Triangle Width

de l'ordonnancement à la théorie des graphes

Luc Libralesso[*1], Vincent Jost[1], Khadija Hadj-Salem[2],
Frédéric Maffray[1], Florian Fontan[1]

21 Février 2019

[1] Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, 38000 Grenoble, France
[2] Université de Tours, LIFAT EA 6300, CNRS, ROOT ERL CNRS 7002, 64 avenue Jean Portalis, 37200 Tours
[*] `luc.libralesso@grenoble-inp.fr`

Description du problème

Optimisation de système de vision embarquée

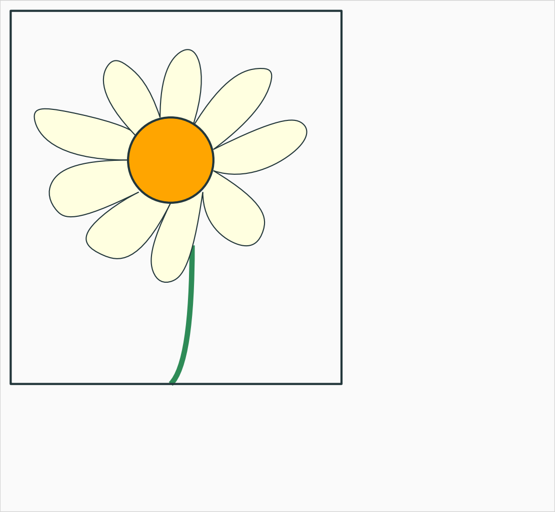

FIG. 1: Illustration du processus

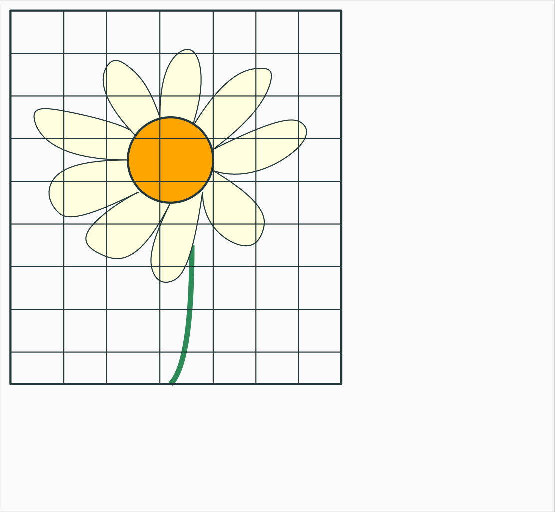

FIG. 2: Illustration du processus - division en parties

Optimisation de système de vision embarquée

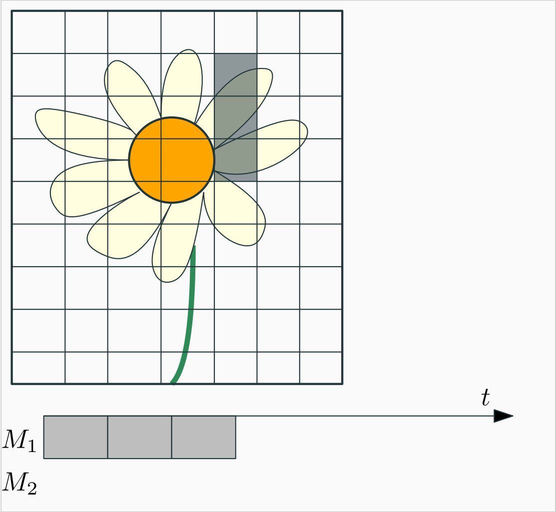

FIG. 3: Illustration du processus - chargement des prérequis pour la tâche 1

Optimisation de système de vision embarquée

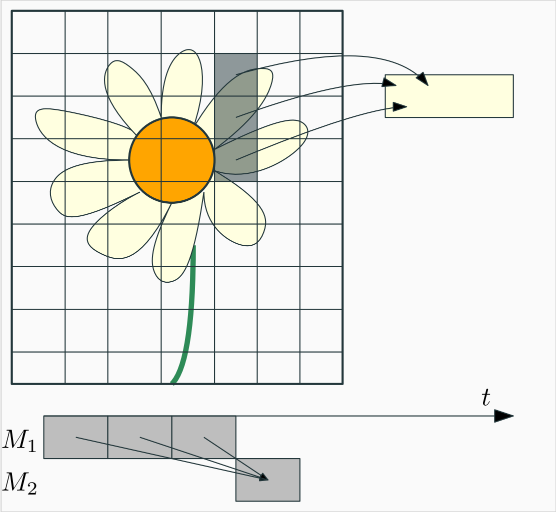

