Optima locaux garantis pour l'approximation différentielle

Jérôme Monnot* Vangelis Th. Paschos Sophie Toulouse
{monnot,paschos,toulouse}@lamsade.dauphine.fr

Résumé

Dans un premier temps, nous introduisons la classe GLO[δ] des problèmes de NP qui garantissent leurs optima locaux pour le rapport différentiel d'approximation polynomiale. Ensuite, nous démontrons qu'un certain nombre de problèmes combinatoires bien connus appartiennent à cette classe tandis que d'autres, aussi bien connus que les précédents, n'y appartiennent pas. Enfin, en s'appuyant sur ces résultats, nous situons la classe GLO[δ] dans le paysage des classes d'approximabilité.

We first introduce the class GLO[δ] ; it includes optimization problems that guarantee the quality of their local optima, with respect to the differential approximation ratio. Next, we show that a certain number of well-known combinatorial problems belong to this class, while other ones do not. Finally, based upon the results obtained, we place GLO[δ] in the scene of the approximability classes.

1 Introduction

Les problèmes d'optimisation auxquels on s'attache dans cet article sont le pendant *optimisation* des problèmes de décision de NP. Un problème Π est caractérisé par l'ensemble I_Π de ses instances, l'union $\text{Sol}_\Pi = \cup_{I \in I_\Pi} \text{Sol}_\Pi(I)$ des ensembles des solutions réalisables associées aux instances I, de la fonction à optimiser m_Π sur ces ensembles et du sens opt_Π de l'optimisation (*maximisation* ou *minimisation*). Par la suite, on notera $|I|$ la taille de l'instance I. Un problème Π appartenant à la classe NPO est un quadruplet $(I_\Pi, \text{Sol}_\Pi, m_\Pi, \text{opt}_\Pi)$ qui vérifie :

1. I_Π est reconnaissable en temps polynomial en $|I|$;
2. $\exists\ p_\Pi$ polynôme tel que, $\forall I \in I_\Pi$, $\text{Sol}_\Pi(I) \subseteq \{0,1\}^{p_\Pi(|I|)}$;
3. $\forall I$, $\forall s$, on sait décider en temps polynomial en $|I|$ si s est réalisable pour I ; de plus, $\forall I \in I_\Pi$, on sait déterminer une solution $\text{triv}_\Pi(I)$ en temps polynomial en $|I|$;
4. $m_\Pi : I_\Pi \times \text{Sol}_\Pi \to \mathbb{N}$ est calculable en temps polynomial en $|I|$;
5. $\text{opt}_\Pi \in \{\min, \max\}$.

Etant donné une instance I d'un problème, deux valeurs particulières nous intéressent pour la résolution de I : celles de la meilleure et de la pire solution. Soit Π un problème de NPO et I une instance de Π ; on définit les valeurs $\beta_\Pi(I)$ et $\omega_\Pi(I)$ par $\beta_\Pi(I) = \text{opt}_\Pi\,\{m_\Pi(I,s) : s \in \text{Sol}_\Pi(I)\}$ et $\omega_\Pi(I) = \overline{\text{opt}}_\Pi\,\{m_\Pi(I,s) : s \in \text{Sol}_\Pi(I)\}$ où $\forall \Pi \in \text{NPO}$, $\text{opt}_\Pi = \max \Rightarrow \overline{\text{opt}}_\Pi = \min$ et $\text{opt}_\Pi = \min \Rightarrow \overline{\text{opt}}_\Pi = \max$.

La sous-classe des problèmes qui nous intéressent plus particulièrement dans cet article est la classe NPO-PB des problèmes polynomialement bornés de NP. Cette classe, introduite dans [18], est la classe des problèmes pour lesquels il existe un polynôme q_Π tel que $\forall I \in I_\Pi$, $\beta_\Pi(I) \leqslant q_\Pi(|I|)$.

