



**HAL**  
open science

# Élection et anneaux unidirectionnels en présence d'homonymes

Anaïs Durand

► **To cite this version:**

Anaïs Durand. Élection et anneaux unidirectionnels en présence d'homonymes. ALGOTEL 2017 - 19èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, May 2017, Quiberon, France. hal-01513893

**HAL Id: hal-01513893**

**<https://hal.science/hal-01513893>**

Submitted on 25 Apr 2017

**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.

# Élection et anneaux unidirectionnels en présence d'homonymes

Anaïs Durand

VERIMAG UMR 5104, Université Grenoble Alpes, France

---

Cet article est un résumé de deux articles [1, 2] portant sur l'élection de leader dans les anneaux unidirectionnels en présence de processus homonymes, c'est-à-dire des anneaux unidirectionnels où les processus sont nommés par des étiquettes qui ne sont pas nécessairement uniques. Nous étudions la résolution de ce problème dans des classes d'anneaux où l'étiquetage des processus est asymétrique. Nous proposons trois algorithmes pour des classes où une borne sur la multiplicité des étiquettes (c'est-à-dire, le nombre de processus partageant la même étiquette) est connue.

**Mots-clés :** Algorithmes distribués, élection de leader, anneau unidirectionnel, homonymes.

---

## 1 Introduction

Angluin [3] a démontré l'impossibilité de résoudre l'élection de leader déterministe dans un réseau anonyme. Depuis, deux directions de recherche ont principalement été étudiées pour casser les symétries : proposer des solutions probabilistes pour les réseaux anonymes ou des solutions déterministes pour les réseaux identifiés. Récemment, la notion de *processus homonymes* a été introduite [7] comme modèle intermédiaire entre les réseaux anonymes et les réseaux identifiés. Dans ce modèle, chaque processus est nommé par une *étiquette* qui peut ne pas être unique.

Notre étude est centrée sur des anneaux unidirectionnels en modèle à message, c'est-à-dire des anneaux orientés où chaque processus peut uniquement envoyer des messages à son voisin de droite et uniquement en recevoir de son voisin de gauche. Nous supposons que les liens de communication sont asynchrones, FIFO et fiables. Et nous nous intéressons à deux variantes du problème d'élection de leader, en présence de processus homonymes, dans ce modèle : l'élection de leader avec *terminaison explicite*, où les processus terminent leur exécution en temps fini, et l'élection de leader avec *terminaison implicite*, où les processus ne terminent pas mais où seul un nombre fini de messages est échangé.

**État de l'art.** Plusieurs travaux récents [4–6] étudient le problème de l'élection de leader dans les anneaux en présence de processus homonymes.

Dans [6], Flocchini *et al* étudie l'*élection de leader faible* dans les anneaux bidirectionnels. L'élection de leader faible consiste à élire au mieux un unique processus, sinon deux processus voisins. Ils démontrent que même en connaissant le nombre de processus  $n$ , il est impossible de résoudre l'élection de leader faible avec terminaison explicite dans les anneaux bidirectionnels où l'étiquetage est symétrique. Un anneau étiqueté est dit *symétrique* s'il existe une rotation non triviale (c'est-à-dire non multiple de sa taille) des étiquettes qui donne le même étiquetage. Dans le cas contraire, l'anneau est dit *asymétrique*. Deux algorithmes résolvant l'élection de leader faible avec terminaison explicite pour les anneaux asymétriques de  $n$  processus sont ensuite proposés, en supposant que  $n$  est premier et qu'il n'y a que deux étiquettes possibles, 0 et 1. Le premier algorithme suppose en plus une orientation commune à tout processus, alors que le second généralise le premier pour des anneaux sans orientation.

Delporte *et al* [4] étudient l'élection de leader dans les anneaux bidirectionnels. Ils démontrent qu'il y a une solution déterministe à l'élection de leader avec terminaison implicite dans ces anneaux si et seulement si le nombre d'étiquettes différentes est strictement plus grand que le plus grand diviseur non trivial de  $n$ . Sous cette hypothèse, ils proposent deux algorithmes. Le premier résout l'élection de leader avec

