

L'Algorithme de Mazurkiewicz

1 Modèle

Dans le modèle considéré, un pas de calcul permet à un sommet de modifier son état et celui de ses voisins en fonction de son propre état et de l'état de ses voisins. Dans ce modèle, un algorithme peut être décrit par un ensemble de règles de réétiquetage dont le support est une étoile. Ces règles sont de la forme de la règle présentée sur la Figure 1.

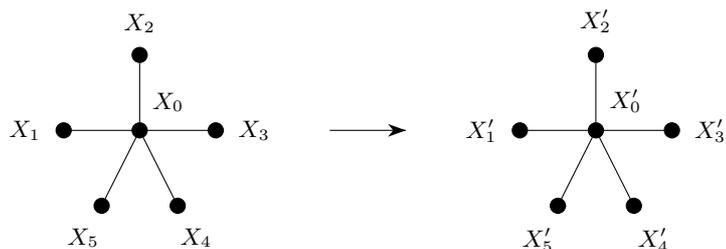


Fig. 1. La forme d'une règle de réétiquetage dans le modèle de Mazurkiewicz.

Cependant, comme pour l'algorithme d'élection dans les arbres, on utilisera des règles décrites de manière générique.

2 Revêtements

Définition 1 Un homomorphisme φ d'un graphe G dans un graphe H est une application de $V(G)$ dans $V(H)$ qui préserve les relations d'adjacence entre sommets, i.e., pour toute arête $\{v, w\} \in E(G)$, $\{\varphi(v), \varphi(w)\} \in E(H)$.

Un exemple d'homomorphisme est présenté sur la Figure 2.

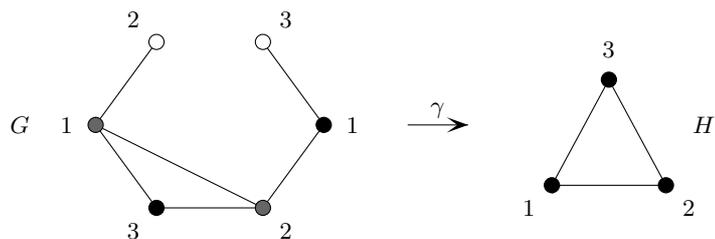


Fig. 2. La fonction γ qui envoie chaque sommet de G étiqueté i sur l'unique sommet de H dont l'étiquette est i est un homomorphisme de G dans H .

Définition 2 Un homomorphisme φ d'un graphe G dans un graphe H est un isomorphisme si φ est une fonction bijective et si φ^{-1} est aussi un isomorphisme.

Les homomorphismes localement bijectifs, i.e., les revêtements, sont les homomorphismes qui permettent de donner des conditions nécessaires que doivent vérifier les graphes admettant un algorithme d'élection dans le modèle de Mazurkiewicz.

Définition 3 Un graphe G est un revêtement d'un graphe H à travers un homomorphisme $\gamma: G \rightarrow H$ si pour tout sommet $v \in V(G)$, γ induit une bijection entre $N_G(u)$ et $N_H(\gamma(u))$, i.e., si les conditions suivantes sont vérifiées :

- $|N_G(u)| = |N_H(\gamma(u))|$,
- $\gamma(N_G(u)) = N_H(\gamma(u))$.

On dit alors que l'homomorphisme γ est localement bijectif

Un graphe G est un revêtement propre de H si γ n'est pas un isomorphisme et G est minimal pour les revêtements si G n'est un revêtement propre d'aucun autre graphe.

Exemple 4 Le graphe G de la Figure 3 est un revêtement de H à travers l'homomorphisme γ .

Le graphe G est un revêtement propre de H et n'est donc pas minimal pour les revêtements; le graphe H est minimal pour les revêtements.

Le graphe G de la Figure 2 n'est pas un revêtement de H à travers γ , car les sommets blancs et gris de G n'ont pas autant de voisins que leurs images (les blancs en ont moins, les gris en ont plus).

