Projet de Recherche

Résumé de l'article

Extension du Leaf Set

Projet de Recherche

TOULLALAN Antoine MENDAS Rosa

Lundi 15 Février (6e réunion)

Qu'est ce que ASCENT ?

L'article présente ASCENT, un outil qui s'ajoute aux DHT qui contienne potentiellement des noeuds mal intentionnés.
ASCENT fournit un certificat attestant qu'une transaction a bien été effectuée entre un client et un service dans un temps donné.
On a ainsi une garantie que les données certifiées contenues dans le DHT ne sont pas corrompues.

Description du système

ASCENT est mis en place dans un certain système comportant les éléments suivants :
Le premier élément de base est évidemment un **DHT** composée de noeuds. Chaque noeud est associé à une clé privée et une clé publique qui servent à les identifier notamment lors des transactions.
Le deuxième élément est un **système de réputation** qui identifie les noeuds "honnêtes" et les noeuds mal intentionnés (on utilisera le système WTR), ce système est associé à une fonction R(X)=Y ou X est un noeud et Y la probabilité que X soit un noeud de confiance.

Description du système

Le troisième élément est un **ensemble de noeuds de confiance**, qui permet à chaque noeud de confiance d'avoir une table contenant une partie des autres noeuds de confiance (pour cela on utilisera l'algotithme CORPS)->l'ensemble des noeuds de confiance sont organisés dans un anneau de confiance ainsi chaque noeud de confiance K est associé avec un noeud Kroot qui se situe dans l'anneau de confiance.

Fonctionnement de ASCENT

Si un noeud n'a pas de table de noeuds de confiance , on prend un noeud K et on demande à Kroot sa table d'anneaux de confiance, on regarde si cette table est bien valide (elle contient bien des noeuds de confiance) si la table n'est pas valide le noeud construit sa table de noeuds valide de zero.

Fonctionnement de ASCENT

Soit un service associé à un noeud , pour utiliser ce service càd effectuer une transaction avec ce noeud, le noeud A va d'abord envoyer un message "RequestInit" à l'ensemble des noeuds de confiance S de Sroot,(Sroot étant le noeuds le plus susceptible d'être associé au service). Si au moins la moitié des noeuds de S répondent favorablement, La transaction a lieu.
Lors de la transaction le noeud A signe chaque message avec sa clé privée et au moins la majorité des noeuds de S doit lui donner la même réponse sinon la transaction prend fin.

Fonctionnement de ASCENT

Après le "RequestInit" de A, les noeuds de S vont chercher à obtenir la certification de A qui se situe dans un autre ensemble C de noeuds. Lorsqu'un noeud de C recoit les notifications d'au moins la moitié des noeuds de S concernant la transaction, alors le noeud de C émet le certificat. Lorsque les certificats sont reçus, les noeuds S répondent favorablement et la transaction a lieu.
Il y a des algorithmes permettant de ne pas perdre de certificats de noeuds lors de départs de noeuds.

Réseaux de Petri Symétriques

- Stocker les configurations d'un système sans tenir compte des symétries
- Système complet trop complexe pour être formellement modélisé
- On s'intéresse à une seule transaction et donc à un acteur et son leafset : celui offrant le service, et celui traitant de la certification