Classe		Classe	Borne Inf. Temps	Temps	Nbr. Msgs	Mémoire
$\mathcal{U}^*$	EL-TE impossible	$\mathcal{U}^* \cap \mathcal{K}_k$	$\Omega(kn)$	$n(k+2)$	$O(n^2 + kn)$	$\lceil \log(k+1) \rceil + 2b + 4$
$\mathcal{A}$	EL-TE impossible	$\mathcal{A} \cap \mathcal{K}_k$	$\Omega(kn)$	$(2k+2)n$	$n^2(2k+1)$	$2(k+1)nb + 2b + 3$
$\mathcal{K}_k$	EL-TI impossible			$O(k^2n^2)$	$O(k^2n^2)$	$2 \lceil \log k \rceil + 3b + 5$

**TABLE 1:** Résumé des résultats. L'élection de leader avec terminaison explicite et l'élection de leader avec terminaison implicite sont respectivement notées EL-TE et EL-TI.

terminaison implicite sans hypothèse supplémentaire. Le second résout l'élection de leader avec terminaison explicite en supposant que les processus connaissent  $n$ . Ce dernier algorithme est asymptotiquement optimal en nombre de messages,  $O(n \log n)$ .

Enfin, Dobrev et Pelc [5] étudient une généralisation de l'élection de leader dans les anneaux unidirectionnels et bidirectionnels. Ils supposent que les processus connaissent une borne inférieure  $m$  et une borne supérieure  $M$  sur  $n$ . Leur généralisation consiste à déterminer, de manière distribuée, si l'élection avec terminaison explicite est possible, et, le cas échéant, la réaliser. Ils proposent ainsi des algorithmes synchrones pour les anneaux unidirectionnels et bidirectionnels résolvant ce problème en  $O(M)$  unités de temps et utilisant  $O(n \log n)$  messages. Ils proposent également un algorithme asynchrone pour les anneaux bidirectionnels qui utilisent  $O(nM)$  messages et montrent que cette complexité est optimale. Aucune complexité en temps n'est donnée pour ce dernier algorithme.

**Contributions.** Nous nous intéressons ici à l'élection de leader déterministe dans des anneaux unidirectionnels où, contrairement à [4–6], les processus ne connaissent ni le nombre de processus,  $n$ , ni aucune borne sur  $n$ . Tout d'abord, l'élection de leader avec terminaison implicite reste impossible à résoudre sans hypothèse supplémentaire. En effet, si on considère le cas d'un anneau où l'étiquetage est symétrique, il est impossible de distinguer deux processus ayant la même étiquette au cours d'une exécution synchrone. Nous nous intéressons alors à la classe  $\mathcal{A}$  des anneaux unidirectionnels asymétriques. Notre premier résultat a été de démontrer que, pour un anneau unidirectionnel asymétrique qui possède au moins une étiquette unique (nous notons  $\mathcal{U}^* \subseteq \mathcal{A}$  l'ensemble de ces anneaux), l'élection de leader avec terminaison explicite est impossible à résoudre. Maintenant, nous introduisons une connaissance supplémentaire : nous supposons que les processus connaissent une borne supérieure,  $k$ , sur les multiplicités des étiquettes (par *multiplicité d'une étiquette*, on entend le nombre de processus de l'anneau portant cette étiquette). Nous notons  $\mathcal{K}_k$  la classe de tous les anneaux unidirectionnels avec multiplicités bornées par  $k$ . Sous cette dernière hypothèse, l'élection de leader avec terminaison explicite dans les anneaux unidirectionnels asymétriques devient possible, c'est-à-dire qu'il existe des algorithmes résolvant le problème pour tous les anneaux de la classe  $\mathcal{A} \cap \mathcal{K}_k$ . Par ailleurs, nous avons aussi démontré l'impossibilité de résoudre l'élection de leader avec terminaison implicite dans la classe  $\mathcal{K}_k$ .

En plus de ces résultats d'impossibilité, nous avons proposé trois algorithmes d'élection de leader avec terminaison explicite. Le premier résout le problème pour la classe  $\mathcal{U}^* \cap \mathcal{K}_k$ , c'est-à-dire, pour tout anneau unidirectionnel où au moins une étiquette est unique et où les multiplicités sont bornées par  $k$ . Cet algorithme termine en au plus  $n(k+2)$  unités de temps, requiert  $\lceil \log k + 1 \rceil + 2b + 4$  bits par processus et  $O(n^2 + kn)$  messages sont échangés pendant une exécution, où  $b$  est le nombre de bits nécessaires pour stocker une étiquette. En outre, nous avons démontré une borne inférieure en  $\Omega(kn)$  unités de temps pour cette classe. Ainsi, cet algorithme est *asymptotiquement optimal en temps*.

