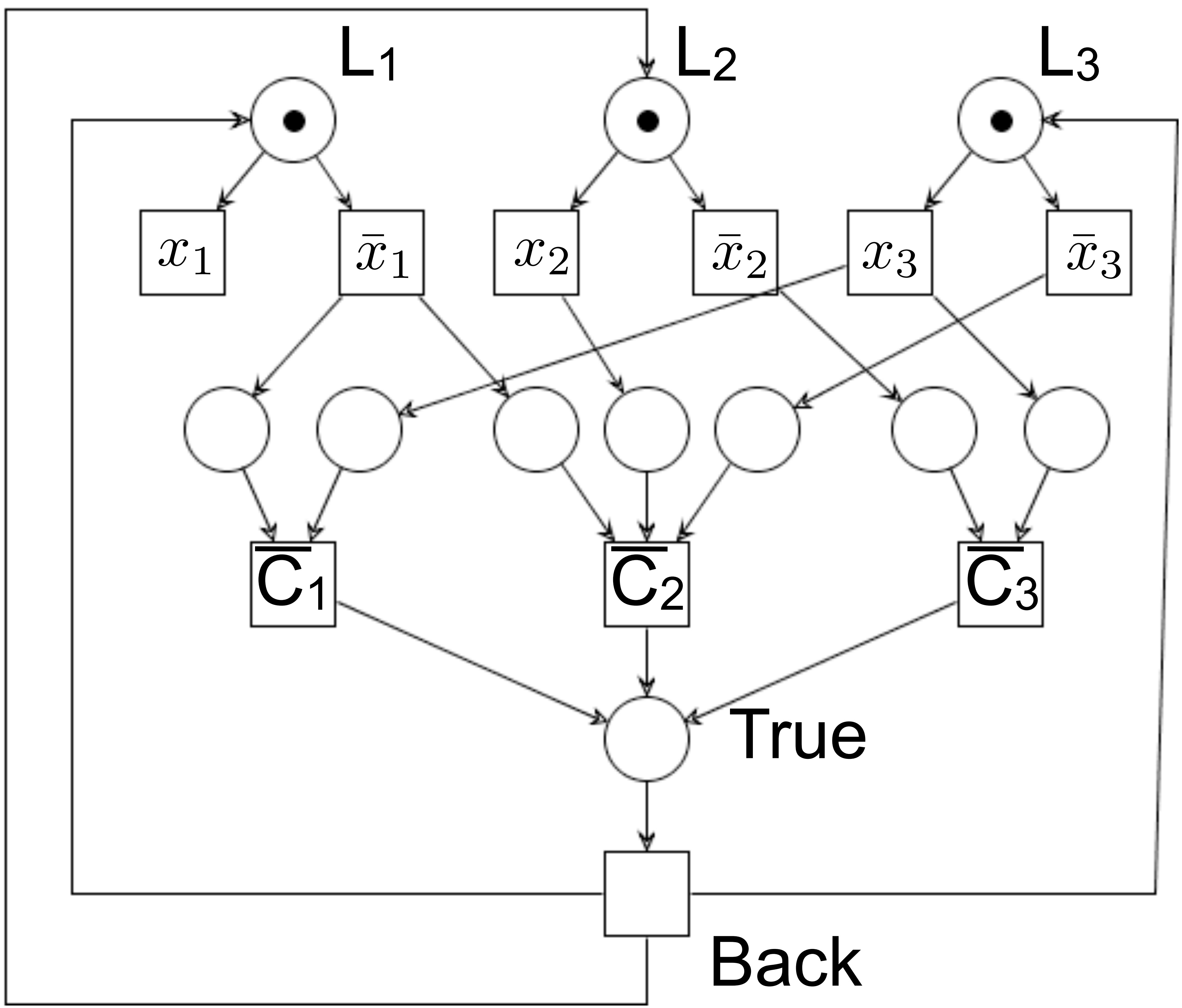
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



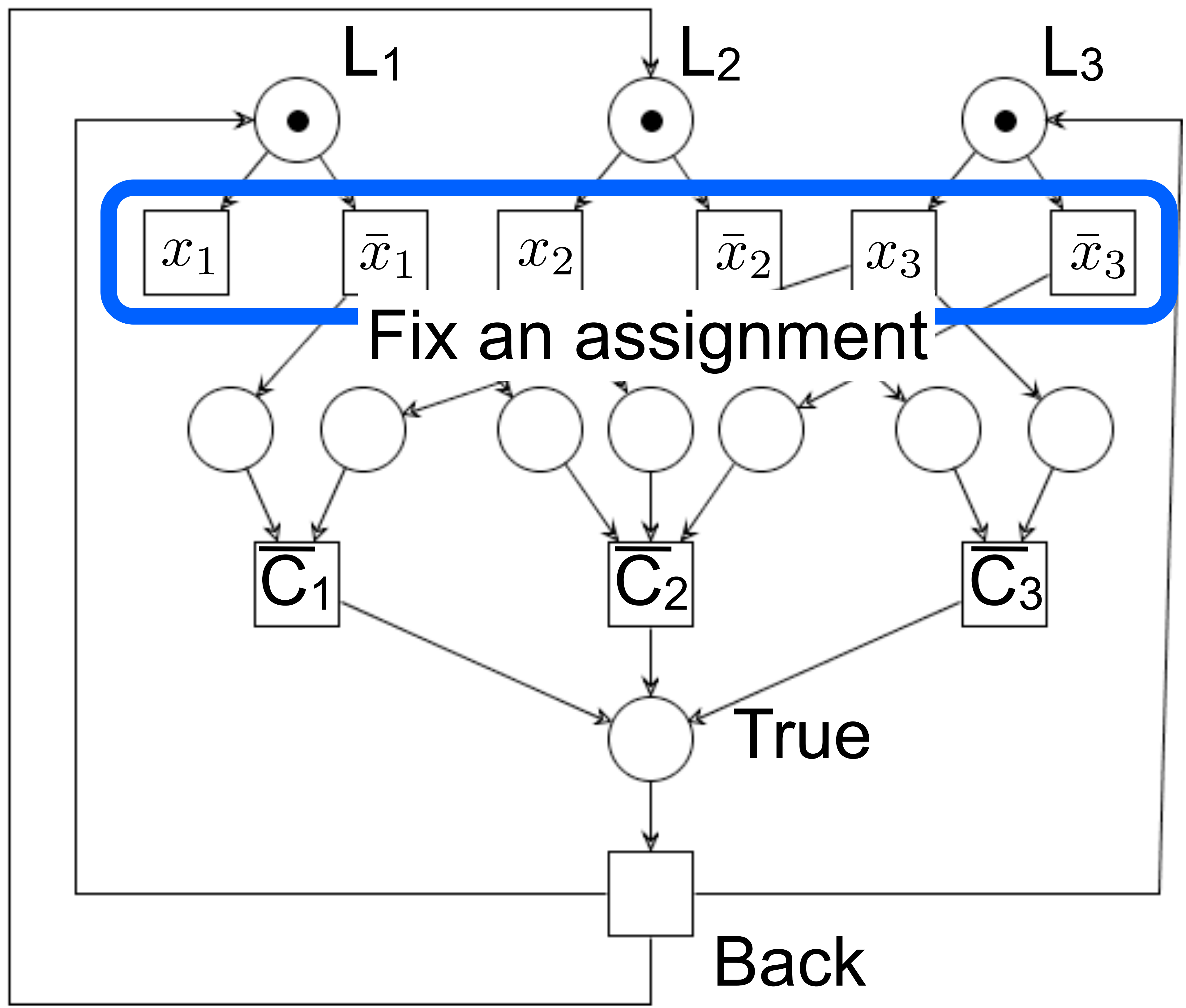
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



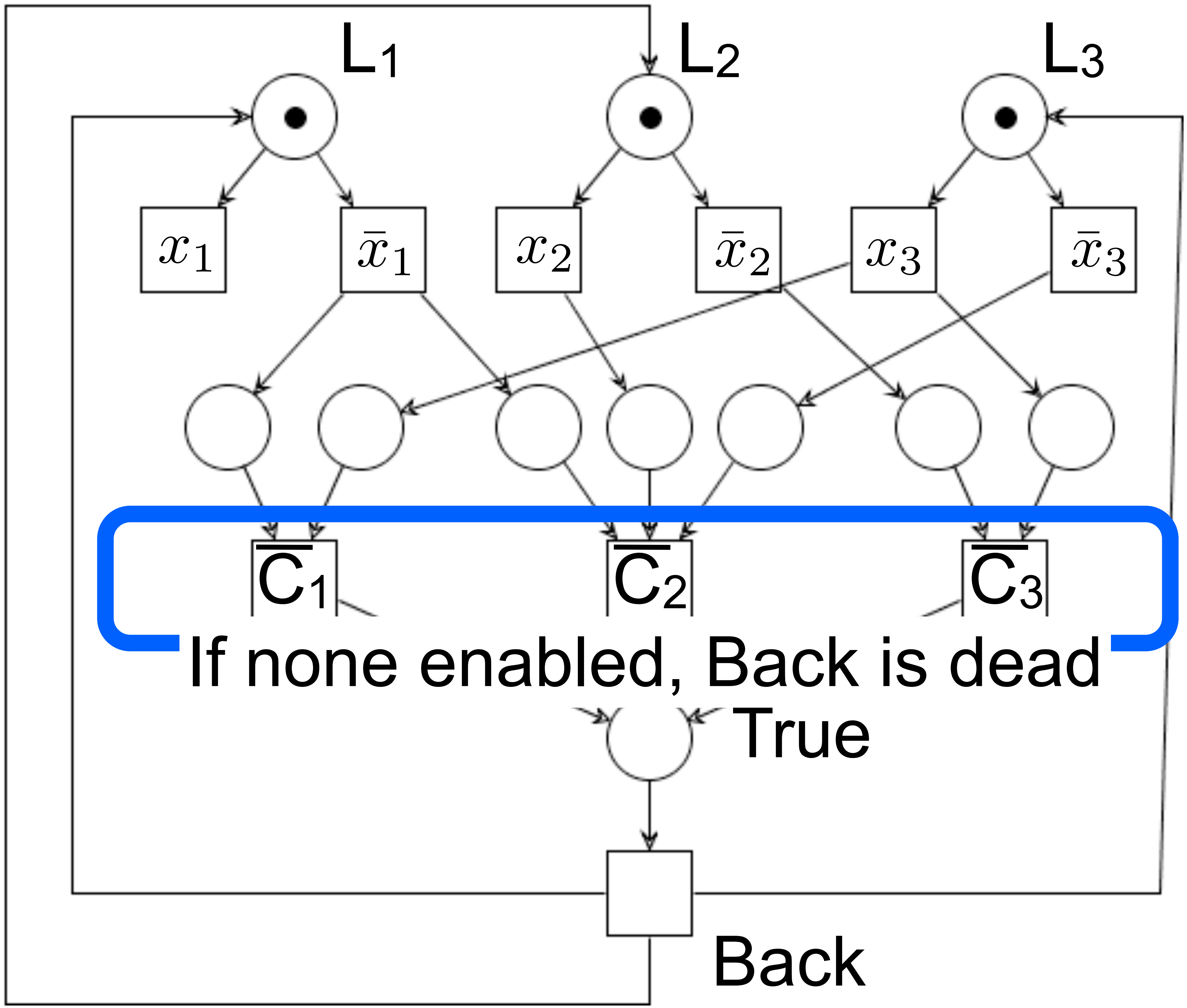
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



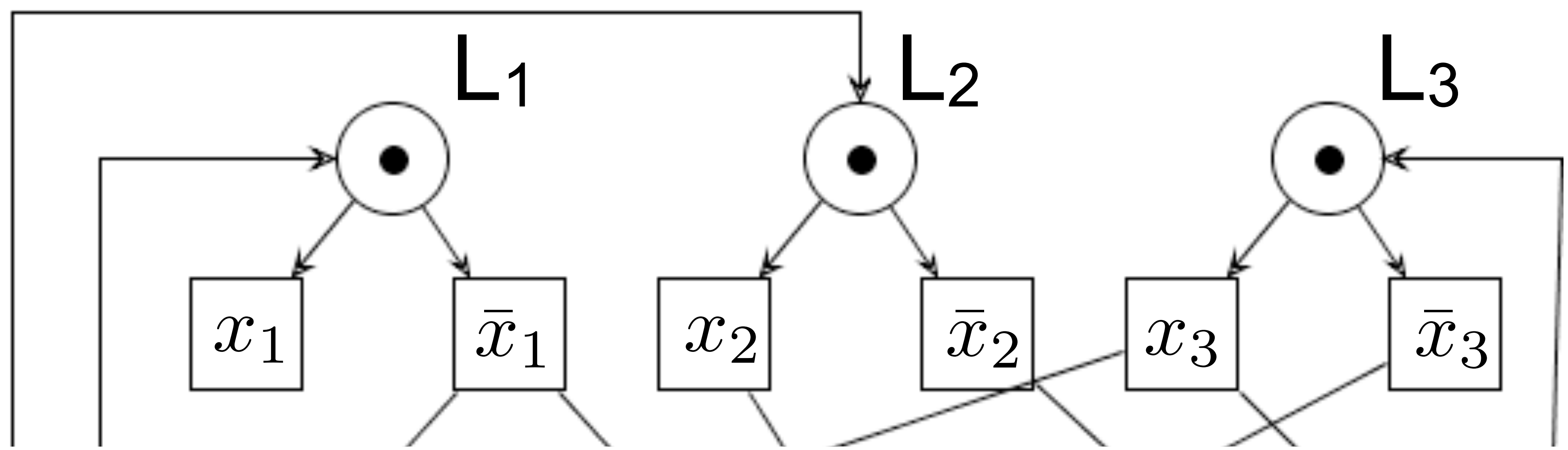
$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$

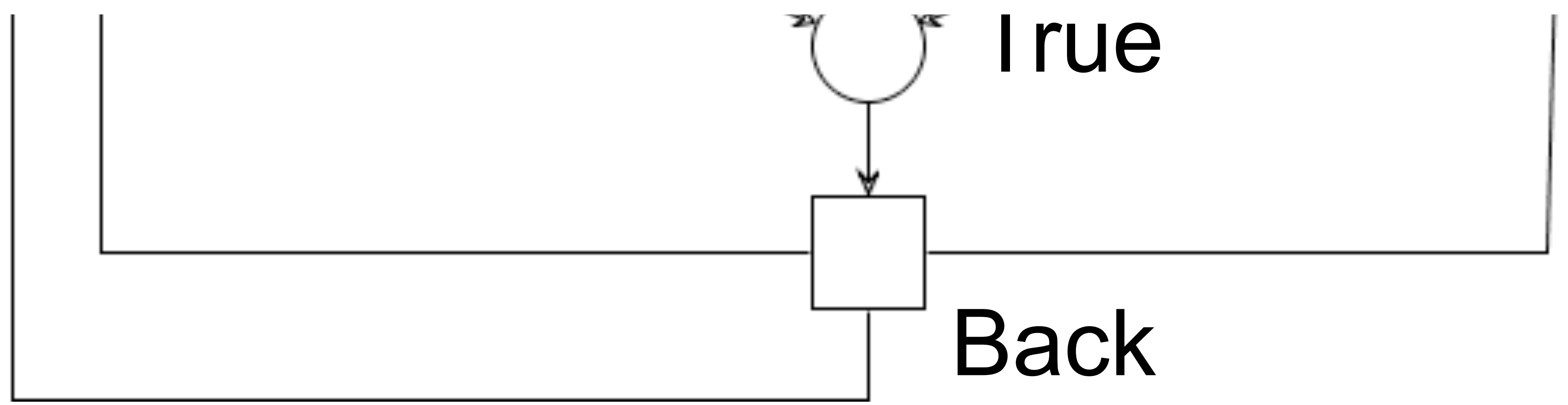


$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$

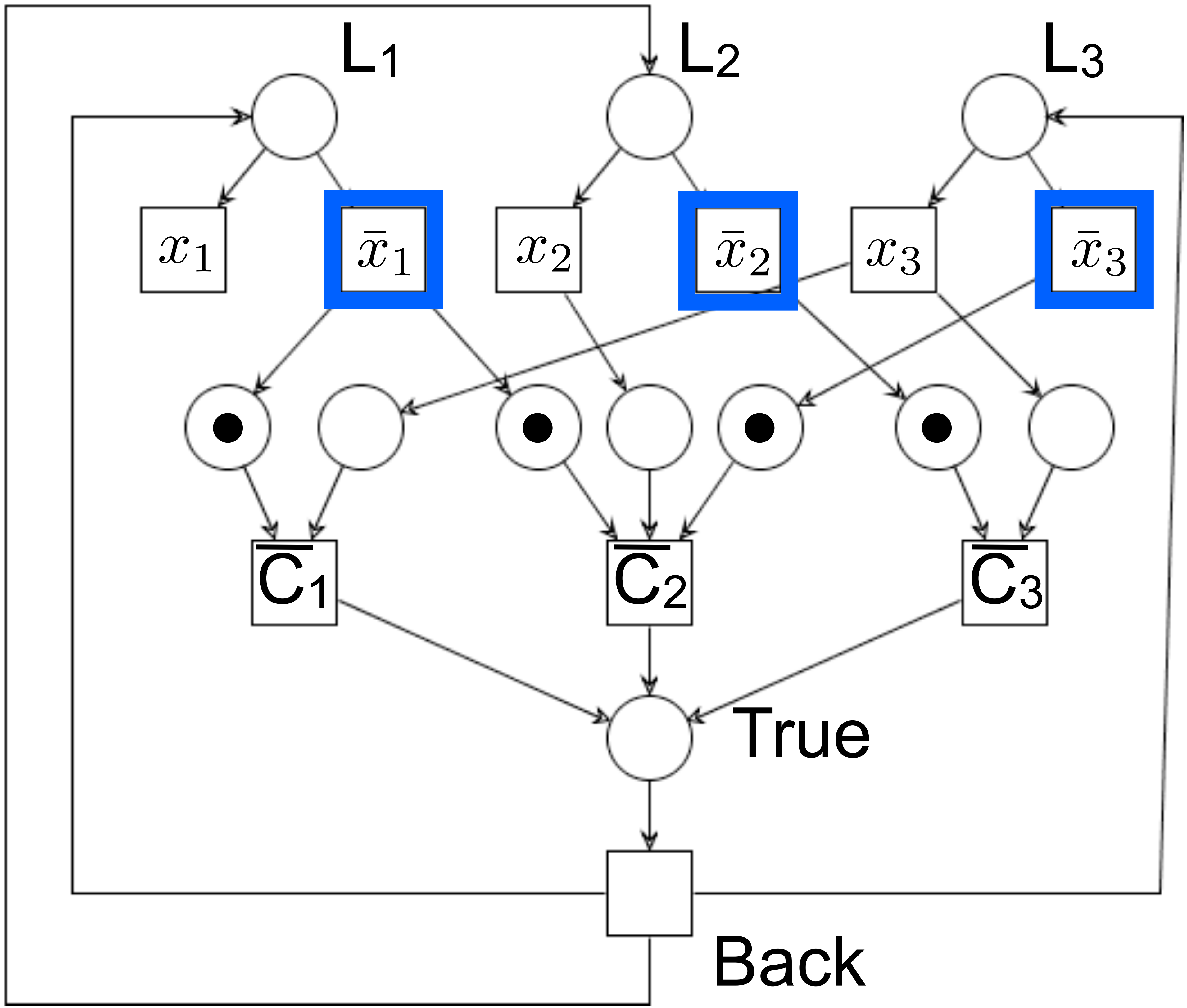


If ϕ is satisfiable, then the net is not live

If the net is not live, then ϕ is satisfiable



$$\neg\phi = (\overline{x}_1 \wedge x_3) \vee (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_2 \wedge x_3)$$



Main consequence

No polynomial algorithm to decide liveness of a free-choice system exists

(unless P=NP)

Exercise

Draw the net corresponding to the formula

$$x_2 \;\wedge\; (x_1 \vee \overline{x}_3 \vee \overline{x}_4) \;\wedge\; (x_1 \vee \overline{x}_2) \;\wedge\; (\overline{x}_1 \vee x_4) \;\wedge\; (\overline{x}_2 \vee \overline{x}_4)$$

Is it satisfiable?

Live and bounded free-choice nets

Rank Theorem

Theorem:

A free-choice system (P,T,F,M0) is live and bounded

iff

1. it has at least one place and one transition
2. it is connected
3. M_0 marks every proper siphon
4. it has a positive S-invariant
5. it has a positive T-invariant
6. rank(N) = $|C_N|$ - 1

(where C_N is the set of clusters)

A polynomial algorithm for maximal siphon

A polynomial algorithm for computing maximal siphon in R

Input: A net $N = (P, T, F, M_0)$, $R \subseteq P$
Output: $Q \subseteq R$

$Q := R$
while $(\exists p \in Q,\ \exists t \in \bullet p,\ t \notin Q\bullet)$
    $Q := Q \setminus \{p\}$
return Q

Q is a **siphon** if $\bullet Q \subseteq Q\bullet$

A polynomial algorithm for maximal unmarked siphon

3. M_0 marks every proper siphon

Input: A net $N = (P, T, F, M_0)$, $R = \{ p \mid M_0(p) = 0 \}$
Output: $Q \subseteq R$ maximal unmarked siphon

$Q := R$
while $(\exists p \in Q,\ \exists t \in \bullet p,\ t \notin Q\bullet)$
$\quad Q := Q \setminus \{p\}$
return Q

If Q is empty then M_0 marks every proper siphon

Main consequence

Given a free-choice system, the problem to decide if it is live and bounded can be solved in polynomial time

