ACTIVITÉ DU N° 119

TRANSPORT DE PRODUITS CHIMIQUES
SOLUTION

Simon MODESTE[1]
Institut Montpelliérain Alexander Grothendieck
Université de Montpellier, CNRS

Énoncé

Une entreprise de produits chimiques doit effectuer une livraison de 16 produits (codés de A à P).

Chaque produit a des compatibilités et des incompatibilités avec les autres produits (c'est-à-dire qu'il peut ou pas être transporté dans le même camion qu'eux).

Cela est résumé dans le tableau suivant :

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	-	☠	☠	✔	✔	✔	☠	✔	✔	✔	✔	✔	✔	☠	✔	✔
B	☠	-	☠	☠	✔	☠	☠	✔	✔	✔	✔	✔	✔	✔	✔	✔
C	☠	☠	-	✔	✔	✔	☠	✔	✔	✔	✔	✔	✔	✔	☠	✔
D	✔	☠	✔	-	☠	☠	✔	✔	✔	✔	✔	☠	✔	✔	✔	✔
E	✔	✔	✔	☠	-	✔	✔	✔	✔	✔	✔	✔	☠	✔	✔	✔
F	✔	☠	✔	☠	✔	-	☠	✔	✔	✔	☠	☠	✔	✔	✔	✔
G	☠	☠	☠	✔	✔	☠	-	☠	☠	☠	☠	✔	✔	✔	✔	✔
H	✔	✔	✔	✔	✔	✔	☠	-	✔	✔	✔	✔	✔	✔	✔	✔
I	✔	✔	✔	✔	✔	✔	☠	✔	-	☠	✔	☠	✔	✔	☠	✔
J	✔	✔	✔	✔	✔	✔	☠	✔	☠	-	☠	✔	☠	☠	☠	✔
K	✔	✔	✔	✔	✔	☠	☠	✔	✔	☠	-	☠	☠	✔	✔	✔
L	✔	✔	✔	☠	✔	☠	✔	✔	☠	✔	☠	-	☠	✔	✔	✔
M	✔	✔	✔	✔	☠	✔	✔	✔	✔	☠	☠	☠	-	☠	✔	✔
N	☠	✔	✔	✔	✔	✔	✔	✔	✔	☠	✔	✔	☠	-	☠	✔
O	✔	✔	☠	✔	✔	✔	✔	✔	☠	☠	✔	✔	✔	☠	-	☠
P	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	☠	-

Quel est le nombre minimal de camions à utiliser pour transporter tous les produits ?

[1] simon.modeste@umontpellier.fr

Solution

1. Un début de solution

Il s'agit d'un problème d'optimisation : il est évident que des solutions de transport existent (quitte à prendre un camion par produit) et que le nombre de camions possible est minoré (il faut au moins un camion). Dit autrement, un camion (au moins) est nécessaire, et 16 camions suffisent.

La question est donc de trouver le plus petit nombre de camions suffisant, ou le plus grand nombre de camions nécessaire. Finalement, l'optimum sera le nombre de camions nécessaire et suffisant pour transporter tous les produits chimiques.

Ici, on peut déjà améliorer notre encadrement de la meilleure solution.

Assez simplement, puisque tous les produits ne sont pas compatibles entre eux, on peut affirmer qu'il faudra au moins 2 camions. Et même, si l'on regarde déjà les produits A, B et C, on voit qu'ils sont deux à deux incompatibles, ainsi il faudra au moins 3 camions (mais est-ce suffisant ?)

Plus généralement, on peut faire l'observation suivante :

> *Si l'on trouve un sous-ensemble de n produits deux à deux incompatibles, on peut affirmer qu'il faudra au moins n camions (autrement dit, l'optimum est supérieur ou égal à n).*

Quant à trouver une solution meilleure que 16 camions, on peut essayer de produire des solutions acceptables (c'est-à-dire sans produits incompatibles dans un même camion). Par exemple, en répartissant les produits comme ceci :

{A, E, I} ; {B, H, K} ; {C, D, N} ; {F, M, O} ; {J, L, P} ; {G},

on peut vérifier qu'il n'y a aucune incompatibilité dans chaque ensemble (représentant le chargement d'un camion) et donc voir que 6 camions suffisent.