Pour la classe plus générale  $\mathcal{A} \cap \mathcal{K}_k$  des anneaux unidirectionnels asymétriques avec multiplicités bornées par  $k$ , nous avons proposé deux algorithmes résolvant l'élection de leader avec terminaison explicite illustrant le compromis classique entre occupation mémoire et complexité en temps. Le premier est asymptotiquement optimal en temps, avec au plus  $(2k+2)n$  unités de temps (en effet,  $\mathcal{U}^* \cap \mathcal{K}_k$  étant inclus dans  $\mathcal{A} \cap \mathcal{K}_k$ , la borne inférieure pour  $\mathcal{U}^* \cap \mathcal{K}_k$  l'est aussi pour  $\mathcal{A} \cap \mathcal{K}_k$ ). Cependant, cet algorithme nécessite  $2(k+1)nb + 2b + 3$  bits par processus, et au plus  $n^2(2k+1)$  messages sont échangés durant une exécution. Au contraire, notre seconde solution nécessite seulement  $2 \lceil \log k \rceil + 3b + 5$  bits par processus. Par contre, sa complexité en temps est en  $O(k^2n^2)$  et sa complexité en messages est en  $O(k^2n^2)$ .

**Plan.** Nos résultats sont résumés dans la table 1. Par manque de place, nous présentons ici uniquement une petite partie d'entre eux. Précisément, nous rappelons dans la section 2 les définitions des deux notions d'élection considérées ici. Les sections 3 et 4 sont consacrées aux bornes inférieures et aux résultats d'impossibilité. Les idées principales de notre solution pour la classe  $\mathcal{U}^* \cap \mathcal{K}_k$  sont exposées dans la section 5.

## 2 Election de Leader

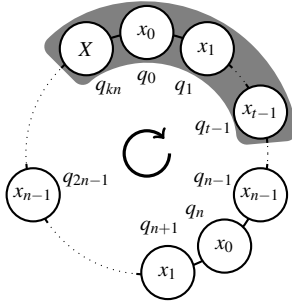
Un algorithme résout l'élection de leader avec terminaison implicite si

1. toutes ses exécutions sont finies, c'est-à-dire qu'en un temps fini, plus aucun message n'est échangé ;
2. dans toute configuration terminale, un unique processus se déclare leader et tout processus connaît l'étiquette du leader.

Un algorithme résout l'élection de leader avec terminaison explicite si

1. il résout l'élection de leader avec terminaison implicite ;
2. la décision de se déclarer leader est irrévocable, c'est-à-dire qu'aucun processus ne peut pas passer du statut leader à non-leader (par conséquent, dans toute configuration, il y a au plus un processus se déclarant leader) ; et
3. les processus finissent par décider localement de la terminaison de l'élection, c'est-à-dire qu'ils finissent par s'arrêter une fois le leader élu et son étiquette connue.

## 3 Bornes inférieures pour les classes $\mathcal{U}^* \cap \mathcal{K}_k$ et $\mathcal{A} \cap \mathcal{K}_k$ , avec $k \geq 2$ .



**FIGURE 1:**  $R_{n,k}$ . En gris, les processus qui peuvent avoir reçu des informations de  $q_{kn}$  après  $t \geq 0$  unités de temps.

Nous avons démontré une borne inférieure en  $\Omega(kn)$  sur la complexité en temps de l'élection de leader avec terminaison explicite dans  $\mathcal{U}^* \cap \mathcal{K}_k$ , où  $k$  est une constante supérieure ou égale à 2. Pour cela, nous considérons un algorithme ALG qui résout ce problème. Ensuite, nous étudions l'exécution synchrone  $e$  de ALG dans un anneau unidirectionnel  $R_n$  où les processus  $p_0, \dots, p_{n-1}$  sont étiquetés de manière unique  $x_0, \dots, x_{n-1}$ , c'est-à-dire  $R_n \in \mathcal{K}_1 \subseteq \mathcal{U}^* \cap \mathcal{K}_k$ . Par hypothèse,  $e$  termine et élit un leader, notons le  $p_\ell$ . Soit  $T$  le temps d'exécution de  $e$ . Soit  $R_{n,k}$  l'anneau unidirectionnel de  $nk + 1$  processus  $q_0, \dots, q_{nk}$  dont l'étiquetage consiste en la séquence  $x_0, \dots, x_{n-1}$  répétée  $k$  fois suivie de la dernière étiquette  $X$ , différente de toutes les autres. Cette construction est présentée dans la figure 1. Par définition,  $R_{n,k}$  appartient aussi à  $\mathcal{U}^* \cap \mathcal{K}_k$ . Considérons maintenant l'exécution synchrone  $e'$  de ALG sur  $R_{n,k}$ .

