



**HAL**  
open science

## Trouver des spanners peu denses dans les cliques temporelles

Arnaud Casteigts, Joseph Peters, Jason Schoeters

► **To cite this version:**

Arnaud Casteigts, Joseph Peters, Jason Schoeters. Trouver des spanners peu denses dans les cliques temporelles. ALGOTEL 2019 - 21èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2019, Saint Laurent de la Cabrerisse, France. hal-02117940

**HAL Id: hal-02117940**

**<https://hal.science/hal-02117940>**

Submitted on 2 May 2019

**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.

# Trouver des spanners peu denses dans les cliques temporelles <sup>†</sup>

Arnaud Casteigts<sup>1</sup> et Joseph G. Peters<sup>2</sup> et Jason Schoeters<sup>1</sup>

<sup>1</sup>LaBRI, Université de Bordeaux, France

<sup>2</sup>School of Computing Science, Simon Fraser University, Canada

---

Nous nous intéressons à une question posée par Kempe, Kleinberg, et Kumar (STOC 2000) : existe-t-il toujours un sous-ensemble d'arêtes *peu dense* préservant la connexité temporelle dans un graphe dynamique  $\mathcal{G}$  ? On appelle ces sous-ensembles des *spanners temporels* de  $\mathcal{G}$ . Récemment, Axiotis et Fotakis (ICALP 2016) ont répondu négativement, montrant qu'il existe une famille de graphes dynamiques denses n'admettant pas de spanner temporel peu dense. La question est donc de savoir si des spanners peu denses existent pour des classes spécifiques de graphes denses. Dans cet article, nous répondons positivement pour les graphes complets en montrant que ces graphes admettent toujours un spanner de taille  $O(n \log n)$ . Il s'agit de la première réponse positive de ce type depuis le travail de Kempe *et al.*

**Mots-clés :** réseaux dynamiques, connexité temporelle, spanners

---

## 1 Introduction

L'étude des réseaux hautement dynamiques a gagné en intérêt avec l'émergence de contextes technologiques où les entités se déplacent et communiquent (véhicules, robots, drones, etc.). Ces réseaux peuvent être modélisés par des graphes dynamiques, aussi appelés graphes temporels, qui se représentent de différentes manières. Ici, nous considérons un modèle simple qui consiste à étiqueter les arêtes d'un graphe par des entiers qui indiquent quand l'arête est présente (en temps discret). Lorsque le réseau est fortement dynamique, les propriétés d'intérêt s'expriment souvent au cours du temps, plutôt qu'à un instant donné. Par exemple, un graphe peut ne jamais être connexe mais offrir une forme de connexité basée sur la notion de chemin temporel (ou *trajet*), à savoir un chemin qui tolère des pauses entre sauts consécutifs et peut donc se réaliser à travers le temps. Cette *connexité temporelle* a été considérée, par exemple, par Awerbuch et Even [2] (1984) et étudiée plus systématiquement au début des années 2000 dans les travaux de Kempe, Kleinberg et Kumar [6], et Bui-Xuan, Ferreira et Jarry [4]. Dans la dernière section de [6], la question suivante est posée :

“Given a temporally connected network  $G = (V, E)$  on  $n$  nodes, is there a subset  $E' \subseteq E$  consisting of (*few*) edges so that the temporal network on the subgraph  $(V, E')$  is also temporally connected? In other words, do all temporal networks have sparse subgraphs preserving this basic connectivity property?”

Cette question est restée ouverte une quinzaine d'années, jusqu'à ce que, récemment, Axiotis et Fotakis [3] y répondent négativement en exhibant une famille de graphes temporels de taille  $\Theta(n^2)$  n'admettant pas de spanner de taille  $o(n^2)$ . Du côté positif, Akrida *et al.* [1] ont montré que l'on peut toujours enlever  $\lfloor n/4 \rfloor$  arêtes sans compromettre la connexité temporelle lorsque le graphe sous-jacent est complet. On parle alors de *clique temporelle*.