On utilise en fait l'idée suivante :

> *Si l'on trouve une répartition acceptable des produits en N camions, alors on sait que l'optimum est inférieur ou égal à N.*

On peut aussi construire la répartition plus systématiquement, en considérant les produits dans un certain ordre (alphabétique par exemple), et en organisant leur rangement un par un dans des camions, en essayant de mettre chaque produit dans un camion où il est compatible avec les produits déjà chargés, et si ce n'est pas possible, en prévoyant un camion supplémentaire.

Ce type de méthode est appelé algorithme « glouton »[2] : on se donne un ordre sur les produits, et pour chaque produit, pris dans l'ordre, on fait un choix qui optimise localement (c'est-à-dire, à chaque fois qu'on regarde un nouveau produit, on utilise si possible un camion déjà disponible — alors qu'il aurait peut-être été judicieux, à un moment donné, de prendre un nouveau camion).

Ici, cela donne, en chargeant chaque produit dans le premier camion possible (lire colonne par colonne) :

[2] On parle d'algorithme glouton (*greedy algorithm* en anglais) pour un algorithme qui aborde un problème pas à pas, sans revenir en arrière, et qui fait des choix localement optimaux, en espérant que cela produira une solution optimale ou assez proche l'optimal (ce qui peut ne pas toujours être le cas).

A → camion 1	E → camion 2	I → camion 1	M → camion 3
B → camion 2	F → camion 3	J → camion 2	N → camion 4
C → camion 3	G → camion 4	K → camion 1	O → camion 5
D → camion 1	H → camion 1	L → camion 2	P → camion 1

On obtient la répartition :

{A, D, H, I, K, P} ; {B, E, J, L} ; {C, F, M} ; {G, N} ; {O},

qui est meilleure que la précédente. Mais peut-on encore faire mieux ?

On pourrait rechercher la solution de façon exhaustive. On peut par exemple appliquer la méthode « gloutonne » à toutes les permutations possibles de produits (on est sûr alors de rencontrer une façon de prendre les produit dans le bon ordre pour utiliser le moins de camions), mais cela fait 16! cas à traiter. Ou encore, on peut tester toutes les partitions possibles de l'ensemble de 16 produits, et trouver, parmi celles qui sont acceptables, celle qui utilise le moins de camions. Mais là encore, cela représenterait beaucoup de cas à traiter.

On pourrait continuer d'explorer le problème « à la main », astucieusement, et parvenir à trouver la solution optimale. Mais essayons de représenter ce problème autrement qu'avec un tableau.

2. Une traduction en théorie des graphes

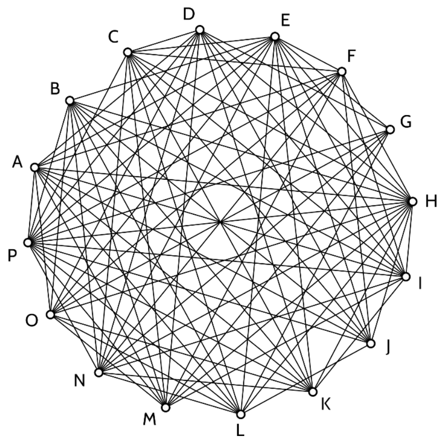
Le tableau des compatibilités et incompatibilités donné dans l'énoncé décrit une relation binaire entre les produits (et même symétrique puisque la relation d'incompatibilité ou de compatibilité est symétrique). On peut avoir envie de représenter la situation avec un *graphe fini simple non orienté*.