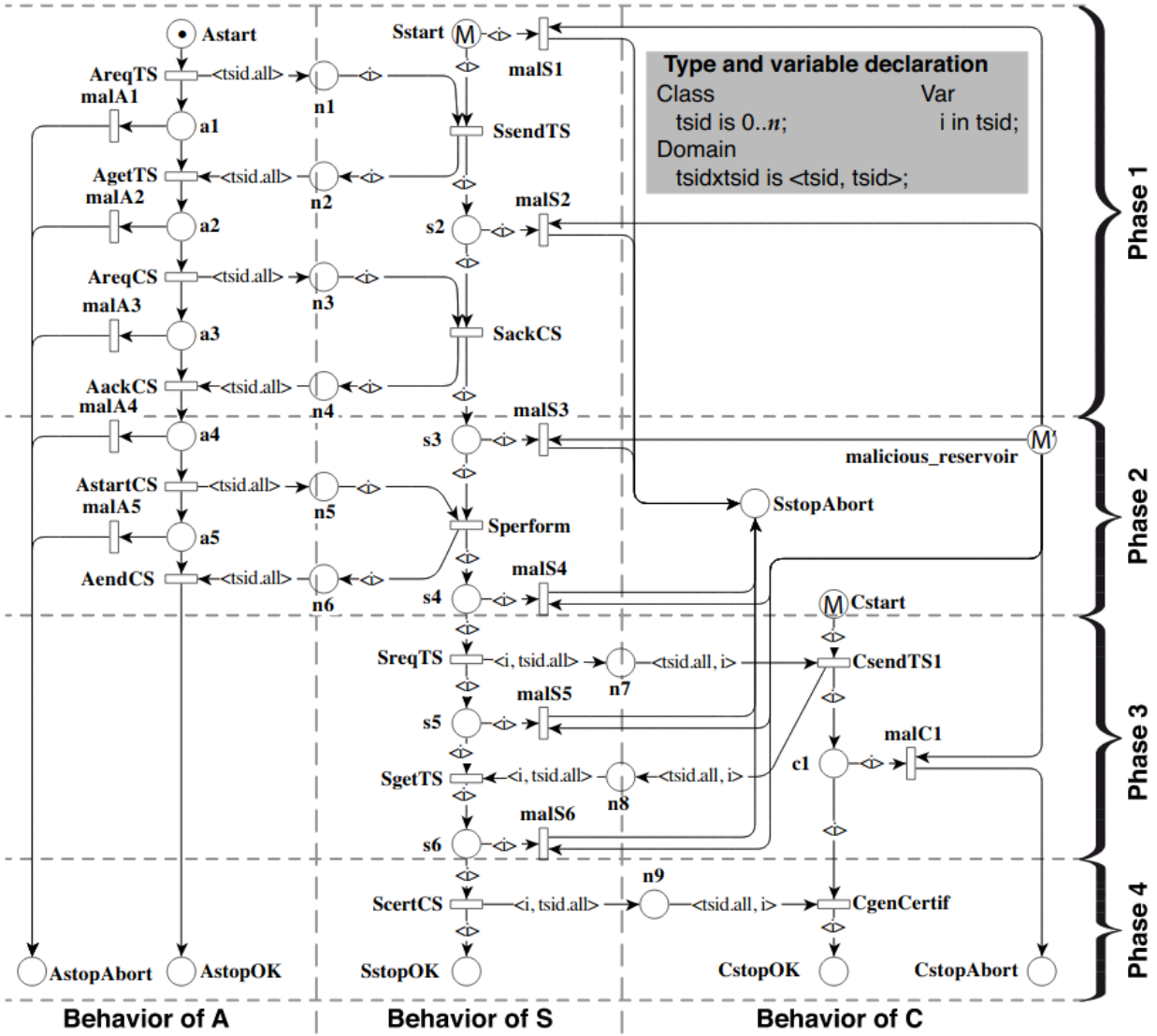
On émet pour hypothèses que :
L'algorithme DHT nous fournit un nombre constant de nœuds dans le leafset
Chaque acteur attend un nombre n de réponses au lieu des $|L|/2 + 1$ habituelles
Le service est basé sur une communication bidirectionnelle