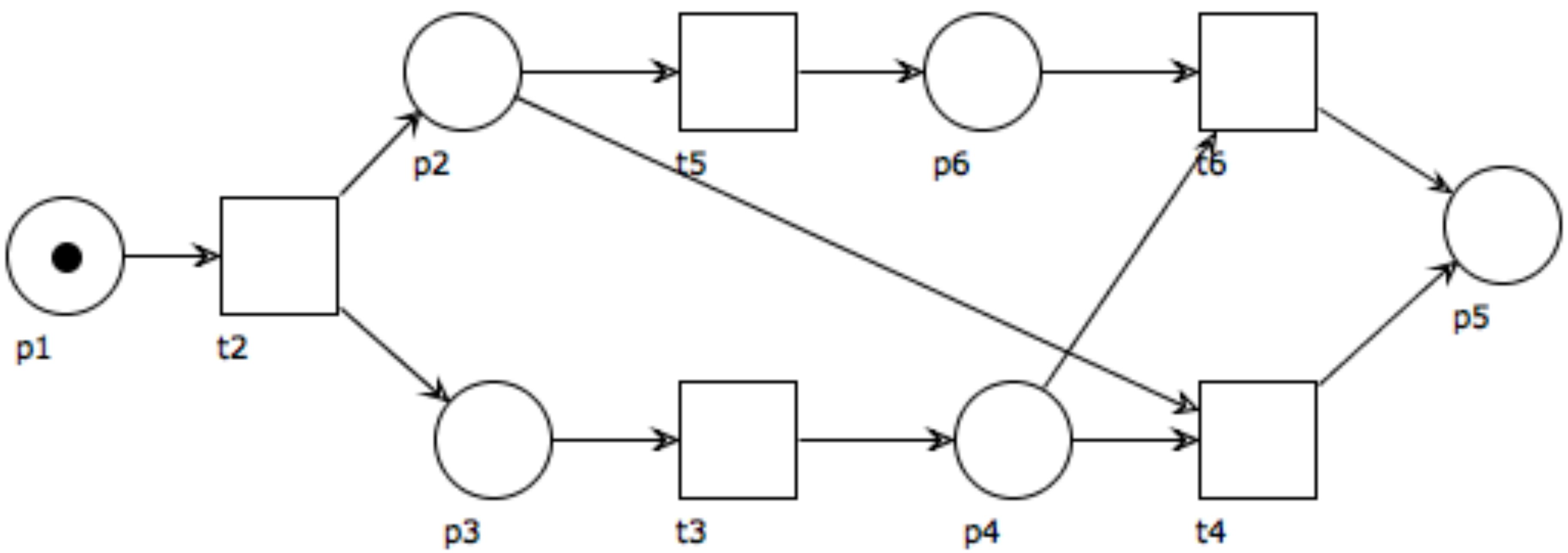
Free-Choice vs Soundness

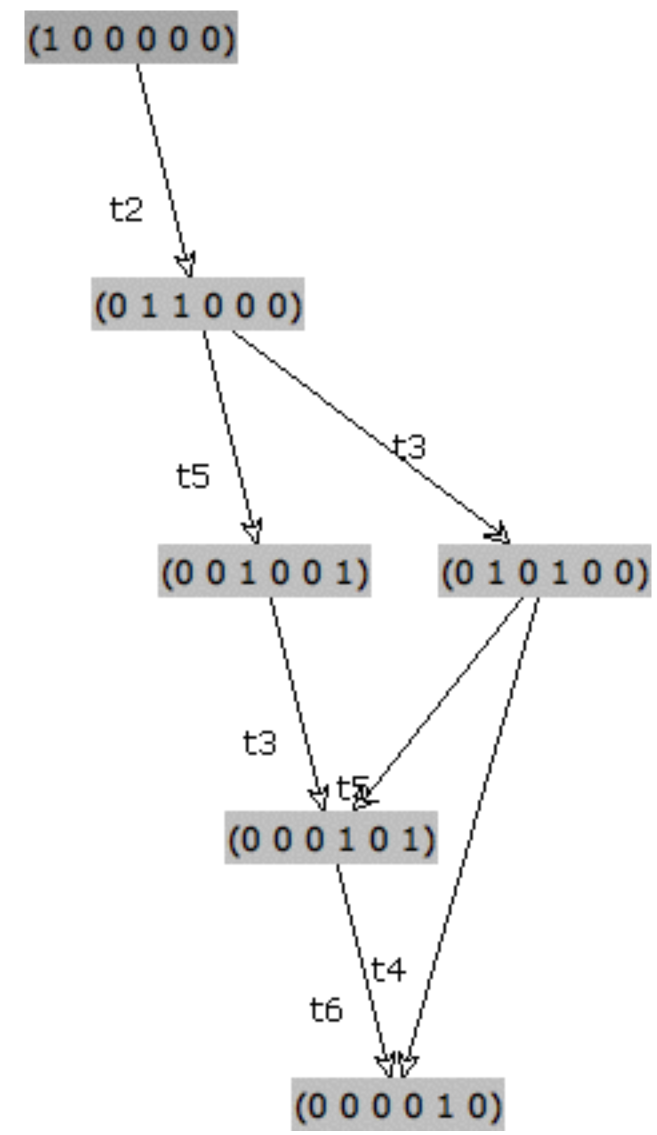
Note that free-choice is orthogonal to soundness:

there exists WF-nets that are free-choice but not sound

there exists WF-nets that are sound but not free-choice

Example: sound but not free-choice





Exercise

Draw a workflow net that is free-choice but not sound

Coverability

S-Coverability analysis

A case is often composed by parallel threads of control (each thread imposing some order over its tasks)

The notion of S-coverability allows to reveal such threads

A technique to find a positive S-invariant

Decompose the free-choice net N in suitable S-nets so
that any place of N belongs to an S-net
(the same place can appear in more S-nets)

Each S-net provides a uniform S-invariant

A positive S-invariant is obtained
as the sum of the S-invariants of each subnet

S-component

Definition: Let $N = (P, T, F)$ and $\emptyset \subset X \subseteq P \cup T$
Let $N' = (P \cap X, T \cap X, F \cap (X \times X))$ be a subnet of N.
N' is an **S-component** if

1. it is a strongly connected S-net
2. for every place $p \in X \cap P$, we have $\bullet p \cup p\bullet \subseteq X$

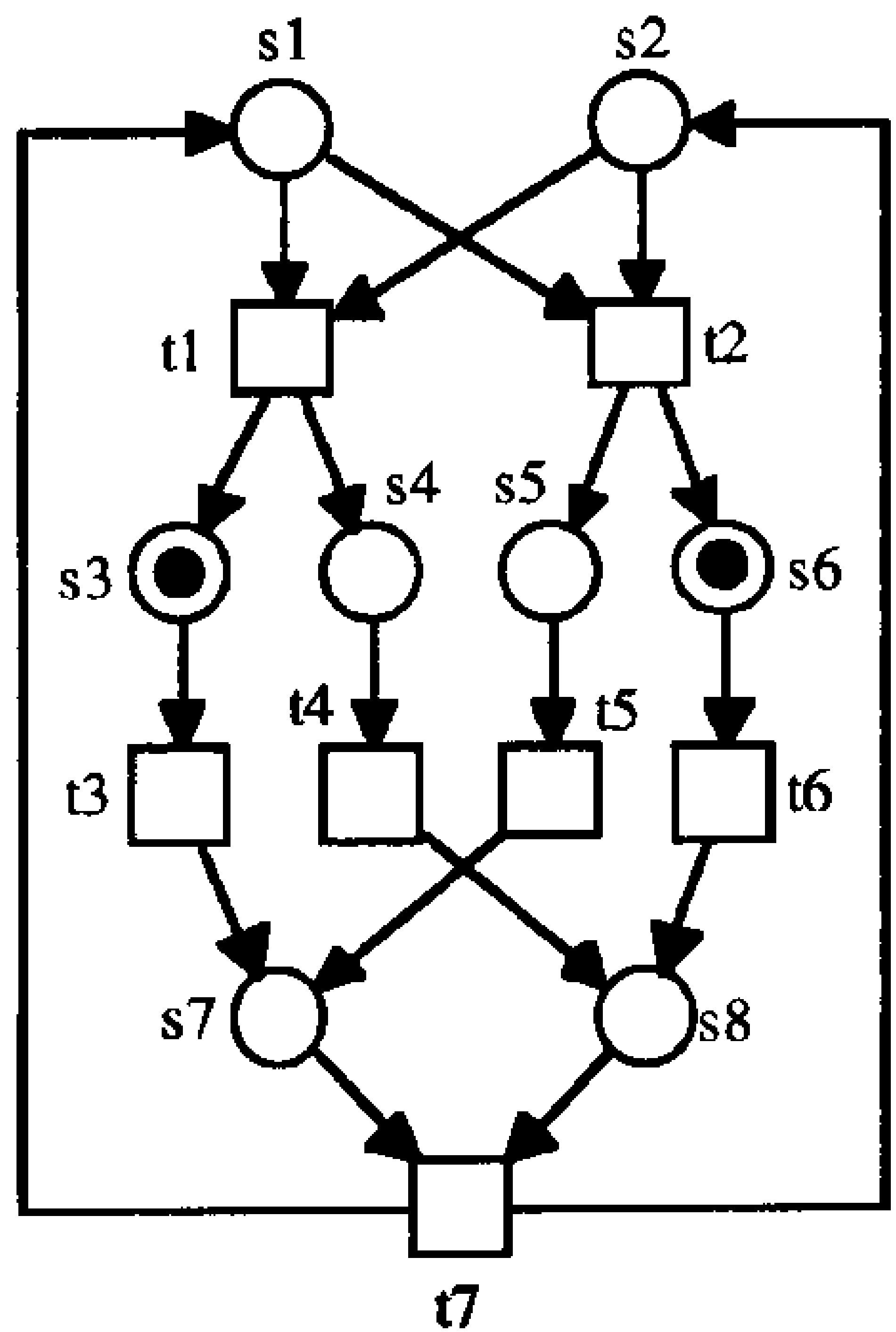
S-cover

Definition: Let **C** be a set of S-components of a net N

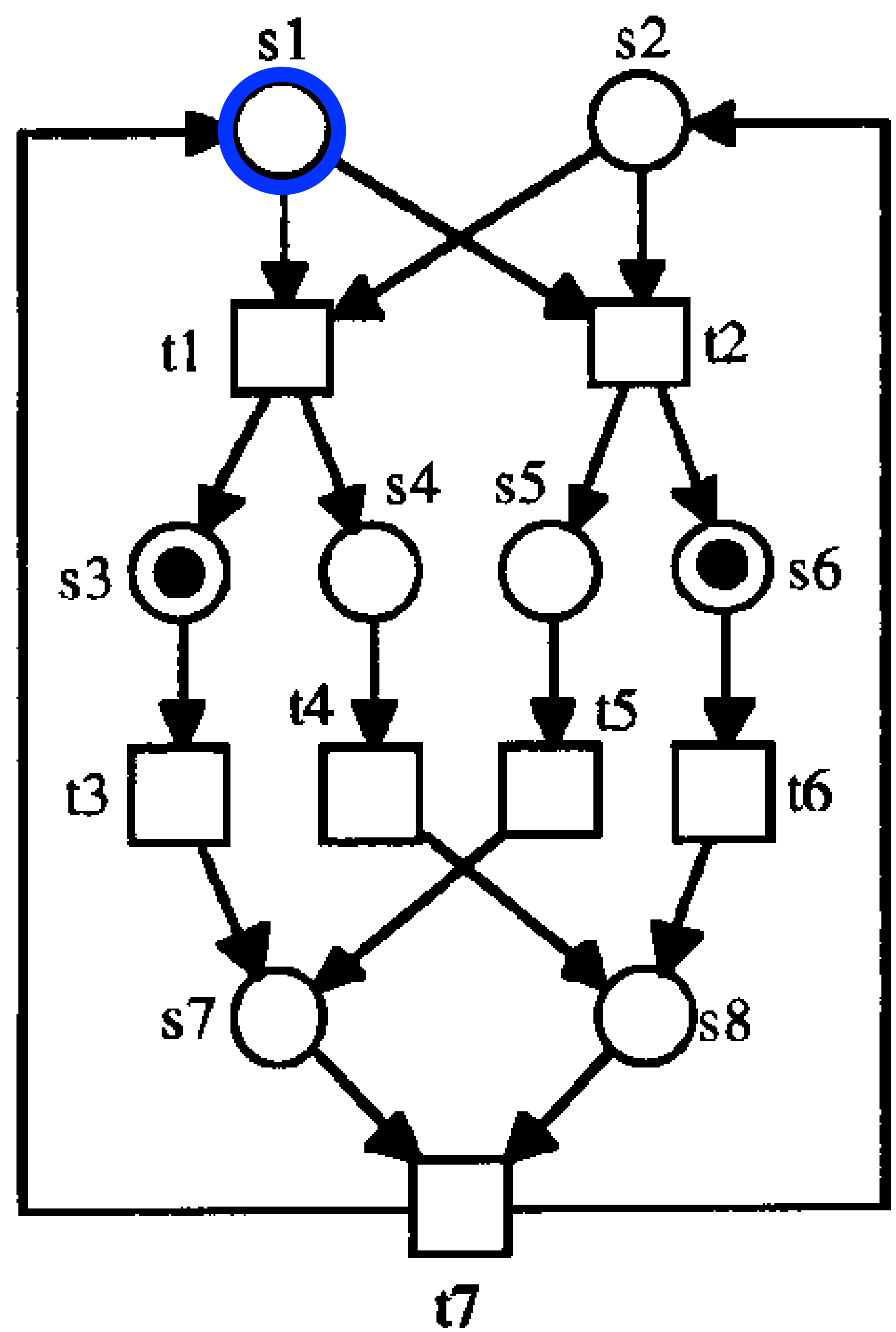
C is an **S-cover** if every place p of N belongs to one or more S-components in **C**

