HAL
open science

Réification des accélérations pour la construction de Karp et Miller

Alain Finkel, Serge Haddad, Igor Khmeniltsky

To cite this version:

Alain Finkel, Serge Haddad, Igor Khmeniltsky. Réification des accélérations pour la construction de Karp et Miller. Modélisation des Systèmes Réactifs (MSR 19), Nov 2019, Angers, France, Nov 2019, Angers, France. hal-02431913

HAL Id: hal-02431913

https://hal.science/hal-02431913v1

Submitted on 8 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Réification des accélérations pour la construction de Karp et Miller

Alain Finkel[1], Serge Haddad[12], and Igor Khmeniltsky[12]

[1] LSV, ENS Paris-Saclay, CNRS, Université Paris-Saclay, France
{finkel,haddad,khmelnitsky}@lsv.fr
[2] Inria, France

Résumé

L'algorithme de Karp et Miller est basé sur une exploration de l'arbre d'accessibilité d'un réseau de Petri où on accélère les séquences de transitions à incidence positive. Les noeuds de l'arbre de Karp et Miller sont étiquetés par un ensemble d'ω-marquages représentant l'ensemble (potentiellement infini) de couverture Cover. Cet ensemble d'ω-marquages permet de décider la couverture d'un marquage ou la finitude de l'ensemble d'accessibilité. Les arcs de l'arbre de Karp et Miller sont étiquetés par des transitions mais la sémantique de celles-ci n'est pas toujours la sémantique de base ce qui rend la preuve de l'algorithme relativement compliquée. Nous introduisons ici trois nouveaux concepts : l'abstraction, l'accélération et la séquence d'exploration. En particulier, nous généralisons la définition des transitions aux ω-transitions afin de pouvoir représenter une accélération par une ω-transition. La notion d'abstraction permet de simplifier grandement la preuve de correction de l'algorithme de Karp et Miller. D'autre part, au prix d'un surcoût minime en mémoire, évalué théoriquement, nous proposons une variante « accélérée » de l'algorithme de Karp et Miller avec un gain escompté en temps d'exécution et dont la preuve de correction se déduit très simplement de notre preuve de l'algorithme de Karp et Miller.

1 Introduction

La couverture et l'algorithme de Karp et Miller. L'ensemble de couverture (noté aussi couverture et *Cover* en anglais) d'un réseau de Petri muni d'un marquage initial est la clôture inférieure (pour l'ordre habituel sur les vecteurs d'entiers) de l'ensemble des marquages accessibles. Une représentation finie effective de la couverture permet de décider plusieurs problèmes parmi lesquels : Un marquage donné peut-il être couvert par un marquage accessible (le problème de la couverture) ? L'ensemble des marquages accessibles est-il fini ? Quelles sont les places non bornées ?

Karp et Miller ont montré en 1969 qu'une représentation finie de la couverture dans les réseaux de Petri et les systèmes d'addition de vecteurs était calculable par un algorithme construisant un arbre fini (KMT) [9] dont l'ensemble fini C des étiquettes des noeuds (des ω-marquages) représente la couverture. Plus précisément, la clôture inférieure (dans $\mathbb{N}^P$) de C, notée $\downarrow C$, coincide avec la couverture. Cet ensemble C n'est pas unique car il dépend de l'ordre choisi pour explorer les successeurs des noeuds de l'arbre. De plus, il n'est pas minimal en le nombre d'éléments car il peut contenir des ω-marquages comparables donc redondants.

La preuve originale de l'algorithme de Karp et Miller est incomplète ainsi que Hack l'avait déjà remarqué en 1974 [8]. De plus les preuves des variantes de l'algorithme de Karp et Miller (voir plus bas) sont difficiles et ne réutilisent pas la preuve de Karp et Miller. Motivés par l'absence de preuve complète et certifiée, Yamamoto et al ont écrit une preuve formalisée en COQ de la correction de l'algorithme de Karp et Miller [15] mais pas de ces variantes.

Clover ou la représentation canonique de la couverture. Il est possible d'associer à tout réseau de Petri marqué une représentation finie et *canonique* de la couverture. En effet,

tout ensemble clos inférieurement dans $\mathbb{N}^P$ est égal à la clôture inférieure (dans $\mathbb{N}^P$) d'un sous-ensemble fini d'ω-marquages incomparables dans $\mathbb{N}_\omega^P$. Ainsi on peut associer à tout réseau de Petri marqué une *unique* représentation finie de sa couverture [3]. Cette approche se généralise aux systèmes de transitions monotones munis d'un belordre (c'est-à-dire aux systèmes bien structurés) [2] et même aux systèmes de transitions monotones munis d'un ordre sans antichaîne infinie [1]. Cette représentation finie est appelée *Clover* (pour *Clo*sure of the C*over*) dans [4]. Elle est minimale, elle est constituée d'éléments maximaux (ces éléments représentent des ensembles clos inférieurement et dirigés, appelés idéaux) et elle est unique. Clover se calcule à partir de C en conservant uniquement les éléments maximaux. Une fois Clover calculé, on peut répondre à plusieurs questions de couverture sans réappliquer à chaque fois un algorithme de couverture car il suffit de comparer le marquage qu'on veut couvrir avec les marquages de Clover, ce qui prend un temps proportionnel à la taille de Clover et non plus nécessairement doublement exponentiel. Clover permet aussi de décider des questions plus générales que les problèmes de couverture et de borne des places. Illustrons ce point avec la question de ω-couverture suivante (qui représente une infinité de questions de couverture) : le marquage $(n, 2, 5, n)$ est-il couvrable pour tout $n \geq 0$? Cette propriété est vérifiée si et seulement si $(\omega, 2, 5, \omega)$ est plus petit qu'un ω-marquage de Clover, et elle est testable en un temps proportionnel à la taille de Clover.

L'ensemble Clover permet donc de résoudre de nombreux problèmes sans réappeler à chaque fois un algorithme doublement exponentiel (de couverture ou de calcul des bornes des places). Se pose donc maintenant la question de trouver des algorithmes efficaces pour calculer Clover.

Les variantes de l'algorithme de Karp et Miller. L'idée développée dans [3] est de modifier l'algorithme de Karp et Miller afin qu'à tout moment de l'exécution de l'algorithme, l'ensemble courant des étiquettes forme une antichaîne maximale (un ensemble d'éléments maximaux deux à deux incomparables). L'algorithme consiste à accélérer, comme l'algorithme original, le noeud courant puis à *détruire* tous les sous-arbres dont le marquage de la racine est strictement couvert par le marquage du noeud courant. A l'inverse si le marquage du noeud courant est couvert par le marquage d'un noeud existant alors l'exploration à partir du noeud courant est stoppée. Malheureusement cet algorithme contient un bug, identifié en 2005, et pour certaines exécutions calcule une sous-approximation stricte de Clover [5, 6]. Depuis 2005, trois principaux algorithmes (avec des variantes) ont été proposés [6, 12, 14, 11] pour calculer Clover sans construire complètement l'arbre de Karp et Miller. Une évaluation empirique de ces trois algorithmes a donné de très bonnes performances sur la plupart des études de cas communément analysées mais aucune borne théorique du surcoût en mémoire de ces trois algorithmes par rapport l'algorithme développé dans [3] n'est connue.

Nos contributions. Tout d'abord nous donnons une preuve simple et élégante de l'algorithme de Karp et Miller basée sur trois nouveaux concepts : *abstraction*, *accélération* et *séquence d'exploration*. Puis nous proposons une version accélérée de l'algorithme de Karp et Miller.

— Une abstraction est une ω-transition (i.e. une transition généralisée) (1) dont l'incidence arrière et l'incidence vis à vis d'une place peut être égale à ω (i.e. appartient à $\mathbb{N}_\omega$) et (2) qui dispose d'une famille infinie de séquences de transitions « justifiant » l'introduction des ω. Nous montrons que le franchissement à partir d'un ω-marquage dont l'idéal associé est inclus dans Cover conduit à un ω-marquage dont l'idéal associé est également inclus dans Cover. Nous prouvons ensuite que la concaténation d'abstractions est encore une abstraction. Une accélération est une abstraction dont l'incidence vis à vis de chaque place est soit nulle soit égale à ω. Nous établissons que toute abstraction à incidence positive se transforme en une accélération en substituant aux composantes strictement positives de l'incidence la valeur ω.

— Le preuve de l'algorithme de Karp et Miller est alors très simple, moyennant l'ajout de

variables fantômes (i.e. sans effet sur l'exécution de l'algorithme). La preuve de terminaison repose sur le belordre de $\mathbb{N}_\omega^P$. La preuve de la consistance est une conséquence quasi-immédiate des propriétés des abstractions et des accélérations. La démonstration inductive de la complétude également courte s'appuie sur la notion de séquence d'exploration détaillée plus loin.