Modèle formel

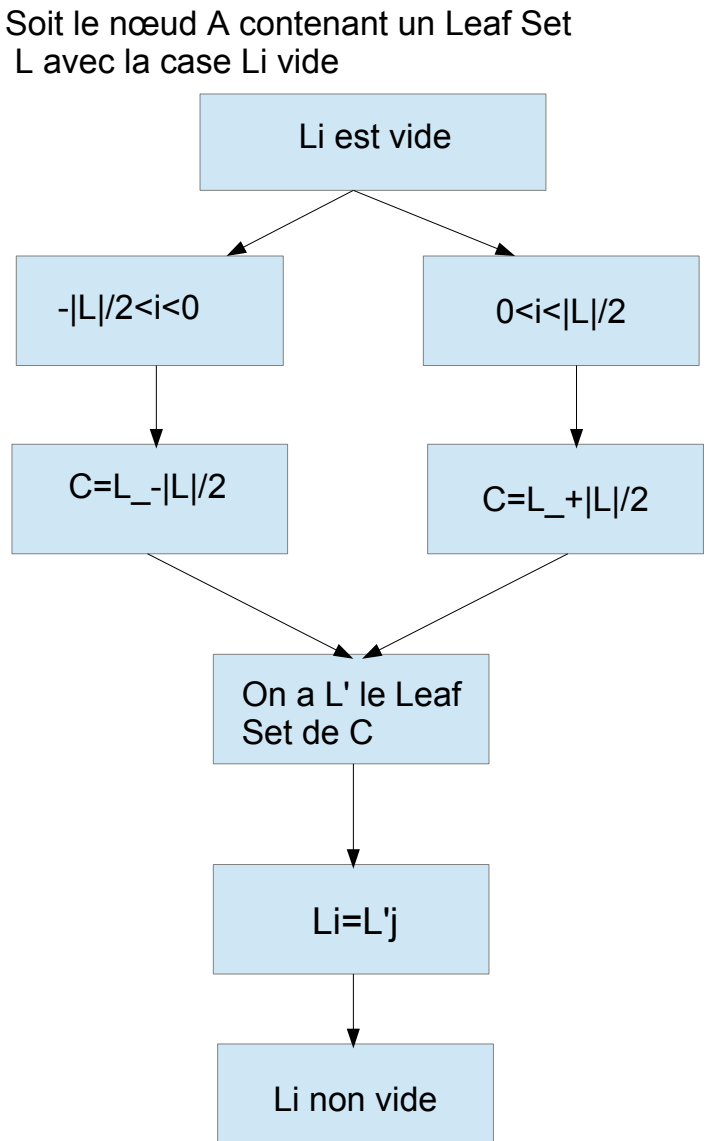
Projet de Recherche

Résumé de l'article

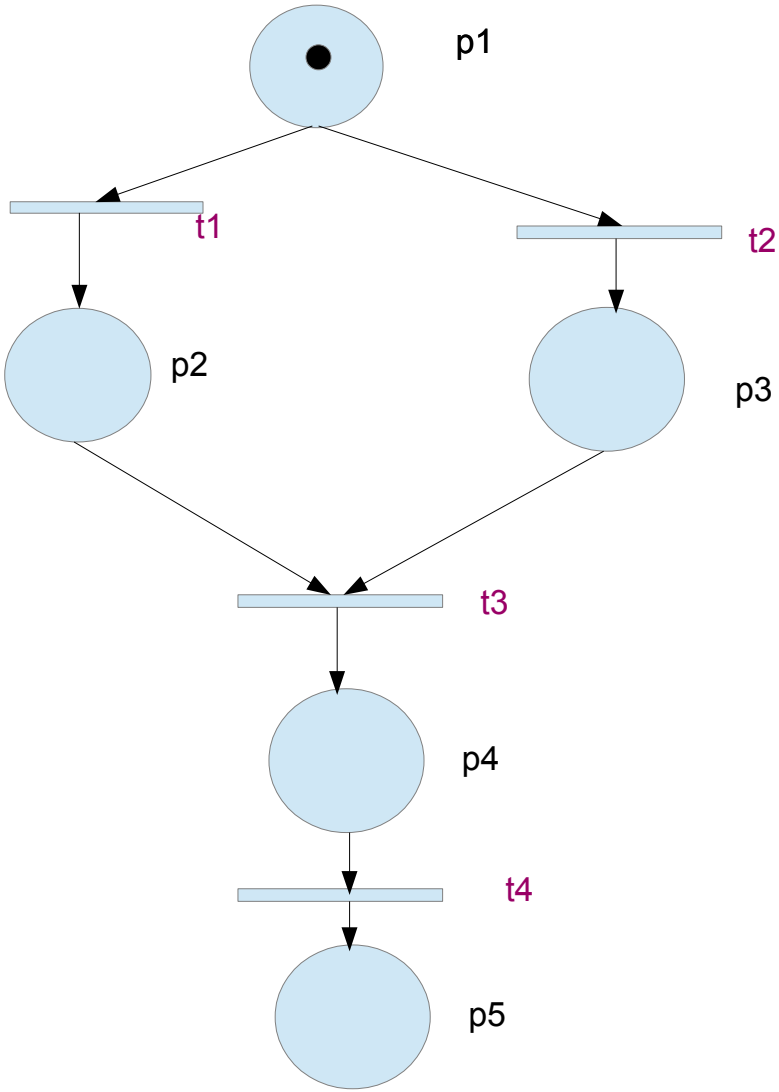
Extension du Leaf Set



extension du LeafSet



extension du LeafSet : Petri



p1 : Li est vide, C non choisit

p2 : C= $L_{-|L|/2}$

p3 : C= $L_{+|L|/2}$

p4 : L' est le Leafset de C

p5 : On assigne L'_j à Li, Li non vide

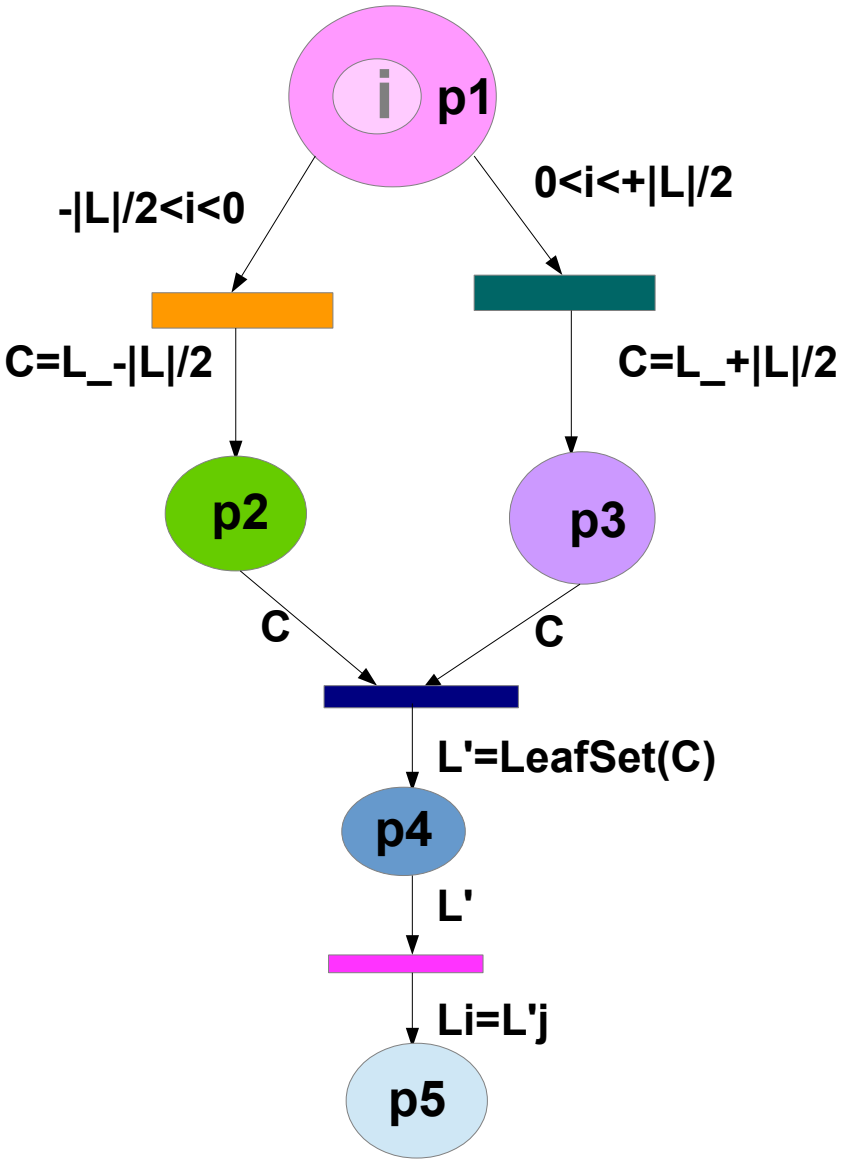
t1 : On a -|L|/2<i<0

t2 : On a 0<i<+|L|/2

t3 : C transmet son Leaf Set