We say that N is **covered by S-components** if it has an S-cover

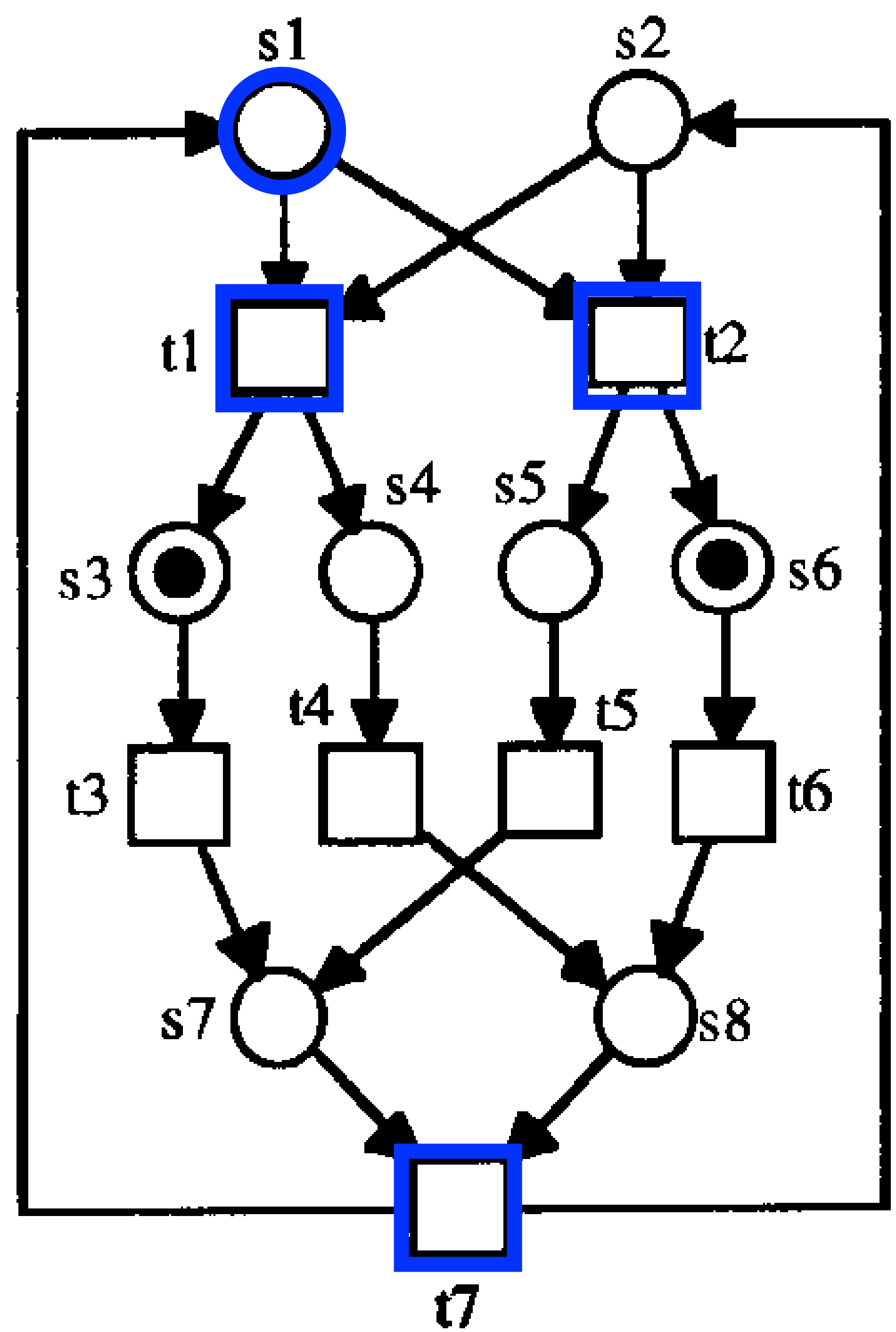
S-cover: example



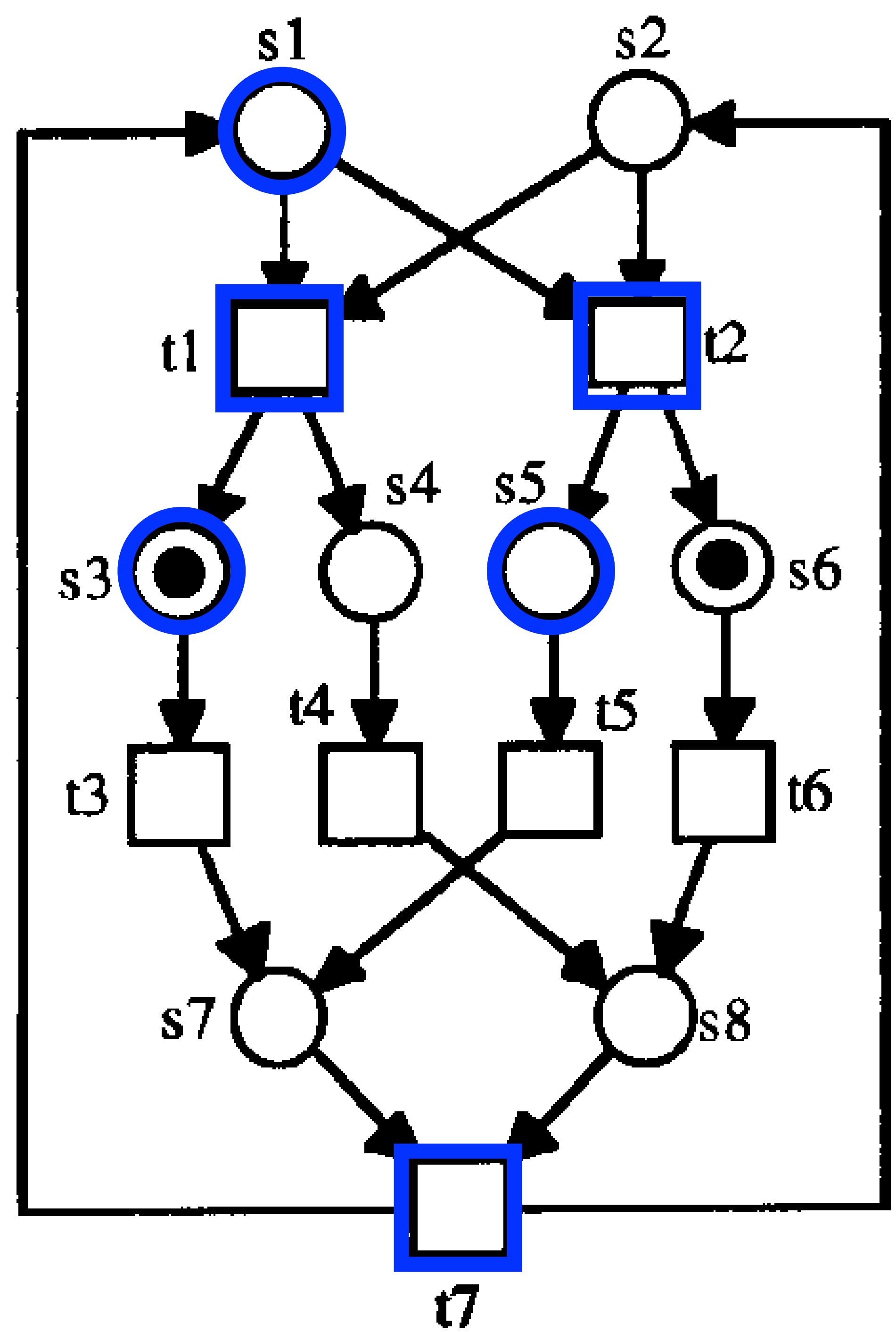
S-cover: example



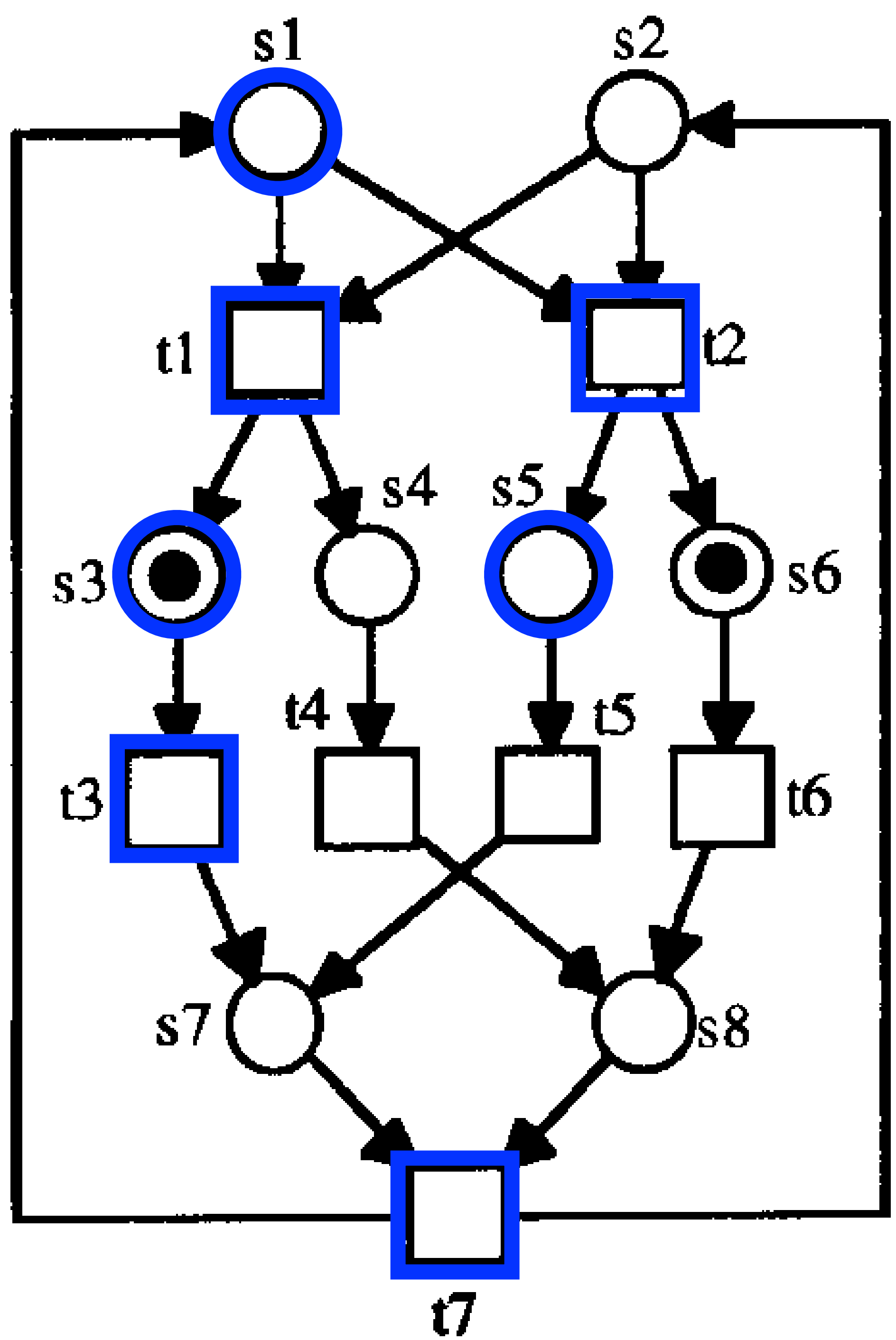
S-cover: example



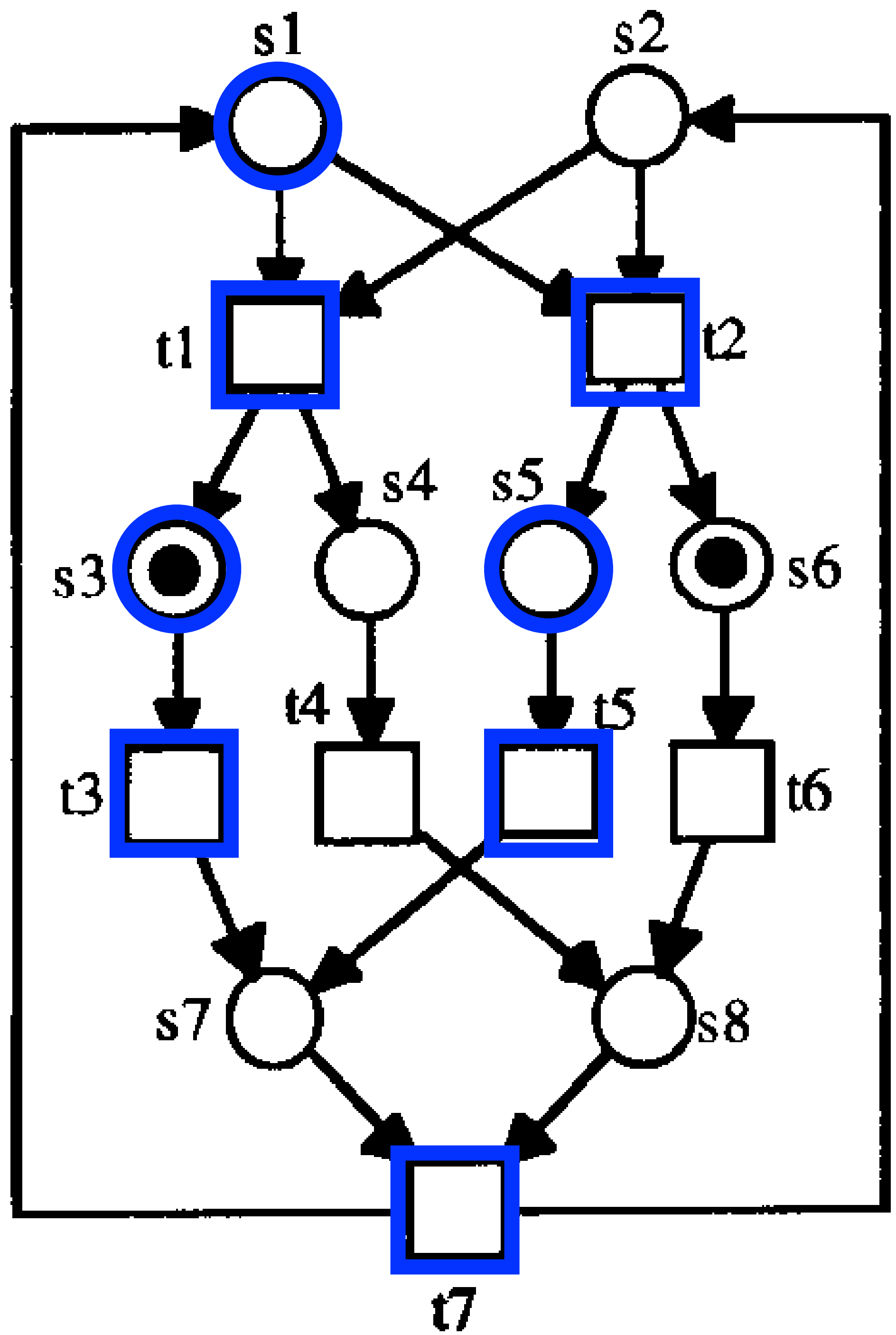
S-cover: example



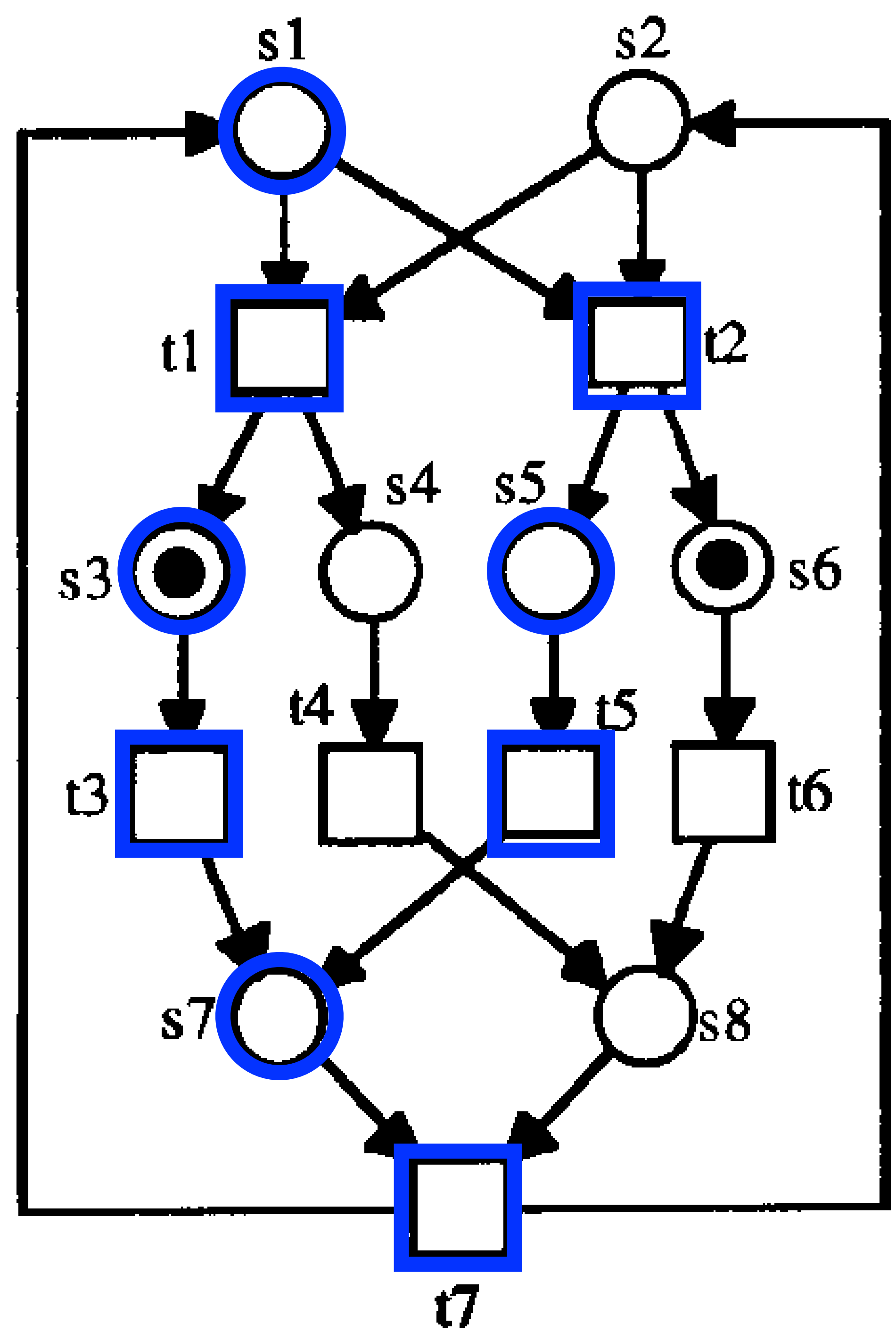
S-cover: example



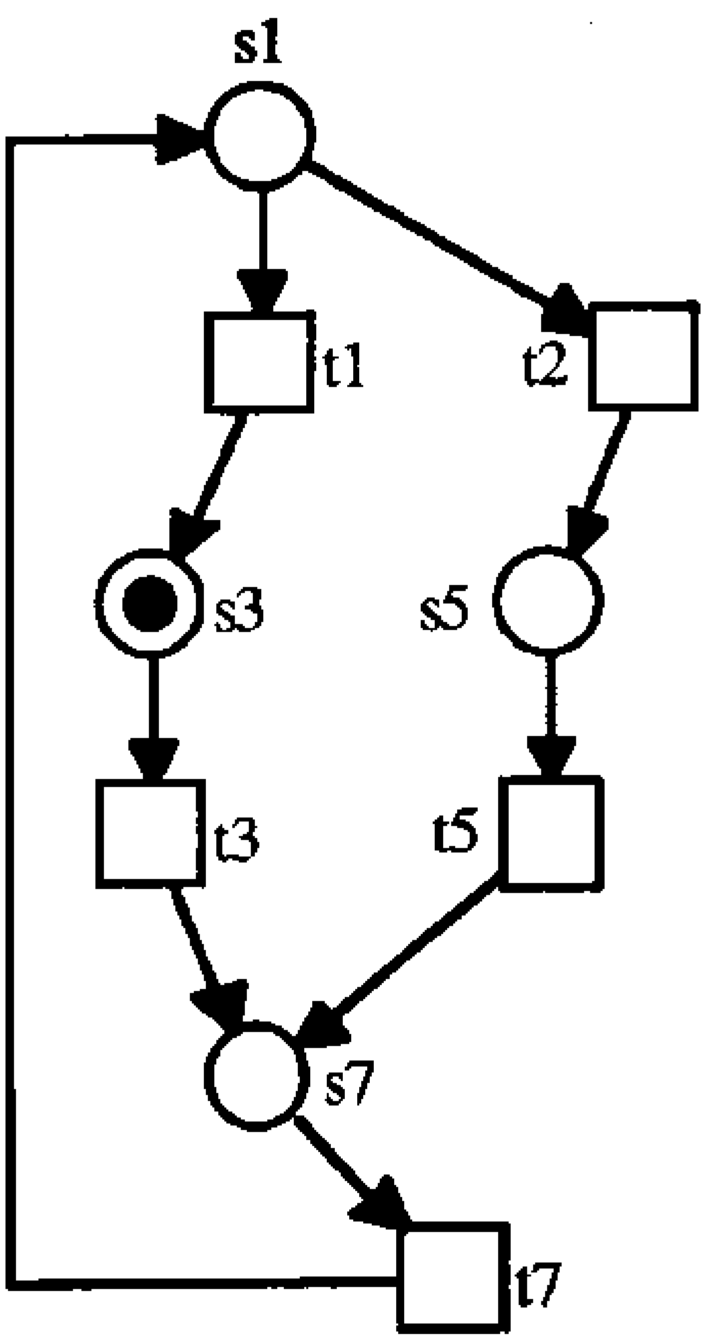
S-cover: example

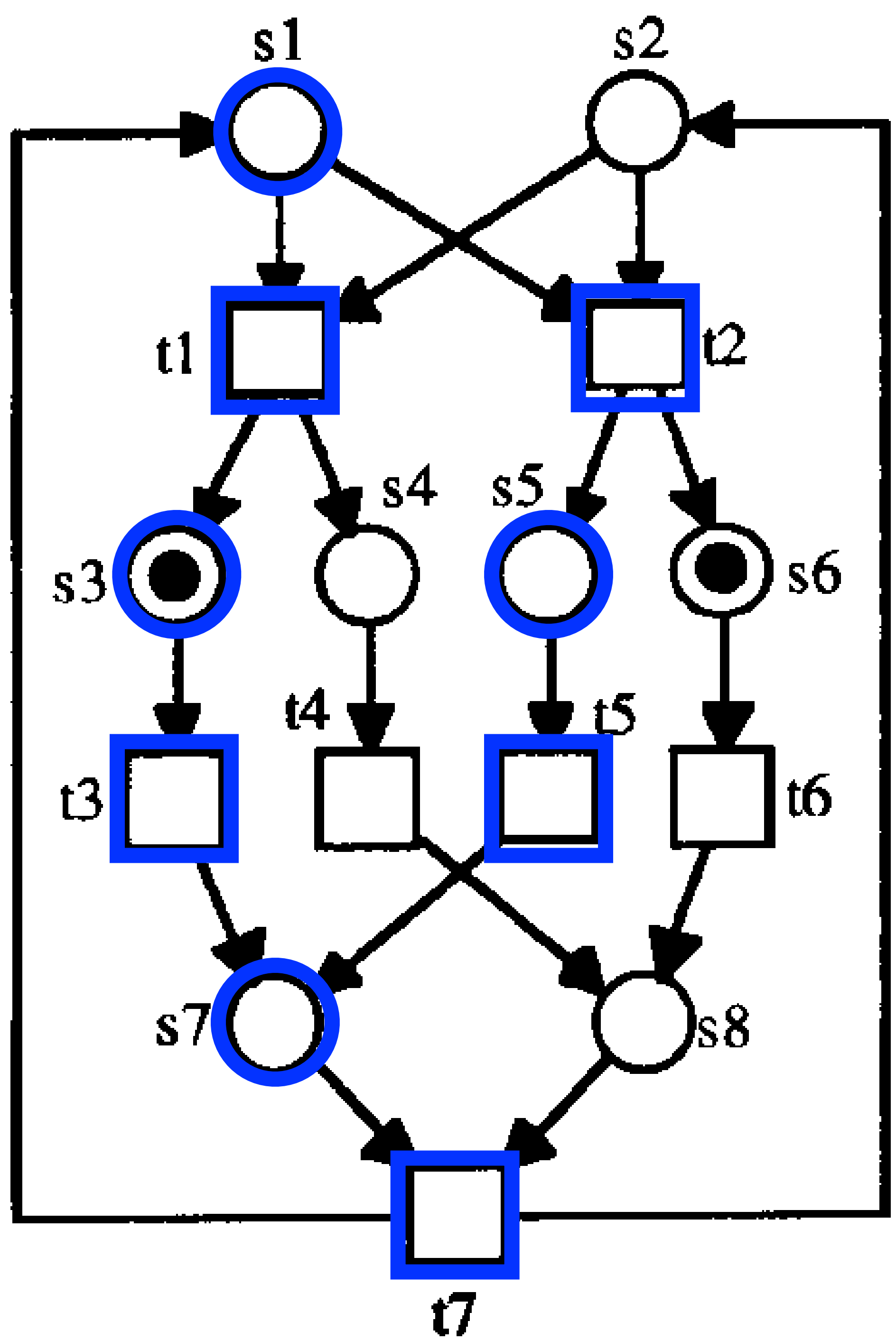


S-cover: example

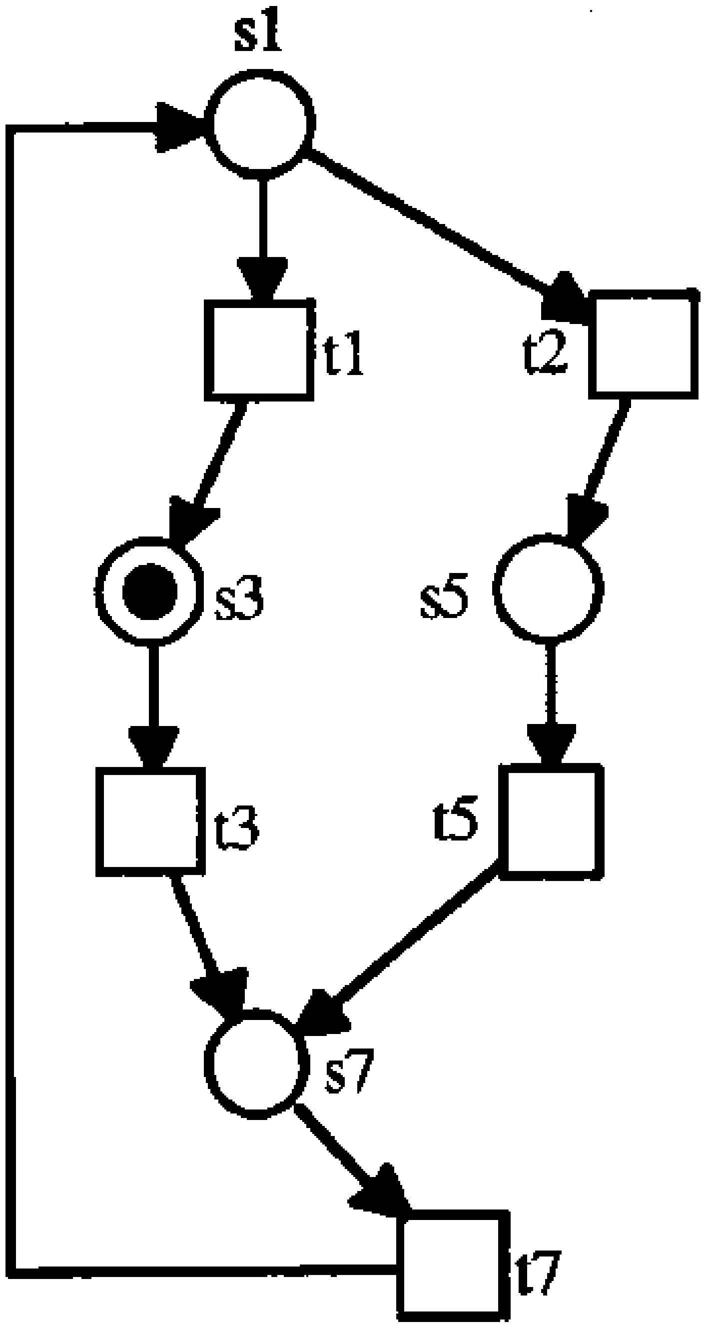


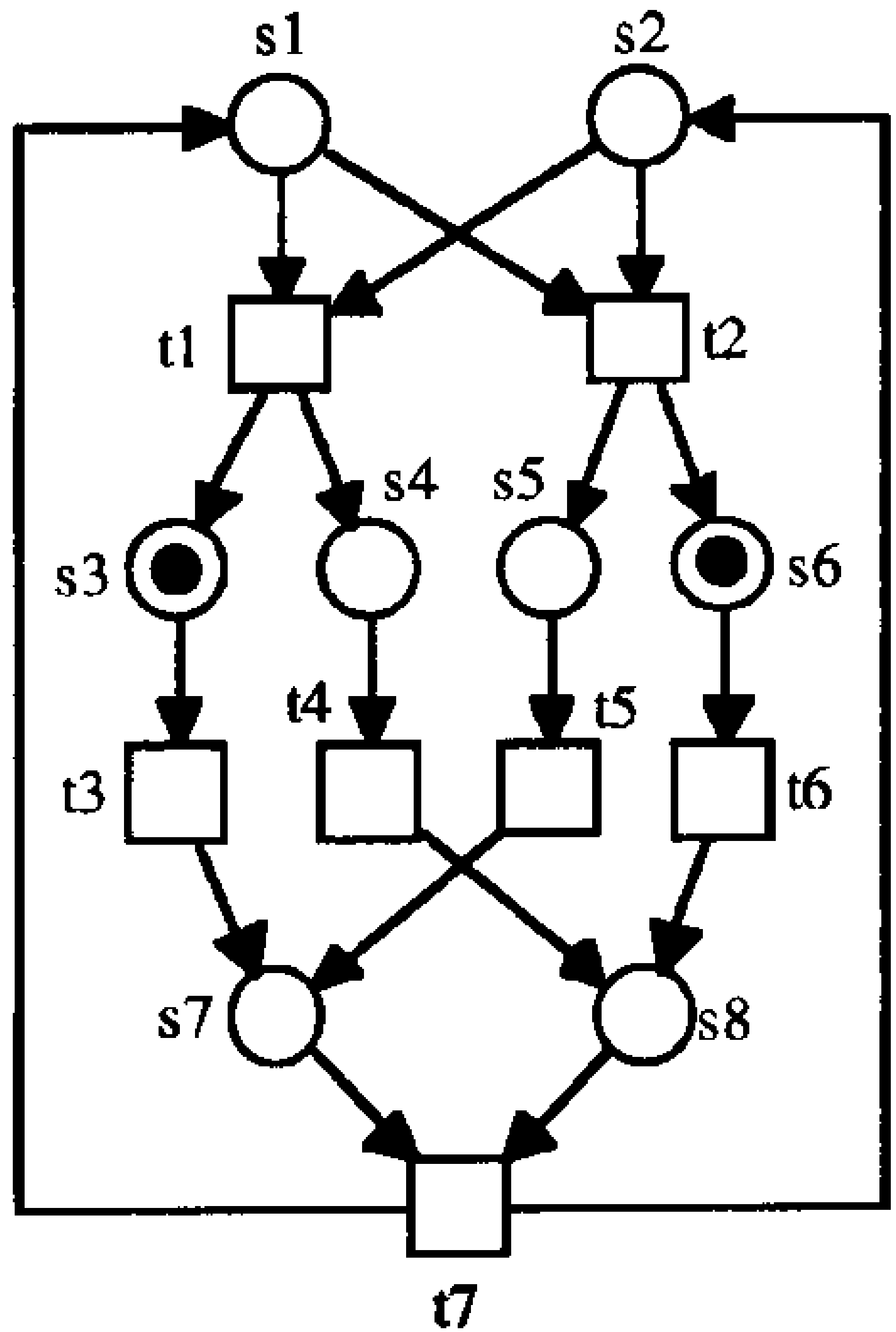
S-cover: example



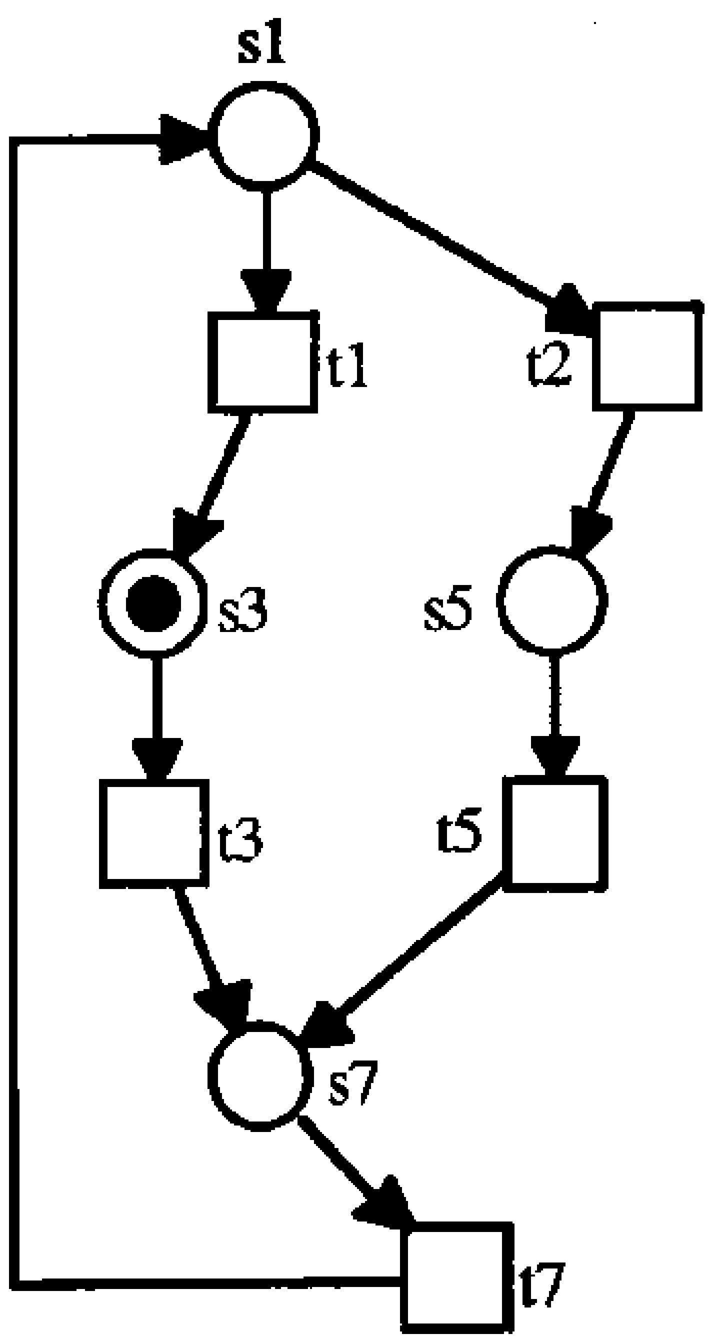


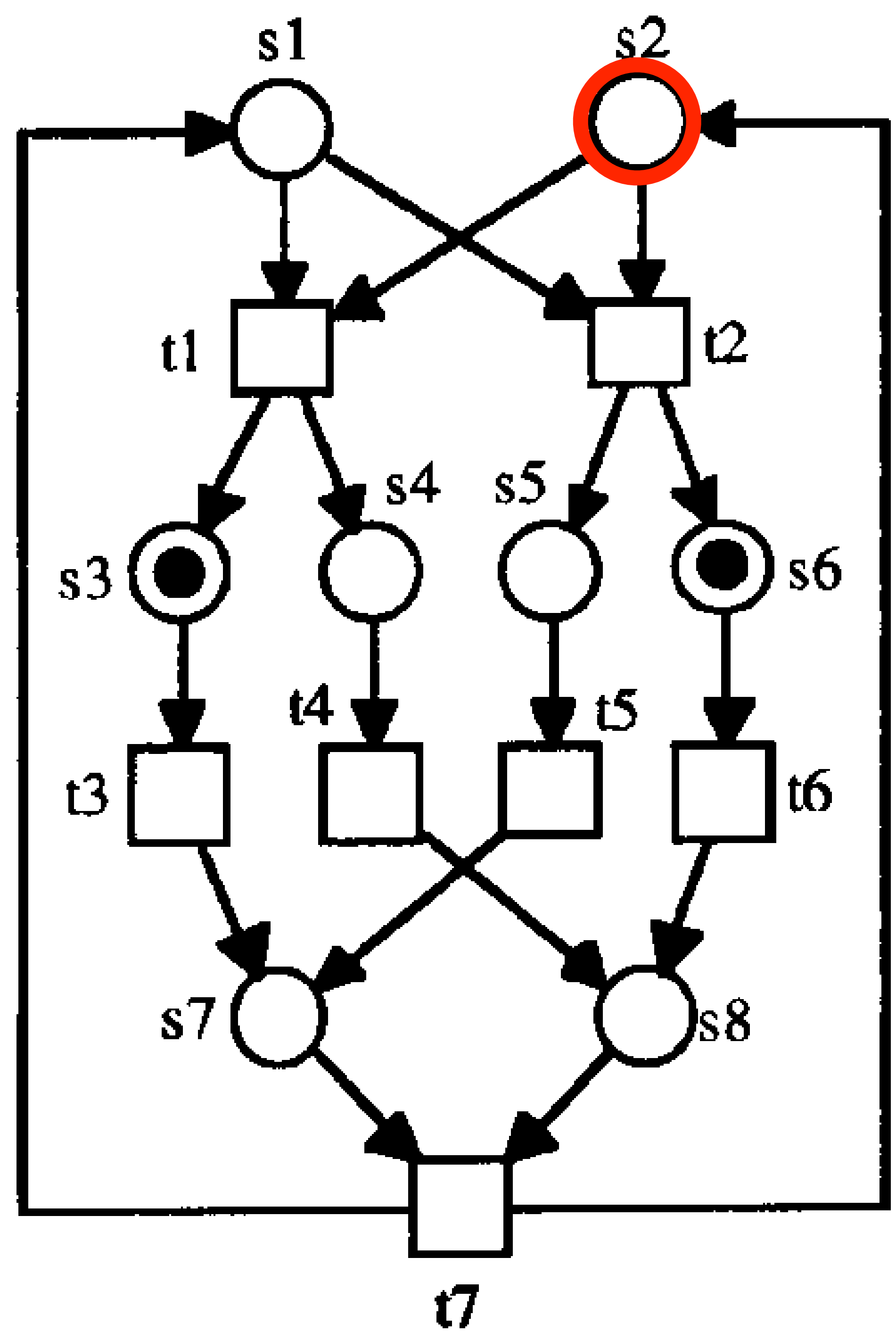
S-cover: example



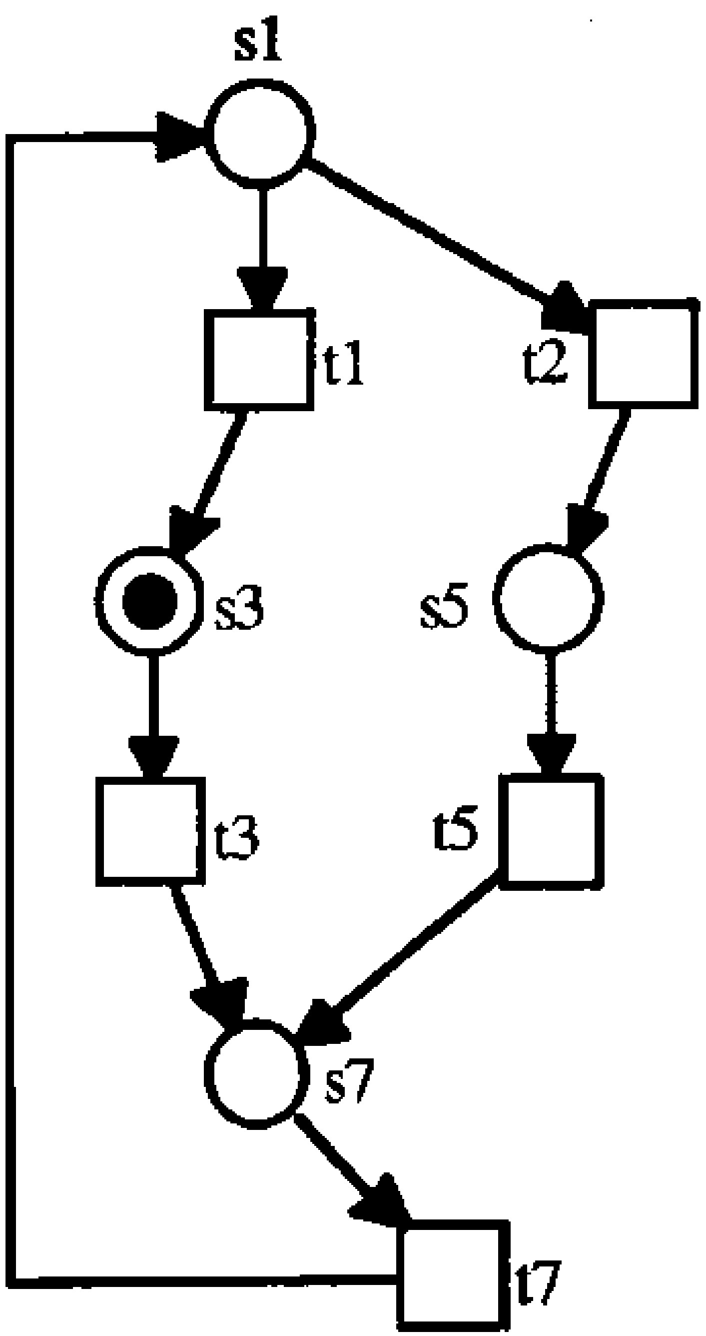