— Nous approfondissons ensuite l'étude des accélérations. L'ensemble des accélérations muni d'un ordre naturel est un belordre : il peut être représenté par sa base finie d'éléments minimaux. Nous montrons que les coefficients entiers des accélérations minimales sont bornés par une expression $B(e, d)$ qui est polynomiale en la taille e des matrices d'incidence et doublement exponentielle en le nombre de places d. Nous montrons aussi comment transformer (*tronquer*) une accélération quelconque en une accélération dont les coefficients entiers sont bornés par $B(e, d)$. Nous proposons alors une version accélérée de l'algorithme de Karp et Miller avec un gain escompté en temps d'exécution. Le principe général est le suivant : lorsqu'on découvre une accélération, on la tronque et on la mémorise. Puis à chaque étape de l'algorithme, on augmente le marquage du noeud courant par le franchissement des accélérations franchissables. En raison de la troncature des accélérations, notre version accélérée de l'algorithme de Karp et Miller demande un surcoût minime en mémoire, comparé à l'occupation mémoire de l'algorithme de Karp et Miller et à sa complexité intrinsèque non-primitive récursive. De plus, la preuve de correction de notre variante accélérée se déduit de façon immédiate de notre preuve originale.

Organisation. Dans la section 2, nous introduisons et étudions les abstractions et les accélérations d'un réseau de Petri. Nous établissons ensuite la preuve de l'algorithme de Karp et Miller dans la section 3. Dans la section 4, nous décrivons notre version de l'algorithme de Karp et Miller. Enfin nous concluons et proposons des perspectives à notre travail dans la section 5.

2 Abstractions de couverture

2.1 Réseaux de Petri : accessibilité et couverture

Nous définissons ici les réseaux de Petri de manière différente mais équivalente à la définition usuelle, à savoir à l'aide de la matrice d'incidence arrière $\mathbf{Pre}$ et de la matrice d'incidence $\mathbf{C}$. La matrice d'incidence avant est implicitement définie par $\mathbf{C} + \mathbf{Pre}$. Ce choix est justifié par l'introduction des abstractions à la section 2.2.

Définition 1. *Un réseau de Petri (RdP) est un tuple* $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{C} \rangle$ *où :*

— P *est l'ensemble fini des* places *;*

— T *est l'ensemble fini des* transitions *avec* $P \cap T = \emptyset$ *;*

— $\mathbf{Pre} \in \mathbb{N}^{P \times T}$ *est la* matrice d'incidence arrière *;*

— $\mathbf{C} \in \mathbb{Z}^{P \times T}$ *est la* matrice d'incidence *qui vérifie :*
pour tout $p \in P$ *et tout* $t \in T$, $\mathbf{C}(p, t) + \mathbf{Pre}(p, t) \geq 0$.

Un RdP marqué $(\mathcal{N}, \mathbf{m}_0)$ *est un RdP* $\mathcal{N}$ *muni d'un* marquage *initial* $\mathbf{m}_0 \in \mathbb{N}^P$.

Le vecteur colonne de la matrice $\mathbf{Pre}$ (resp. $\mathbf{C}$) indicé par $t \in T$ est noté $\mathbf{Pre}(t)$ (resp. $\mathbf{C}(t)$). Une transition $t \in T$ est *franchissable* depuis un marquage $\mathbf{m} \in \mathbb{N}^P$ si $\mathbf{m} \geq \mathbf{Pre}(t)$. Lorsque t est franchissable depuis $\mathbf{m}$, son *franchissement* conduit au marquage $\mathbf{m}' \stackrel{\text{def}}{=} \mathbf{m} + \mathbf{C}(t)$, ce qu'on note $\mathbf{m} \xrightarrow{t} \mathbf{m}'$. On étend la franchissabilité et le franchissement à une séquence $\sigma \in T^*$ par récurrence sur sa longueur. La séquence vide ε est toujours franchissable et ne modifie pas le

marquage. La séquence $\sigma = t\sigma'$, avec $t \in T$ et $\sigma' \in T^*$ est franchissable depuis $\mathbf{m}$ si $\mathbf{m} \xrightarrow{t} \mathbf{m}'$ et σ' est franchissable depuis $\mathbf{m}'$. Le franchissement de σ depuis $\mathbf{m}$ conduit au marquage $\mathbf{m}''$ atteint par σ' depuis $\mathbf{m}'$. On note ce franchissement par $\mathbf{m} \xrightarrow{\sigma} \mathbf{m}''$.

Définition 2. *Soit* $(\mathcal{N}, \mathbf{m}_0)$ *un RdP marqué.* L'ensemble d'accessibilité $Reach(\mathcal{N}, \mathbf{m}_0)$ *est défini par :*

$$Reach(\mathcal{N}, \mathbf{m}_0) = \{\mathbf{m} \mid \exists \sigma \in T^* \ \mathbf{m}_0 \xrightarrow{\sigma} \mathbf{m}\}$$

Avant d'introduire l'ensemble de couverture d'un RdP, nous rappelons quelques éléments relatifs aux ensembles ordonnés. Soit $(X, \leq)$ un ensemble ordonné, nous dirons aussi que X est ordonné quand l'ordre est implicite. La *clôture* inférieure (resp. supérieure) d'un ensemble $E \subseteq X$ est notée $\downarrow E$ (resp. $\uparrow E$) et définie par :

$$\downarrow E = \{x \in X \mid \exists y \in E \ y \geq x\} \qquad (\text{resp. } \uparrow E = \{x \in X \mid \exists y \in E \ y \leq x\})$$

Un ensemble $E \subseteq X$ est *clos supérieurement* (resp. inférieurement) si $E = \uparrow E$ (resp. $E = \downarrow E$).

Une *antichaine* E est un ensemble qui vérifie : $\forall x \neq y \in E \ \neg(x \leq y \vee y \leq x)$. X est dit *FAC* si toutes ses antichaines sont finies. Un ensemble $E \subset X$ est *dirigé* si pour tout $x, y \in E$ il existe $z \in E$ tel que $x \leq z$ et $y \leq z$. Un *idéal* est un ensemble clos inférieurement et dirigé. Il existe une caractérisation très intéressante des ensembles FAC : un ensemble est FAC si et seulement si il est égal à une union finie d'idéaux (on trouvera une preuve de ce résultat bien connu des mathématiciens dans [1]). Pour un ensemble $E \subseteq X$, il peut exister plusieurs ensembles finis d'idéaux dont l'union est égale à E. Parmi tous ces ensembles finis, on peut choisir *l'unique* ensemble des idéaux *maximaux* (pour l'inclusion) qui a de plus la propriété d'être une partition de E : cet ensemble est donc canoniquement associé à E.

Rappelons qu'un ensemble ordonné $(X, \leq)$ est *bien fondé* si toutes ses suites strictement décroissantes sont finies et que $(X, \leq)$ est un *belordre* s'il est FAC et bien fondé. On peut donner une autre caractérisation d'un belordre : un ensemble $(X, \leq)$ est un belordre si et seulement si : pour toute suite $(x_n)_{n \in \mathbb{N}}$ de X, il existe une sous-suite infinie croissante. Rappelons enfin que $(\mathbb{N}, \leq)$ et $(\mathbb{N}^P, \leq)$ sont des belordres.

Nous sommes maintenant en mesure d'introduire la *couverture* (appelée aussi l'ensemble de couverture) d'un réseau et d'en étudier quelques propriétés.

Définition 3. *Soit* $(\mathcal{N}, \mathbf{m}_0)$ *un RdP marqué.* L'ensemble de couverture $Cover(\mathcal{N}, \mathbf{m}_0)$ *est défini par :*

$$Cover(\mathcal{N}, \mathbf{m}_0) = \downarrow Reach(\mathcal{N}, \mathbf{m}_0)$$

Puisque l'ensemble de couverture est clos inférieurement et que $\mathbb{N}^P$ est FAC, il peut s'exprimer comme une union finie d'idéaux. Les idéaux de $\mathbb{N}^P$ peuvent être définis de manière élégante comme suit. On étend d'abord les entiers naturels et relatifs : $\mathbb{N}_\omega = \mathbb{N} \cup \{\omega\}$ et $\mathbb{Z}_\omega = \mathbb{Z} \cup \{\omega\}$. Puis on étend la relation d'ordre et l'addition à $\mathbb{Z}_\omega$: pour tout $n \in \mathbb{Z}$, $\omega > n$ et pour tout $n \in \mathbb{Z}_\omega$, $n + \omega = \omega + n = \omega$. $\mathbb{N}_\omega^P$ muni de cet ordre étendu est aussi un belordre et ses élements sont appelés des *ω-marquages*. Il y a une correspondance bi-univoque entre les idéaux de $\mathbb{N}^P$ et les ω-marquages. Soit $\mathbf{m} \in \mathbb{N}_\omega^P$. Définissons $[\![\mathbf{m}]\!]$ par :

$$[\![\mathbf{m}]\!] = \{\mathbf{m}' \in \mathbb{N}^P \mid \mathbf{m}' \leq \mathbf{m}\}$$

$[\![\mathbf{m}]\!]$ est un idéal de $\mathbb{N}^P$ (et tout idéal peut être défini ainsi). En vertu des propriétés énoncées plus haut, nous sommes en mesure de définir formellement le Clover d'un réseau de Petri.