Fig. 4: Illustration du processus - exécution de la tâche 1

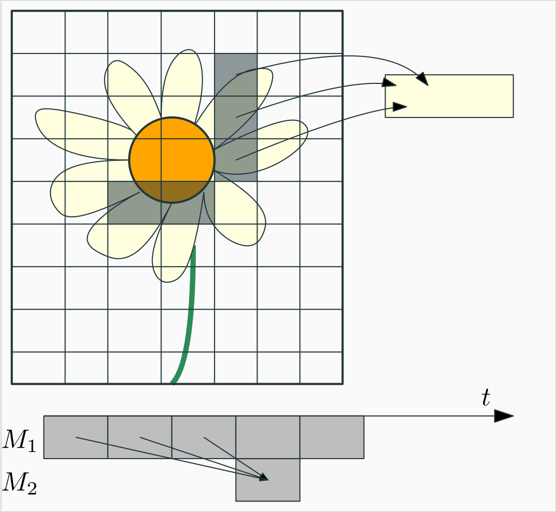

FIG. 5: Illustration du processus - chargement des prérequis pour la tâche 2

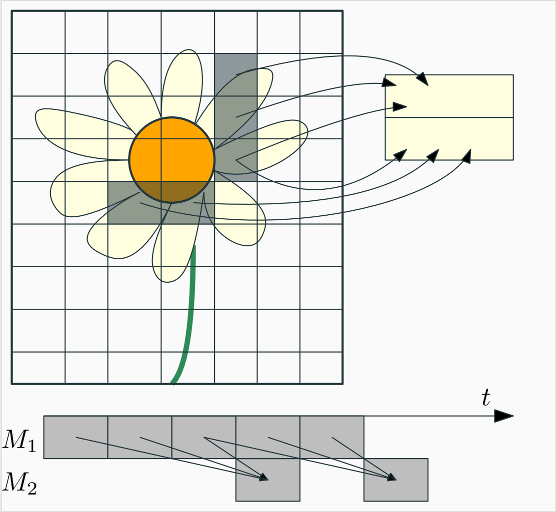

FIG. 6: Illustration du processus - exécution de la tâche 2

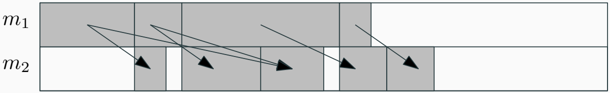

Fig. 7: triangle width - version scheduling

Entrées:

- $\mathcal{X}$: ensemble de parties d'image
- $\mathcal{Y}$: ensemble de tâches à effectuer
- relation de précédence

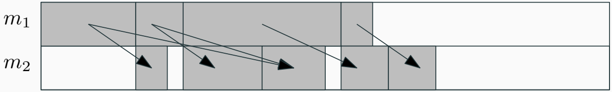

FIG. 7: triangle width - version scheduling

Entrées:

- $\mathcal{X}$: ensemble de parties d'image
- $\mathcal{Y}$: ensemble de tâches à effectuer
- relation de précédence

Sorties:

- Ordres $\sigma_{\mathcal{X}}$,$\sigma_{\mathcal{Y}}$ sur les parties/tâches.
- Sous contrainte de précédence
- minimise la durée totale (Makespan)