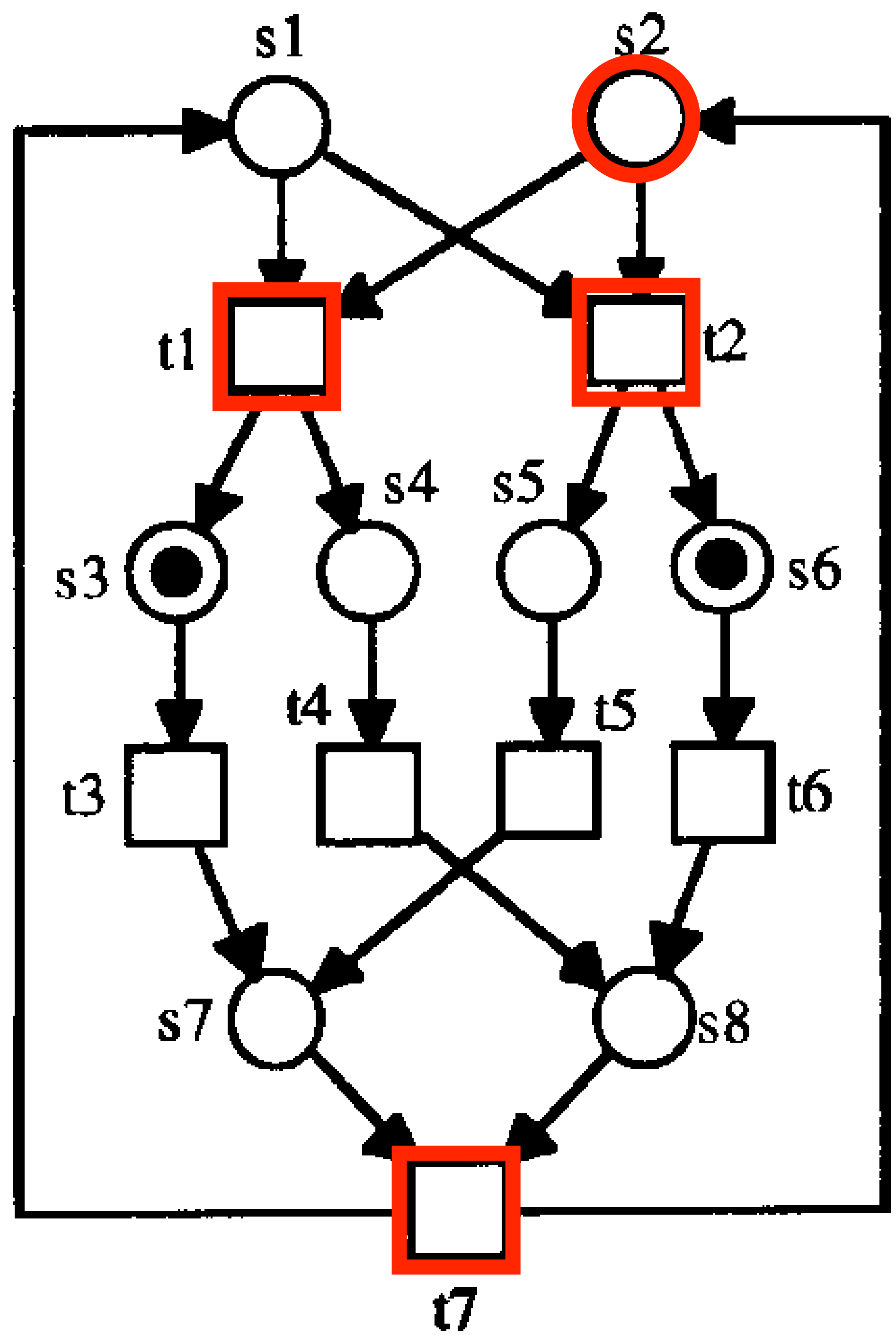
S-cover: example



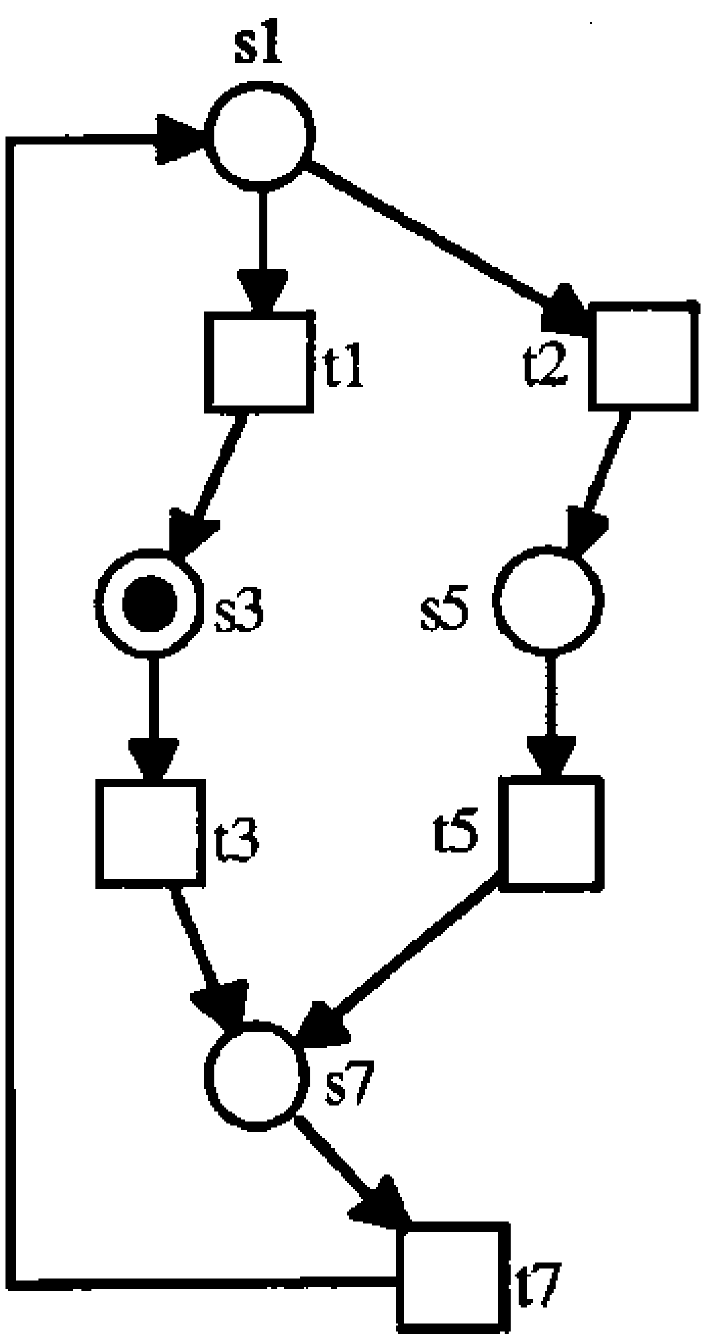


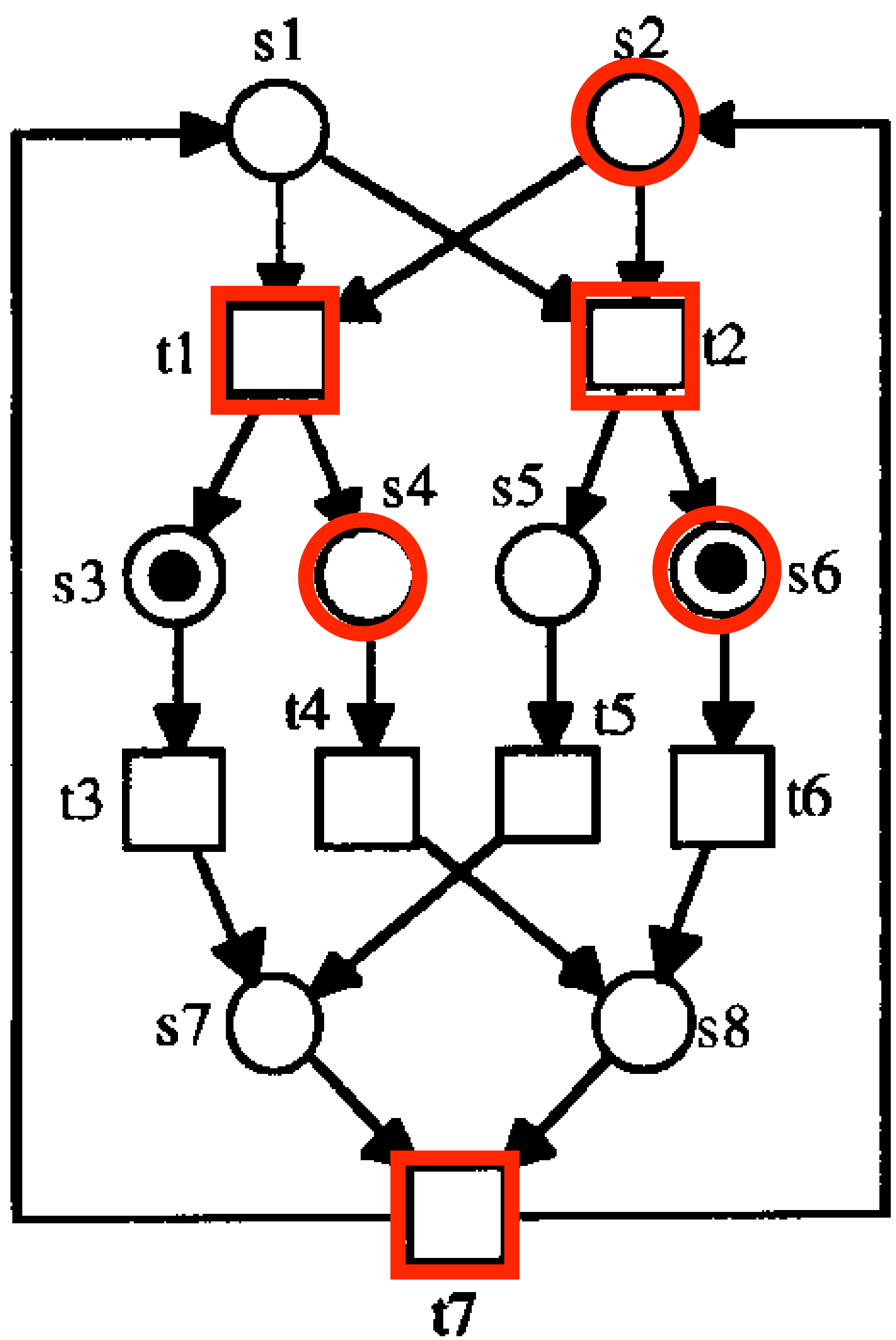
S-cover: example



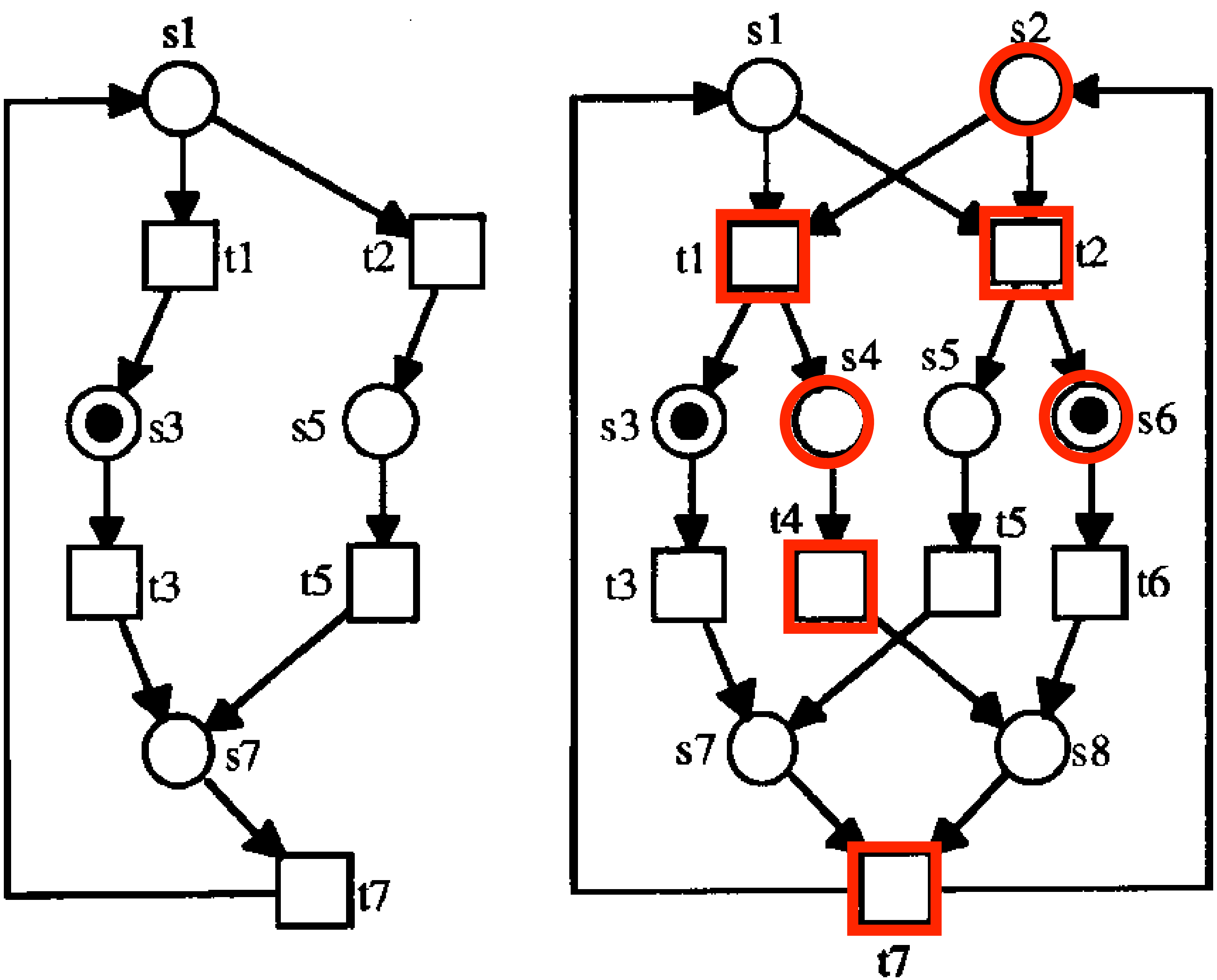


S-cover: example

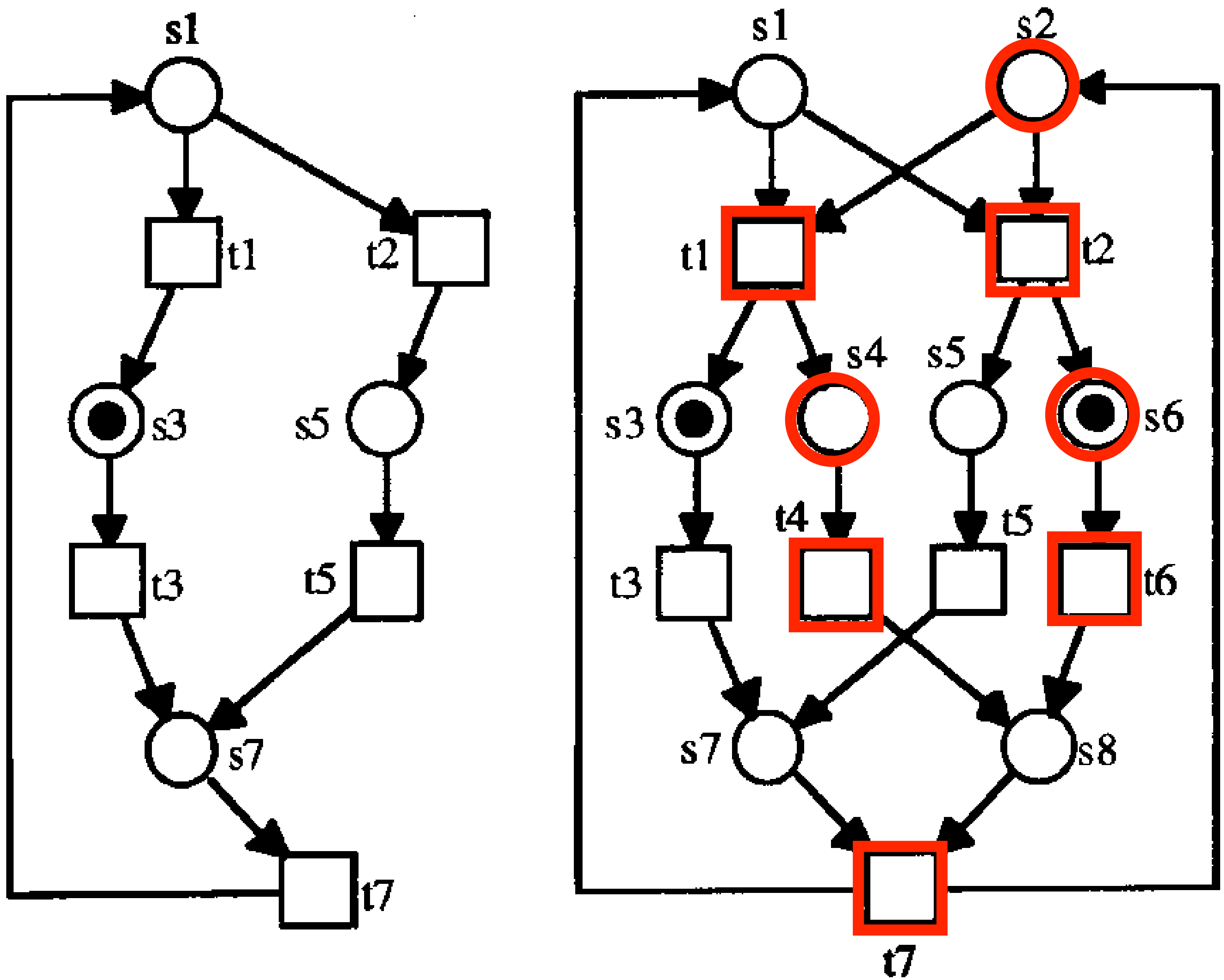




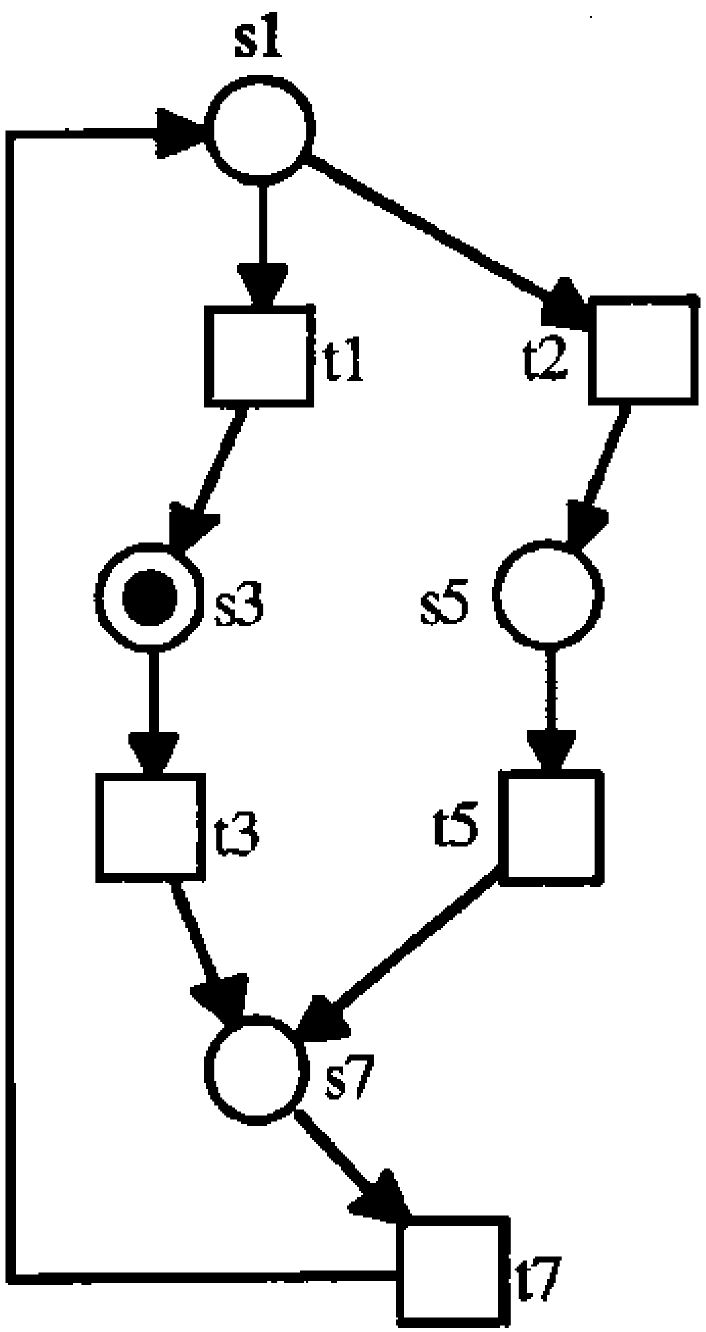
S-cover: example

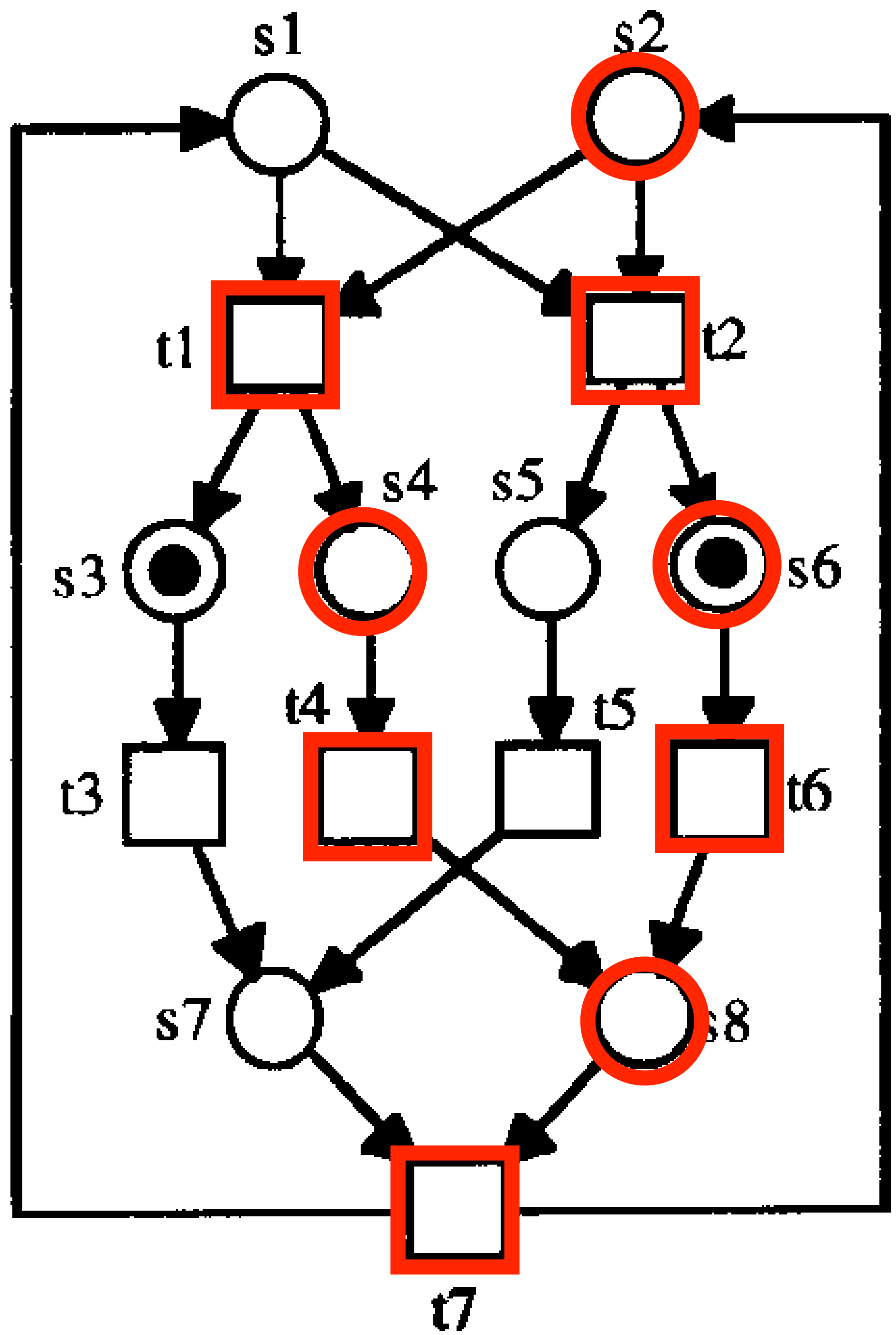


S-cover: example

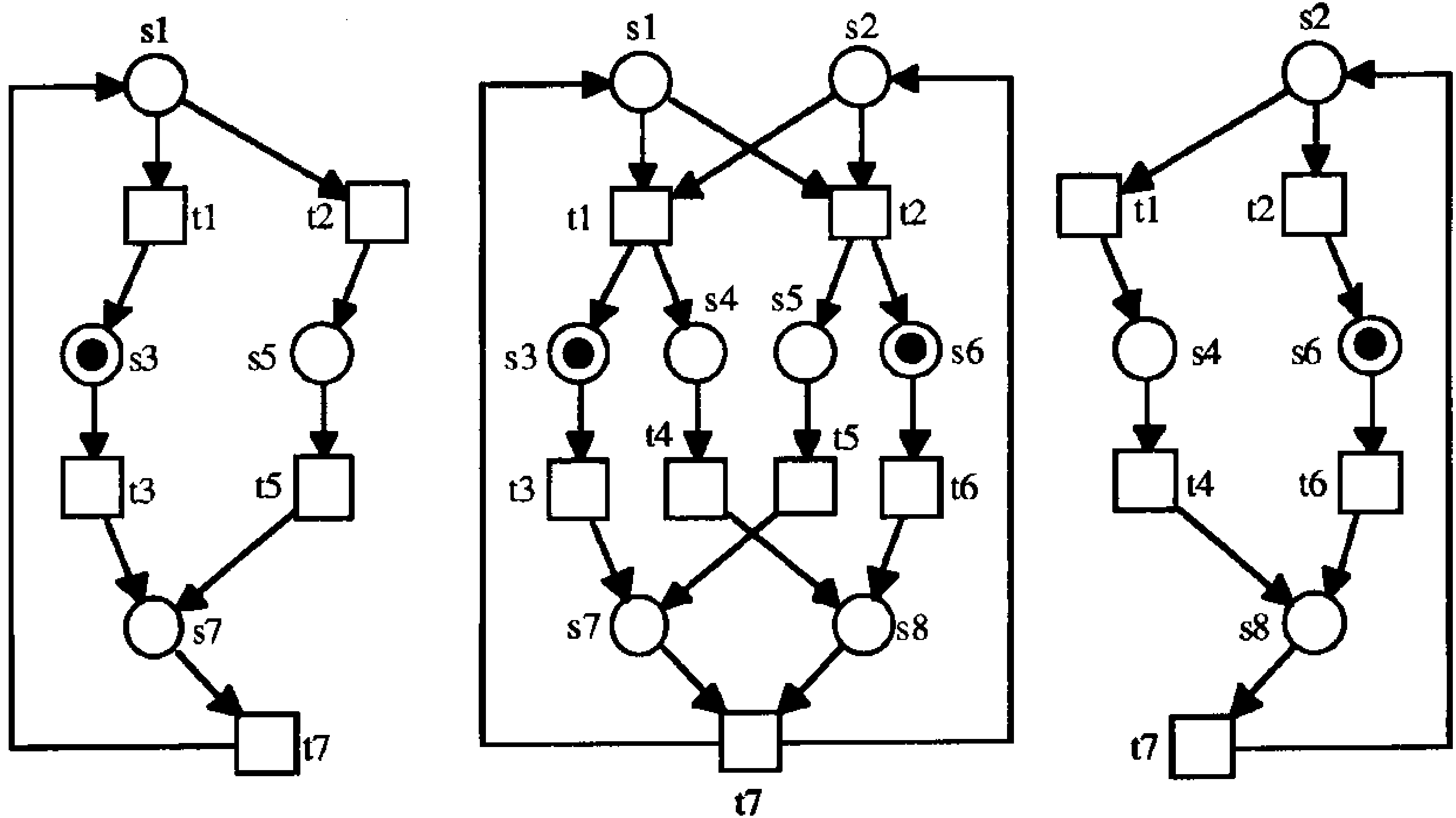


S-cover: example





S-cover: example



S-coverability theorem

Theorem: If a free-choice net N is live and bounded then N is S-coverable

(proof omitted)

Consequence:

free-choice + not S-coverable => not (live and bounded)

A technique to find a positive T-invariant