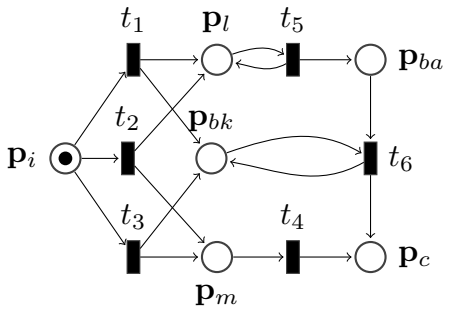

FIGURE 1 – Un RdP non borné

Définition 4. *Soit* $(\mathcal{N}, \mathbf{m}_0)$ *un RdP marqué. Alors* $Clover(\mathcal{N}, \mathbf{m}_0) \subseteq \mathbb{N}_\omega^p$ *est l'ensemble des idéaux maximaux tel que :*

$$Cover(\mathcal{N}, \mathbf{m}_0) = \bigcup_{\mathbf{m} \in Clover(\mathcal{N}, \mathbf{m}_0)} [\![\mathbf{m}]\!]$$

Remark : On peut montrer que $Clover(\mathcal{N}, \mathbf{m}_0)$ est ainsi bien défini de façon unique et qu'il est le *plus petit* (en nombre d'éléments) ensemble fini parmi tous les ensembles finis d'idéaux dont l'union est égale à $Cover(\mathcal{N}, \mathbf{m}_0)$.

On pourra trouver dans [1] une définition plus générale de *Clover* pour les systèmes de transitions bien structurés. L'un des buts de l'algorithme de Karp et Miller est le calcul de $Clover(\mathcal{N}, \mathbf{m}_0)$.

Exemple. *Le réseau de la figure 1 est non borné. Son Clover est l'ensemble suivant à quatre éléments :*

$$\{p_i, p_{bk} + p_m, p_l + p_m + \omega p_{ba}, p_l + p_{bk} + \omega p_{ba} + \omega p_c\}$$

Par exemple, le marquage $p_l + p_{bk} + \alpha p_{ba} + \beta p_c$ *est atteint donc couvert par la séquence* $t_1 t_5^{\alpha+\beta} t_6^{\beta}$.

2.2 Abstraction et accélération

Afin d'introduire les abstractions et les accélerations, nous généralisons les transitions pour prendre en compte la possibilité de marquer une place avec ω jetons.

Définition 5. *Soit* P *un ensemble de places. Une* ω*-transition* $\mathbf{a}$ *est définie par :*

— $\mathbf{Pre}(\mathbf{a}) \in \mathbb{N}_\omega^P$ *son* incidence arrière *;*

— $\mathbf{C}(\mathbf{a}) \in \mathbb{Z}_\omega^P$ *son* incidence *avec* $\mathbf{Pre}(\mathbf{a}) + \mathbf{C}(\mathbf{a}) \geq 0$.

Par souci d'homogénéité, on notera $\mathbf{Pre}(\mathbf{a})(p)$ (resp. $\mathbf{C}(\mathbf{a})(p)$) par $\mathbf{Pre}(p, \mathbf{a})$ (resp. $\mathbf{C}(p, \mathbf{a})$). Une ω-transition $\mathbf{a}$ est franchissable depuis un ω-marquage $\mathbf{m} \in \mathbb{N}_\omega^P$ si $\mathbf{m} \geq \mathbf{Pre}(\mathbf{a})$. Lorsque $\mathbf{a}$ est franchissable depuis $\mathbf{m}$, son franchissement conduit à l'ω-marquage $\mathbf{m}' \stackrel{\text{def}}{=} \mathbf{m} + \mathbf{C}(\mathbf{a})$, ce qu'on note comme précédemment $\mathbf{m} \xrightarrow{\mathbf{a}} \mathbf{m}'$. On remarque que si $\mathbf{Pre}(p, \mathbf{a}) = \omega$ alors quelque soit la valeur de $\mathbf{C}(p, \mathbf{a})$, $\mathbf{m}'(\mathbf{a}) = \omega$. Aussi sans perte de généralité, on suppose que pour toute ω-transition $\mathbf{a}$, $\mathbf{Pre}(p, \mathbf{a}) = \omega$ implique $\mathbf{C}(p, \mathbf{a}) = \omega$.

Afin de définir les abstractions, nous définissons les incidences d'une séquence σ d'ω-transitions par récurrence sur la longueur. Comme précédemment, nous introduisons $\mathbf{Pre}(p,\sigma) \stackrel{\text{def}}{=} \mathbf{Pre}(\sigma)(p)$ et $\mathbf{C}(p,\sigma) \stackrel{\text{def}}{=} \mathbf{C}(\sigma)(p)$. Le cas de base correspond à la définition d'une ω-transition. Soit $\sigma = t\sigma'$, avec t une ω-transition et σ' une séquence d'ω-transitions, alors :

- $\mathbf{C}(\sigma) = \mathbf{C}(t) + \mathbf{C}(\sigma')$;
- Pour tout $p \in P$
  - si $\mathbf{C}(p,t) = \omega$ alors $\mathbf{Pre}(p,\sigma) = \mathbf{Pre}(p,t)$;
  - sinon si $\mathbf{Pre}(p,\sigma') = \omega$ alors $\mathbf{Pre}(p,\sigma) = \omega$;
  - sinon $\mathbf{Pre}(p,\sigma) = \max(\mathbf{Pre}(p,t), \mathbf{Pre}(p,\sigma') - \mathbf{C}(p,t))$.

On vérifie immediatement que σ est franchissable depuis $\mathbf{m}$ si et seulement si $\mathbf{m} \geq \mathbf{Pre}(\sigma)$ et dans ce cas, $\mathbf{m} \xrightarrow{\sigma} \mathbf{m} + \mathbf{C}(\sigma)$.

Une *abstraction* d'un RdP est une ω-transition qui reflète de manière concise le comportement du réseau du point de vue de la couverture (voir la proposition 7). On remarquera qu'une transition t d'un RdP est par construction (avec $\sigma_n = t$) une abstraction.

Définition 6. *Soit $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{C}\rangle$ un RdP et $\mathbf{a}$ une ω-transition. $\mathbf{a}$ est une* abstraction *si pour tout $n \geq 0$, il existe $\sigma_n \in T^*$ telle que pour tout $p \in P$ avec $\mathbf{Pre}(p,\mathbf{a}) \in \mathbb{N}$:*

1. *$\mathbf{Pre}(p,\sigma_n) \leq \mathbf{Pre}(p,\mathbf{a})$;*
2. *Si $\mathbf{C}(p,\mathbf{a}) \in \mathbb{Z}$ alors $\mathbf{C}(p,\sigma_n) \geq \mathbf{C}(p,\mathbf{a})$;*
3. *Si $\mathbf{C}(p,\mathbf{a}) = \omega$ alors $\mathbf{C}(p,\sigma_n) \geq n$.*

La proposition suivante justifie l'intérêt des abstractions.

Proposition 7. *Soit $(\mathcal{N}, \mathbf{m}_0)$ un RdP marqué, $\mathbf{a}$ une abstraction et $\mathbf{m}$ un ω-marquage tels que : $[\![\mathbf{m}]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$ et $\mathbf{m} \xrightarrow{\mathbf{a}} \mathbf{m}'$. Alors $[\![\mathbf{m}']\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$.*

Preuve. Soit $\mathbf{m}^* \in [\![\mathbf{m}']\!]$. Posons $n = \max(\mathbf{m}^*(p) \mid \mathbf{m}'(p) = \omega)$
et $\ell = \max(\mathbf{Pre}(p,\sigma_n), n - \mathbf{C}(p,\sigma_n) \mid \mathbf{m}(p) = \omega)$. Définissons $\mathbf{m}^\sharp \in [\![\mathbf{m}]\!]$ ainsi :

- Si $\mathbf{m}(p) < \omega$ alors $\mathbf{m}^\sharp(p) = \mathbf{m}(p)$;
- Sinon $\mathbf{m}^\sharp(p) = \ell$.

Vérifions que σ_n est franchissable depuis $\mathbf{m}^\sharp$. Soit $p \in P$,

- Si $\mathbf{m}(p) < \omega$ alors $\mathbf{m}^\sharp(p) = \mathbf{m}(p) \geq \mathbf{Pre}(p,\mathbf{a}) \geq \mathbf{Pre}(p,\sigma_n)$;
- Sinon $\mathbf{m}^\sharp(p) = \ell \geq \mathbf{Pre}(p,\sigma_n)$.

Démontrons que $\mathbf{m}^\sharp + \mathbf{C}(\sigma_n) \geq \mathbf{m}^*$. Soit $p \in P$,

- Si $\mathbf{m}(p) < \omega$ et $\mathbf{C}(p,\mathbf{a}) < \omega$ alors $\mathbf{m}^\sharp(p) + \mathbf{C}(p,\sigma_n) \geq \mathbf{m}(p) + \mathbf{C}(p,\mathbf{a}) = \mathbf{m}'(p) \geq \mathbf{m}^*(p)$;
- Si $\mathbf{m}(p) < \omega$ et $\mathbf{C}(p,\mathbf{a}) = \omega$ alors $\mathbf{m}^\sharp(p) + \mathbf{C}(p,\sigma_n) \geq \mathbf{C}(p,\sigma_n) \geq n \geq \mathbf{m}^*(p)$;
- Si $\mathbf{m}(p) = \omega$ alors $\mathbf{m}^\sharp(p) + \mathbf{C}(p,\sigma_n) \geq n - \mathbf{C}(p,\sigma_n) + \mathbf{C}(p,\sigma_n) = n \geq \mathbf{m}^*(p)$.