### 1.1 Contributions

Dans ce travail, nous montrons qu'il est toujours possible de trouver un spanner temporel de taille  $O(n \log n)$  dans les cliques temporelles, apportant ainsi la première réponse positive (bien que restreinte à

---

<sup>†</sup>Version courte d'un article accepté à ICALP 2019 [5], aussi accessible sur <https://arxiv.org/abs/1810.00104>

une famille de graphes) depuis la question de Kempe *et al.* En toute généralité, plusieurs modèles de cliques temporelles peuvent être envisagés, selon que les arêtes ont une seule ou plusieurs dates de présence, et selon que l'on considère des trajets dont les dates augmentent strictement ou pas (trajets stricts ou non-stricts). Nous montrons dans [5] que la partie intéressante du problème se réduit au cas particulier où chaque arête n'a qu'une date et les dates incidentes à un même sommet sont différentes – on parle alors d'étiquetage (et par extension, de clique temporelle) *simple*. Un exemple de clique temporelle simple et un de ses spanners sont donnés Figure 1.

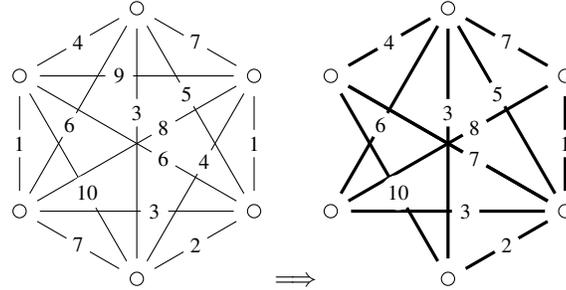


FIGURE 1: Exemple d'une clique temporelle et un de ses spanners temporels. Dans ce cas, le spanner n'est pas minimal, car on peut continuer à lui enlever des arêtes sans compromettre la connectivité temporelle.

Dans cette version courte, nous présentons les principales étapes d'un algorithme permettant de calculer un spanner de taille  $O(n \log n)$  dans n'importe quelle clique temporelle simple. La plupart des démonstrations sont omises et nous invitons le lecteur ou la lectrice à consulter [5] pour de plus amples détails. Néanmoins, nous tentons de transmettre la majorité des idées importantes.

## 2 Algorithme

L'algorithme se décompose principalement en deux phases. La première phase construit un spanner  $S$  de taille  $\binom{n}{2}/2 + O(n)$  (essentiellement la moitié des arêtes) basé sur notre technique de *feux d'artifices* (voir Sous-section 2.2). En fonction de la structure de  $S$ , soit le problème est ensuite réduit à une instance ayant un sommet de moins et l'algorithme recommence la première partie (récursion), soit le spanner doit posséder une structure très particulière qui permet à la seconde phase de le sparsifier jusqu'à  $O(n \log n)$  arêtes. Le cas où l'algorithme récurse jusqu'au bout sans entrer dans la seconde phase donne un spanner de taille  $O(n)$ . Les deux parties utilisent les notions de délégation entre sommets et de démontabilité, que nous présentons donc pour commencer.

### 2.1 Délégations et démontabilité

Étant donné un sommet  $v$ , on note  $e^-(v)$  (resp.  $e^+(v)$ ) l'arête incidente à  $v$  ayant la plus petite (resp. plus grande) étiquette/date.

**Remarque 1.** Si  $e^-(v) = \{u, v\}$ , alors le sommet  $u$  peut joindre tous les autres via le sommet  $v$  (par un trajet). Symétriquement, si  $e^+(v) = \{u, v\}$ , alors tous les sommets peuvent joindre  $u$  via  $v$ .