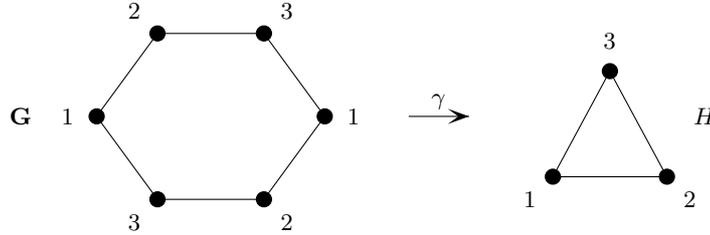


Fig. 3. Le graphe G est un revêtement de H à travers l'homomorphisme γ qui envoie chaque sommet de G étiqueté i sur l'unique sommet de H dont l'étiquette est i .

Dans le lemme suivant, on montre que si G est un revêtement de H , pour tout sommet v de $V(H)$, l'image inverse de l'étoile de H centrée en v est une union disjointe d'étoiles de G .

Lemme 5 On considère des graphes G, H tels que G est un revêtement de H à travers un homomorphisme γ . Soit v un sommet de $V(H)$ et soient u_1, u_2 deux sommets distincts de $\gamma^{-1}(v)$. Pour tous sommets $u'_1 \in N_G(u_1) \cup \{u_1\}$ et $u'_2 \in N_G(u_2) \cup \{u_2\}$, $u'_1 \neq u'_2$.

Preuve : On considère des graphes G, H tels que G est un revêtement de H à travers un homomorphisme γ . Soit v un sommet de $V(H)$ et soient u_1, u_2 deux sommets de $\gamma^{-1}(v)$. Puisque H est un graphe, on sait que $u_2 \notin N_G(u_1)$. S'il existe $u \in N_G(u_1) \cap N_G(u_2)$, alors γ n'est pas localement bijectif en u puisque u a deux voisins qui ont la même image par γ . \square

Puisqu'on ne considère que des graphes connexes, tout homomorphisme localement bijectif est surjectif.

Proposition 6 Si un graphe G est un revêtement d'un graphe connexe H à travers γ , alors γ est surjectif.

Preuve : Soit G un revêtement d'un graphe connexe H à travers un homomorphisme γ .

On considère un sommet $v \in \gamma(V(G))$. Il existe donc $u \in V(G)$ tel que $\gamma(u) = v$. Puisque γ induit une bijection entre $N_G(u)$ et $N_H(v)$, pour tout sommet $v' \in N_H(v)$, il existe un sommet $u' \in N_G(u)$ telle que

$\gamma(u') = v'$ et donc $v' \in \gamma(V(G))$. Ainsi, puisque H est connexe, on sait que γ est un homomorphisme surjectif de G dans H . \square

Si un graphe G est un revêtement d'un graphe H connexe, alors tous les sommets de H ont le même nombre d'antécédents, qui est appelé le *nombre de feuillets* du revêtement.

Proposition 7 *Si un graphe G est un revêtement d'un graphe connexe H à travers γ , alors il existe une constante q telle que pour tout $v \in V(H)$, $|\gamma^{-1}(v)| = q$.*

Cette constante q est appelée le nombre de feuillets du revêtement.

Preuve : On considère un graphe G qui est un revêtement d'un graphe connexe H à travers un homomorphisme γ . On considère un sommet $v \in V(H)$ et un sommet $v' \in N_H(v)$.

Puisque γ est un homomorphisme localement bijectif, pour tout sommet $u \in \gamma^{-1}(v)$, il existe un unique sommet $u' \in N_G(u)$ tel que $\gamma(u') = v'$. De plus, puisque γ est localement bijectif en u' , pour tout sommet $u'' \in N_G(u')$ différent de u , $\gamma(u'') \neq \gamma(u)$. Par conséquent, $|\gamma^{-1}(v)| \leq |\gamma^{-1}(v')|$ et par symétrie, $|\gamma^{-1}(v)| = |\gamma^{-1}(v')|$. Ainsi, puisque le graphe H est connexe, pour tout $v, v' \in V(H)$, $|\gamma^{-1}(v)| = |\gamma^{-1}(v')|$. \square

Exemple 8 *Voici quelques exemples de graphes connexes qui sont minimaux pour les revêtements.*

- les graphes dont le nombre d'arêtes et le nombre de sommets sont premiers entre eux,
- les arbres,
- les anneaux de taille première.

3 Résultat d'impossibilité

On présente maintenant le lemme qui met en évidence le lien entre les calculs locaux et les revêtements. Ce lemme a été prouvé par Angluin en 1980.