■

Une façon simple de construire de nouvelles abstractions consiste à les concaténer.

Proposition 8. *Soit $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{C}\rangle$ un RdP et σ une séquence d'abstractions. Alors l'ω-transition $\mathbf{a}$ définie par $\mathbf{Pre}(\mathbf{a}) = \mathbf{Pre}(\sigma)$ et $\mathbf{C}(\mathbf{a}) = \mathbf{C}(\sigma)$ est une abstraction.*

Preuve. Nous établissons ce résultat par récurrence sur la longueur de σ. Le cas de base est immédiat. Soit $\sigma = b\sigma'$ et (par hypothèse de récurrence) soit $\{\sigma'_n\}_{n\in\mathbb{N}}$ une famille de séquences de transitions associées à σ'. Soit $\{\sigma_{n,b}\}_{n\in\mathbb{N}}$ une famille de séquences de transitions associées à b. Fixons $n \in \mathbb{N}$ et définissons $n' = \max(n, \max(n - \mathbf{C}(p,b) \mid \mathbf{C}(p,b) < \omega = \mathbf{C}(p,\sigma')))$. Définissons $\ell = \max(\mathbf{Pre}(p,\sigma'_{n'}), n - \mathbf{C}(p,\sigma'_{n'}) \mid \mathbf{Pre}(p,b) < \omega = \mathbf{C}(p,b))$. Vérifions que $\sigma_{\ell,b}\sigma'_{n'}$ satisfait les conditions de la définition 6. Soit $p \in P$, $\mathbf{Pre}(p,\mathbf{a}) < \omega$ si et seulement si (1) $\mathbf{Pre}(p,b) < \omega$ et $\mathbf{C}(p,b) = \omega$ ou (2) $\mathbf{Pre}(p,b) < \omega$ et $\mathbf{C}(p,b) < \omega$ et $\mathbf{Pre}(p,\sigma') < \omega$.

Cas $\mathbf{Pre}(p,b) < \omega$ et $\mathbf{C}(p,b) = \omega$. Par conséquent, $\mathbf{Pre}(p,\mathbf{a}) = \mathbf{Pre}(p,b)$ et $\mathbf{C}(p,\mathbf{a}) = \omega$.
On a donc $\mathbf{Pre}(\sigma_{\ell,b}) \leq \mathbf{Pre}(p,b) = \mathbf{Pre}(p,\mathbf{a})$.
De plus $\mathbf{Pre}(p,\sigma_{\ell,b}) + \mathbf{C}(p,\sigma_{\ell,b}) \geq \mathbf{C}(p,\sigma_{\ell,b}) \geq \ell \geq \mathbf{Pre}(p,\sigma'_{n'})$.
Enfin $\mathbf{C}(p,\sigma_{\ell,b}) + \mathbf{C}(p,\sigma'_{n'}) \geq \ell + \mathbf{C}(p,\sigma'_{n'}) \geq n - \mathbf{C}(p,\sigma'_{n'}) + \mathbf{C}(p,\sigma'_{n'}) \geq n$.

Cas $\mathbf{Pre}(p,b) < \omega$ et $\mathbf{C}(p,b) < \omega$ et $\mathbf{Pre}(p,\sigma') < \omega$.
D'où $\mathbf{Pre}(p,\mathbf{a}) = \max(\mathbf{Pre}(p,b), \mathbf{Pre}(p,\sigma') - \mathbf{C}(p,b))$. Or :

$$\begin{aligned}
\mathbf{Pre}(p,\sigma_{\ell,b}\sigma'_{n'}) = & \quad \max(\mathbf{Pre}(p,\sigma_{\ell,b}), \mathbf{Pre}(p,\sigma'_{n'}) - \mathbf{C}(p,\sigma_{\ell,b})) \\
\leq & \quad \max(\mathbf{Pre}(p,b), \mathbf{Pre}(p,\sigma') - \mathbf{C}(p,b)) \\
= & \quad \mathbf{Pre}(p,\mathbf{a})
\end{aligned}$$

Il y a maintenant deux sous-cas à considérer.

∘ $\mathbf{C}(p,\sigma') < \omega$. Par conséquent, $\mathbf{C}(p,\mathbf{a}) = \mathbf{C}(p,b) + \mathbf{C}(p,\sigma')$.
Or $\mathbf{C}(p,\sigma_{\ell,b}\sigma'_n) = \mathbf{C}(p,\sigma_{\ell,b}) + \mathbf{C}(p,\sigma'_n) \geq \mathbf{C}(p,b) + \mathbf{C}(p,\sigma') = \mathbf{C}(p,\mathbf{a})$.

∘ $\mathbf{C}(p,\sigma') = \omega$. Par conséquent, $\mathbf{C}(p,\mathbf{a}) = \omega$.
Or $\mathbf{C}(p,\sigma_{\ell,b}\sigma'_{n'}) = \mathbf{C}(p,\sigma_{\ell,b}) + \mathbf{C}(p,\sigma'_{n'}) \geq \mathbf{C}(p,b) + n - \mathbf{C}(p,b) = n$.
Par conséquent $\mathbf{a}$ est une abstraction.

■

Nous introduisons maintenant le concept sous-jacent à la construction de Karp et Miller.

Définition 9. *Soit $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{C}\rangle$ un RdP. On dit que* $\mathbf{a}$ *est une* accélération *si* $\mathbf{a}$ *est une abstraction telle que* $\mathbf{C}(\mathbf{a}) \in \{0,\omega\}^P$.

La proposition suivante fournit un moyen d'obtenir une accélération à partir d'une abstraction quelconque.

Proposition 10. *Soit $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{C}\rangle$ un RdP et* $\mathbf{a}$ *une abstraction. Définissons* $\mathbf{a}'$ *une ω-transition ainsi. Pour tout $p \in P$:*

— *Si* $\mathbf{C}(p,\mathbf{a}) < 0$ *alors* $\mathbf{Pre}(p,\mathbf{a}') = \mathbf{C}(p,\mathbf{a}') = \omega$ *;*

— *Si* $\mathbf{C}(p,\mathbf{a}) = 0$ *alors* $\mathbf{Pre}(p,\mathbf{a}') = \mathbf{Pre}(p,\mathbf{a})$ *et* $\mathbf{C}(p,\mathbf{a}') = 0$ *;*

— *Si* $\mathbf{C}(p,\mathbf{a}) > 0$ *alors* $\mathbf{Pre}(p,\mathbf{a}') = \mathbf{Pre}(p,\mathbf{a})$ *et* $\mathbf{C}(p,\mathbf{a}') = \omega$.

Alors $\mathbf{a}'$ *est une accélération.*

Preuve. Considérons $\{\sigma_n\}_{n\in\mathbb{N}}$ une famille associée à l'abstraction $\mathbf{a}$. Nous allons démontrer que la famille $\{\sigma^n_n\}_{n\in\mathbb{N}}$ vérifie les conditions de la définition 6 relativement à $\mathbf{a}'$. Pour tout n :

— Soit $p \in P$ tel que $\mathbf{Pre}(p,\mathbf{a}') < \omega$.
Ceci implique que $\mathbf{C}(p,\mathbf{a}) \geq 0$ et que $\mathbf{Pre}(p,\mathbf{a}') = \mathbf{Pre}(p,\mathbf{a})$.
Puisque $\mathbf{C}(p,\sigma_n) \geq \mathbf{C}(p,\mathbf{a}) \geq 0$, $\mathbf{Pre}(p,\sigma^n_n) = \mathbf{Pre}(p,\sigma_n) \leq \mathbf{Pre}(p,\mathbf{a}) = \mathbf{Pre}(p,\mathbf{a}')$;

— Soit $p \in P$ tel que $\mathbf{C}(p,\mathbf{a}') = 0$. On a donc $0 = \mathbf{C}(p,\mathbf{a}) \leq \mathbf{C}(\sigma_n)$.
Par conséquent, $0 \leq n\mathbf{C}(\sigma_n) = \mathbf{C}(\sigma^n_n)$;

— Soit $p \in P$ tel que $\mathbf{Pre}(p, \mathbf{a}') < \omega$ et $\mathbf{C}(p, \mathbf{a}') = \omega$. Ceci implique que $\mathbf{C}(p, \mathbf{a}) > 0$. On a donc $1 \leq \mathbf{C}(p, \mathbf{a}) \leq \mathbf{C}(\sigma_n)$. Par conséquent, $n \leq n\mathbf{C}(\sigma_n) = \mathbf{C}(\sigma_n^n)$.

■

3 L'algorithme de Karp et Miller

L'algorithme 1 est l'algorithme de Karp-Miller auquel nous avons ajouté Acc et δ des variables « fantômes » (i.e. sans influence sur le comportement de l'algorithme) qui en facilitent grandement la preuve. Nous décrivons de manière synthétique cet algorithme. Il maintient un arbre orienté $Tr = (V, E, \lambda, \delta)$ dont les sommets (V) sont étiquetés par un ω-marquage (fonction λ) et les arcs sont étiquetés par une séquence d'ω-transitions appartenant à $T\mathsf{Acc}^*$. Il maintient un sous-ensemble de sommets (Front) qui restent encore à explorer. Afin de faciliter l'écriture de l'algorithme, nous avons introduit $\mathsf{Anc}(u)$ l'ensemble des ancêtres de u (en excluant u).