Decompose the free-choice net N in suitable T-nets so that any transition of N belongs to a T-net (the same transition can appear in more T-nets)

Each T-net provides a uniform T-invariant

A positive T-invariant is obtained as the sum of the T-invariants of each subnet

T-component

Definition: Let $N = (P, T, F)$ and $\emptyset \subset X \subseteq P \cup T$
Let $N' = (P \cap X, T \cap X, F \cap (X \times X))$ be a subnet of N.
N' is a **T-component** if

1. it is a strongly connected T-net
2. for every transition $t \in X \cap T$, we have $\bullet t \cup t \bullet \subseteq X$

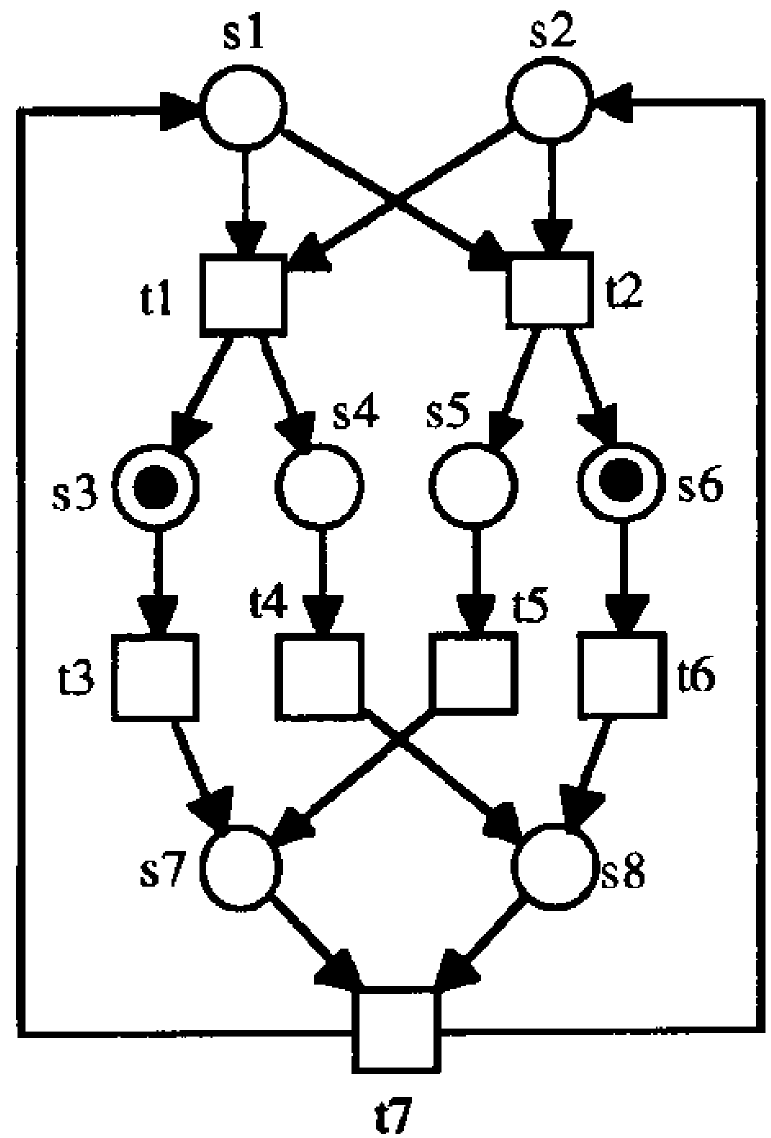
T-cover

Definition: Let **C** be a set of T-components of a net N

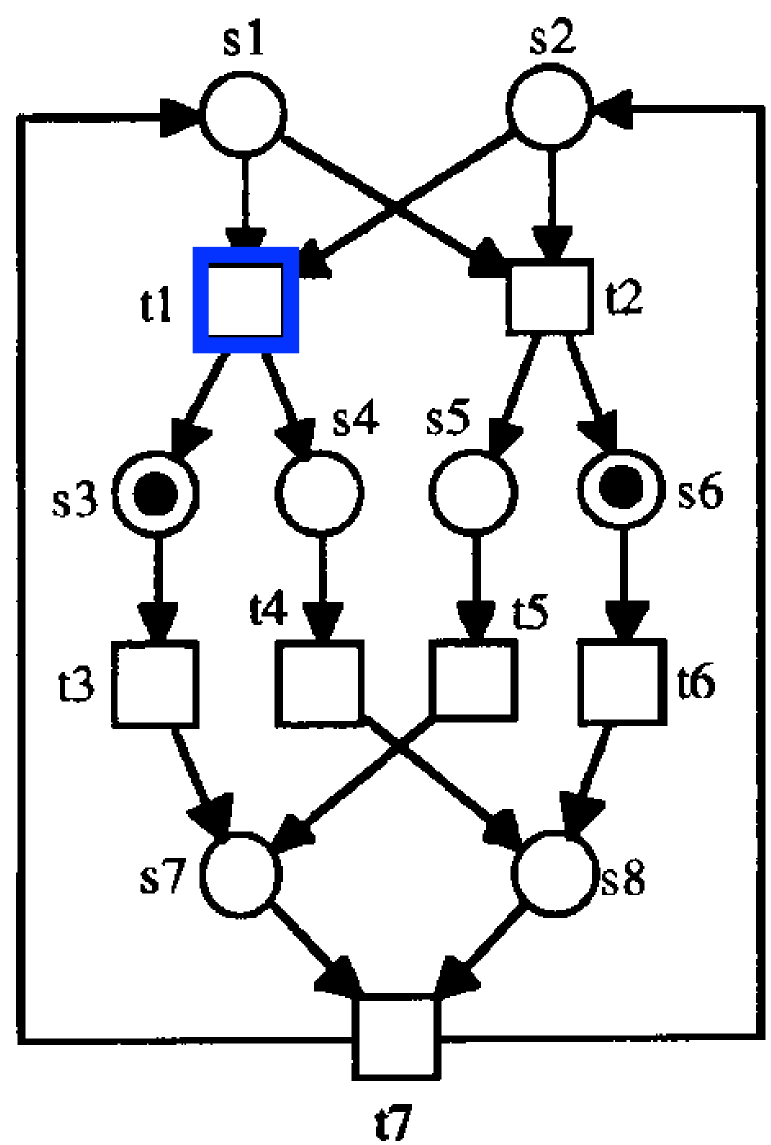
C is a **T-cover** if every transition t of N belongs to one or more T-components in **C**