- Toutes les durées sont quelconques ($\in \mathcal{R}^+$)
- Nous nous intéressons au cas:
  - $duree(x \in \mathcal{X}) = \alpha$
  - $duree(y \in \mathcal{Y}) = \beta$
- et au cas où toutes les durées sont unitaires
- tous ces cas sont $\mathcal{NP}$-Hard

Représentation sous forme de matrice

1	4	3	2	5	6			
		d	e	f		a	c	b

Fig. 8: Exemple d'ordonnancement

toutes les durées sont unitaires

Représentation sous forme de matrice

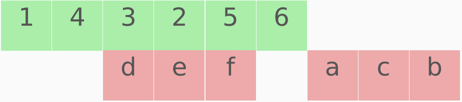

FIG. 8: Exemple d'ordonnancement

toutes les durées sont unitaires

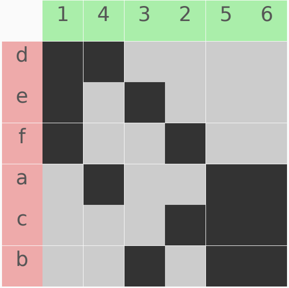

FIG. 9: Exemple de matrice pour triangle width

THÉORÈME:

$$\text{min. makespan} \iff \text{max. triangle supérieur droit}$$

Reformulation en visualisation de matrice

Play time !

`http://librallu.gitlab.io/hypergraph-viz`

Reformulation en parcours de graphe

THÉORÈME:

Une permutation suffit à encoder le problème

Reformulation en parcours de graphe

Théorème:

Une permutation suffit à encoder le problème

Intérêt: On définit un problème de visite de graphe équivalent à triangle width.