De nouveau, par hypothèse,  $e'$  termine et il n'existe aucune configuration de  $e'$  où deux processus se déclarent leader, l'élection étant irrévocable. Or, après  $t \geq 0$  unités de temps, seuls les processus  $q_i$  avec  $i \in \{0, \dots, t\}$  peuvent avoir reçu une information du processus  $q_{kn}$  d'étiquette  $X$ , (cf. les processus dans la zone grise sur la figure 1). Ainsi, on a la propriété suivante : (\*) Après  $t \geq 0$  unités de temps, tous les processus  $q_i$ , avec  $i \in \{t, \dots, kn - 1\}$ , ont le même état que le processus  $p_{i \bmod n}$  dans  $e$  après  $t$  unités de temps. Supposons maintenant par contradiction que  $T \leq (k-2)n$ . Soit  $j_1 = (k-2)n + \ell$  et  $j_2 = (k-1)n + \ell$ . D'après (\*), après  $T$  unités de temps dans  $e'$ ,  $q_{j_1}$  et  $q_{j_2}$  ont le même état que  $p_\ell$  dans la configuration terminale de  $e$ , i.e.,  $q_{j_1}$  et  $q_{j_2}$  se déclarent leader, ce qui contredit l'irrévocabilité de l'élection. Ainsi, l'exécution  $e$  dure au moins  $(k-2)n + 1$  unités de temps.

Notez que, par définition,  $\mathcal{U}^* \subseteq \mathcal{A}$ . Ainsi, cette borne inférieure est aussi valable pour  $\mathcal{A} \cap \mathcal{K}_k$ , avec  $k \geq 2$ .

## 4 Impossibilité pour les classes $\mathcal{U}^*$ , $\mathcal{A}$ et $\mathcal{K}_k$ avec $k \geq 2$ .

La borne inférieure précédente nous permet de démontrer notre résultat d'impossibilité pour la classe  $\mathcal{U}^*$ . L'idée de la preuve est que  $\mathcal{U}^*$  contient l'union infinie des classes  $\mathcal{U}^* \cap \mathcal{K}_k$  pour tout  $k \geq 1$ . Ainsi, comme la borne inférieure sur chaque classe  $\mathcal{U}^* \cap \mathcal{K}_k$  avec  $k \geq 2$  dépend du paramètre  $k$ , la borne inférieure sur le temps d'exécution de tout algorithme pour  $\mathcal{U}^*$  est nécessairement non bornée, d'où le résultat d'impossibilité. Comme  $\mathcal{U}^* \subseteq \mathcal{A}$ , ce résultat d'impossibilité est aussi valable pour  $\mathcal{A}$ .

Concernant la classe  $\mathcal{K}_k$  avec  $k \geq 2$ , elle contient en particulier des anneaux symétriques de  $i \times k$  processus avec  $i \geq 1$ , dont la séquence d'étiquettes est  $l_1, l_2, \dots, l_i$  répétée  $k$  fois. L'exécution synchrone de

tout algorithme sur de tels anneaux ne permet pas de différencier les processus d'étiquettes identiques. Par conséquent, il est impossible de résoudre l'élection de leader avec terminaison implicite dans cette classe.

## 5 Élection de leader avec terminaison explicite pour $\mathcal{U}^* \cap \mathcal{K}_k$

Les performances de cette solution sont données dans la table 1. Son but est d'élire, parmi les processus d'étiquettes uniques, celui (noté  $\ell$ ) de plus petite étiquette. Nous utilisons pour cela des compteurs dont le but est d'évaluer approximativement les multiplicités des étiquettes. Chaque processus  $p$  possède une étiquette  $p.id$  et un compteur entier  $p.cnt$ ; tous les messages échangés contiennent une étiquette et un compteur entier qui reflète lui aussi la multiplicité de l'étiquette.