We say that N is **covered by T-components** if it has a T-cover

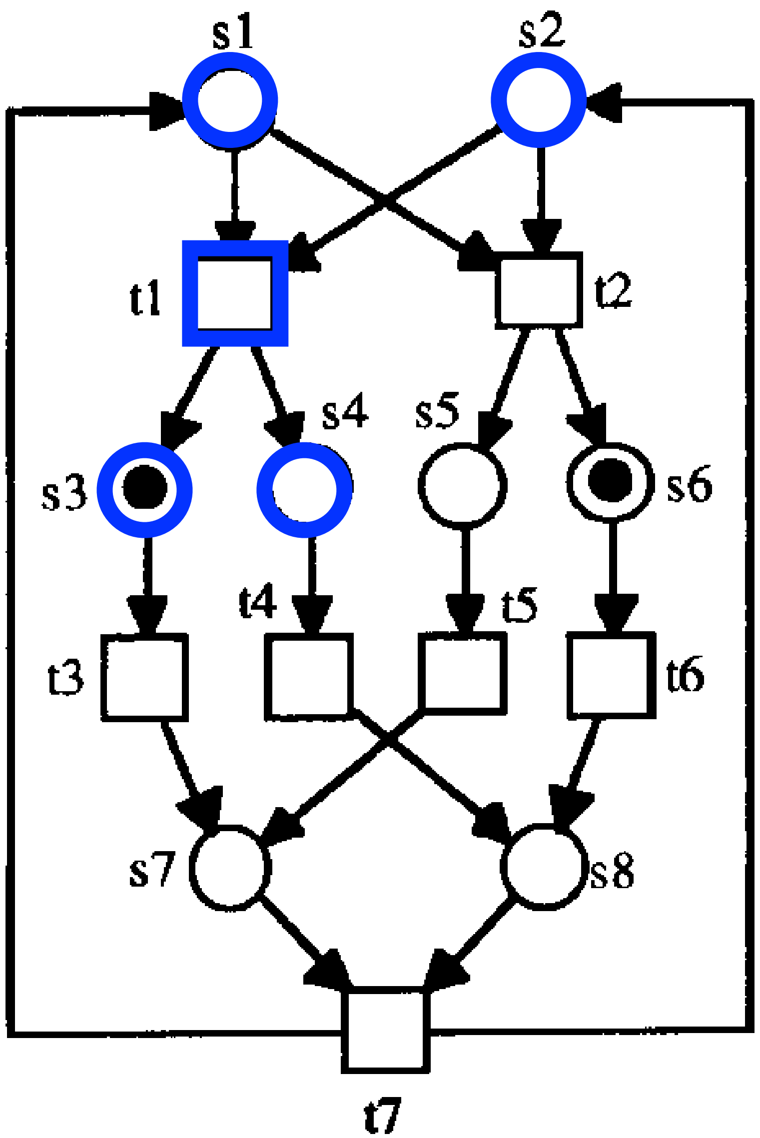
T-cover: example



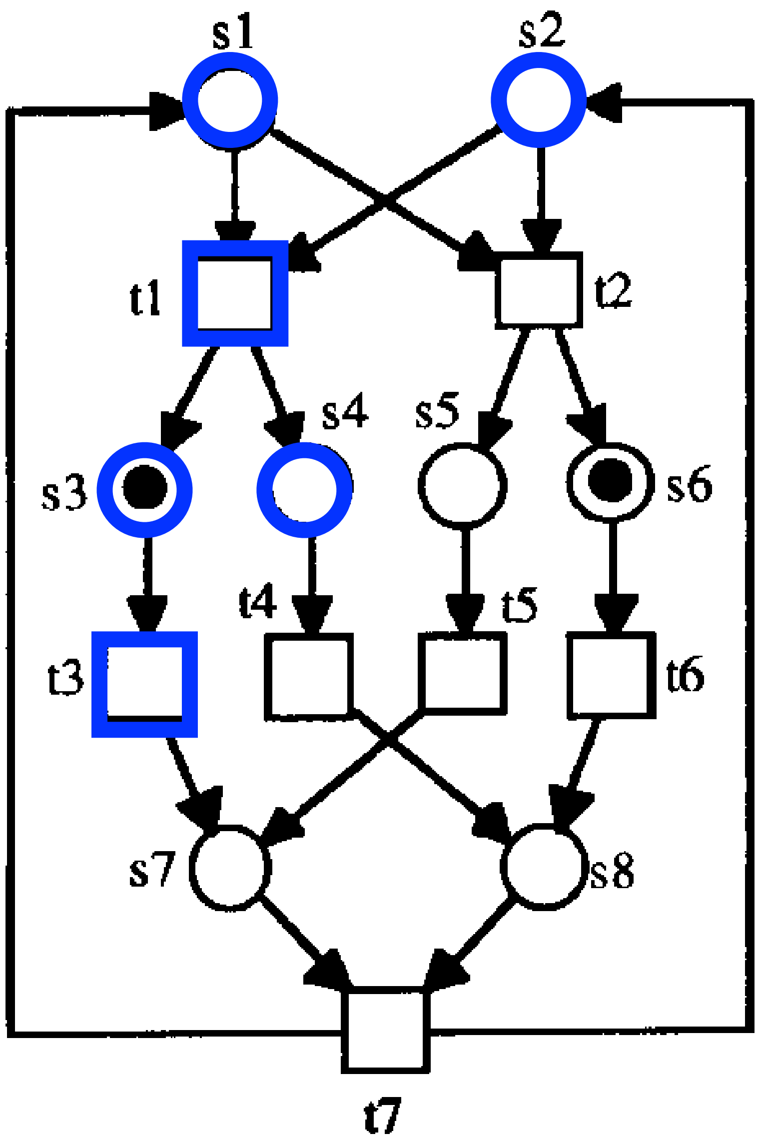
T-cover: example



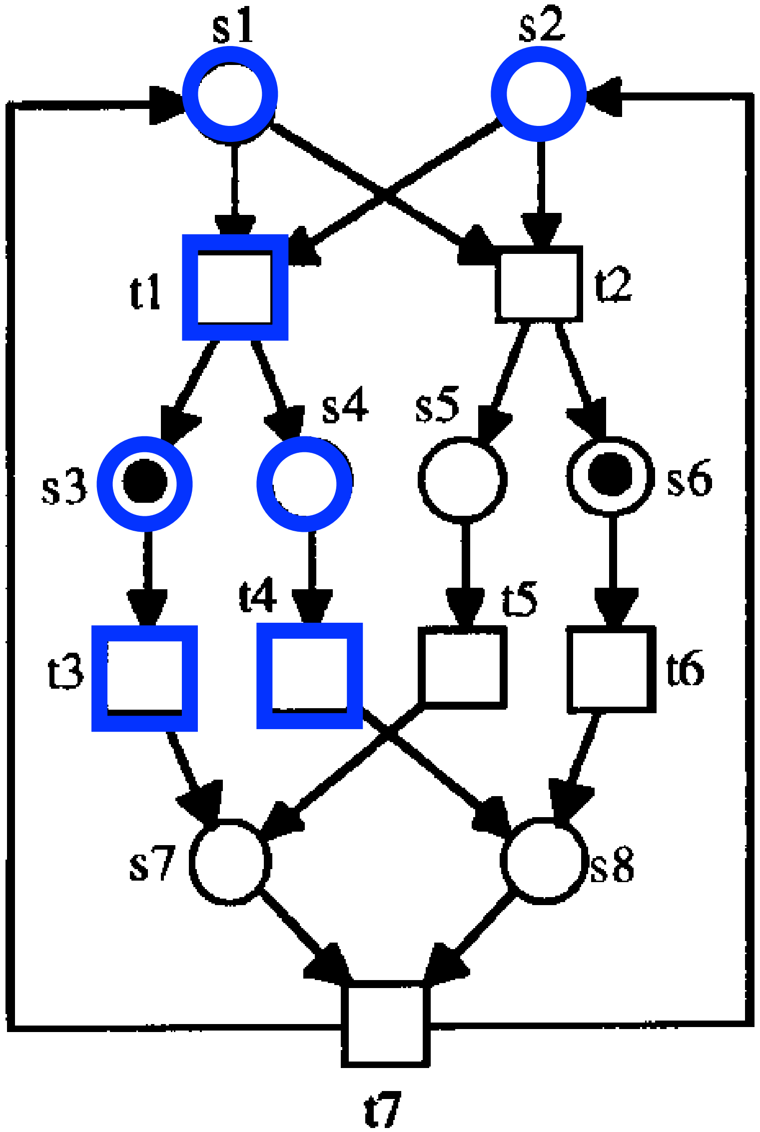
T-cover: example



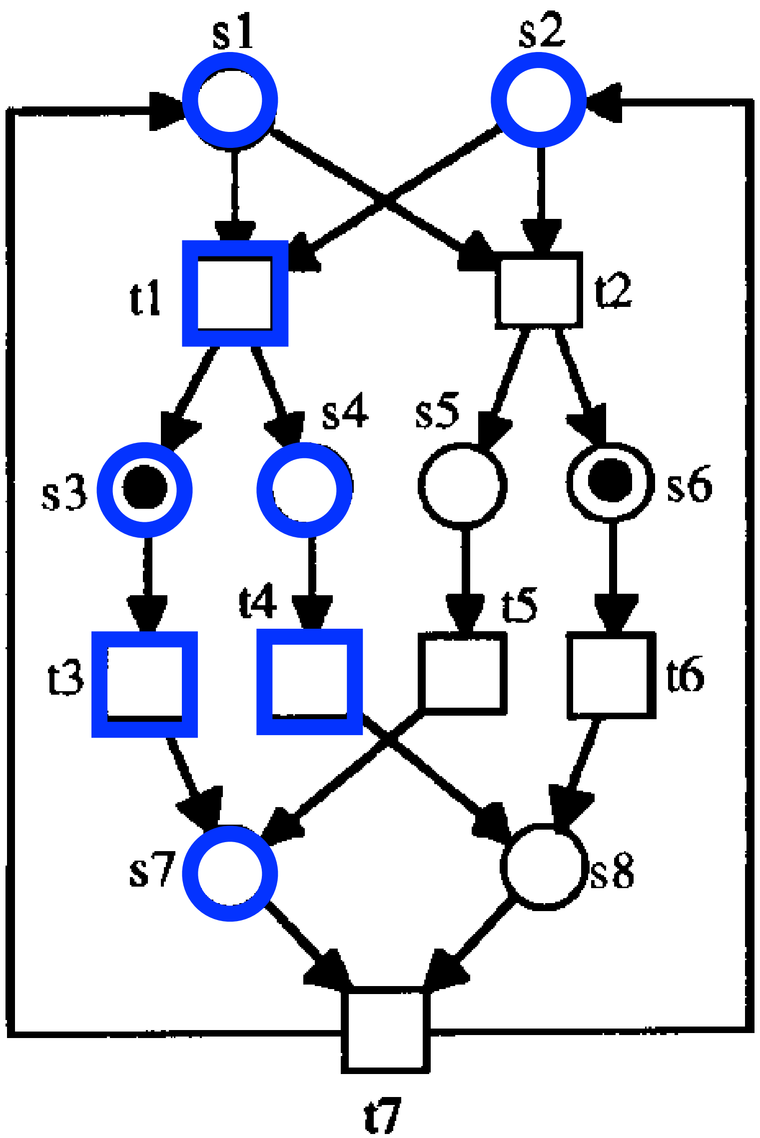
T-cover: example



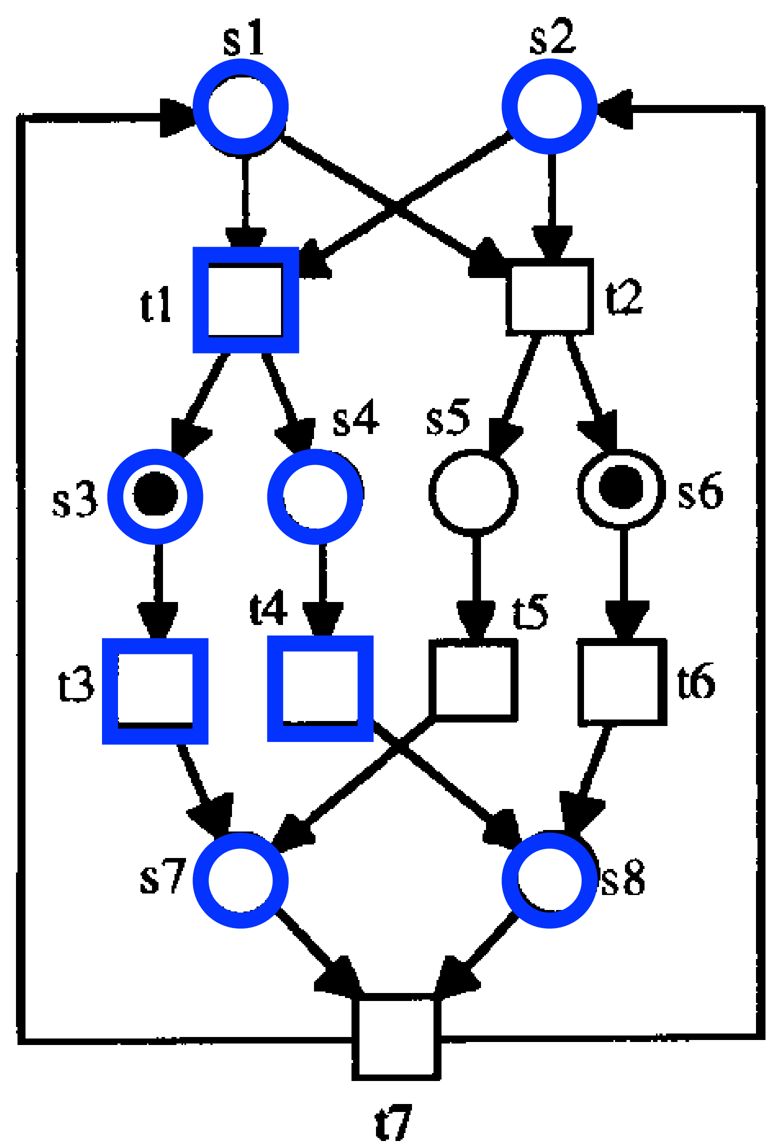
T-cover: example



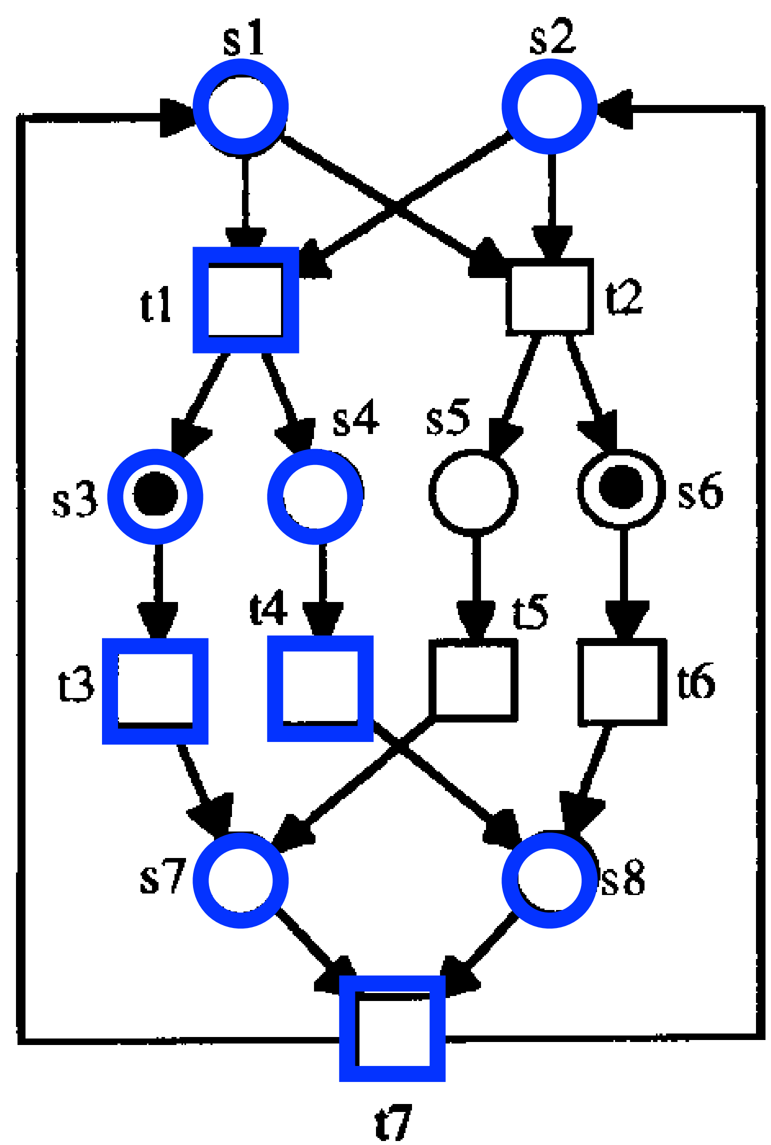
T-cover: example



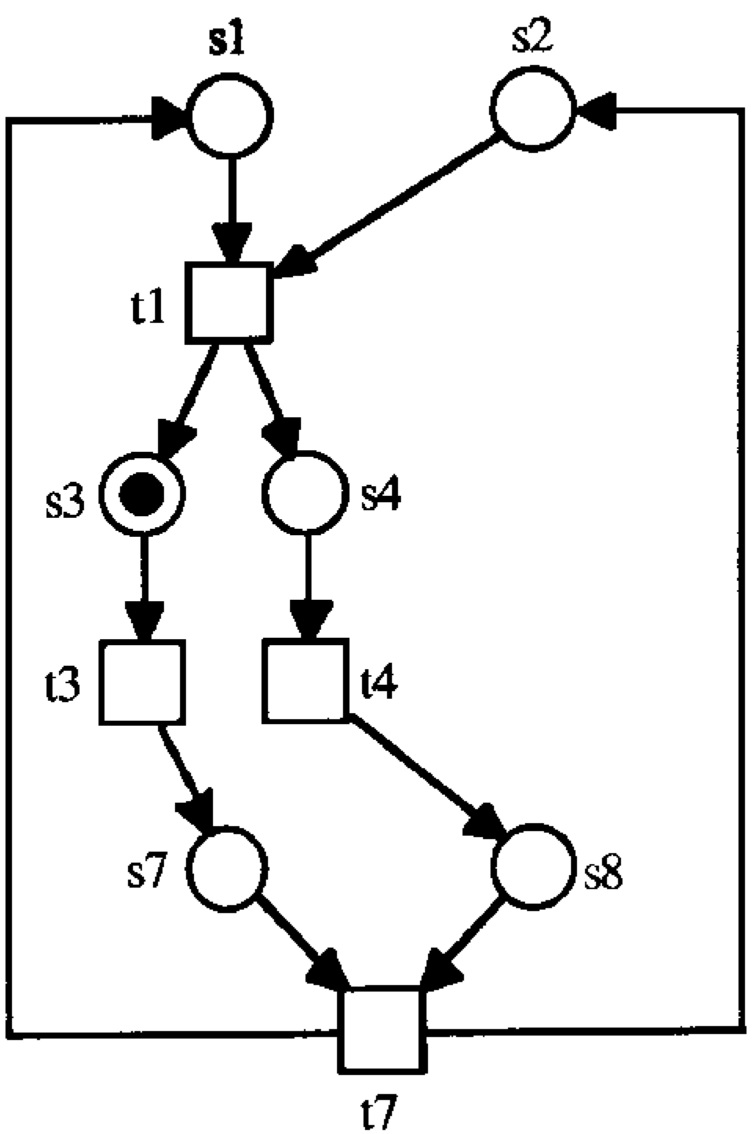
T-cover: example

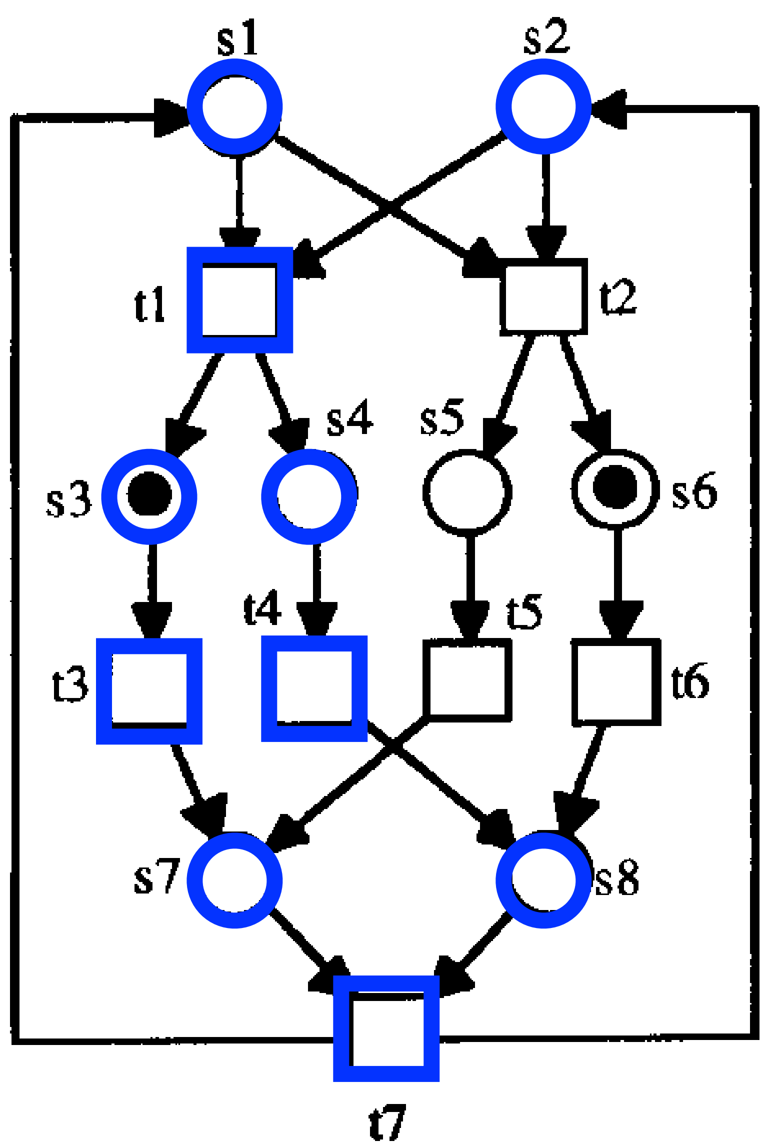


T-cover: example

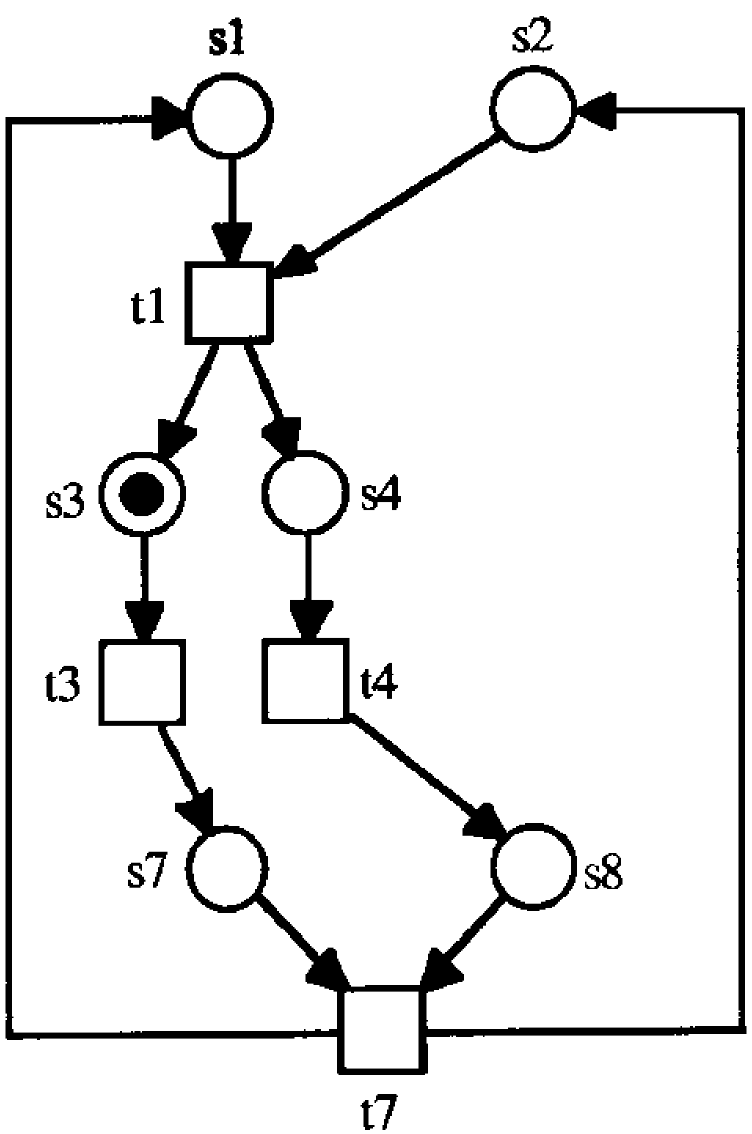


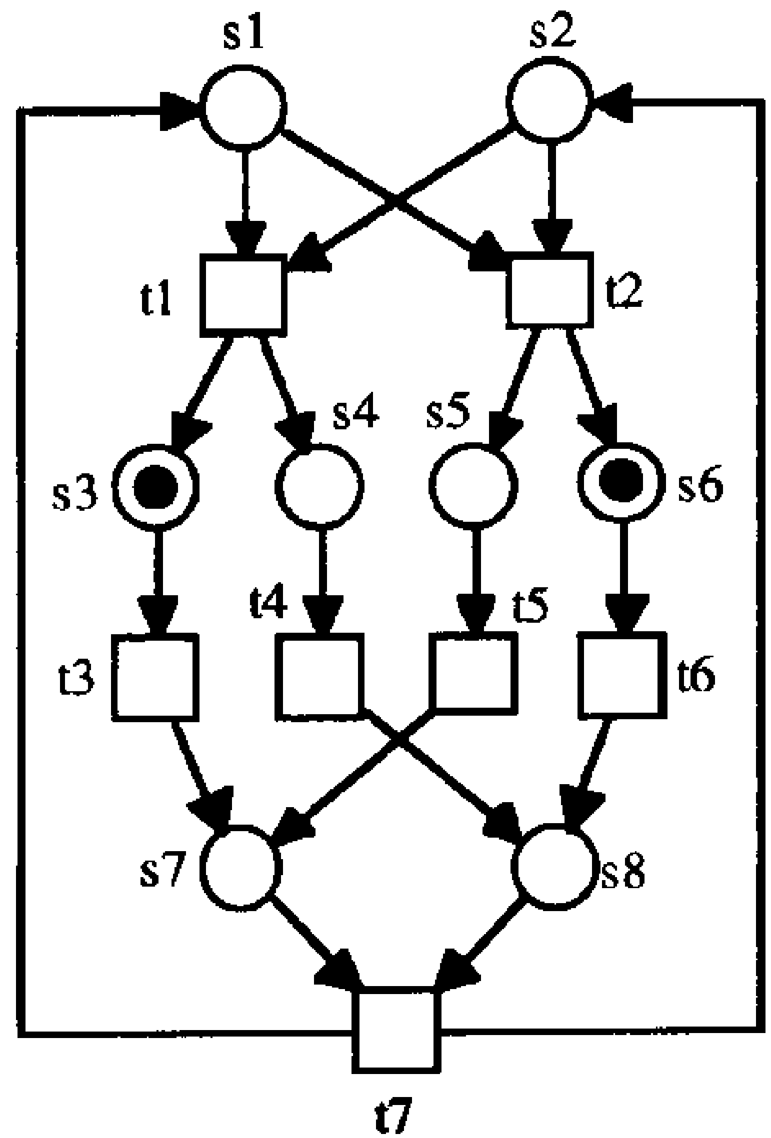
T-cover: example



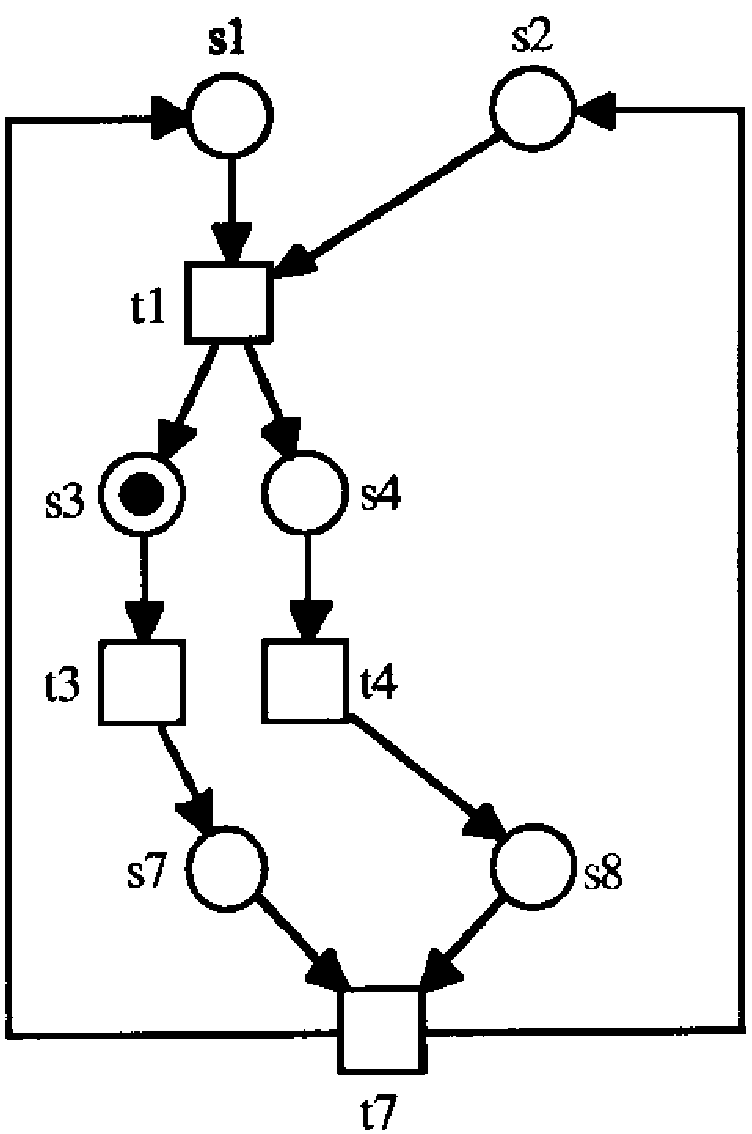


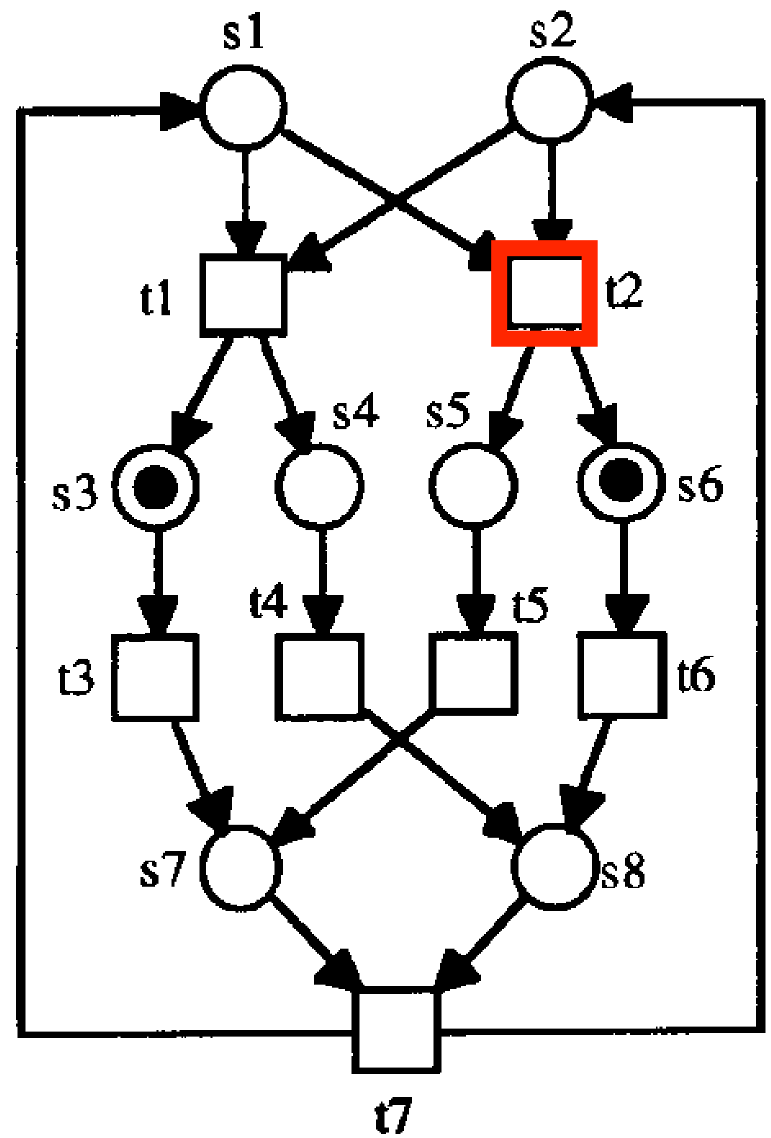