Tant que Front n'est pas vide, l'algorithme choisit un sommet $u \in \mathsf{Front}$. Trois cas se présentent alors :

— Le marquage de u est inférieur ou égal à celui d'un ancêtre u' : alors u est extrait de Front et de V et l'arc entrant en u est détruit.

— Le marquage de u est supérieur à celui d'un ancêtre u' et pour au moins une place p, $\lambda(u')(p) < \lambda(u)(p) < \omega$. Alors pour toutes ces places p, on substitue au marquage entier la valeur ω. *Dans notre version*, on définit aussi une ω-transition $\mathbf{a}$ à partir de la séquence d'ω-transitions qui étiquettent le chemin de u' à u en substituant aux incidences de place strictement positives la valeur ω et on concatène cette accélération à la séquence qui étiquette l'arc entrant en u.

— Sinon on détermine les transitions franchissables et on les franchit pour créer les fils de u qui sont insérés dans Front. Le sommet u est extrait de Front. *Dans notre version*, l'arc entrant dans un nouveau sommet est étiqueté par la transition qui a été franchie.

Lorsque Front est vide, l'algorithme se termine. L'ensemble $Clover(\mathcal{N}, \mathbf{m}_0)$ correspond aux ω-marquages maximaux associés aux sommets de V.

Exemple. *La figure 2 montre l'arbre de Karp et Miller correspondant au réseau de la figure 1. Décrivons son déroulement lors du développement de la branche la plus à gauche. A partir du marquage initial, on franchit t_1 ce qui conduit à $p_l + p_{bk}$, incomparable avec $\mathbf{m}_0$. On poursuit donc l'exploration. Seule t_5 est franchissable et conduit au marquage $p_l + p_{bk} + p_{ba}$. Une accélération $\mathbf{a}_1$ est découverte avec $\mathbf{Pre}(\mathbf{a}_1) = p_l$ et $\mathbf{C}(\mathbf{a}_1) = \omega p_{ba}$. Le marquage courant est alors modifié conformément au franchissement de $\mathbf{a}_1$. Ce sommet est examiné à nouveau lors d'une itération ultérieure. Il n'y a plus d'accélération possible. Par conséquent on poursuit l'exploration : t_5 et t_6 sont franchissables. Le sommet associé au franchissement de t_5 a un marquage identique : il sera donc détruit. Le sommet associé au franchissement de t_6 donne lieu a une nouvelle accélération : $\mathbf{a}_2$ avec $\mathbf{Pre}(\mathbf{a}_2) = p_{bk} + \omega p_l$ et $\mathbf{C}(\mathbf{a}_2) = \omega p_c$. Depuis cet ultime marquage t_5 et t_6, sont franchissables et conduisent au même marquage ce qui termine le développement de la branche. Observons que $\mathbf{a}_1$ est re-découvert à deux reprises lors de la construction. Le Cover calculé ici est représenté par un ensemble de 11 noeuds et donc 11 ω-marquages. Cet ensemble de 11 ω-marquages est redondant mais on peut calculer Clover en conservant seulement les éléments maximaux, soit l'ensemble suivant à 4 élements : $\{p_i, p_l + p_{bk} + \omega p_{ba} + \omega p_c, p_{bk} + p_m, p_l + p_m + \omega p_{ba}\}$.*

Nous allons maintenant établir la correction de l'algorithme de Karp et Miller à savoir :

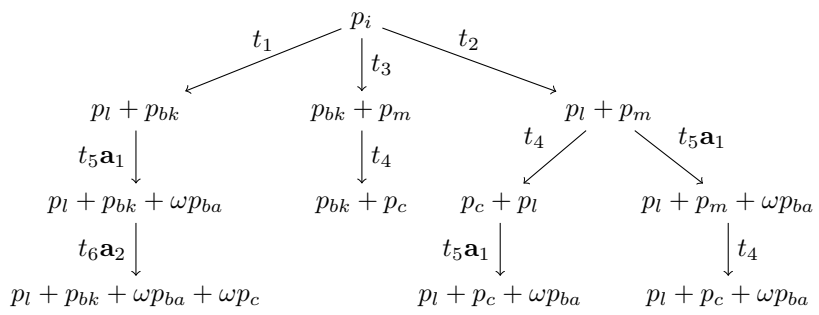

FIGURE 2 – Un arbre de Karp et Miller

Algorithme 1 : L'algorithme de Karp et Miller

```
KarpMiller(𝒩, m_0)
Input : Un RdP marqué (𝒩, m_0)
Data : V ensemble de sommets ; E ⊆ V × V ; Front ⊆ V ; λ : V → ℕ_ω^p ; δ : E → TAcc* ;
Tr = (V, E, λ, δ) un arbre étiqueté ; Acc un ensemble d'ω-transitions ;
u, u', u'' des sommets ; a une accélération ;
Output : Un arbre étiqueté Tr = (V, E, λ, δ)
V ← {r} ; E ← ∅ ; Front ← {r} ; λ(r) ← m_0 ; Acc ← ∅ ;
while Front ≠ ∅ do
    Choisir u ∈ Front
    if ∃u' ∈ Anc(u) t.q. λ(u') ≥ λ(u) then
        Front ← Front \ {u} ; V ← V \ {u} ; E ← E \ V × {u} // λ(u) est couvert
    else if ∃u' ∈ Anc(u) t.q. λ(u') < λ(u) ∧ ∃p λ(u')(p) < λ(u)(p) < ω then
        // Une accélération est trouvée entre u et l'un des ancestres de u
        Soit γ ∈ E* le chemin de u' à u dans Tr
        a ← NewAcceleration()
        foreach p ∈ P do
            if C(p, δ(γ)) < 0 then Pre(p, a) ← ω ; C(p, a) ← ω
            if C(p, δ(γ)) = 0 then Pre(p, a) ← Pre(p, δ(γ)) ; C(p, a) ← 0
            if C(p, δ(γ)) > 0 then Pre(p, a) ← Pre(p, δ(γ)) ; C(p, a) ← ω ; λ(u)(p) ← ω
        end
        Soit (u'', u) l'arc conduisant à u dans Tr
        δ((u'', u)) ← δ((u'', u)) · a ; Acc ← Acc ∪ {a}
    else
        Front ← Front \ {u}
        foreach t ∈ T t.q. λ(u) ≥ Pre(t) do
            // Ajout des fils de u
            u' ← NewNode() ; V ← V ∪ {u'} ; Front ← Front ∪ {u'} ; E ← E ∪ {(u, u')}
            λ(u') ← λ(u) + C(t) ; δ((u, u')) ← t
        end
    end
end
return Tr
```

— sa terminaison ;
— sa consistance : $\bigcup_{v \in V} [\![\lambda(v)]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$;
— sa complétude : $Cover(\mathcal{N}, \mathbf{m}_0) \subseteq \bigcup_{v \in V} [\![\lambda(v)]\!]$.

La terminaison repose sur le fait que $\mathbb{N}_\omega^P$ est un belordre.

Proposition 11 (terminaison)**.** *L'algorithme 1 termine.*

Preuve. Supposons que l'algorithme ne termine pas. Un sommet de l'arbre ne peut être choisi dans la boucle qu'au plus $|P| + 1$ fois. En effet il ne reste dans Front que s'il a fait l'objet d'une accélération ce qui implique que le marquage associé a, au moins, une composante de plus égale à ω. Par conséquent l'algorithme construit un arbre dont le nombre de sommets est infini. Chaque sommet a au plus $\mid T \mid$ fils. Par application du lemme de König, cet arbre a une branche infinie.
Soit $\mathbf{m}_0, \mathbf{m}_1, \ldots$ les marquages associés aux sommets de cette branche. $\mathbb{N}_\omega^P$ est un belordre. On peut donc en extraire une sous-suite croissante $\mathbf{m}_{\alpha(0)} \leq \mathbf{m}_{\alpha(1)} \leq \cdots$. On ne peut avoir l'égalité entre deux sommets consécutifs car alors le deuxième sommet aurait été détruit. On en déduit donc que (1) soit une accélération a été détectée le deuxième marquage a déjà, au moins, une composante égale à ω de plus, (2) soit le deuxième marquage a déjà , au moins, une composante égale à ω de plus. Ainsi chaque marquage a, au moins, une composante égale à ω de moins que le marquage suivant. Donc cette suite contient au plus $|P| + 1$ éléments : ce qui contredit l'hypothèse.

■

Le lemme suivant illustre l'intérêt d'avoir introduit les variables fantômes.

Lemme 12. *Pour tout arc $(u, v) \in E$, on a $\lambda(u) \xrightarrow{\delta(u,v)} \lambda(v)$.*

Preuve. Il y a deux cas à considérer.

• **Création de l'arc** (u, v)**.** Ceci se fait lors de la construction des successeurs de u. Par conséquent, il existe une transition $t \in T$ telle que $\lambda(u) \xrightarrow{t} \lambda(v)$ et cette transition étiquette l'arc.