On dit qu'un graphe étiqueté $\mathbf{G} = (G, \lambda)$ est un revêtement de $\mathbf{H} = (H, \eta)$ à travers γ si G est un revêtement de H à travers γ et si γ préserve l'étiquetage, i.e., pour tout $v \in V(G)$, $\lambda(v) = \eta(\gamma(v))$.

Lemme 9 (Lemme de relèvement) *On considère un graphe \mathbf{G} qui est un revêtement d'un graphe \mathbf{H} à travers un homomorphisme γ et un algorithme \mathcal{A} utilisant des calculs locaux. S'il existe une exécution de \mathcal{A} sur \mathbf{H} qui permet d'atteindre une configuration \mathbf{H}' , alors il existe une exécution de \mathcal{A} sur \mathbf{G} qui permet d'atteindre une configuration \mathbf{G}' tel que \mathbf{G}' est un revêtement de \mathbf{H}' à travers γ .*

Preuve : Il suffit de prouver ce lemme pour un pas de calcul. On considère deux graphes (G, λ) et (H, η) tels que (G, λ) est un revêtement de (H, η) à travers γ . On considère un pas de réétiquetage qui modifie les étiquettes d'une étoile $B_H(v)$ pour un sommet $v \in V(H)$. On note η' l'étiquetage de H obtenu après l'application de ce pas de réétiquetage.

Puisque γ est un homomorphisme localement bijectif, on sait que pour tout sommet $u \in \gamma^{-1}(v)$, $(B_G(u), \lambda)$ est isomorphe à $(B_H(v), \eta)$. De plus, d'après le Lemme 5, on sait que pour tous $u, u' \in \gamma^{-1}(v)$, les étoiles $B_G(u)$ et $B_G(u')$ sont disjointes. On peut donc appliquer le pas de réétiquetage sur chacune des étoiles $B_G(u)$ pour tout $u \in \gamma^{-1}(v)$ de telle sorte que tout sommet $u' \in V(B_G(u))$ soit réétiqueté avec la même étiquette que $\gamma(u')$. On note λ' l'étiquetage de G obtenu après l'application de tous ces réétiquetages. Le graphe (G, λ') ainsi obtenu est un revêtement de (H, η') à travers γ . \square

On déduit de ce lemme de relèvement le résultat d'impossibilité suivant qui a aussi été prouvé par Angluin en 1980.

Proposition 10 *Soit G un graphe qui n'est pas minimal pour les revêtements. Il n'existe pas d'algorithme d'élection pour le graphe G utilisant des calculs locaux.*

Preuve : On considère un graphe H qui n'est pas isomorphe à G et tel que G soit un revêtement de H à travers un homomorphisme γ . Étant donné un algorithme \mathcal{A} utilisant des calculs locaux sur les étoiles fermées, on considère une exécution de \mathcal{A} sur H . Si cette exécution est infinie, alors d'après le Lemme 9, il existe une exécution infinie de \mathcal{A} sur G ; auquel cas, \mathcal{A} n'est pas un algorithme d'élection.

On suppose maintenant qu'il existe une exécution finie de \mathcal{A} sur H et on considère l'étiquetage final η' de H . D'après le Lemme 9, il existe une exécution de \mathcal{A} sur G qui permet d'atteindre une configuration λ' de G' telle que $\mathbf{G}' = (G, \lambda')$ est un pseudo-revêtement de $\mathbf{H}' = (H, \eta')$ à travers γ . Si \mathbf{G}' n'est pas une configuration finale de \mathcal{A} (autrement dit, si on peut encore appliquer un pas de calcul), alors il existe un sommet $u \in V(G)$ tel qu'on puisse appliquer une règle de réétiquetage sur l'étoile $B_G(u)$. Dans ce cas là, on peut aussi appliquer cette même règle de réétiquetage sur l'étoile $B_H(\gamma(u))$ et la configuration \mathbf{H}' n'est pas une configuration finale de \mathcal{A} . Par conséquent, \mathbf{G}' est une configuration finale de \mathcal{A} . Mais puisque \mathbf{G}' n'est pas isomorphe à \mathbf{H}' , on sait d'après la Proposition 7 que chaque étiquette de \mathbf{H}' apparaît au moins deux fois dans \mathbf{G}' et par conséquent, \mathcal{A} n'est pas un algorithme d'élection. \square

4 Algorithme d'énumération

On va maintenant décrire l'algorithme d'énumération \mathcal{M} de Mazurkiewicz qui permet de résoudre le problème de l'énumération sur les graphes minimaux pour les revêtements.