Chaque processus  $p$  commence par initialiser son compteur  $p.cnt$  à 0, puis initie un jeton en envoyant un message contenant son étiquette et la valeur de son compteur, 0. L'ensemble des jetons va circuler dans l'anneau. Lorsqu'un processus reçoit un jeton contenant son étiquette, il incrémente son propre compteur ainsi que le compteur du jeton. Grâce à ces compteurs, les processus, initialement tous actifs, vont petit à petit devenir passifs jusqu'à ce que seul  $\ell$ , le leader, reste actif.

*Élimination des processus non uniques.* La première règle pour désactiver un processus est la suivante. Si un processus  $p$  reçoit un jeton contenant une valeur de compteur plus petite que son propre compteur,  $p.cnt$ , alors  $p$  peut en déduire que son étiquette n'est pas unique et devient passif. Lorsqu'un processus  $p$  devient passif, le jeton qu'il a initié devient inutile :  $p$  ne retransmet plus les jetons portant son étiquette. Ainsi, tous les processus d'étiquette non unique deviennent passifs lors du premier tour de leur jeton. Lorsque les jetons initiés par les autres processus ont fait une fois le tour de l'anneau, le compteur dans chacun de ces jetons vaut 1 et leurs initiateurs sont encore actifs.

*Élection du processus d'étiquette unique minimum.* Pour élire le leader parmi les processus encore actifs, on procède comme suit. Si un processus  $p$  reçoit un jeton contenant une étiquette plus petite que la sienne et une valeur de compteur  $c$  à la fois strictement positive et identique à la valeur son propre compteur, i.e.,  $p.cnt = c \wedge c > 0$  (cette condition assure que l'on ne se trouve pas dans le premier tour de l'anneau),  $p$  en déduit que son étiquette n'est pas la plus petite étiquette unique et devient passif.

*Terminaison explicite.* Ainsi, après que les jetons restants aient fait au plus deux tours de l'anneau, seul  $\ell$  est encore actif, mais  $\ell$  ne peut pas immédiatement le déterminer et se déclarer leader. Cependant, lorsque  $\ell$  reçoit un jeton contenant son étiquette et la valeur de compteur  $k$ ,  $\ell$  peut en déduire qu'il n'est pas dans le premier tour (les compteurs étant initialisés à 0, aucun jeton ne peut atteindre la valeur  $k$  lors du premier tour). Donc, son étiquette est nécessairement unique et en conséquence, ce jeton a effectué  $k + 1 > 2$  tours de l'anneau. Ainsi,  $\ell$  déduit qu'il est le seul processus encore actif et se déclare leader. Il envoie alors un dernier jeton  $\langle \ell.id, k + 1 \rangle$  pour informer les autres processus de l'étiquette du processus élu. Lorsque les autres processus reçoivent ce jeton, ils le retransmettent puis s'arrêtent explicitement. Lorsque  $\ell$  reçoit  $\langle \ell.id, k + 1 \rangle$ , il s'arrête à son tour, mettant fin à l'exécution.

## Références

- [1] K. Altisen, A. K. Datta, S. Devismes, A. Durand, and L. L. Larmore. Leader election in rings with bounded multiplicity (short paper). In *SSS 2016*, pages 1–6, Lyon, France, November 7-10 2016.
- [2] K. Altisen, A. K. Datta, S. Devismes, A. Durand, and L. L. Larmore. Leader election in asymmetric labeled unidirectional rings. In *IPDPS 2017*, Orlando, Florida, USA, May 29 - June 2 2017. To appear.
- [3] D. Angluin. Local and global properties in networks of processors. In *STOC*, pages 82–93, 1980.
- [4] C. Delporte-Gallet, H. Fauconnier, and H. Tran-The. Leader election in rings with homonyms. In *NETYS*, pages 9–24, 2014.
- [5] S. Dobrev and A. Pelc. Leader election in rings with nonunique labels. *Fundam. Inform.*, 59(4) :333–347, 2004.
- [6] P. Flocchini, E. Kranakis, D. Krizanc, F. L. Luccio, and N. Santoro. Sorting and election in anonymous asynchronous rings. *JPDC*, 64(2) :254–265, 2004.
- [7] M. Yamashita and T. Kameda. Electing a leader when processor identity numbers are not distinct (extended abstract). In *WDAG*, pages 303–314, 1989.