• **Modification de** $\lambda(v)$**.** Ceci se fait lors de la découverte d'une accélération $\mathbf{a}$ entre un ancêtre u' de v et v. Notons $\mathbf{m}^-$ le marquage associé à v avant sa mise à jour et $\mathbf{m}^+$ le marquage associé à v après sa mise à jour. Par induction, la séquence d'ω-transitions le long du chemin de u' à v est franchissable depuis $\lambda(u')$, donc aussi depuis $\mathbf{m}^-$. Cette séquence a même précondition que $\mathbf{a}$ excepté éventuellement sur les places p telles que $\mathbf{m}^-(p) = \omega$. Donc $\mathbf{a}$ est franchissable depuis $\mathbf{m}^-$ et par construction $\mathbf{m}^- \xrightarrow{\mathbf{a}} \mathbf{m}^+$.

■

Le lemme suivant est basé sur la préservation des abstractions par concaténation et la construction d'accélérations à partir d'abstractions.

Lemme 13. *Toute ω-transition $\mathbf{a} \in$ Acc est une accélération.*

Preuve. La preuve se fait par induction selon l'ordre d'insertion dans Acc. Soit $\mathbf{a} \in Acc$ une ω-transition. Notons σ la séquence correspondante au chemin dans l'arbre qui a conduit à la création de $\mathbf{a}$. σ est une séquence d'abstractions (par hypothèse d'induction). En vertu de la proposition 8, c'est une abstraction. La construction de $\mathbf{a}$ à partir de σ correspond à la proposition 10. $\mathbf{a}$ est donc une accélération. ■

Le consistance de l'algorithme est maintenant une conséquence naturelle des lemmes précédents.

Proposition 14 (consistance)**.** *Pour tout $v \in V$, $[\![\lambda(v)]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$.*

Preuve. La preuve se fait par induction sur la longueur du chemin de r à u. Le marquage associé à r est $\mathbf{m}_0$. Or $[\![\mathbf{m}_0]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$. Notons u le père de v. Par hypothèse d'induction, $[\![\lambda(u)]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$. En vertu du lemme 12, $\lambda(u) \xrightarrow{\delta(u,v)} \lambda(v)$. En vertu du lemme 13, $\delta(u, v)$ est une séquence d'abstractions. En vertu de la proposition 8, $\delta(u, v)$ est donc une abstraction. En vertu de la proposition 7, $[\![\lambda(v)]\!] \subseteq Cover(\mathcal{N}, \mathbf{m}_0)$.

■

Afin de faciliter la preuve de complétude, nous introduisons la notion de séquence d'exploration relatif à l'arbre de couverture durant sa construction.

Définition 15. *Une séquence de franchissement de transitions $\mathbf{m} \xrightarrow{\sigma} \mathbf{m}'$ est une* séquence d'exploration *de Tr s'il existe $v \in \mathsf{Front}$ avec $\lambda(v) = \mathbf{m}$ et pour tout marquage $\mathbf{m}''$ visité par la séquence et tout $v \in V \setminus \mathsf{Front}$, on a : $\mathbf{m}'' \not\leq \lambda(v)$.*

Lemme 16. *Pour tout $\mathbf{m} \in Cover(\mathcal{N}, \mathbf{m}_0)$ au début de chaque tour de la boucle principale,*

1. *Soit il existe $v \in V \setminus \mathsf{Front}$ tel que $\mathbf{m} \in [\![\lambda(v)]\!]$;*
2. *Soit il existe une séquence d'exploration $\mathbf{m}_1 \xrightarrow{\sigma} \mathbf{m}_2 \geq \mathbf{m}$*

Preuve. Nous établissons ce résultat par induction sur le nombre de tours de boucle déjà effectués.

• Pour tout $\mathbf{m} \in Cover(\mathcal{N}, \mathbf{m}_0)$, il existe une séquence $\mathbf{m}_0 \xrightarrow{\sigma} \mathbf{m}_2 \geq \mathbf{m}$.
Or initialement $V = \mathsf{Front} = \{r\}$ et $\lambda(r) = \mathbf{m}_0$. Par conséquent, l'assertion 2 est vérifiée.

• Supposons le résultat établi au moment de débuter un tour de la boucle. Fixons $\mathbf{m} \in Cover(\mathcal{N}, \mathbf{m}_0)$. Si $\mathbf{m}$ satisfait l'assertion 1, elle reste satisfaite jusqu'à la terminaison de l'algorithme.

Supposons que $\mathbf{m}$ satisfait l'assertion 2. Notons la séquence d'exploration $\mathbf{m}_1 \xrightarrow{\sigma} \mathbf{m}_2 \geq \mathbf{m}$ avec $w \in \mathsf{Front}$ vérifiant $\lambda(w) = \mathbf{m}_1$. Considérons les différentes alternatives.

∘ $\exists u' \in \mathsf{Anc}(u)$ t.q. $\lambda(u') \geq \lambda(u)$. Ceci implique que $u \neq w$ et la séquence d'exploration est toujours valide.

∘ $\lambda(u)$ est modifié par une accélération. Si $u \neq w$, la séquence d'exploration est toujours valide. Si $u = w$ alors, puisque $\lambda(u)$ a été augmenté, σ est toujours franchissable à partir de $\lambda(u)$. Chaque marquage visité est supérieur ou égal au marquage correspondant de la séquence d'exploration. Donc cette nouvelle séquence est une séquence d'exploration qui couvre $\mathbf{m}$.

∘ u est retiré du Front et on calcule ses fils. Il y a maintenant deux sous-cas à considérer.

— Soit pour tout marquage visité $\mathbf{m}'$ par σ, $\mathbf{m}' \not\leq \lambda(u)$. Par conséquent, la séquence d'exploration est toujours valide.

— Dans le cas contraire, considérons $\mathbf{m}'$ le dernier marquage visité qui vérifie $\mathbf{m}' \leq \lambda(u)$ et le suffixe de la séquence $\mathbf{m}' \xrightarrow{\sigma'} \mathbf{m}_2$.
Si $\sigma' = \varepsilon$ alors $\mathbf{m} \leq \mathbf{m}_2 = \mathbf{m}' \leq \lambda(u)$. Par conséquent $\mathbf{m}$ vérifie l'assertion 1.
Sinon $\mathbf{m}' \xrightarrow{t} \mathbf{m}'' \xrightarrow{\sigma''} \mathbf{m}_2$. Puisque $\mathbf{m}' \leq \lambda(u)$, u a un fils $v \in \mathsf{Front}$ tel que $\lambda(u) \xrightarrow{t} \lambda(v) \geq \mathbf{m}''$. Par conséquent, $\lambda(v) \xrightarrow{\sigma''} \mathbf{m}^* \geq \mathbf{m}$ pour un certain $\mathbf{m}^*$ et vu le choix de $\mathbf{m}'$ cette séquence est une séquence d'exploration.

■

Proposition 17 (complétude)**.** *Lorsque l'algorithme 1 termine, $Cover(\mathcal{N}, \mathbf{m}_0) \subseteq \bigcup_{v \in V} [\![\lambda(v)]\!]$.*

Preuve. Lorsque l'algorithme 1 termine, l'ensemble Front est vide. Le résultat est alors une conséquence immédiate du lemme 16. ■

4 Une amélioration de l'algorithme

Afin de présenter une amélioration de l'algorithme, nous allons étudier plus en détail l'ensemble des accélérations et pour débuter nous équipons les ω-transitions d'un ordre naturel.

Définition 18. *Soit P un ensemble de places et deux ω-transitions $\mathbf{a}$ et $\mathbf{a}'$.*

$$\mathbf{a} \leq \mathbf{a}' \text{ si et seulement si } \mathbf{Pre}(\mathbf{a}) \leq \mathbf{Pre}(\mathbf{a}') \wedge \mathbf{C}(\mathbf{a}) \geq \mathbf{C}(\mathbf{a}')$$

Autrement dit, $\mathbf{a} \leq \mathbf{a}'$ si étant donné un ω-marquage $\mathbf{m}$, si $\mathbf{a}'$ est franchissable depuis $\mathbf{m}$ alors $\mathbf{a}$ est aussi franchissable et son franchissement conduit à un marquage supérieur ou égal à celui atteint par le franchissement de $\mathbf{a}'$.

Proposition 19. *Soit $\mathcal{N}$ un RdP. Alors l'ensemble des abstractions de $\mathcal{N}$ est clos supérieurement. De même, l'ensemble des accélérations est clos supérieurement dans l'ensemble des ω-transitions à incidence dans $\{0, \omega\}^P$.*

Preuve. Soit $\mathbf{a}$ une abstraction et $\mathbf{a}' \geq \mathbf{a}$ une ω-transition. Soit $\{\sigma_n\}_{n\in\mathbb{N}}$ une famille des séquences associée à $\mathbf{a}$. Soit $n_0 = \max(\mathbf{C}(p, \mathbf{a}') \mid \mathbf{C}(p, \mathbf{a}') \in \mathbb{N})$ avec par convention $\max(\emptyset) = 0$. Nous allons montrer que la famille $\{\sigma_{\max(n,n_0)}\}_{n\in\mathbb{N}}$ peut être associée à $\mathbf{a}'$. Soit p tel que $\mathbf{Pre}(p, \mathbf{a}') \in \mathbb{N}$. Ceci implique $\mathbf{Pre}(p, \mathbf{a}) \in \mathbb{N}$. D'autre part :

- $\mathbf{Pre}(p, \sigma_{\max(n,n_0)}) \leq \mathbf{Pre}(p, \mathbf{a}) \leq \mathbf{Pre}(p, \mathbf{a}')$;
- Si $\mathbf{C}(p, \mathbf{a}') \in \mathbb{Z}$ et $\mathbf{C}(p, \mathbf{a}) \in \mathbb{Z}$ alors $\mathbf{C}(p, \sigma_{\max(n,n_0)}) \geq \mathbf{C}(p, \mathbf{a}) \geq \mathbf{C}(p, \mathbf{a}')$;
- Si $\mathbf{C}(p, \mathbf{a}') \in \mathbb{Z}$ et $\mathbf{C}(p, \mathbf{a}) = \omega$ alors $\mathbf{C}(p, \sigma_{\max(n,n_0)}) \geq n_0 \geq \mathbf{C}(p, \mathbf{a}')$;
- Si $\mathbf{C}(p, \mathbf{a}') = \omega$ alors $\mathbf{C}(p, \mathbf{a}) = \omega$ et $\mathbf{C}(p, \sigma_{\max(n,n_0)}) \geq n$.