Durant l'exécution de l'algorithme, chaque sommet v essaie d'obtenir une identité qui est un numéro entre 1 et $|V(G)|$. À chaque fois qu'un sommet modifie son numéro, il en informe immédiatement ses voisins. Chaque sommet v peut donc tenir à jour la liste des numéros de ses voisins, qui sera appelée la *vue locale* de v . Lorsqu'un sommet v modifie son numéro ou sa vue locale, il diffuse dans le réseau son numéro accompagné de son étiquette initiale et de sa vue locale.

Si un sommet u découvre qu'un autre sommet v a le même numéro que lui, alors le sommet u doit décider s'il modifie son identité. Pour cela, il compare son étiquette $\lambda(u)$ et sa vue locale avec l'étiquette $\lambda(v)$ et la vue locale de v : si l'étiquette de u est plus faible que l'étiquette de v ou si les deux sommets ont la même étiquette et que la vue locale de u est plus « faible » (pour un ordre qu'on expliquera par la suite), alors le sommet u choisit un nouveau numéro (sa nouvelle identité temporaire) et informe ses voisins de ce changement de numéro. Ensuite, les numéros et les vues locales qui ont été modifiées sont diffusés à nouveau dans le graphe. Lorsque l'exécution est terminée, si le graphe \mathbf{G} est minimal pour les revêtements, alors chaque sommet a un numéro unique.

Étiquettes On considère un graphe G . Lors de l'exécution, chaque sommet va obtenir une étiquette de la forme $(n(v), N(v), M(v))$ qui représente les informations suivantes :

- $n(v) \in \mathbb{N}$ est le *numéro* courant du sommet v qui est modifié lors de l'exécution de l'algorithme,
- $N(v) \in \mathcal{P}_{\text{fin}}(\mathbb{N})$ ($\mathcal{P}_{\text{fin}}(\mathbb{N})$ est l'ensemble des ensembles finis d'entiers.) est la *vue locale* du sommet v qui contient des informations sur les voisins de v ; c'est un ensemble fini d'entiers. À tout moment de l'exécution, pour chaque voisin v' de v tel que $n(v') \neq 0$, la vue locale de v contient $n(v')$.
- $M(v) \subseteq \mathbb{N} \times \mathcal{P}_{\text{fin}}(\mathbb{N})$ est la *boîte-aux-lettres* de v . Elle va contenir toute l'information reçue par v lors de l'exécution de l'algorithme, i.e., les couples de numéros et de vues locales qui auront été diffusées par tous les sommets du graphe.

Initialement, chaque sommet a une étiquette de la forme $(0, \emptyset, \emptyset)$ qui signifie qu'au début de l'algorithme, v n'a pas choisi de numéro et qu'il n'a aucune information à propos de ses voisins, ni à propos des autres sommets du graphe.

Un Ordre sur les Vues Locales Les bonnes propriétés de l'algorithme de Mazurkiewicz sont basées sur un ordre total sur les vues locales. Cet ordre permet à un sommet u de modifier son numéro s'il découvre qu'il existe un autre sommet avec le même numéro, la même étiquette et une vue locale « plus forte ». Afin d'éviter des exécutions infinies, il faut que lorsque la vue locale d'un sommet est modifiée, elle ne puisse pas

devenir plus faible, et ce pour éviter qu'un sommet ne modifie son numéro à cause d'un message qu'il ait lui-même envoyé lors d'une étape précédente.

Ensuite, étant données deux ensembles $N_1, N_2 \in \mathcal{P}_{\text{fin}}(\mathbb{N})$ distincts, on dit que $N_1 \prec N_2$ si le maximum de la différence symétrique $N_1 \Delta N_2 = (N_1 \setminus N_2) \cup (N_2 \setminus N_1)$ appartient à N_2 .