Ainsi, il arrive qu'un sommet  $u$  puisse déléguer à un voisin  $v$  la responsabilité de joindre tous les autres sommets, en se bornant lui-même à joindre  $v$ . On dit que  $u$  *délègue ses émissions* à  $v$  et que  $v$  joue le rôle d'*émetteur* pour  $u$ . Symétriquement, un sommet  $u$  peut *déléguer ses réceptions* à un sommet  $v$ , de sorte que les autres sommets se contentent de joindre  $v$  pour joindre  $u$  (via  $e^+(v)$ ). On dira que  $v$  est un *récepteur* pour  $u$  (on parle de "collector" dans [5]). Ces notions peuvent en réalité se généraliser à des trajets. Il n'est pas nécessaire pour cela que chaque saut du trajet arrive par une arête localement minimum (resp. sorte par une arête localement maximum); il suffit que cette propriété soit vérifiée pour le dernier saut vers l'émetteur (resp. premier saut depuis le récepteur).

On dit qu'un sommet  $v$  est *démontable* (à  $k$  sauts) s'il peut déléguer à la fois ses émissions et ses réceptions par des trajets d'au plus  $k$  sauts (potentiellement à des sommets différents). Le concept de démontabilité

## Trouver des spanners peu denses dans les cliques temporelles

suggère un certain type de spanner, construit en sélectionnant les arêtes des deux trajets de délégation, puis en réduisant le problème restant à la construction d'un spanner dans  $\mathcal{G}[V \setminus v]$ , c.à.d. dans la sous-clique de  $\mathcal{G}$  résultante du retrait de  $v$ .

**Théorème 1 (Démontabilité).** *Soit  $\mathcal{G}$  une clique temporelle et  $u, v, w$  trois sommets tels que  $u$  délègue ses émissions à  $v$  via les arêtes  $C_1$  et ses réceptions à  $w$  via les arêtes  $C_2$ . Supposons  $S'$  un spanner temporel de  $\mathcal{G}' = \mathcal{G}[V \setminus u]$ . Alors  $S = S' \cup C_1 \cup C_2$  est un spanner temporel de  $\mathcal{G}$ .*

Ainsi, si  $k$  est constant et que l'on arrive à démonter le graphe récursivement, cela résulte en un spanner de taille  $\Theta(n)$ . Malheureusement, nous montrons dans [5] qu'une famille infinie d'instances ne sont pas démontable, mais ce concept est toutefois utilisé de manière plus élaborée dans notre algorithme, présenté maintenant.

## 2.2 Feux d'artifices (arbres de délégation)

L'idée des feux d'artifices est de mutualiser les délégations de sorte que plusieurs sommets délèguent leurs émissions (ou réceptions) à un seul et même sommet. Nous construisons donc des arbres de délégations d'émission  $\mathcal{T}^-$  et de réceptions  $\mathcal{T}^+$  appelés *feux d'artifices*, qui sont ensuite combinés pour donner un spanner de taille au plus  $\binom{n}{2}/2 + O(n)$ . Les arbres sont construits en deux étapes, données ici pour le cas des émissions (les réceptions fonctionnent de manière symétrique). Pour commencer, on définit le graphe orienté  $G^- = (V, E^-)$  tel que  $(u, v) \in E^-$  si et seulement si  $\{u, v\} = e^-(v)$  pour n'importe quel  $v$ .

**Lemme 1.** *Des chemins orientés dans  $G^-$  correspondent à des trajets dans  $\mathcal{G}$ .*

Par construction,  $E^-$  induit un ensemble disjoint d'arbres orientés d'une source vers possiblement plusieurs puits. Par une suite de transformations préservant des propriétés de délégabilité, on modifie ces arbres en un ensemble  $\mathcal{T}^-$  d'arbres potentiellement plus petits, mais où le nombre de puits dans chaque arbre est exactement un et chaque arbre contient au moins deux sommets. Cela permet aux sommets de chaque arbre de délèguer leurs émissions au sommet puits de ce même arbre (appelé ici aussi *émetteur*). Symétriquement, on construit un graphe  $G^+$  dont les arêtes  $E^+$  induisent un ensemble d'arbres que nous modifions en  $\mathcal{T}^+$  de sorte que les sommets d'un même arbre peuvent tous délèguer leurs réceptions au même *récepteur*.