t4 : On choisit la case L'_j de L' qui correspond le mieux à Li

extension du LeafSet : Petri coloré

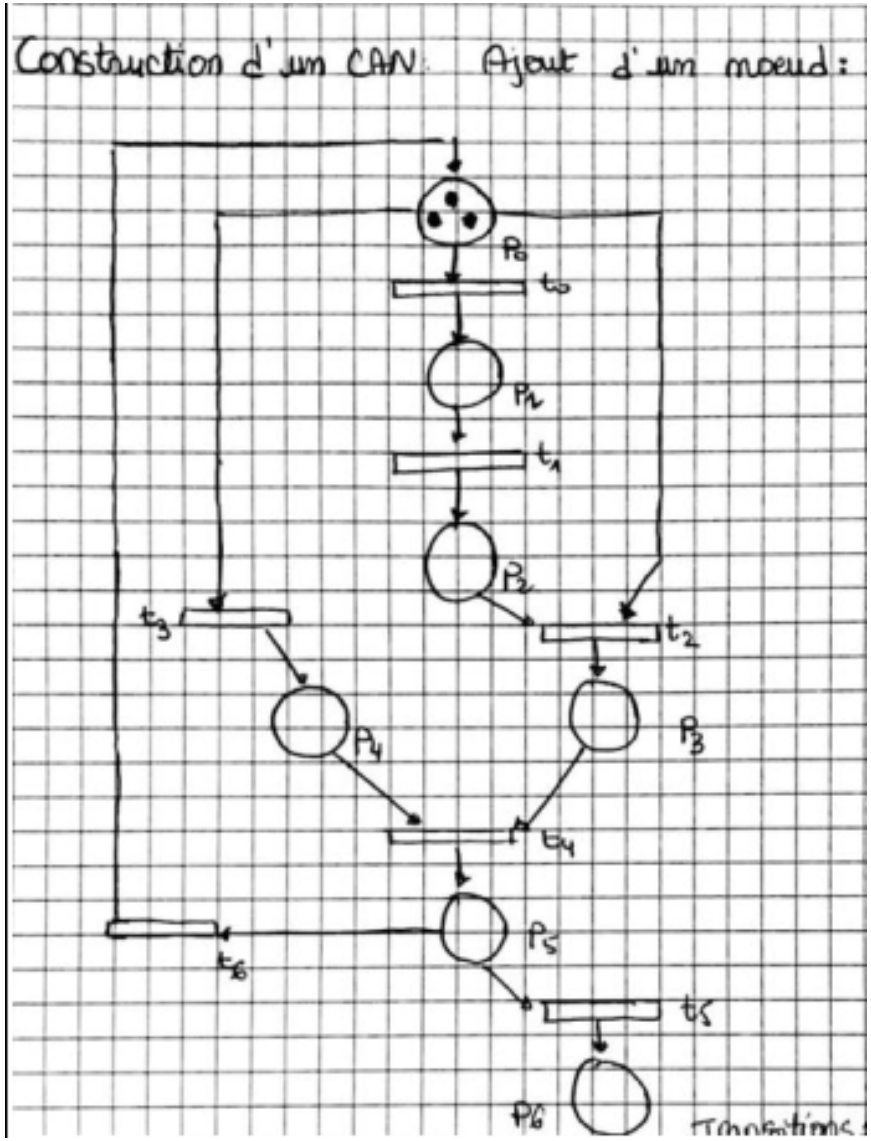


Présentation de la modélisation

Projet de Recherche

Résumé de l'article

Extension du Leaf Set



Présentation de la modélisation

P0 : Noeud à inserer
P1 : Noeud Bootstrap
P2 : Liste des noeuds actifs
P3 : Noeud aléatoire
P4 : Point aléatoire
P5 : Noeud responsable de la zone contenant P
P6 : Noeud dans CAN

T0 : Trouver frâce à DNS IP du Bootstrap c
T1 : c établi une liste des noeuds actifs du CAN
T2 : Noeud choisi aléatoirement pour le routage
T3 : Noeud à inserer choisi aléatoirement point P
T4 : Router un message JOIN vers le noeud x responsable de la zone contenant P
T5 : x divise sa zone et accorde la moitié
T6 : x refuse de diviser sa zone

Sources

- https ://hal.sorbonne-universite.fr/hal-01547514/document
- https ://pages.lip6.fr/Olivier.Marin/Publis/corps-CJ2011.pdf