Un *graphe fini simple non orienté* (appelé simplement *graphe* dans la suite) est la donnée d'un ensemble fini de sommets, et d'un ensemble fini d'arêtes qui sont des paires de sommets. On note $G=(V,E)$ le graphe G d'ensemble de sommets V et d'ensemble d'arêtes E. Un graphe peut se représenter dans le plan, avec des points pour représenter ses sommets, et des lignes joignant ces points pour chaque arête.

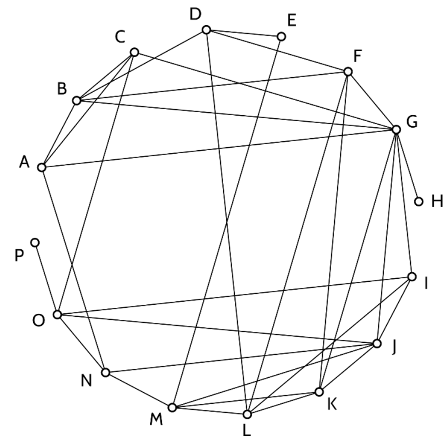
Ici, les sommets du graphe sont les produits {A, … , P} et les arêtes peuvent être les couples de produits compatibles ou les couples de produits incompatibles. Cela produit deux interprétations de la situation en termes de graphes, donnés ci-après.

Notre problème de produits chimiques se traduit alors en recherche d'une partition des sommets du graphe, la plus petite possible (en nombre d'ensemble de la partition), où chaque ensemble de la partition représente un lot de produits pouvant être transportés dans un même camion, autrement dit :

- dans le premier cas, il faut que tous les produits du même ensemble soit tous reliés deux à deux par une arête (c'est-à-dire qu'ils soient tous compatibles) ;
- dans le second, il faut qu'aucune arête ne relie deux d'entre eux (c'est-à-dire qu'il n'y ait aucune incompatibilité).



Graphe G_C *des relations de compatibilité.*

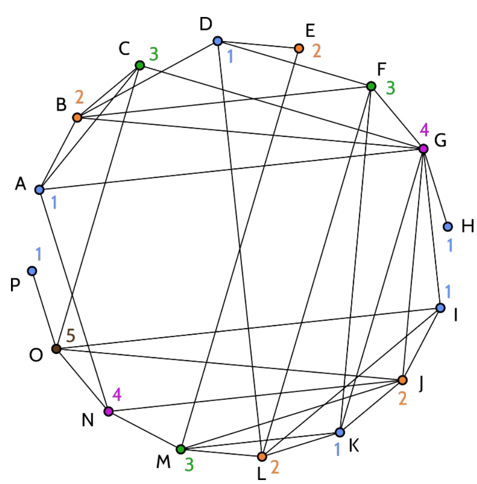


Graphe G_I *des relations d'incompatibilité.*

Bien que ces deux représentations en graphes soient complètement équivalentes sur le plan mathématique, on perçoit que la seconde sera plus facile à manipuler. D'une part, le graphe G_I contient beaucoup moins d'arêtes ; d'autre part, nous avons vu qu'identifier des ensembles de produits incompatibles entre eux (comme A, B et C) est très utile, et repérer des ensembles de sommets possédant toutes les arêtes possibles (ce qu'on appelle des *cliques*) peut s'avérer plus facile que repérer des ensembles de sommets n'ayant aucune arête entre eux (on appelle cela des *stables*). Nous allons donc conserver la seconde modélisation, avec le graphe G_I.

En fait, ceci est un problème classique de théorie des graphes, appelé *coloration de graphe* : on cherche à attribuer une couleur à chacun des sommets (chaque couleur représente un camion) de façon à ce qu'il n'y ait pas d'arêtes dont les sommets soient de même couleur (sinon il y aurait deux produits incompatibles dans un même camion). On cherche le nombre minimal de couleurs nécessaire pour réaliser une telle coloration (le minimum de camions nécessaire). Ce nombre est appelé le *nombre chromatique* d'un graphe G, noté $\chi(G)$.