**Lemme 2.** *Le nombre d'émetteurs dans  $\mathcal{T}^-$  est au plus  $\frac{n}{2}$  et le nombre de collecteurs dans  $\mathcal{T}^+$  est au plus  $\frac{n}{2}$ .*

En combinant les deux structures, on définit un spanner  $S$  dans lequel on inclut les arêtes correspondant aux deux arbres, ainsi que toutes les arêtes *entre* émetteurs et collecteurs. Ainsi, via  $S$ , chaque sommet peut joindre tous les autres sommets par un trajet de longueur au plus 3, passant par un émetteur, puis un récepteur, et enfin le sommet destination. On montre ainsi que  $S$  est de taille au plus  $\binom{n}{2}/2 + O(n)$ .

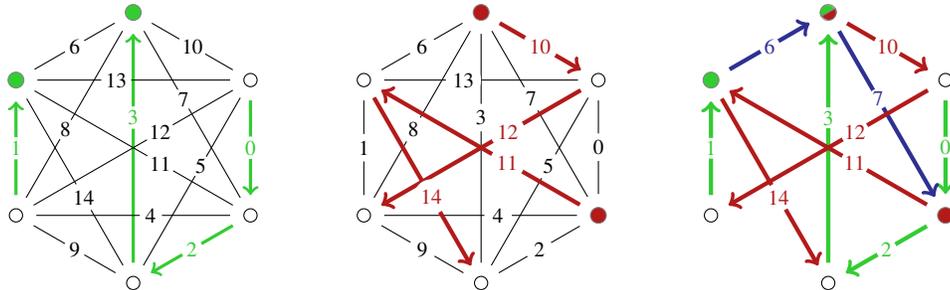


FIGURE 2: Exemple d'arbres de délégations  $\mathcal{T}^-$  (en vert) et  $\mathcal{T}^+$  (en rouge), dont la combinaison induit un spanner si l'on y adjoint les arêtes entre émetteurs et récepteurs (en bleu).

## 2.3 Récursion ou sparsification

Une fois obtenu un spanner  $S$  à base de feux d'artifices, deux cas sont possibles selon la configuration des émetteurs  $X^-$  et des récepteurs  $X^+$ .

**Cas 1** ( $X^- \cup X^+ \neq V$ ). Dans ce cas, au moins un sommet  $v$  n'est ni émetteur, ni récepteur. On utilise alors une propriété de la transformation de  $E^-$  vers  $T^-$  (et de  $E^+$  à  $T^+$ ) qui implique que  $v$  est démontable à 2 sauts (cette propriété n'est pas évoquée ci-dessus, elle correspond au Lemmes 3 et 5 dans [5]). Il est donc possible de sélectionner les arêtes des trajets correspondants (au plus quatre) pour une future inclusion au spanner, puis d'effectuer une récursion des feux d'artifices sur  $\mathcal{G}[V \setminus v]$ .

**Cas 2** ( $X^- \cup X^+ = V$ ). Dans ce cas, combiné avec le Lemme 2, nous avons donc exactement  $n/2$  émetteurs et  $n/2$  récepteurs, les deux étant disjoints car leur union doit contenir les  $n$  sommets. Nous observons que le spanner issu des feux d'artifices est alors nécessairement un graphe biparti complet dont les parties sont  $X^-$  et  $X^+$ . On considère alors deux scénarios selon qu'un émetteur  $v$  puisse déléguer ses émissions à un autre émetteur via les arêtes du graphe biparti. Si tel est le cas,  $v$  peut être démonté car on sait déjà qu'il a aussi délégué ses réceptions à un récepteur de  $X^+$  (et l'algorithme effectue là aussi une récursion sur  $\mathcal{G}[V \setminus v]$ ). Dans le cas contraire, on montre que cette impossibilité résulte en une propriété structurelle très forte, que nous exploitons pour finir de sparsifier le graphe par une technique de *délégations en couches*.