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

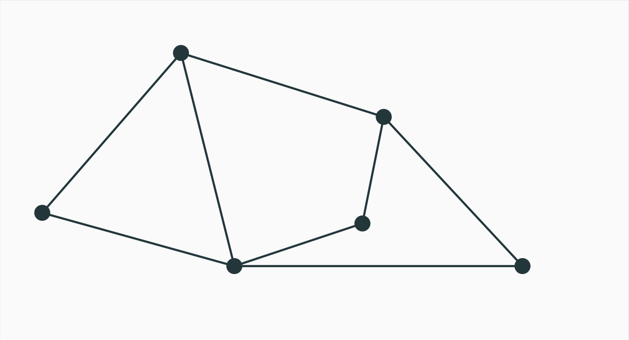

Fig. 10: $i = 0$, fuel: 10

- Les sommets coûtent 3
- Les arêtes rapportent 1

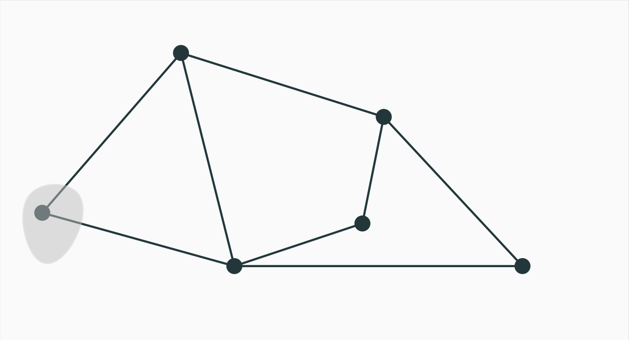

FIG. 11: $i = 1$, fuel: $10 - 3 = 7$

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

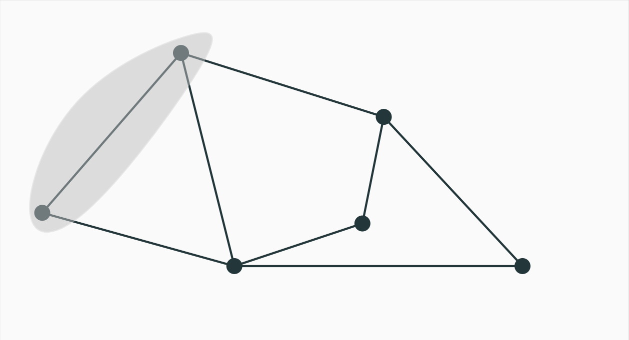

Fig. 12: $i = 2$, fuel: $7 - 3 + 1 = 5$

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

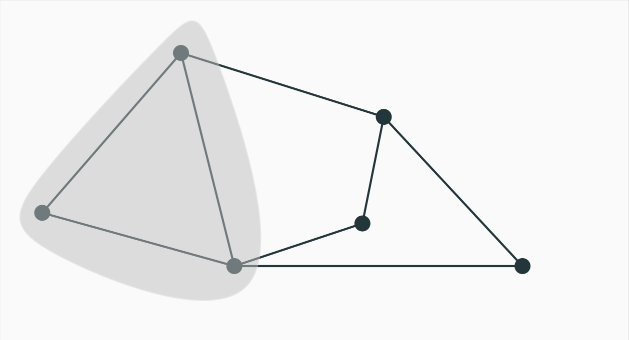

Fig. 13: $i = 3$, fuel: $5 - 3 + 2 = 4$

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

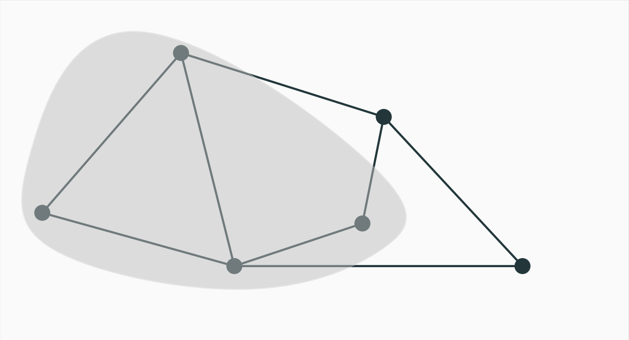

Fig. 14: $i = 4$, fuel: $4 - 3 + 1 = 2$

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

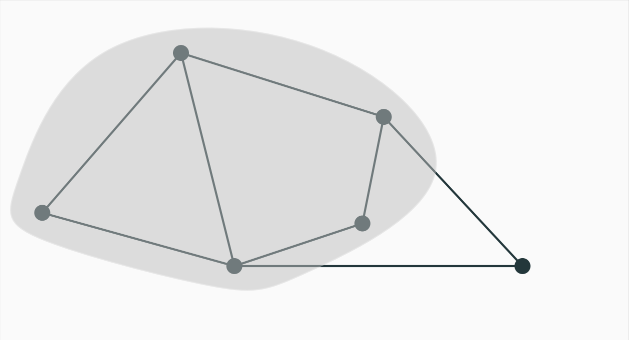

Fig. 15: $i = 5$, fuel: $2 - 3 + 2 = 1$

Reformulation en parcours de (Hyper-)graphe

- Les sommets coûtent 3
- Les arêtes rapportent 1

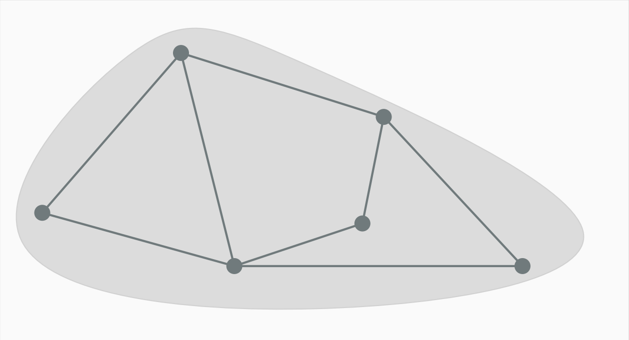

Fig. 16: $i = 6$, fuel: $1 - 3 + 2 = 0$

Quelques résultats

Le problème est $\mathcal{NP}$-Difficile

Réduction depuis Clique:
clique de taille k dans $G \iff$ visite avec coût de k

$$fuel = \frac{k(k+3)}{2}$$

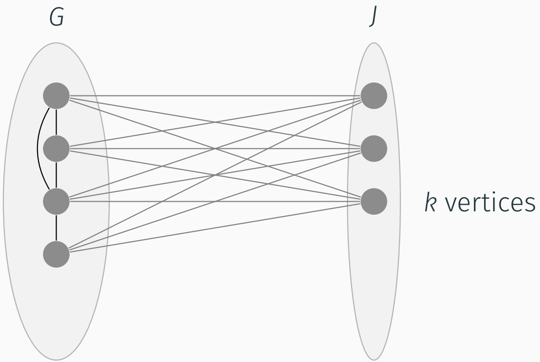

Fig. 17: graphe Gadget

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)
- Fertin, Rusu, Vialette ([FRV15]) réfutent la preuve en 2015

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)
- Fertin, Rusu, Vialette ([FRV15]) réfutent la preuve en 2015
- donnent une nouvelle preuve

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)
- Fertin, Rusu, Vialette ([FRV15]) réfutent la preuve en 2015
- donnent une nouvelle preuve
- elle aussi longue...

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)
- Fertin, Rusu, Vialette ([FRV15]) réfutent la preuve en 2015
- donnent une nouvelle preuve
- elle aussi longue…

Permutation de matrice 0/1 pour la rendre triangulaire