On peut aussi voir cet ordre comme l'ordre lexicographique usuel sur les ensembles ordonnés N_1 et N_2 . On note n_1, n_2, \dots, n_k et n'_1, n'_2, \dots, n'_l les éléments respectifs de N_1 et de N_2 dans l'ordre décroissant : $n_1 \geq n_2 \geq \dots \geq n_k$ et $n'_1 \geq n'_2 \geq \dots \geq n'_l$. Alors $N_1 \prec N_2$ si l'une des conditions suivantes est vérifiée :

- $k < l$ et pour tout $i \in [1, k]$, $n_i = n'_i$,
- $n_i < n'_i$ où i est le plus petit indice pour lequel $n_i \neq n'_i$.

Si $N(u) \prec N(v)$, alors on dit que la vue locale $N(v)$ de v est *plus forte* que celle de u et que $N(u)$ est *plus faible* que $N(v)$. On note $N(u) \preceq N(v)$ lorsque $N(u) = N(v)$ ou $N(u) \prec N(v)$.

Les Règles de Réétiquetage On décrit maintenant l'algorithme d'énumération grâce à des règles de réétiquetage.

Les règles sont décrites pour une étoile $B(v_0)$ de centre v_0 . L'étiquette d'un sommet $v \in N_G(v_0) \cup \{v_0\}$ avant l'application de la règle est notée $(n(v), N(v), M(v))$ et on note $(n'(v), N'(v), M'(v))$ l'étiquette de v après l'application de la règle de réétiquetage. Par ailleurs, afin de rendre plus lisible les règles, on ne mentionne pas les différents champs des étiquettes qui ne sont pas modifiés.

La première règle permet aux sommets d'une même étoile d'échanger les informations dont ils disposent (i.e., contenues dans leur boîte-aux-lettres) à propos des étiquettes présentes dans le graphe. Cette règle ne peut être appliquée que si un tel échange d'information est nécessaire (i.e., tous les sommets de $B(v_0)$ n'ont pas la même boîte-aux-lettres).

\mathcal{M}_1 : **Règle de Diffusion**

Précondition :

- $\exists v \in N_G(v_0)$ tel que $M(v) \neq M(v_0)$.

Réétiquetage :

- $\forall v \in V(B(v_0)), M'(v) := \bigcup_{w \in V(B(v_0))} M(w)$.

La deuxième règle permet à un sommet v_0 de changer de numéro s'il n'a pas encore de numéro (i.e., $n(v_0) = 0$) ou s'il sait qu'il existe un sommet dans le graphe qui a le même numéro que lui et qui a une vue locale plus forte que la sienne. Dans ce cas là, v_0 choisit un nouveau numéro et modifie la vue locale de ses voisins. Toutes les boîtes-aux-lettres des sommets de l'étoile $B(v_0)$ sont modifiées : on ajoute à chaque boîte-aux-lettres tous les couples $(n'(v), N'(v))$ pour tous les sommets v' de l'étoile $B(v_0)$.

\mathcal{M}_2 : **Règle de Renommage**

Précondition :

- $\forall v \in N_G(v_0), M(v) = M(v_0)$,
- $n(v_0) = 0$ ou
- $\exists (n(v_0), N) \in M(v_0)$ tel que $N(v_0) \prec N$

Réétiquetage :

- $n'(v_0) := 1 + \max\{n' \mid \exists (n', N') \in M(v_0)\}$;
- $\forall v \in N_G(v_0), N'(v) := N(v) \setminus \{n(v_0)\} \cup \{n'(v_0)\}$;
- $\forall v \in V(B(v_0)), M'(v) := M(v) \cup \bigcup_{w \in V(B(v_0))} \{(n'(w), N'(w))\}$.

4.1 Correction de l'Algorithme d'Énumération

On considère un graphe G . Pour tout sommet $v \in V(G)$, on note $(n_i(v), N_i(v), M_i(v))$ l'étiquette du sommet v après la i ème étape de réétiquetage de l'algorithme \mathcal{M} décrit ci-dessus. On présente d'abord quelques propriétés qui sont satisfaites par n'importe quelle exécution de l'algorithme.

Propriétés Satisfaites lors de l'Exécution Le lemme suivant, qui peut être facilement prouvé par une récurrence sur le nombre d'étapes, rappelle quelques propriétés qui sont toujours satisfaites par l'étiquetage.

