

Algorithmes d'élection dans les arbres par calculs locaux

1 Calculs locaux sur les étoiles

On décrit l'algorithme d'élection dans les arbres anonymes par calculs locaux en utilisant des règles « génériques ». Une règle générique permet de décrire la règle de réétiquetage d'une étoile, quel que soit le degré du centre de l'étoile. En général, on considère une boule générique $(B(v_0), \lambda)$ de centre v_0 et la règle est décrite par une précondition portant sur les étiquettes présentes dans $(B(v_0), \lambda)$ et un réétiquetage $(B(v_0), \lambda')$. La règle peut être appliquée dans un graphe étiqueté $\mathbf{G} = (G, \lambda)$ sur une étoile $B_G(u)$ de centre u si la précondition est vérifiée par $B_G(u)$ et les étiquettes des sommets de $B_G(u)$ sont alors modifiées en fonction du réétiquetage λ' . En général, on ne mentionne pas dans le réétiquetage les étiquettes des sommets qui ne sont pas modifiées.

L'algorithme est décrit par les deux règles \mathcal{E}_1 et \mathcal{E}_2 présentées ci-dessous. Cet algorithme est un algorithme d'élection pour les arbres où tous les sommets sont initialement étiquetés A .

\mathcal{E}_1 : Règle d'Élagage

Précondition :

- $\lambda(v_0) = A$,
- $\exists! v \in V(B(v_0)) \setminus \{v_0\}$ tel que $\lambda(v) = A$.

Réétiquetage :

- $\lambda'(v_0) := \text{NON-ÉLU}$.

\mathcal{E}_2 : Règle d'Élection

Précondition :

- $\lambda(v_0) = A$,
- $\nexists v \in V(B(v_0)) \setminus \{v_0\}$ tel que $\lambda(v) = A$.

Réétiquetage :

- $\lambda'(v_0) := \text{ÉLU}$.

La règle \mathcal{E}_1 peut être appliquée dans un graphe \mathbf{G} sur une étoile $B_G(u)$ de centre u si l'étiquette de u est A et si u a un unique voisin étiqueté A . Lorsque cette règle est appliquée, seule l'étiquette de u est modifiée et devient NON-ÉLU.

La règle \mathcal{E}_2 peut être appliquée dans un graphe \mathbf{G} sur une étoile $B_G(u)$ de centre u si l'étiquette de u est A et si u n'a aucun voisin étiqueté A . Lorsque cette règle est appliquée, seule l'étiquette de u est modifiée et devient ÉLU.

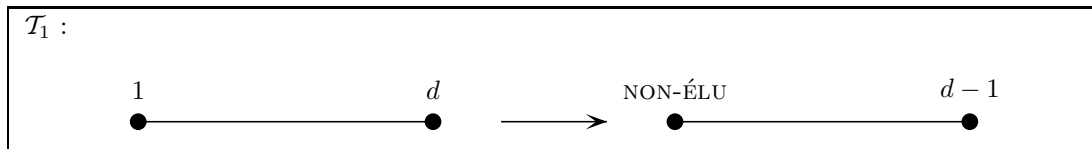
Une exécution de cet algorithme est présentée sur la Figure 1. On va montrer que pour tout arbre $\mathbf{T} = (T, \lambda)$ où tous les sommets sont étiquetés A (i.e., $\forall v \in V(T), \lambda(v) = A$), toute exécution de l'algorithme décrit par les règles \mathcal{E}_1 et \mathcal{E}_2 permet de résoudre l'élection dans \mathbf{T} .

On considère un arbre \mathbf{T} et une exécution de l'algorithme décrit précédemment sur \mathbf{T} . Pour chaque étape i de l'exécution, on note $\lambda(v)$ l'étiquette du sommet v après le i ème pas de réétiquetage.

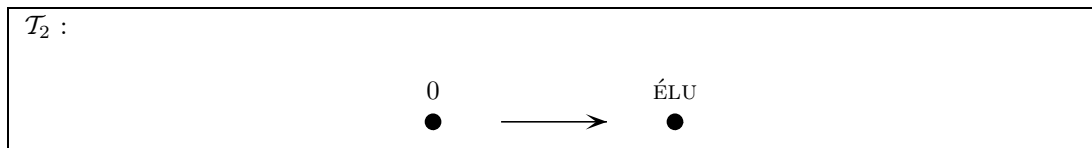
On observe qu'à chaque application d'une des deux règles, le nombre de sommets qui n'ont pas d'étiquettes finales diminue strictement. On est donc assuré que toute exécution de l'algorithme termine. On montre dans le lemme suivant un invariant qui permet de prouver la correction de l'algorithme.

Lemma 1. *Pour toute étape i , le graphe induit par l'ensemble des sommets étiquetés A est un arbre et s'il existe un sommet étiqueté A alors aucun sommet n'a l'étiquette ÉLU.*

Démonstration. On montre ce lemme par récurrence sur i . Initialement, tous les sommets sont étiquetés A et puisque \mathbf{T} est un arbre, la propriété est bien vérifiée. On suppose que la propriété est vérifiée à l'étape i .



La seconde règle permet à un sommet qui n'a plus aucun voisin actif dans l'arbre de prendre l'étiquette ÉLU.



La preuve de l'algorithme est la même que précédemment, en ajoutant comme invariant que l'étiquette d'un sommet est son nombre de voisins « actifs » (dont l'étiquette n'est pas ÉLU ou NON-ÉLU).