La solution obtenue par l'algorithme glouton précédent peut se représenter sur le graphe (et on note qu'il est assez aisé de vérifier qu'elle est acceptable (aucun sommet n'est adjacent à un sommet de même couleur).



Une coloration du graphe d'incompatibilité, correspondant à la répartition acceptable des produits en 5 camions (les couleurs sont numérotées de 1 à 5).

Mais peut-on trouver une meilleure coloration du graphe ? Il s'agit de savoir si la solution de notre problème est 3, 4 ou 5 couleurs (puisqu'on a déjà vu qu'il faut au moins 3 camions/couleurs, et que 5 camions/couleurs suffisent).

Avant de répondre à cette question, prenons un moment pour étudier un peu plus en détail le nombre chromatique d'un graphe dans un cadre plus général. L'intérêt est que de nombreux problèmes peuvent se modéliser avec des graphes, et qu'on peut trouver des problèmes très divers qui correspondent à un problème de coloration de graphes, si l'on fait les bons choix de modélisation. Ainsi, étudier ce problème pour lui même nous permettra non seulement de résoudre notre problème de transport de produits chimiques, mais pourra servir dans d'autres situations.

3. Généralisons (et formalisons) un peu

Pour un graphe $G=(V,E)$, une *coloration* est l'association d'une couleur à chaque sommet de façon que deux sommets adjacents ne soient pas de même couleur. Autrement dit, une coloration est une fonction $\mathscr{C}: V \rightarrow \{1, \ldots, p\}$ (*p* entier naturel) telle que pour toute arête $\{s, t\}$ de E, $\mathscr{C}(s) \neq \mathscr{C}(t)$. (par convention, on représente, comme fait plus haut, les couleurs par des entiers)

On dit qu'un graphe G est *p-coloriable* si on peut le colorier avec seulement p couleurs. Pour tout graphe G, on peut trouver un N tel que G est N-coloriable (au moins avec N égal au nombre de sommets). Ainsi, on peut définir $\chi(G)$ le plus petit entier p tel que G est p-coloriable. $\chi(G)$ est appelé *nombre chromatique* de G, et une coloration de G avec $\chi(G)$ couleurs est dite *optimale*.

Le problème du nombre chromatique d'un graphe peut donc se formaliser par le problème suivant :

> Données : un graphe $G=(V,E)$ où V est l'ensemble des sommets et E l'ensemble des arêtes.
>
> Question : Déterminer $\chi(G)$.

Que peut-on dire de $\chi(G)$ et comment peut-on résoudre le problème ci-dessous ? Nous invitons les lecteur·rice·s à prendre un temps pour étudier cette question. Et notamment : $\chi(G)$ peut-il être aussi grand que l'on veut ? Peut-on l'encadrer ? Peut-on le calculer facilement à partir du graphe G ? Pour un entier k donné, quels sont les graphes k-coloriables ? Quel lien avec les sous-graphes de G ? Peut-on déterminer $\chi(G)$ pour des familles de graphes classiques ? etc.

Pour commencer, observons que dans un graphe *complet*, c'est-à-dire un graphe dont toutes les paires de sommets sont des arêtes, le nombre chromatique est égal au nombre de sommets puisque chaque sommet va nécessiter une couleur différente de celle des autres sommets.

On peut aussi reformuler en termes de graphes les deux observations faites précédemment :

- *Dans un graphe G, si l'on trouve un sous-ensemble de n sommets, tous deux à deux adjacents (un tel sous-graphe est appelé une clique de taille n), alors il faudra au moins n couleurs pour colorier le graphe G (puisqu'il faut déjà au moins n couleurs distinctes pour ces n sommets dans une coloration de G). Autrement dit,* $\chi(G) \geqslant n$.
- *Dans un graphe G, si l'on arrive à trouver une coloration « acceptable » avec N couleurs, alors on peut affirmer* $\chi(G) \leqslant N$.