Lemme 11 *Pour tout sommet $v \in V(G')$, et pour toute étape i ,*

1. $n_i(v) \neq 0 \implies (n_i(v), N_i(v)) \in M_i(v)$,
2. $\exists n \in N_i(v) \iff \exists v' \in N_G(v)$ tel que $n_i(v') = n > 0$,
3. $\forall n \in N_i(v), n \neq n_i(v)$ et $\exists(n, N) \in M_i(v)$,
4. $\forall v', v'' \in N_G(v), n_i(v), n_i(v') > 0 \implies n_i(v') \neq n_i(v'')$.

L'algorithme \mathcal{M} a des propriétés de monotonie intéressantes qui sont données dans le lemme suivant.

Lemme 12 *Pour chaque sommet v et chaque étape i ,*

- $n_i(v) \leq n_{i+1}(v)$,
- $N_i(v) \preceq N_{i+1}(v)$,
- $M_i(v) \subseteq M_{i+1}(v)$.

De plus, à chaque étape i , il existe un sommet v telle qu'au moins une de ces inégalités (ou inclusions) est stricte pour v .

Preuve : La propriété est trivialement vraie pour les sommets qui ne sont pas réétiquetés lors de la $(i+1)$ ème étape. De plus, il est facile de voir que quelque soit la règle appliquée à l'étape $i+1$, on a toujours $M_i(v) \subseteq M_{i+1}(v)$ pour tout sommet $v \in V(G)$.

Pour chaque sommet v tel que $n_i(v) \neq n_{i+1}(v)$, alors la règle \mathcal{M}_2 a été appliquée sur l'étoile $B(v)$ et on sait que $n_{i+1}(v) = 1 + \max\{n' \mid \exists(n', N') \in M_i(v)\}$. De plus, ou bien $n_i(v) = 0 < n_{i+1}(v)$, ou alors d'après le Lemme 11, $(n_i(v), N_i(v)) \in M_i(v)$ et donc $n_i(v) < n_{i+1}(v)$.

Pour chaque sommet v tel que $N_i(v) \neq N_{i+1}(v)$, alors la règle \mathcal{M}_2 a été appliquée sur une étoile $B(v_0)$ telle que $v_0 \in N_G(v)$. On sait que $N_{i+1}(v) = N_i(v) \setminus \{n_i(v_0)\} \cup \{n_{i+1}(v_0)\}$ et que pour tout $(n, N) \in M_i(v)$, $n_{i+1}(v_0) > n$. Ainsi, on a $\max N_{i+1}(v) \triangle N_i(v) = n_{i+1}(v_0) \in N_{i+1}(v)$. Par conséquent, $N_i(v) \prec N_{i+1}(v)$.

Puisque chaque application d'une règle modifie l'étiquette d'au moins un sommet v , on sait que l'une de ces inégalités est stricte pour v . \square

Les informations dont dispose chaque sommet v dans sa boîte-aux-lettres permettent d'obtenir des informations vérifiées par la configuration globale du graphe. Les deux lemmes suivants permettent de prouver que si un sommet v connaît un numéro m à une étape i (i.e., il existe N tels que $(m, N) \in M_i(v)$), alors pour chaque $m' \leq m$, il existe un sommet w tel que $n_i(w) = m'$. On montre d'abord que si v connaît un numéro m , alors il existe un sommet w tel que $n_i(w) = m$.

Lemme 13 *Pour chaque sommet $v \in V(G)$ et chaque étape i , pour tout $(m, N) \in M_i(v)$, il existe un sommet $w \in V(G)$ tel que $n_i(w) = m$.*

Preuve : On remarque d'abord qu'un élément (m, N) est ajouté à une étape i dans $\bigcup_{v \in V(G)} M_i(v)$ seulement s'il existe un sommet v tel que $n_i(v) = m$ et $N_i(v) = N$.

Étant donné un sommet v , une étape i et un couple $(m, N) \in M_i(v)$, on note $U = \{(u, j) \in V(G) \times \mathbb{N} \mid j \leq i, n_j(u) = m\}$. On considère ensuite l'ensemble $U' = \{(u, j) \in U \mid \forall(u', j') \in U, N_{j'}(u') \prec N_j(u) \text{ ou } N_{j'}(u') = N_j(u) \text{ et } j' \leq j\}$. Puisque $(m, N) \in M_i(v)$, U et U' sont deux ensembles non-vides. On remarque aisément qu'il existe i_0 tel que pour tout $(u, j) \in U'$, $j = i_0$.