La démonstration s'applique aussi aux accélérations.

■

Proposition 20. *Soit $\mathcal{N}$ un RdP, l'ensemble des accélérations de $\mathcal{N}$ est un belordre.*

Preuve. L'ensemble des accélérations est un sous-ensemble de $\mathbb{N}^P \times \{0, \omega\}^P$ avec l'ordre obtenu par produit cartésien répété de $(\mathbb{N}, \leq)$ et $(\{0, \omega\}, \geq)$. Ces deux ensembles étant des belordres et le produit cartésien préservant cette propriété, la proposition s'en suit. ■

Observons que l'ensemble des accélérations est non vide puisqu'il contient l'accélération $\mathbf{a}$ définie par $\mathbf{Pre}(\mathbf{a}) = \mathbf{C}(\mathbf{a}) = 0$ dont la famille associée $\{\sigma_n\}$ est définie par : pour tout n, $\sigma_n = \varepsilon$. Puisque l'ensemble des accélérations est un belordre et qu'il est clos supérieurement, il est égal à la clôture supérieure de l'ensemble fini des accélérations *minimales*. Nous allons donc maintenant étudier la taille d'une accélération minimale. Etant donné un RdP, on note $d = |P|$ et $e = \max_{p,t}(\max(\mathbf{Pre}(p, t), \mathbf{Pre}(p, t) + \mathbf{C}(p, t))$.

Nous allons utiliser le résultat suivant de Jérôme Leroux (publié sur HAL en juin 2019) qui donne une borne à la longueur des séquences de transitions les plus courtes qui relient deux marquages $\mathbf{m}_1$ et $\mathbf{m}_2$ mutuellement accessibles.

Théorème 21. *(Theorem 2, [10]) Soit $\mathcal{N}$ un RdP, $\mathbf{m}_1, \mathbf{m}_2$ des marquages, σ_1, σ_2 des séquences de transitions telles que $\mathbf{m}_1 \xrightarrow{\sigma_1} \mathbf{m}_2 \xrightarrow{\sigma_2} \mathbf{m}_1$. Alors il existe σ'_1, σ'_2 telles que $\mathbf{m}_1 \xrightarrow{\sigma'_1} \mathbf{m}_2 \xrightarrow{\sigma'_2} \mathbf{m}_1$ vérifiant :*

$$|\sigma'_1\sigma'_2| \leq ||\mathbf{m}_1 - \mathbf{m}_2||_\infty (3de)^{(d+1)^{2d+4}}$$

On en déduit une borne supérieure sur la taille des accélérations minimales. Soit $\mathbf{v} \in \mathbb{N}_\omega^P$. On note $||\mathbf{v}||_\infty = \max(\mathbf{v}(p) \mid \mathbf{v}(p) \in \mathbb{N})$.

Proposition 22. *Soit $\mathcal{N}$ un RdP et* $\mathbf{a}$ *une accélération minimale.*
Alors $||\mathbf{Pre}(\mathbf{a})||_\infty \leq e(3de)^{(d+1)^{2d+4}}$.

Preuve. On considère le réseau $\mathcal{N}' = \langle P', T', \mathbf{Pre}', \mathbf{C}' \rangle$ obtenu à partir de $\mathcal{N}$ en supprimant l'ensemble de places $\{p \mid \mathbf{Pre}(p, \mathbf{a}) = \omega\}$ et en ajoutant l'ensemble de transitions $T_1 = \{t_p \mid p \in P'\}$ avec $\mathbf{Pre}(t_p) = 0$ et $\mathbf{C}(t_p) = -p$.
On note $P_1 = \{p \mid \mathbf{Pre}(p, \mathbf{a}) < \omega = \mathbf{C}(p, \mathbf{a})\}$. On introduit $\mathbf{m}_1$ le marquage obtenu en restreignant $\mathbf{Pre}(\mathbf{a})$ à P' et $\mathbf{m}_2 = \mathbf{m}_1 + \sum_{p \in P_1} p$. Observons que $d' \leq d$ et $e' = e$.
Soit $\{\sigma_n\}_{n \in \mathbb{N}}$ la famille de séquences associée à $\mathbf{a}$. Considérons $n^* = ||\mathbf{Pre}(\mathbf{a})||_\infty + 1$. Alors σ_{n^*} est franchissable dans $\mathcal{N}'$ depuis $\mathbf{m}_1$ et son franchissement conduit à un marquage qui couvre $\mathbf{m}_2$. En concaténant des transitions de T_1, on obtient une séquence de franchissement dans $\mathcal{N}'$ $\mathbf{m}_1 \xrightarrow{\sigma_1} \mathbf{m}_2$. Par le même procédé, on obtient une séquence $\mathbf{m}_2 \xrightarrow{\sigma_2} \mathbf{m}_1$.
Appliquons le Théorème 21. Il existe une séquence σ_1' avec $\mathbf{m}_1 \xrightarrow{\sigma_1'} \mathbf{m}_2$ et $|\sigma_1'| \leq (3de)^{(d+1)^{2d+4}}$ car $||\mathbf{m}_1 - \mathbf{m}_2||_\infty = 1$. En supprimant les transitions de T_1 dans σ_1', on obtient une séquence $\sigma_1'' \in T^*$ avec $\mathbf{m}_1 \xrightarrow{\sigma_1''} \mathbf{m}_2' \geq \mathbf{m}_2$ avec $|\sigma_1''| \leq (3de)^{(d+1)^{2d+4}}$.
L'ω-transition $\mathbf{a}'$, définie par $\mathbf{Pre}(p, \mathbf{a}') = \mathbf{Pre}(p, \sigma_1'')$ pour tout $p \in P'$, $\mathbf{Pre}(p, \mathbf{a}') = \omega$ pour $p \in P \setminus P'$ et $\mathbf{C}(\mathbf{a}') = \mathbf{C}(\mathbf{a})$, est une accélération dont la famille associée est $\{\sigma_1''^n\}_{n \in \mathbb{N}}$. Par définition de $\mathbf{m}_1$, $\mathbf{a}' \leq \mathbf{a}$. Puisque $\mathbf{a}$ est minimale, $\mathbf{a}' = \mathbf{a}$. Puisque $|\sigma_1''| \leq (3de)^{(d+1)^{2d+4}}$, $||\mathbf{Pre}(\mathbf{a})||_\infty = ||\mathbf{Pre}(\mathbf{a}')||_\infty \leq e(3de)^{(d+1)^{2d+4}}$.

■

Proposition 23. *Soit $\mathcal{N}$ un RdP et* $\mathbf{a}$ *une accélération.*
Alors l'ω-transition $trunc(\mathbf{a})$ *définie par :*

- $\mathbf{C}(trunc(\mathbf{a})) = \mathbf{C}(\mathbf{a})$ *;*
- *pour tout p tel que* $\mathbf{Pre}(p, \mathbf{a}) \neq \omega$, $\mathbf{Pre}(p, trunc(\mathbf{a})) = \min(\mathbf{Pre}(p, \mathbf{a}), e(3de)^{(d+1)^{2d+4}})$ *;*
- *pour tout p tel que* $\mathbf{Pre}(p, \mathbf{a}) = \omega$, $\mathbf{Pre}(p, trunc(\mathbf{a})) = \omega$

est une accélération.

Preuve. Soit $\mathbf{a}' \leq \mathbf{a}$, une accélération minimale. Pour tout p tel que $\mathbf{Pre}(p, \mathbf{a}) \neq \omega$, $Pre(p, \mathbf{a}') \leq e(3de)^{(d+1)^{2d+4}}$. Par conséquent $\mathbf{a}' \leq trunc(\mathbf{a})$. Puisque l'ensemble des accélérations est clos supérieurement, on en déduit que $trunc(\mathbf{a})$ est une accélération. ■

Nous sommes maintenant en mesure de décrire l'amélioration apportée à la construction de Karp et Miller (cf l'algorithme 2). Tout d'abord lorsqu'on découvre une accélération, on la tronque avant de l'insérer dans Acc. D'autre part, lorsqu'un sommet de Front est choisi, on essaie préalablement de lui appliquer les accélérations de Acc pour augmenter son marquage.