On peut en fait généraliser le premier point : si l'on s'intéresse à un sous-graphe G' d'un graphe $G=(V,E)$ (c'est-à-dire un graphe $G'=(V',E')$ avec $V' \subset V$ et $E' \subset E$) on peut remarquer que :

si G est k-coloriable, alors G' est k-coloriable. Et réciproquement, si G' nécessite au moins k couleurs pour être colorié, alors G aussi. En fait, on peut affirmer plus simplement :

Pour tout graphe G, et tout sous-graphe G' de G, $\chi(G') \leqslant \chi(G)$.

Ceci, et l'observation qu'un graphe complet à n sommets a un nombre chromatique égal à n, permet de retrouver le premier point ci-dessus.

On peut aussi formaliser l'algorithme glouton vu précédemment. Soit un graphe $G=(V,E)$ et $(i_1, i_2, \ldots, i_n)$ une numérotation des sommets de G, définissant un ordre sur V. On appelle algorithme glouton de coloration l'algorithme suivant :

Algorithme glouton de coloration

Tant que tous les sommets ne sont pas coloriés :

- Prendre la première couleur c non utilisée ;
- Parcourir les sommets dans l'ordre fixé et colorier en c chaque sommet non-colorié et n'ayant aucun voisin de couleur c.

Notons que ce n'est pas tout à fait ce que nous avons énoncé précédemment, mais nous laissons aux lecteur·rice·s le soin de vérifier que l'on obtient la même coloration que l'on énumère les couleurs et pour chaque couleur les sommets dans l'ordre en leur attribuant la couleur « active » lorsque c'est possible, ou que l'on énumère les sommets et pour chacun on énumère les couleurs jusqu'à en trouver une acceptable.

Étudions cet algorithme plus en détail.

L'algorithme termine-t-il toujours, et est-il correct ? On voit bien que pour chaque nouvelle couleur, on va parcourir une fois tous les sommets, et on va colorier au moins un nouveau sommet. Ainsi, l'algorithme se terminera au pire après avoir parcouru autant de couleurs que le nombre de sommets : l'algorithme termine toujours. Quant à la coloration produite, puisque chaque couleur est attribuée à un sommet en respectant la règle de coloration (on ne le colorie en k que si ses voisins déjà coloriés n'ont pas déjà la couleur k, et ses voisins non-coloriés prendront une couleur plus grande que k). Ainsi la coloration obtenue est toujours une coloration « acceptable ».

De cette preuve d'algorithme (bien que peu formalisée), découle directement un résultat déjà mentionné plus haut : pour colorier un graphe à n sommets, n couleurs suffisent. Peut-on améliorer cette majoration du nombre chromatique, en observant le comportement de cet algorithme ?

On appelle *degré* d'un sommet s dans un graphe G, le nombre d'arêtes adjacentes à s, autrement dit, le nombre d'arêtes de G dans lesquelles s apparaît. Soit s un sommet d'un graphe G, de degré $deg(s)=p$. Lors de l'exécution de l'algorithme glouton, on peut observer que s sera colorié au plus tard à l'étape de la $(p+1)$-ième couleur. En effet, dans le pire des cas, tous les sommets voisins de s auront été coloriés avant lui, et auront été coloriés chacun d'une couleur différente. Ainsi, un sommet de degré p sera colorié avec la couleur $p+1$ ou une couleur plus petite. Si l'on note $\Delta(G)$ le degré maximal des sommets d'un graphe G, on peut affirmer :