## 2.4 Délégation en couches

La propriété additionnelle dont nous disposons ici est que l'arête minimum d'un émetteur doit aussi être minimum du côté récepteur. De même, l'arête maximum d'un récepteur est aussi maximum pour l'émetteur correspondant. Cette dernière propriété nous permet de réduire la tâche de joindre un récepteur à celle de joindre l'émetteur correspondant. Bien sûr, il est désormais impossible qu'un émetteur  $u$  puisse joindre un autre émetteur  $v$  via l'arête  $e^-(v)$ , son trajet arrivera donc sur  $v$  par une autre arête  $e^i(v)$ , où  $i$  est le rang local de cette arête (c'est à dire la  $i$ ème plus petite). Cela donne lieu à la notion de *délégation partielle*, où  $u$  délègue à  $v$  le soin de joindre les récepteurs (et donc les émetteurs) qu'il peut encore joindre, les autres devant être joints par l'utilisation d'une arête directe depuis  $u$  (qui sera payée en supplément).

Plus précisément, cette partie de l'algorithme procède par itérations portant sur des intervalles de rangs d'arêtes de taille doublante, de sorte qu'à chaque itération la moitié des émetteurs délèguent partiellement leurs émissions à l'autre moitié des émetteurs. Ces émetteurs sont ensuite éliminés après avoir sélectionné les arêtes utilisées pour la délégation et les arêtes vers les récepteurs ratés par cette délégation. Au total,  $\Theta(n)$  arêtes sont ainsi sélectionnées à chaque itération, pour  $\Theta(\log n)$  itérations, soit un spanner de  $\Theta(n \log n)$  arêtes dans l'instance courante de la récursion. Le théorème suivant résume l'ensemble du processus.

**Théorème 2.** *Toute clique temporelle simple admet un spanner de taille  $O(n \log n)$ .*

*Idée de la preuve.* Soit  $n_1$  le nombre de fois où le graphe est démonté à 2 sauts, et  $n_2$  le nombre de sommets restant quand la procédure de délégations en couches démarre (observons que  $n_2 = n - n_1$ ). Alors, le nombre total d'arêtes dans le spanner est de  $\Theta(n_1) + \Theta(n_2 \log n_2)$ , soit  $O(n \log n)$  arêtes.  $\square$

## Références

- [1] Eleni C. Akrida, Leszek Gasieniec, George B. Mertzios, and Paul G. Spirakis. The complexity of optimal design of temporally connected graphs. *Theory of Computing Systems*, 61(3) :907–944, 2017.
- [2] B. Awerbuch and S. Even. Efficient and reliable broadcast is achievable in an eventually connected network. In *Proceedings of the 3rd ACM symposium on Principles of distributed computing (PODC)*, pages 278–281, Vancouver, Canada, 1984. ACM.
- [3] Kyriakos Axiotis and Dimitris Fotakis. On the size and the approximability of minimum temporally connected subgraphs. *arXiv preprint arXiv :1602.06411*, 2016.
- [4] B. Bui-Xuan, A. Ferreira, and A. Jarry. Computing shortest, fastest, and foremost journeys in dynamic networks. *International Journal of Foundations of Computer Science*, 14(2) :267–285, April 2003.
- [5] Arnaud Casteigts, Joseph G. Peters, and Jason Schoeters. Temporal cliques admit sparse spanners. In *46th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 132 of *LIPICs*, pages 129 :1–129 :14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2019.
- [6] David Kempe, Jon Kleinberg, and Amit Kumar. Connectivity and inference problems for temporal networks. *Journal of Computer and System Sciences*, 64(4) :820–842, 2002.