- Question posée par Wilf([Wil93]) en 1993
- Répondue par DasGupta [DJK+98]) en 1998 ($\mathcal{NP}$-Hard)
- Fertin, Rusu, Vialette ([FRV15]) réfutent la preuve en 2015
- donnent une nouvelle preuve
- elle aussi longue...

En utilisant Triangle Width, nouvelle preuve plus courte

Is a matrix permutation-wise diagonal est $\mathcal{NP}$-Difficile

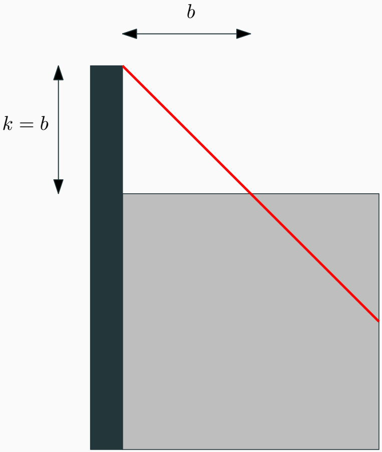

Fig. 18: Matrice Gadget

Quelques mots sur la résolution

Quelques approches utilisées

Méthodes exactes:

- 3 modèles MIP
- modèle CP
- 2 branch and bound

Quelques approches utilisées

Méthodes exactes:

- 3 modèles MIP
- modèle CP
- 2 branch and bound

Heuristiques/Meta-Heuristiques:

- 2 heuristiques constructives
- 2 modèles LocalSolver
- Simulated Annealing

résultats

- Les modèles MIP ne fontionnent pas bien
- Le modèle CP marche très bien
- Les heuristiques donnent de bons résultats en peu de calcul

conclusion

- Triangle width peut s’exprimer sous 3 formes
- Une appli web pour la version matrice
- Un bon outil pour faire des preuves de $\mathcal{NP}$-Hardness.

conclusion

- Triangle width peut s'exprimer sous 3 formes
- Une appli web pour la version matrice
- Un bon outil pour faire des preuves de $\mathcal{NP}$-Hardness.

Problèmes ouverts:

- existe t-il un MIP efficace ?

Questions?

Bibliographie

Bhaskar DasGupta, Tao Jiang, Sampath Kannan, Ming Li, and Elizabeth Sweedyk.
On the complexity and approximation of syntenic distance.
Discrete Applied Mathematics, 88(1-3):59–82, 1998.

Guillaume Fertin, Irena Rusu, and Stéphane Vialette.
Obtaining a triangular matrix by independent row-column permutations.
In *International Symposium on Algorithms and Computation*, pages 165–175. Springer, 2015.

Herbert S Wilf.
On crossing numbers, and some unsolved problems.
Combinatorics, geometry and probability (Cambridge, 1993), pages 557–562, 1993.