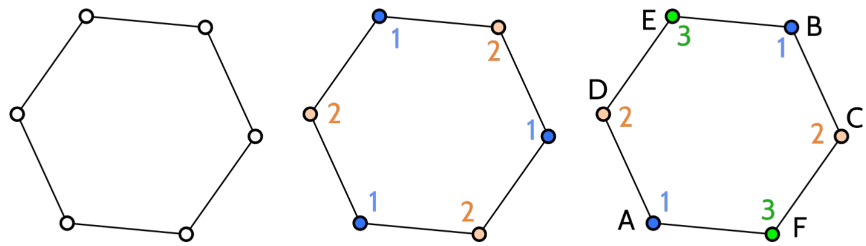
$$\chi(G) \leqslant \Delta(G)+1$$

Et si de plus on note $\omega(G)$ la taille de la plus grande clique de G (le plus grand sous-graphe complet), on peut écrire l'encadrement :

$$\omega(G) \leqslant \chi(G) \leqslant \Delta(G)+1$$

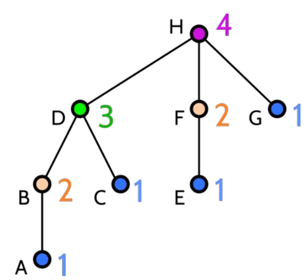
Une autre question se pose, l’algorithme glouton de coloration donne-t-il toujours une coloration optimale (ou donne-t-il au moins une coloration proche de l’optimum) ?

En explorant un peu, on peut trouver des contre-exemples où l’algorithme ne donne pas une coloration optimale. Par exemple, ci-dessous pour le cycle à 6 sommets, si l’on ordonne les sommets en commençant pas deux sommets « opposés ».



Le cycle à 6 sommets (à gauche), sa coloration optimale avec 2 couleurs (au centre), et la coloration non-optimale produite par l’algorithme glouton en suivant l’ordre alphabétique donné (à droite).

Pire, pour tout entier N, on peut construire un graphe dont le nombre chromatique est 2, mais pour lequel l’algorithme glouton produit une coloration à N couleurs pour un certain ordre. L’exemple ci-dessous illustre le cas d’un graphe 2-coloriable colorié avec $N=4$ couleurs par l’algorithme glouton. Les lecteur·rice·s qui veulent pousser la réflexion pourront vérifier que tout arbre est 2-coloriable, et pourront essayer de généraliser l’exemple (avec une récurrence ?) afin de prouver l’énoncé pour N quelconque.



Un graphe 2-coloriable, avec une coloration en 4 couleurs produite par l’algorithme glouton suivant l’ordre alphabétique sur les sommets.

On voit sur cet exemple que c’est le choix de l’ordre des sommets qui produit la coloration non optimale, et que si l’on avait choisi un autre ordre, par exemple en allant de haut en bas (et peu importe l’ordre sur un « étage »), on aurait obtenu la 2-coloration qui est optimale. En fait, cela est toujours vrai : il existe un ordre sur les sommets tel que l’algorithme glouton produise la coloration optimale. Pour s’en convaincre, prenons un graphe G de nombre chromatique $\chi(G)$, et $\mathscr{C}: V \rightarrow \{1, \dots, \chi(G)\}$ une coloration de G. Ordonnons les sommets selon le numéro de leur couleur (peu importe l’ordre pour des sommets de même couleur). Lors de l’exécution de l’algorithme glouton, pour chaque sommet s, la couleur $c(s)$ qui lui sera attribuée vérifie $c(s) \leqslant \mathscr{C}(s)$, puisque la couleur $\mathscr{C}(s)$ n’a pas été attribuée aux voisins déjà coloriés de s.

Une piste d’amélioration de l’algorithme glouton serait alors de choisir un « bon » ordre des sommets. Compte-tenu de l’importance des degrés des sommets dans la coloration (plus un sommet a de voisins, plus il y a de contraintes pour attribuer une couleur), une idée assez « naturelle » peut être de commencer par les sommets les plus contraignants, c’est-à-dire de plus haut degré. Ainsi, on ordonne les sommets du plus haut degré au plus petit pour appliquer l'algorithme glouton de coloration.

L'algorithme obtenu se nomme algorithme de Welsh et Powell. Hélas, cet algorithme non plus ne peut pas garantir d'obtenir une coloration optimale (essayez de bricoler les exemples précédents, en jouant sur les degrés des sommets, pour vous en convaincre).