Exemple. *La figure 3 montre l'arbre de Karp et Miller accéléré correspondant au réseau de la figure 1. Lorsque le sommet obtenu par franchissement de t_2 à partir du marquage initial est examiné, on évalue le franchissement de $\mathbf{a}_1$ et $\mathbf{a}_2$ découverts dans la branche gauche. L'accélération $\mathbf{a}_1$ est franchissable et elle est donc franchie.*

La preuve de terminaison est inchangée tandis que les preuves de consistance et de complétude ne nécessitent que des modifications très mineures pour intégrer le cas de l'application des accélérations. Du point de vue de la taille de l'arbre, le fait d'appliquer les accélérations mémorisées la fait décroître. On pourrait rétorquer qu'il faut conserver les accélérations. Cependant, en raison de la troncature, il y au plus un nombre doublement exponentiel d'accélérations chacune de taille simplement exponentielle : soit une complexité mémoire additionnelle doublement exponentielle. Il faut se rappeler ici que la taille de cet arbre de couverture est au pire cas

Algorithme 2 : Une accélération de l'algorithme de Karp et Miller

```
KarpMillerImproved(𝒩, m_0)
Input : Un RdP marqué (𝒩, m_0)
Data : V ensemble de sommets ; E ⊆ V × V ; Front ⊆ V ; λ : V → ℕ_ω^p ; δ : E → TAcc* ;
Tr = (V, E, λ, δ) un arbre étiqueté ; Acc un ensemble d'ω-transitions ;
u, u', u'' des sommets ; a une accélération ;
Output : Un arbre étiqueté Tr = (V, E, λ, δ)
1  V ← {r} ; E ← ∅ ; Front ← {r} ; λ(r) ← m_0 ; Acc ← ∅ ;
2  while Front ≠ ∅ do
3      Choisir u ∈ Front et soit u'' le prédécesseur de u
4      foreach a ∈ Acc t.q. λ(u) --a--> λ(u) + C(a) > λ(u) do
5          λ(u) ← λ(u) + C(a) ; δ((u'', u)) ← δ((u'', u))a
6      end
7      if ∃u' ∈ Anc(u) t.q. λ(u') ≥ λ(u) then
8          Front ← Front \ {u} ; V ← V \ {u} ; E ← E \ V × {u} // λ(u) est couvert
9      else if ∃u' ∈ Anc(u) t.q. λ(u') < λ(u) ∧ ∃p λ(u')(p) < λ(u)(p) < ω then
           // Une accélération est trouvée entre u et l'un des ancestres de u
10         Soit γ ∈ E* le chemin de u' à u dans Tr
11         a ← NewAcceleration()
12         foreach p ∈ P do
13             if C(p, δ(γ)) < 0 then Pre(p, a) ← ω ; C(p, a) ← ω
14             if C(p, δ(γ)) = 0 then Pre(p, a) ← Pre(p, δ(γ)) ; C(p, a) ← 0
15             if C(p, δ(γ)) > 0 then Pre(p, a) ← Pre(p, δ(γ)) ; C(p, a) ← ω ; λ(u)(p) ← ω
16         end
17         a ← trunc(a)
18         δ((u'', u)) ← δ((u'', u)) · a ; Acc ← Acc ∪ {a}
19     else
20         Front ← Front \ {u}
21         foreach t ∈ T t.q. λ(u) ≥ Pre(t) do
               // Ajout des fils de u
22             u' ← NewNode() ; V ← V ∪ {u'} ; Front ← Front ∪ {u'} ; E ← E ∪ {(u, u')}
23             λ(u') ← λ(u) + C(t) ; δ((u, u')) ← t
24         end
25     end
26 end
27 return Tr
```

non primitive récursive. Par conséquent l'augmentation de la taille mémoire est négligeable et par contre l'effet de ces accélérations peut réduire de manière considérable la taille de l'arbre de couverture et le temps pour le construire. Si cependant l'espace mémoire est une contrainte forte alors il suffit de conserver un sous-ensemble des accélérations car la preuve de l'algorithme modifié est valable pour tout sous-ensemble.

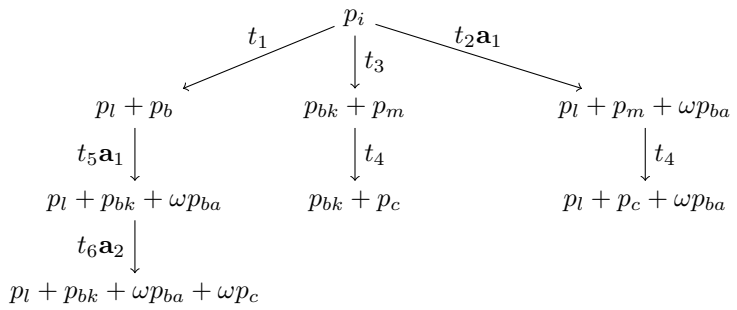

FIGURE 3 – Un arbre accéléré

5 Conclusion

L'étude de l'algorithme de Karp et Miller nous a conduits à deux résultats. Tout d'abord nous avons élaboré une preuve simple et élégante de cet algorithme basée sur les nouvelles notions d'abstraction, d'accélération et de séquence d'exploration. Puis nous avons conçu une version accélérée de cet algorithme qui mémorise toutes les accélérations calculées afin de les ré-appliquer systématiquement.

Nous prévoyons d'implémenter l'algorithme accéléré de Karp et Miller et une version optimisée, vis à vis de l'occupation mémoire, en conservant à chaque étape uniquement les ω-marquages formant une antichaîne. Nous comparerons aussi les performances de ces deux algorithmes avec tous les algorithmes existants calculant Clover. Enfin la notion d'accélération est orthogonale aux améliorations précédemment proposées. Par conséquent, nous développerons des versions combinant ces améliorations avec l'usage des accélérations.

En parallèle, nous étudierons aussi la possibilité de pré-calculer efficacement l'ensemble des accélérations minimales ou à défaut un sous-ensemble significatif.

Enfin il serait intéressant d'introduire et d'appliquer la notion d'accélération pour l'étude d'autres systèmes de transitions bien structurés.

Références

[1] Michael Blondin, Alain Finkel, and Pierre McKenzie. Well behaved transition systems. *Logical Methods in Computer Science*, 13(3) :1–19, September 2017.

[2] A. Finkel. Reduction and covering of infinite reachability trees. *Information and Computation*, 89(2) :144–179, 1990.

[3] A. Finkel. The minimal coverability graph for Petri nets. In *Advances in Petri Nets 1993*, volume 674 of *Lecture Notes in Computer Science*, pages 210–243. Springer, 1993.

[4] A. Finkel and J. Goubault-Larrecq. Forward analysis for WSTS, part II : Complete WSTS. *Logical Methods in Comp. Science*, 8(4), 2012.

[5] Alain Finkel, Gilles Geeraerts, Jean-François Raskin, and Laurent Van Begin. A counter-example the the minimal coverability tree algorithm. Technical Report 535, Université Libre de Bruxelles, Belgium, 2005.

[6] G. Geeraerts, J.-F. Raskin, and L. Van Begin. On the efficient computation of the minimal coverability set of Petri nets. *Int. J. Foundations of Computer Science*, 21(2) :135–165, 2010.

[7] Gilles Geeraerts, Jean-François Raskin, and Laurent Van Begin. On the efficient computation of the minimal coverability set for petri nets. In Kedar S. Namjoshi, Tomohiro Yoneda, Teruo Higashino, and Yoshio Okamura, editors, *Automated Technology for Verification and Analysis, 5th International Symposium, ATVA 2007, Tokyo, Japan, October 22-25, 2007, Proceedings*, volume 4762 of *Lecture Notes in Computer Science*, pages 98–113. Springer, 2007.

[8] Michel Hack. *Decidability questions for Petri Nets.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1976.

[9] R. M. Karp and R. E. Miller. Parallel program schemata. *Journal of Computer and System Sciences*, 3(2) :147–195, 1969.

[10] Jérôme Leroux. Distance Between Mutually Reachable Petri Net Configurations. working paper or preprint, June 2019.

[11] A. Piipponen and A. Valmari. Constructing minimal coverability sets. *Fundamenta Informaticae*, 143(3–4) :393–414, 2016.

[12] P.-A. Reynier and F. Servais. Minimal coverability set for Petri nets : Karp and Miller algorithm with pruning. *Fundamenta Informaticae*, 122(1–2) :1–30, 2013.

[13] Pierre-Alain Reynier and Frédéric Servais. Minimal coverability set for petri nets : Karp and miller algorithm with pruning. In Lars Michael Kristensen and Laure Petrucci, editors, *Applications and Theory of Petri Nets - 32nd International Conference, PETRI NETS 2011, Newcastle, UK, June 20-24, 2011. Proceedings*, volume 6709 of *Lecture Notes in Computer Science*, pages 69–88. Springer, 2011.

[14] A. Valmari and H. Hansen. Old and new algorithms for minimal coverability sets. *Fundamenta Informaticae*, 131(1) :1–25, 2014.

[15] Mitsuharu Yamamoto, Shogo Sekine, and Saki Matsumoto. Formalization of karp-miller tree construction on petri nets. In *Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs*, CPP 2017, pages 66–78, New York, NY, USA, 2017. ACM.