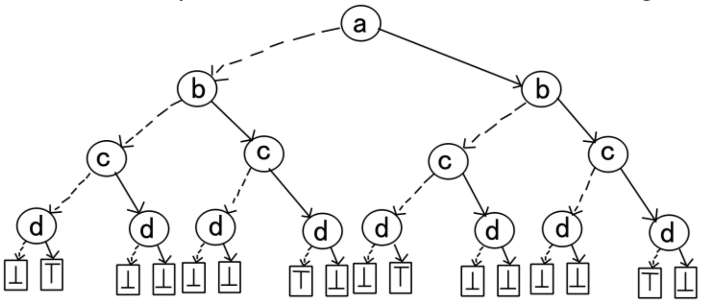
Q17 6.4

[redacted] Encoding and state space computation of Petri Nets
General algorithm/Operations

Définition formelle des SFDD :
Soit T un ensemble de termes. L'ensemble des SFDD S est défini inductivement :
$\perp \in S$: le terminal rejetant
$\top \in S$: le terminal acceptant
$\langle t, \tau, \sigma \rangle \in S \Leftrightarrow t \in T,\ \tau \in S,\ \sigma \in S$
– noeuds avec le terme t, sous-noeud acceptant τ (take node), sous-noeud rejetant σ (skip node)

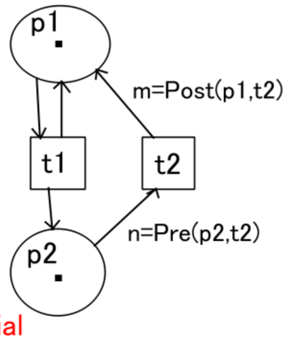
Exemple :
Termes : a<b<c<d
Ensembles :
{a,b,c}
{a,d}
{b,c}
{d}



on va représenter tout arc du petri netpar

But : faire des calculs sur l'ensemble (d'états) plutôt que sur chaque élément (état)

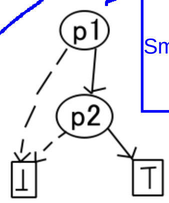
Encoder un réseau de Pétri :
Un réseau de Pétri est défini par <P,T,Pre,Post> où:
P,T sont des ensembles finis disjoints(place, transition)
Pre,Post : P x T->N
L'état d'un réseau de Pétri est le marquage M: P->N
$t \in T$ est tirable $\Leftrightarrow \forall p \in P$: $Pre(p,t) \leq M(p)$
Modification suite au tir :
$\forall p \in P$, $M'(p)=M(p)+Post(p,t)-Pre(p,t)$



Exemple:
(1,1)->t1->(1,2)
(1,1)->t2->(2,0)

But : calculer tous les marquages atteignables depuis un marquage initial
-> implémenter une notion de tirable et de tir dans l'encodage du réseau de Pétri

Encodage Réseau de Pétri safe : un seul jeton par place.
Encodage du marquage : $S_M = \cup\{p\}$ où $p \in P$ et $M(p)=1$ Ex: (1,0,1)-> S_M = {p1,p3}
Ordre total :p1<p2<...<pk
Encodage en SFDD: (1,1)={p1,p2}=

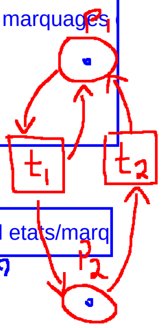


Sm est le set d etats qui ont

State space computation :
Algorithme global sur l'ensemble des marquages (pas sur des états individuels):
s_0 : état initial, ϕ : ensemble de transitions homomorphiques, Résultat : ensemble des états atteignables
begin
s, s_{old}, temp : set of states; , s=états atteignables
$s \leftarrow \{s_0\}$;
Repeat
$s_{old} \leftarrow s$;
$\forall t \in \phi$ do
temp<-t(s) (la transition est appliquée à tous les états)
$s \leftarrow s \cup temp$
until $s=s_{old}$ (point fixe)
return s;

on commence à l état initialon choisit un des marquages

ecxpliqué en bas



Transitions homomorphiques :

marquage m, suite a une transition t

appliquer une transition a un ensemble d etats/marq

Pour un marquage : t(m)=m+post(t)-pre(t)=post(t,pre(t,m)) étendu aux ensembles d'états :
$t(s \cup \{m\}) = t(s) \cup \{post(t,pre(t,m))\}$, s un ensemble d'état, m un état (un marquage)
$t(\emptyset)=\emptyset$, une transition non tirable est représentée par $\{\emptyset\}$
homomorphisme pour t :
$t=post(t) \circ pre(t)$ (enlever les jetons puis les ajouter)
$pre(t)=pre(t,p1) \circ \ldots \circ pre(t,pn)$
$post(t)=post(t,p1) \circ \ldots \circ post(t,pn)$
$pre(t,pi)=\ominus(pi) \circ filter(pi)$ si $Pre(t,pi)\neq 0$ (Si pi est dans la précondition de t)
(id) sinon filter : tirabilité , $\ominus$: suppression (de pi dans S_M)
$post(t,pi)=\oplus(pi)$ si $Post(t,pi)\neq 0$ (Si il y a une production de jeton)
(id) sinon $\oplus$: insertion

recalculer un marquage suite à unetra

Exemple :
$pre(t1,p1) = \ominus(p1) \circ filter(p1)$
$pre(t1,p2)$ = id
$pre(t2,p1)$ = id
$pre(t2,p2) = \ominus(p2) \circ filter(p2)$
$post(t1,p1) = \oplus(p1)$
$post(t1,p2) = \oplus(p2)$
$post(t2,p1) = \oplus(p1)$
$post(t2,p2)$ = id
-> t homomorphisme