Une autre idée serait, au fil de la coloration, d'ajuster l'ordre des sommets non encore coloriés : puisque la contrainte n'est pas tant le degré d'un sommet que le nombre de couleurs différentes déjà présentes chez ses sommets voisins, on peut vouloir traiter les sommets qui ont le plus grand nombre de couleurs dans leur voisinage en priorité. Cela donne l'algorithme appelé *DSATUR*.

À chaque étape de la coloration, pour tout sommet non colorié, on pose $DSAT(s)$ le nombre de couleurs différentes présentes dans les sommets (coloriés) adjacents à s. L'algorithme, sur le même principe que l'algorithme glouton, est le suivant :

Algorithme DSATUR

Au départ, on classe les sommets par degrés décroissants.

On colorie le premier sommet avec la première couleur.

Tant que tous les sommets ne sont pas coloriés :

- Prendre le sommet avec le DSAT le plus élevé (et le plus haut degré si besoin)
- Colorier ce sommet avec la plus petite couleur possible.

Là encore, l'algorithme ne permet pas d'obtenir à coup sûr une coloration optimale.

En réalité, on sait qu'on ne peut pas vraiment trouver un algorithme glouton qui résolve à coup sûr le problème de coloration optimale de graphe. Pour être plus précis, le problème de décision associé (à savoir, étant donné un graphe G et un entier k, déterminer si G est k-coloriable) est un problème *NP-complet* dès lors que $k \geqslant 2$. De façon brève, il n'existe pas d'algorithme polynomial (comme le sont les algorithmes gloutons) pour déterminer le nombre chromatique d'un graphe arbitraire (sous réserve que la conjecture « $P \neq NP$ », à un million de dollars, soit vraie).

Ceci ne nous dit pas que nos algorithmes gloutons ne donnent jamais de solution optimale ou ne permettent jamais de s'en approcher, dans certains cas. Et cela ne nous dit pas non plus qu'il n'existe pas des graphes ou des familles de graphes pour lequel déterminer le nombre chromatique est « facile ». D'ailleurs, qu'en est-il dans notre cas de transport de produits chimiques ?

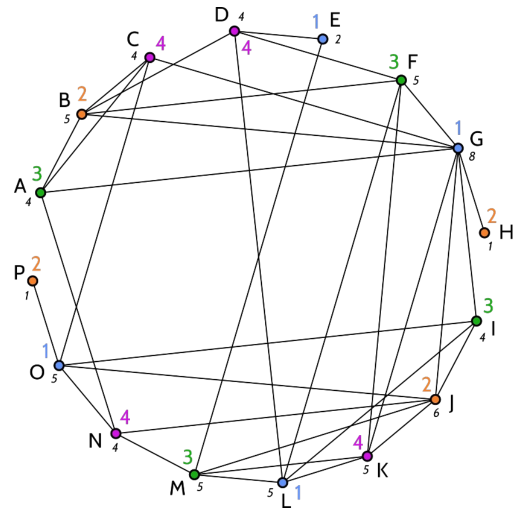
4. Revenons à notre problème initial

Revenons au graphe G_I et appliquons l'inégalité qui encadre le nombre chromatique $\chi(G_I)$ entre la taille de la plus grande clique $\omega(G_I)$ et le degré maximal des sommets plus un, $\Delta(G_I)+1$.

Le degré maximal des sommets est 8, atteint par le sommet G.

Rechercher la plus grande clique d'un graphe n'est pas un problème facile (en fait, c'est aussi un problème NP-complet). Mais on a vu que si l'on trouve une clique, même si ce n'est pas la plus grande, elle donnera un minorant de $\chi(G_I)$. En observant un peu le tableau ou le graphe d'incompatibilités, on peut remarquer que les sommets A, B, C et G sont tous incompatibles deux à deux, ils forment une clique de G_I. On peut donc déjà affirmer $4 \leqslant \chi(G_I) \leqslant 9$. En fait, notre coloration précédente en 5 couleurs nous permet de raffiner cet encadrement : $4 \leqslant \chi(G_I) \leqslant 5$.