Si $i_0 < i$, il existe exactement un élément $(u, i_0) \in U'$ puisqu'à chaque étape, le numéro d'au plus un sommet peut être modifié. Le numéro $n_{i_0}(u) = m$ a donc été modifié à l'étape $i_0 + 1$, mais par maximalité de $N_{i_0}(u)$, la règle \mathcal{M}_2 n'a pas pu être appliquée à u à l'étape i_0 . Par conséquent, $i_0 = i$ et il existe donc un sommet w tel que $n_i(w) = m$. \square

Dans le lemme suivant, on montre que si un sommet v connaît un numéro m , alors il connaît tous les numéros inférieurs à m .

Lemme 14 *Pour chaque sommet v et chaque étape i , pour tout $(m, N) \in M_i(v)$, pour tout $m' \in [1, m]$, il existe $(m', N') \in M_i(v)$.*

Preuve : On montre ce lemme par récurrence sur i . Initialement, la propriété est trivialement vraie. On suppose que la propriété est vérifiée pour $i \geq 0$. La propriété est trivialement vraie à l'étape $i + 1$ pour tout sommet $w \in V(G)$ dont l'étiquette n'est pas modifiée à l'étape $i + 1$. Soit v un sommet dont l'étiquette est modifiée à l'étape $i + 1$.

Si la règle \mathcal{M}_1 a été appliquée à l'étape $i + 1$, alors pour tout $(m, N) \in M_{i+1}(v)$, il existe v' tel que $(m, N) \in M_i(v')$ et $M_i(v') \subseteq M_{i+1}(v)$. Par conséquent, par hypothèse de récurrence, pour tout $m' < m$, il existe $(m', N') \in M_i(v') \subseteq M_{i+1}(v)$. On suppose maintenant que la règle \mathcal{M}_2 a été appliquée à l'étape $i + 1$ sur une étoile $B(v_0)$ qui contient v . Soit $(m, N) \in M_{i+1}(v) \setminus M_i(v)$. Si $m = n_{i+1}(v_0) = 1 + \max\{m' \mid \exists(m', N') \in M_i(v)\}$, pour tout $m' < m$, il existe $(m', N') \in M_i(v) \subseteq M_{i+1}(v)$. Si $m \neq n_{i+1}(v_0)$, alors il existe $v' \in N_G(v_0)$ tel que $m = n_i(v') = n_{i+1}(v')$ et d'après le Lemme 11, on sait qu'il existe $(m, N') \in M_i(v') = M_i(v)$. Ainsi la propriété est vérifiée à l'étape $i + 1$. \square

On veut maintenant montrer que toute exécution de l'algorithme \mathcal{M} termine sur \mathbf{G} . D'après les Lemmes 13 et 14, on voit qu'à chaque étape de l'exécution, les numéros des sommets forment un ensemble $[1, k]$ ou un ensemble $[0, k]$ avec $k \leq |V(G)|$. Par conséquent, d'après le Lemme 12, on sait qu'il existe une étape i_0 telle que pour tout sommet v et toute étape $i \geq i_0$, $n_{i+1}(v) = n_i(v)$.

De plus, pour chaque sommet v et chaque étape i , si $N_i(v)$ contient un couple n' , alors il existe $v' \in N_G(v)$ tel que $n_i(v') = n'$. Par conséquent $N(v)$ ne peut prendre qu'un nombre fini de valeurs et il en est de même pour $M(v)$. Ainsi le nombre de valeurs différentes que peut prendre l'étiquette de chaque sommet est fini (mais dépend de la taille du graphe). Par ailleurs, les étiquettes consécutives de chaque sommet v forment une suite croissante et puisqu'à chaque étape i , l'étiquette d'au moins un sommet est modifiée, toute exécution de l'algorithme termine.

Propriétés Satisfaites par l'Étiquetage Final Puisqu'on sait que l'algorithme termine toujours, on s'intéresse maintenant aux propriétés satisfaites par l'étiquetage final.