T-cover: example



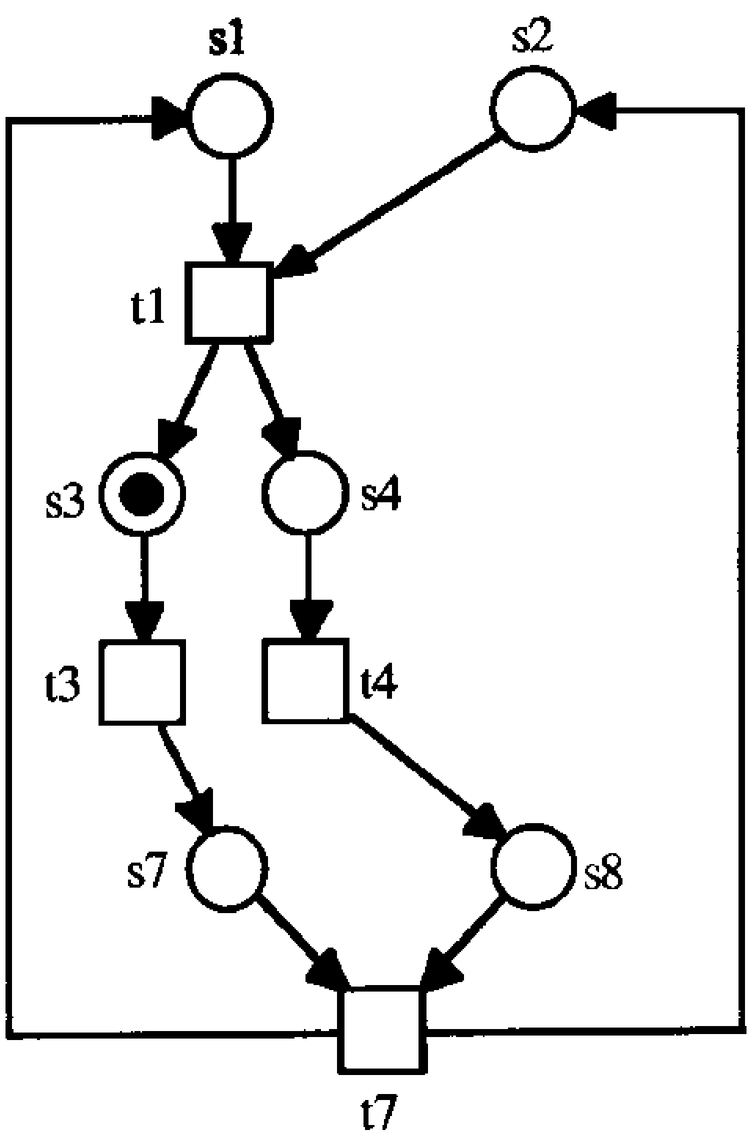


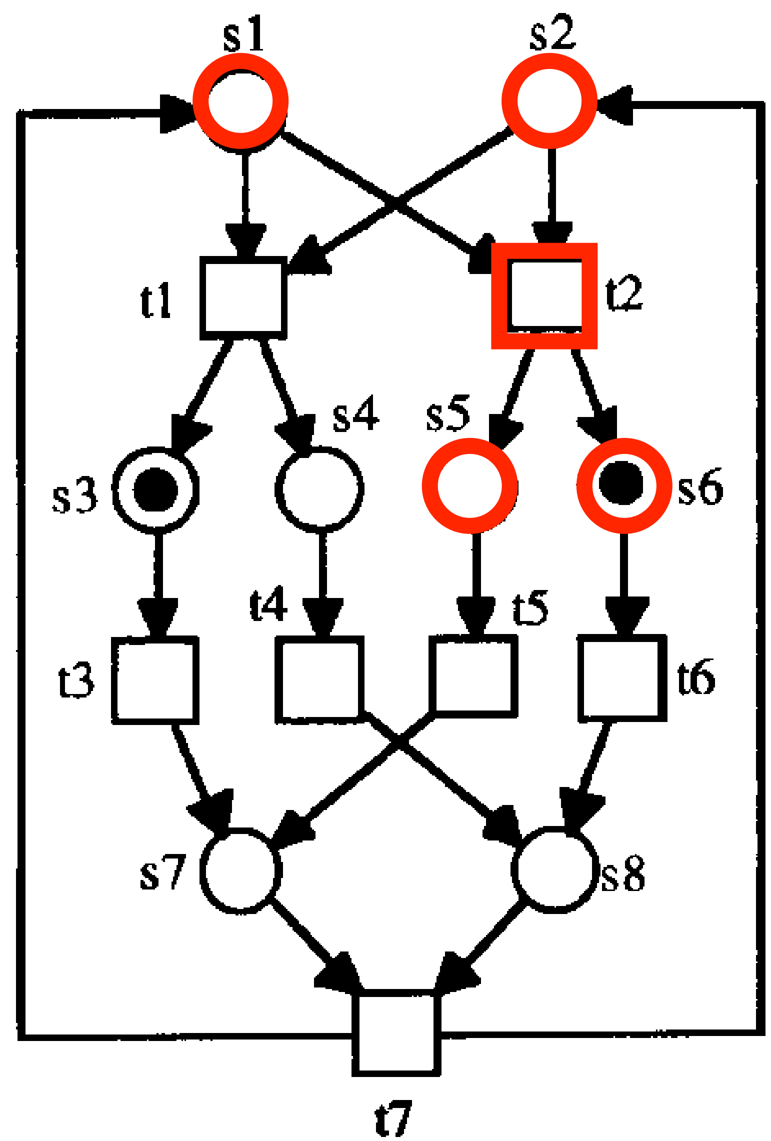
T-cover: example



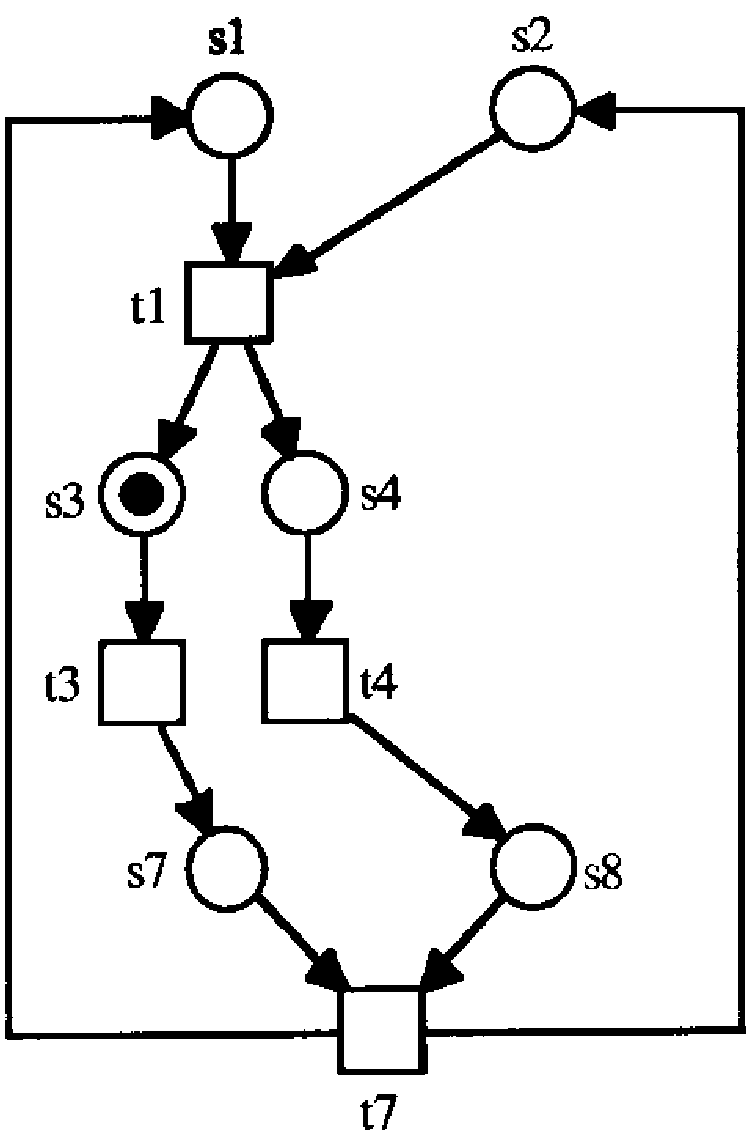


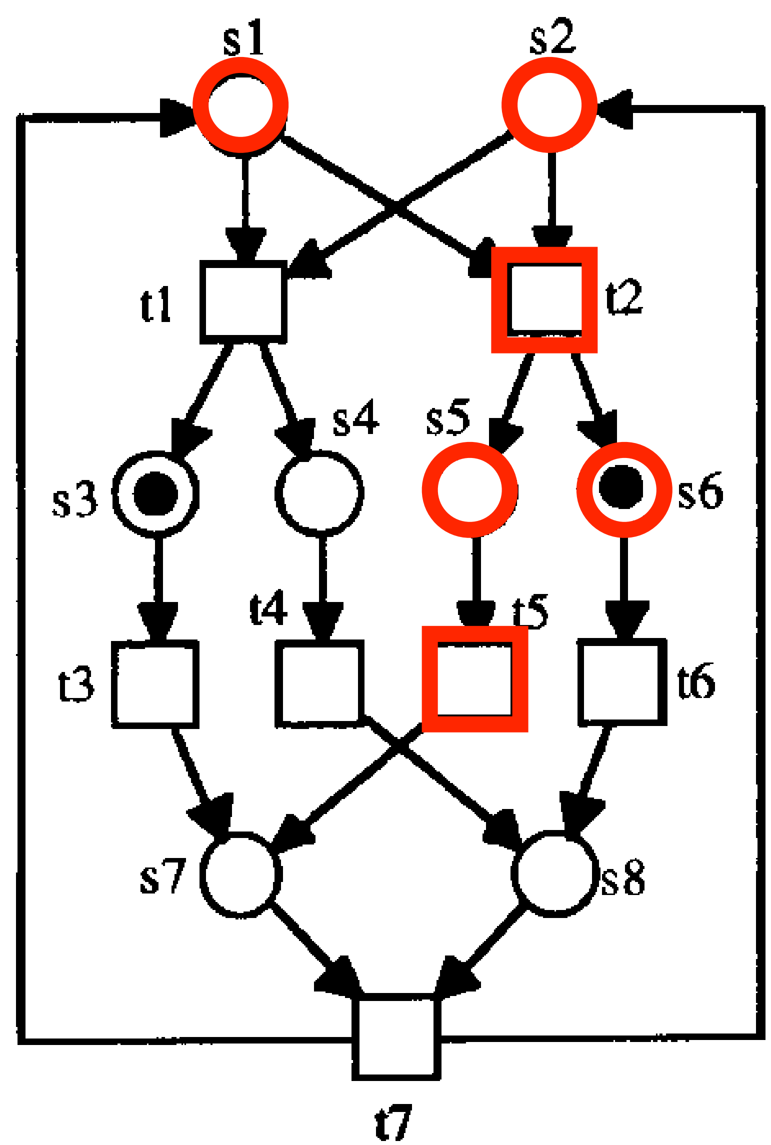
T-cover: example



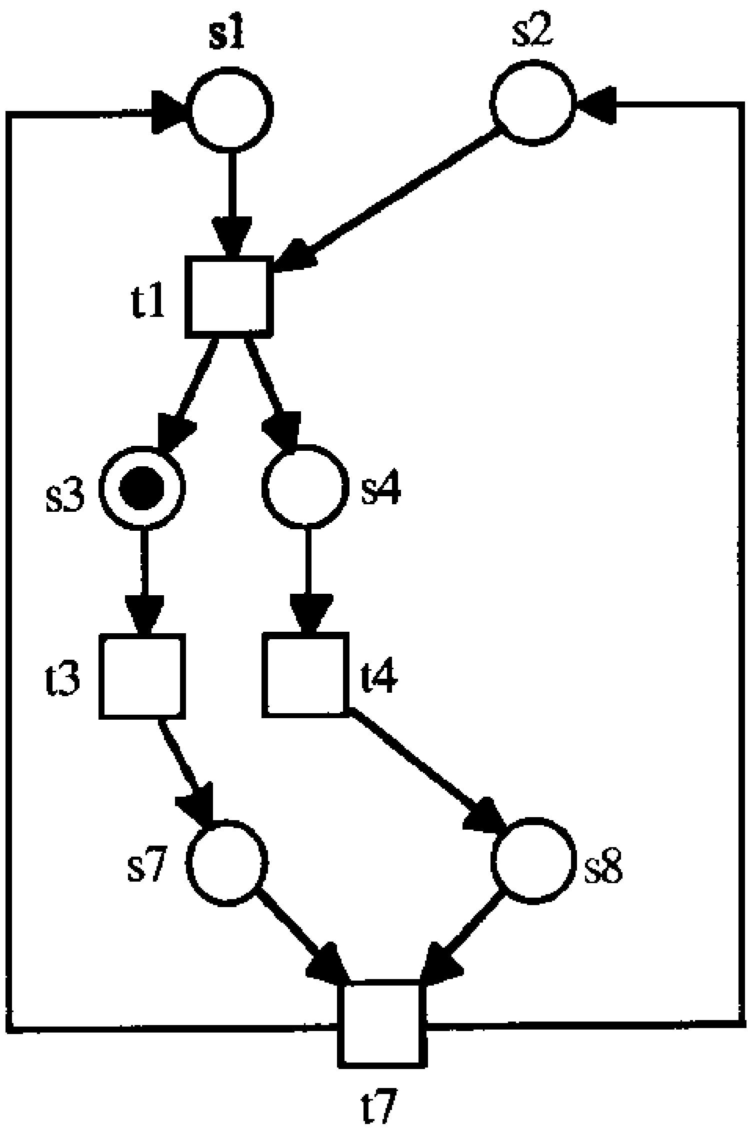


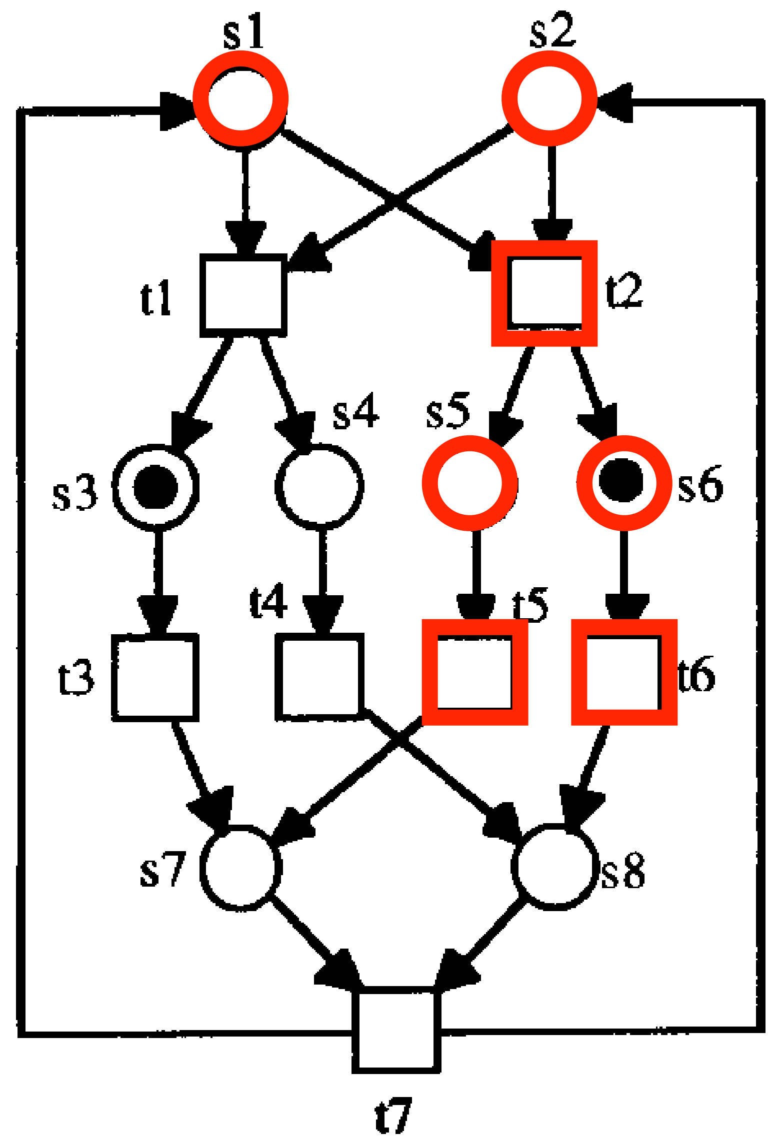
T-cover: example



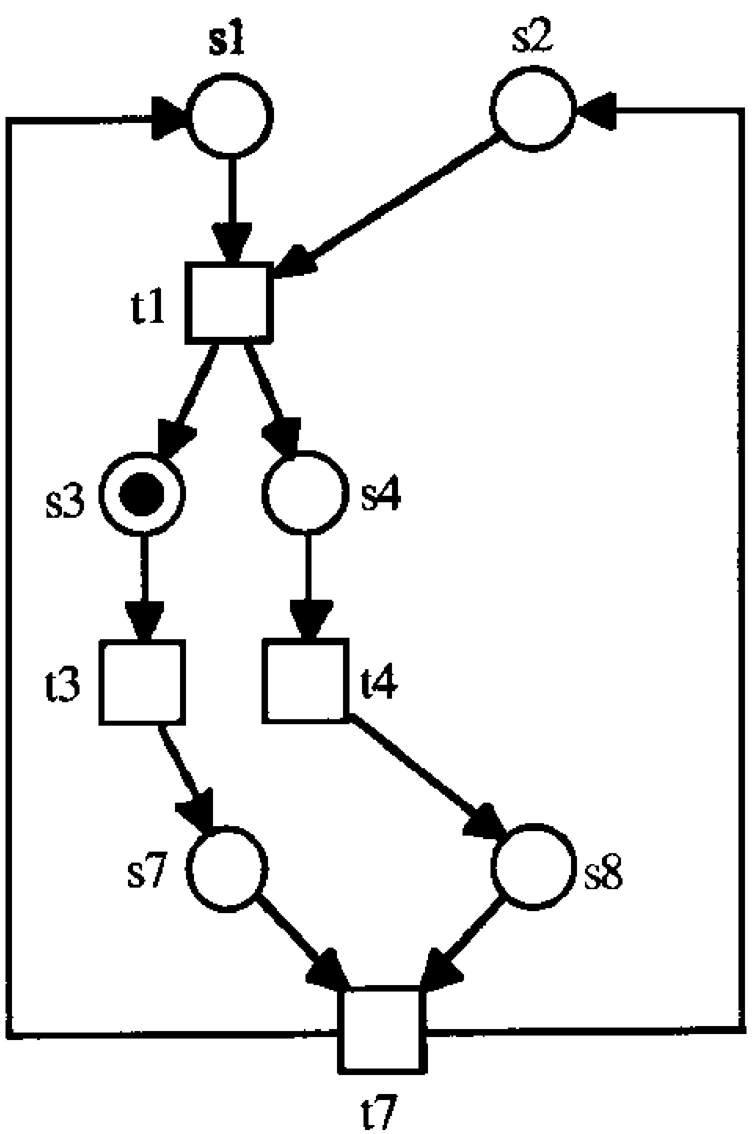


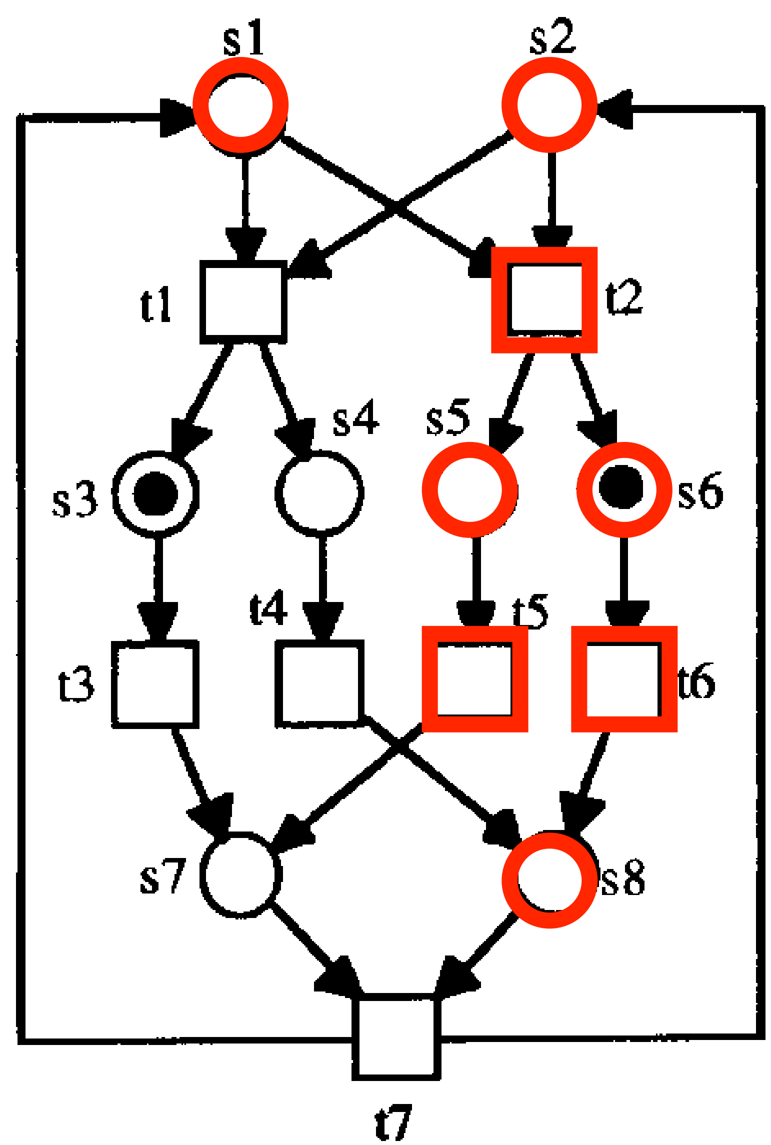
T-cover: example



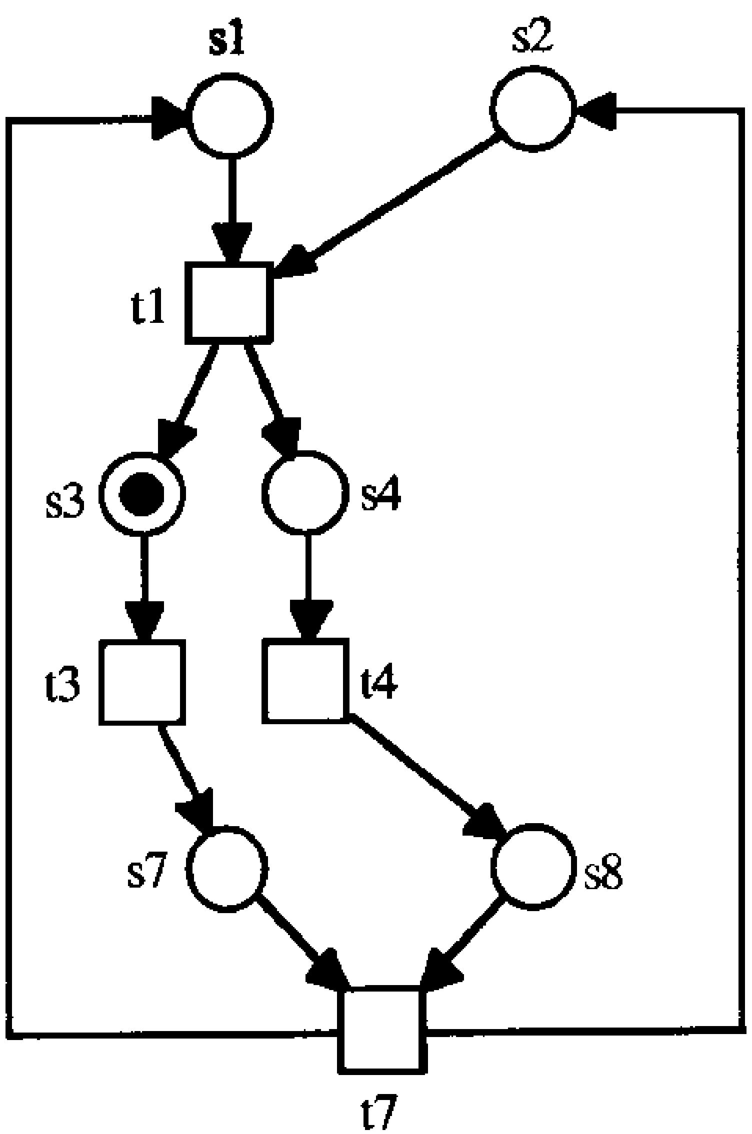


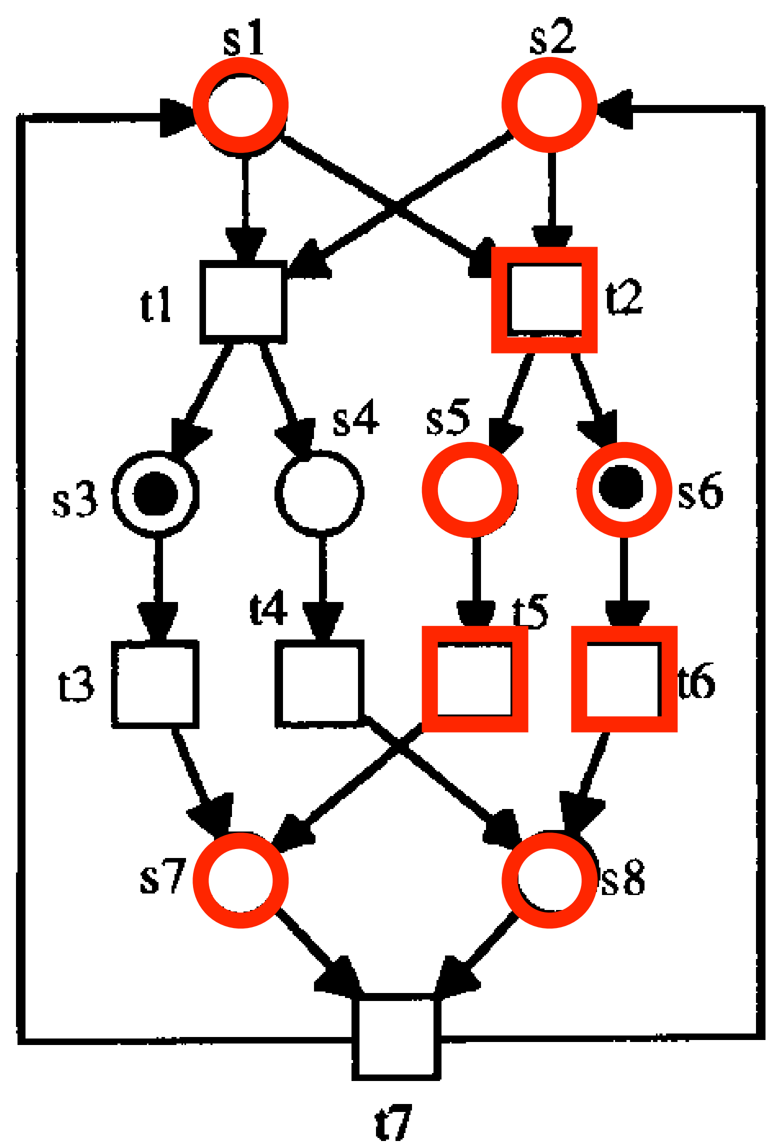
T-cover: example



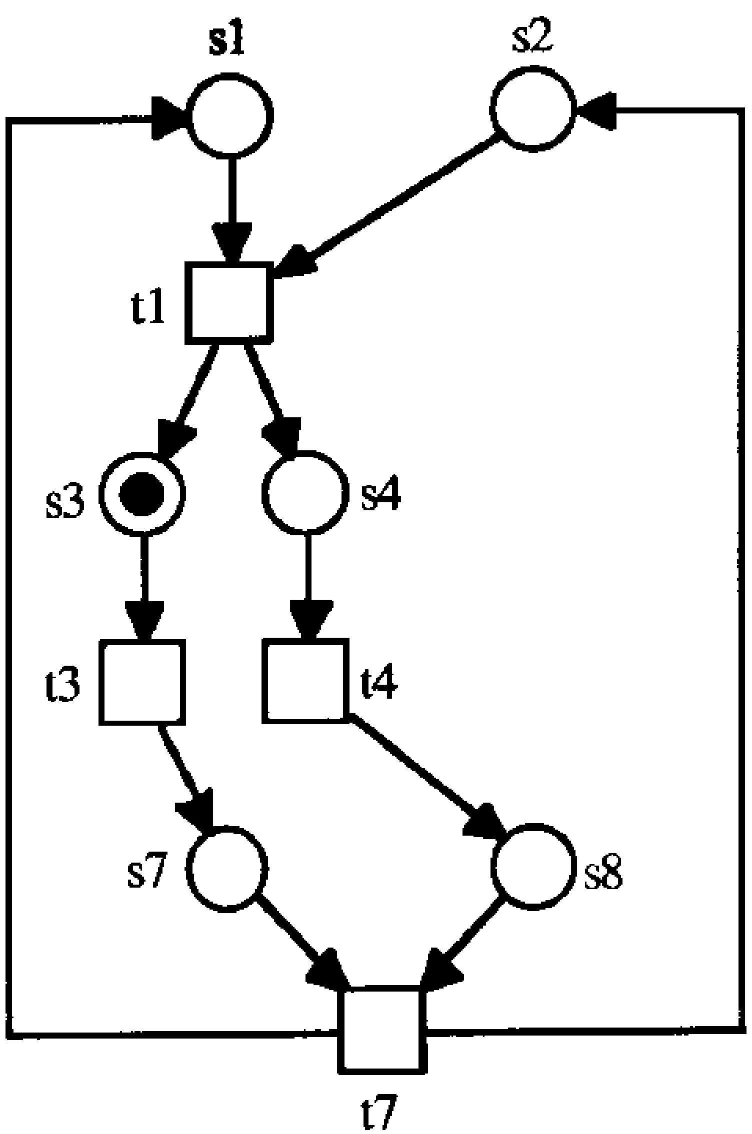


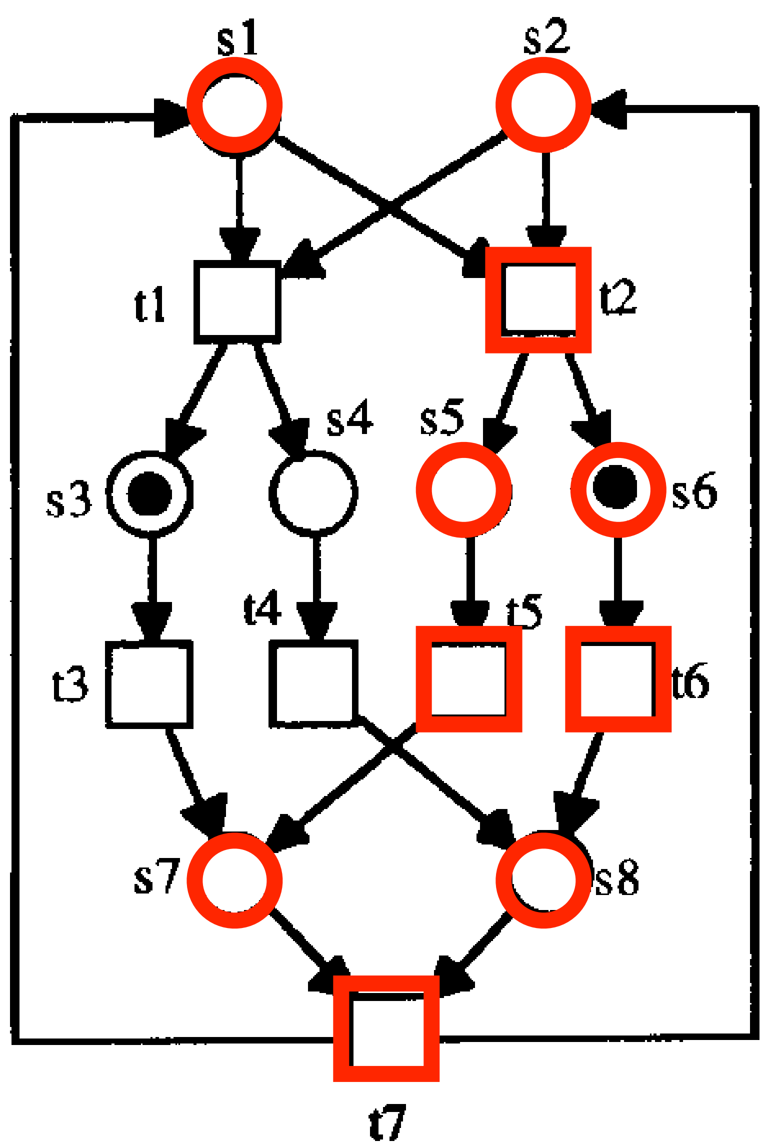
T-cover: example



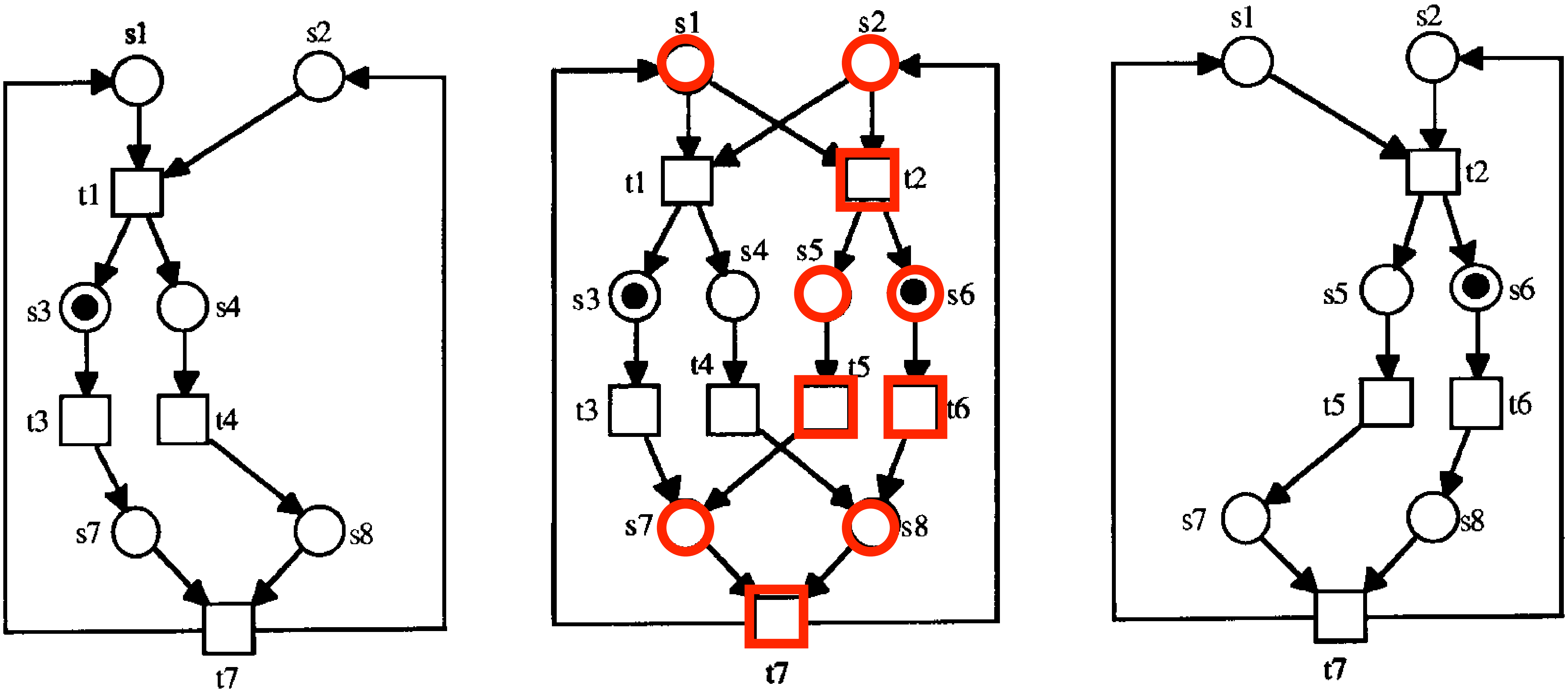


T-cover: example

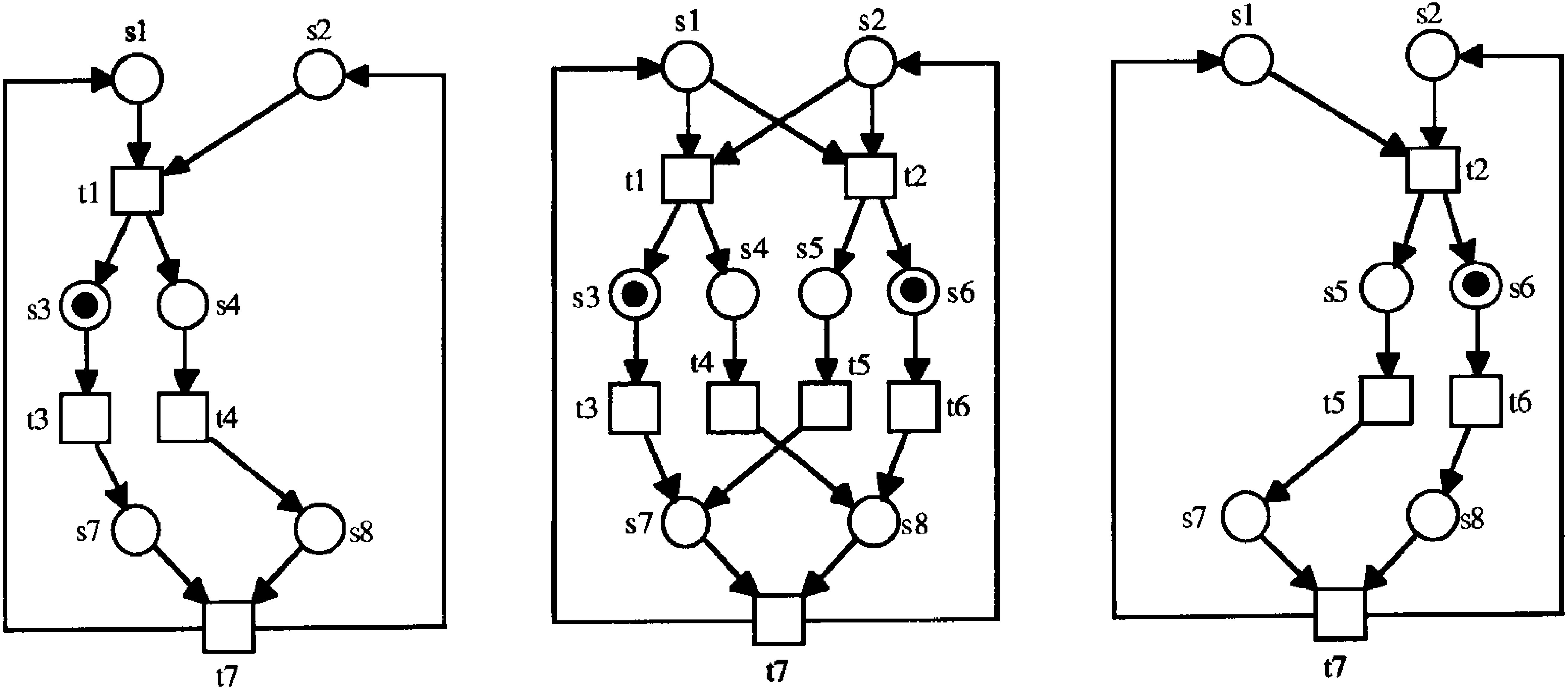