Appliquons l'algorithme de Welsh et Powell au graphe G_1. On traite donc les sommets dans l'ordre décroissant de leurs degrés (et ordre alphabétique en cas d'égalité de degrés) : G, J, B, F, K, L, M, O, A, C, D, I, N, E, H, P. L'algorithme glouton fournit alors la coloration à 4 couleurs suivante.



Le graphe d'incompatibilités G_1 (les degrés des sommets sont indiqués en petit et en italique) et la coloration à 4 couleurs (numérotées de 1 à 4) produite par l'algorithme de Welsh et Powell.

Nous obtenons donc une répartition des produits en 4 camions : {E, G, L, O}, {B, H, J, P}, {A, F, I, M}, {C, D, K, N}. Cette répartition est optimale, puisqu'on a vu qu'il fallait au moins 4 camions.

Solution : la solution optimale de transport des produits utilise 4 camions.

On aurait aussi pu trouver cette solution en manipulant directement le tableau des compatibilités et incompatibilités des produits chimiques, notamment en trouvant une répartition en 4 camions et en trouvant la minoration du nombre de camions par 4. En fait, plusieurs des raisonnements présentés sur les graphes peuvent se mettre en œuvre avec le tableau des produits, notamment si on propose ce problème à des classes de collégien·ne·s ou de lycéen·ne·s, comme activité de résolution de problème.

La coloration de graphe est un problème classique de théorie des graphes, permettant de modéliser de nombreux problèmes ; pour la présentation d'une solution dans Petit x, il était aisé de se ramener aux graphes et intéressant de présenter quelques résultats généraux.

5. Quelques remarques algorithmiques

Prenons un instant pour réfléchir à la place qu'ont eue les algorithmes dans l'étude du nombre chromatique de graphes.

Nous avons rencontré :

- des algorithmes esquissés pour des cas particuliers qui deviennent des moyens de prouver certaines propriétés dans le cas général ;
- des algorithmes qui sont des outils qu'on peut appliquer à des cas particuliers, pour déterminer des propriétés des objets étudiés ;

- des algorithmes qu'on applique à des cas génériques, et qui permettent de produire des résultats généraux ;
- des algorithmes qui interviennent dans une preuve et des preuves qui découlent de l'étude d'algorithmes ;
- des algorithmes objets de preuve (notamment de correction et de terminaison), et supports de raisonnements ;
- des algorithmes qui deviennent objets d'étude à part entière, et dont on explore les propriétés, les limites, l'efficacité…

Ce problème peut être utilisé en formation des enseignants, pour initier à la théorie des graphes bien sûr, mais surtout pour entrer dans une réflexion d'ordre épistémologique sur la place et le rôle que peuvent avoir les algorithmes en mathématiques.

Pour cela, comme pour un usage en classe, certains des choix de variables didactiques faits ici sont importants : taille des données suffisamment grande pour faire émerger des procédures qui esquissent des algorithmes ; représentation des données en tableau et non en graphe, sans donner plus d'importance aux compatibilités qu'aux incompatibilités ; ordre initial des produits ; existence ou non de configurations simples à identifier permettant obtenir des minorations, et existence ou non d'une minoration accessible permettant de prouver l'optimalité.

Pour terminer, mentionnons deux références bibliographiques, autour de la théorie des graphes :

Aldon, G., Meny, J.-M. & Xavier, L. (2005). *Introduction à la théorie des graphes, butinage graphique*. CRDP Lyon.
https://publimath.univ-irem.fr/numerisation/LY/ILY05001/ILY05001.pdf

Cartier, L. (2008). *Le graphe comme outil pour enseigner la preuve et la modélisation*. [Thèse de doctorat, Université Joseph Fourier].
https://theses.hal.science/tel-00416598v1