Lemme 15 *Toute exécution ρ de l'algorithme \mathcal{M} sur un graphe G termine et l'étiquetage final (n_ρ, N_ρ, M_ρ) vérifie les propriétés suivantes :*

1. *il existe un entier $k \leq |V(G)|$ tel que $\{n_\rho(v) \mid v \in V(G)\} = [1, k]$,*

et pour tous sommets v, v' :

2. $M_\rho(v) = M_\rho(v')$,

3. $(n_\rho(v), N_\rho(v)) \in M_\rho(v')$,

4. *si $n_\rho(v) = n_\rho(v')$, alors $N_\rho(v) = N_\rho(v')$,*

5. $\forall w, w' \in N_G(v), n_\rho(w) \neq n_\rho(w')$,

6. $n \in N_\rho(v)$ *si et seulement si il existe $w \in N_G(v)$ tel que $n_\rho(w) = n$; auquel cas, $n_\rho(v) \in N_\rho(w)$.*

Preuve :

1. D'après les Lemmes 13 et 14 et puisque la règle \mathcal{M}_2 ne peut pas être appliquée.

2. Dans le cas contraire, la règle \mathcal{M}_1 peut être appliquée.

3. C'est une conséquence directe de la propriété précédente d'après le Lemme 11.

4. Dans le cas contraire, la règle \mathcal{M}_2 peut être appliquée à v ou à v' .

5. D'après le Lemme 11 et puisque la règle \mathcal{M}_2 ne peut pas être appliquée.

6. D'après le Lemme 11. \square

Grâce au Lemme 15, on peut prouver que l'étiquetage final permet de construire un graphe H tel que G est un revêtement de H .

Proposition 16 *Étant donné un graphe G , on peut construire, à partir de l'étiquetage final obtenu après une exécution ρ de \mathcal{M} , un graphe H tel qu'il existe un homomorphisme localement bijectif de G dans H .*

Preuve : On utilise les notations du Lemme 15.

On considère le graphe H défini par $V(H) = \{m \in \mathbb{N} \mid \exists v \in V(G), n_\rho(v) = m\}$ et $E(H) = \{\{m, m'\} \mid \exists v, v' \in V(G); n_\rho(v) = m, n_\rho(v') = m' \text{ et } \{v, v'\} \in E(G)\}$. D'après le Lemme 11, on sait qu'il n'existe pas d'arête $\{v, v'\} \in E(G)$ telle que $n_\rho(v) = n_\rho(v')$: le graphe H ne contient donc pas de boucle. De plus, de par la définition de $E(H)$, on sait que H ne contient pas d'arêtes multiples.

On considère maintenant la fonction $n_\rho : V(G) \rightarrow V(H)$. Par définition de H , on sait que si $\{v, v'\} \in E(G)$, alors $\{n_\rho(v), n_\rho(v')\} \in E(H)$. Par conséquent, n_ρ est un homomorphisme de G dans H .

D'après le Lemme 15, pour tout sommet v , si $m \in N_H(n_\rho(v))$, alors il existe $w \in N_G(v)$ tel que $n_\rho(w) = m$ et par conséquent, $n_\rho(N_G(v)) = N_H(n_\rho(v))$. De plus, pour tous $w, w' \in N_G(v)$, $n_\rho(w) \neq n_\rho(w')$ et on a donc $|N_G(v)| = |N_H(n_\rho(v))|$. Ainsi, l'homomorphisme n_ρ est localement bijectif et G est un revêtement de H à travers n_ρ . \square

On considère maintenant un graphe G qui est minimal pour les revêtements. Pour chaque exécution ρ de \mathcal{M} sur G , le graphe obtenu à partir de l'étiquetage final est isomorphe à G . Par conséquent, l'ensemble des numéros des sommets est exactement $[1, |V(G)|]$: chaque sommet a un identifiant unique.

De plus, une fois qu'un sommet a obtenu le numéro $|V(G)|$, d'après les Lemmes 13 et 14, il sait que tous les sommets de G ont un numéro unique qui ne va plus être modifié. Dans ce cas là, ce sommet peut prendre l'étiquette ÉLU et diffuser ensuite l'information qu'un sommet a été élu. On a donc obtenu un algorithme d'élection pour le graphe G .

Par ailleurs, d'après la Proposition 10, on sait que pour tout graphe G qui n'est pas minimal pour les revêtements, il n'existe aucun algorithme d'élection utilisant des calculs locaux pour le graphe G . On a donc prouvé le théorème suivant.

Théorème 17 *Pour tout graphe G , il existe un algorithme d'élection pour G si et seulement si G est minimal pour les revêtements.*