T-cover: example



T-cover: example



T-coverability theorem

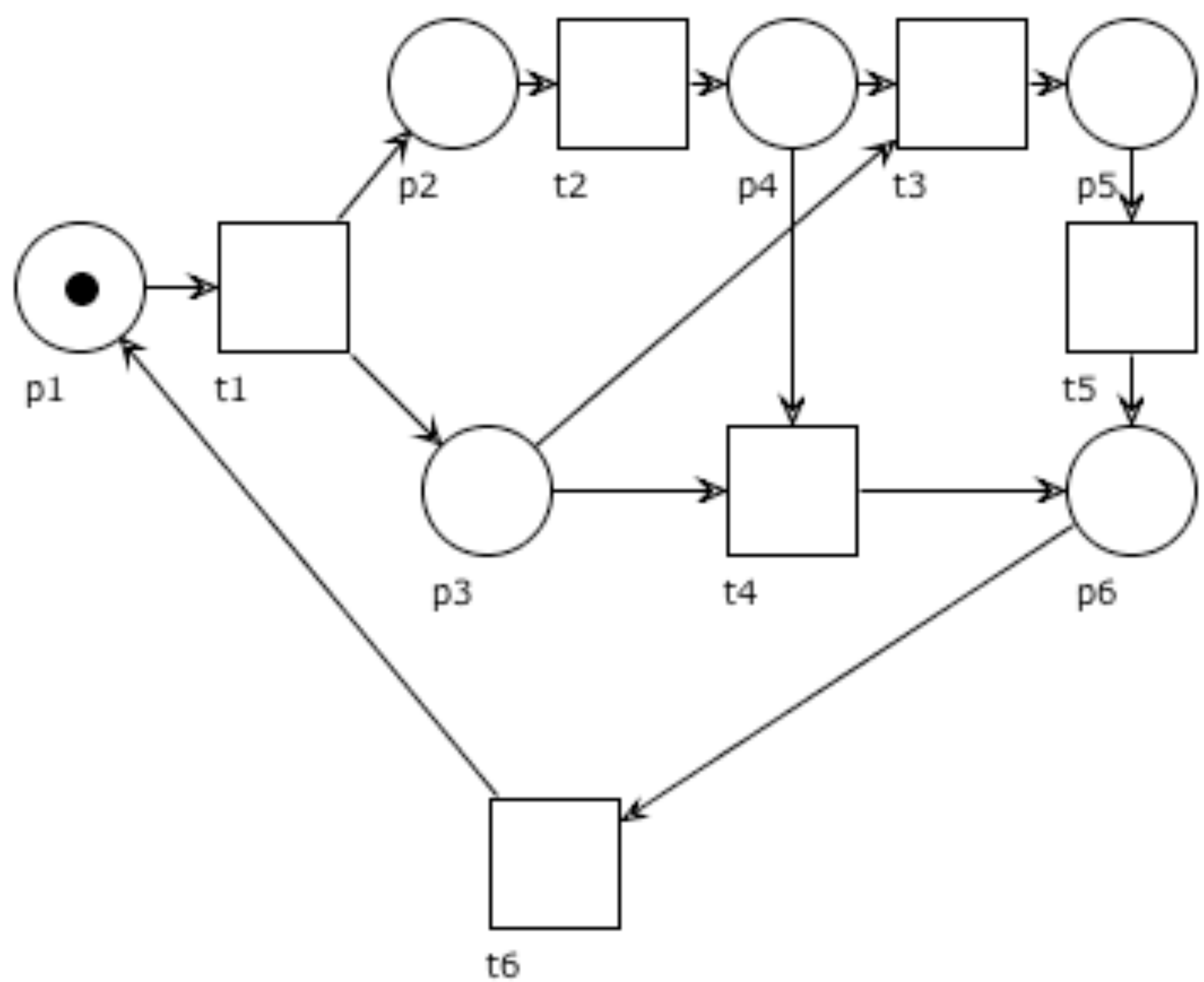
Theorem: If a free-choice net N is live and bounded then N is T-coverable

(proof omitted)

Consequence:
free-choice + not T-coverable => not (live and bounded)

Exercise

Find an S-cover and a T-cover for the net below and derive suitable S- and T-invariants



Compositionality

Compositionality of sound free-choice nets

Lemma:
If a free-choice workflow net N is sound then it is safe

(because N* is S-coverable and M_0=i has just one token)

Proposition:
If N and N' are sound free-choice workflow nets then N[N'/t] is a sound free-choice workflow net

(we just need to show that N[N'/t] is free-choice)