*LAMSADE, Université Paris-Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France

Ici, nous généralisons la définition de [18] en prenant en compte des bornes sur l'ensemble des valeurs réalisables, ce qui se traduira par une borne sur la valeur de la pire solution. Dans ce qui suit, un problème $\Pi \in$ NPO appartient à NPO-PB si et seulement s'il existe un polynôme q_Π tel que, $\forall I \in I_\Pi$, $\max\{\omega_\Pi(I), \beta_\Pi(I)\} \leqslant q_\Pi(|I|)$.

L'idéal serait de savoir résoudre les problèmes de NPO en temps polynomial ; malheureusement, la présomption est forte de penser que P $\neq$ NP ou encore PO $\neq$ NPO où PO, version optimisation de la classe P, désigne l'ensemble des problèmes de NPO que l'on sait résoudre en temps polynomial. C'est pourquoi on essaie d'approcher seulement, mais au mieux, ces problèmes difficiles. L'approximation polynomiale a pour mission de résoudre les problèmes selon les deux exigences : (i) en termes de temps, de garantie d'une exécution rapide (complexité polynomiale des algorithmes), (ii) en termes de performance, de garantie d'un certain niveau d'approximation. Un algorithme approché n'est ni plus ni moins qu'un algorithme qui donne une solution réalisable au problème posé. Un tel algorithme est polynomial s'il se déroule, pour toute instance, en un temps polynomial en la taille de l'instance. Soit Π un problème de NPO et `A` un algorithme approché pour Π ; on notera respectivement pour une instance I de ce problème $\lambda_{\mathtt{A}}(I)$, $\beta_\Pi(I)$ et $\omega_\Pi(I)$ les valeurs de la solution donnée par l'algorithme `A`, de l'optimum et de la pire solution sur I. L'estimation de la qualité d'un algorithme est faite à l'aide de rapports d'approximation.

Le rapport classique consiste à comparer la valeur de la solution donnée par l'algorithme à la valeur de l'optimum. La qualité d'une solution approchée pour I, en approximation classique, serait donc donnée par l'estimation du rapport :

$$\rho_\Pi^{\mathtt{A}}(I) = \begin{cases} \frac{\lambda_{\mathtt{A}}(I)}{\beta_\Pi(I)} & \text{opt}_\Pi = \max \\ \frac{\beta_\Pi(I)}{\lambda_{\mathtt{A}}(I)} & \text{opt}_\Pi = \min \end{cases}$$

et la performance de `A` pour Π par : $\rho_\Pi^{\mathtt{A}} = \inf_{I \in I_\Pi} \left\{\rho_\Pi^{\mathtt{A}}(I)\right\}$.

Le rapport différentiel, moins communément usité mais qui a déjà donné lieu à des résultats convaincants, se réfère à deux points de repère : l'optimum mais aussi la pire des solutions. En différentiel, il est tout aussi important de s'éloigner de la valeur d'une pire solution que de s'approcher de la valeur d'une solution optimale. Sur une instance I, il est donné par :

$$\delta_\Pi^{\mathtt{A}}(I) = \frac{|\omega_\Pi(I) - \lambda_{\mathtt{A}}(I)|}{|\omega_\Pi(I) - \beta_\Pi(I)|}$$

et la performance de `A` pour Π par : $\delta_\Pi^{\mathtt{A}} = \inf_{I \in I_\Pi} \left\{\delta_\Pi^{\mathtt{A}}(I)\right\}$. Il est alors équivalent de démontrer :

$$\delta_\Pi^{\mathtt{A}} \geqslant \epsilon \Longleftrightarrow \begin{cases} \lambda_{\mathtt{A}}(I) \leqslant \epsilon\beta_\Pi(I) + (1-\epsilon)\omega_\Pi(I) & \text{opt}_\Pi = \min \\ \lambda_{\mathtt{A}}(I) \geqslant \epsilon\beta_\Pi(I) + (1-\epsilon)\omega_\Pi(I) & \text{opt}_\Pi = \max \end{cases} \quad (1)$$

L'objectif principal de la théorie de l'approximation polynomiale est de classer les algorithmes en fonction du meilleur rapport (connu) qu'ils garantissent ; c'est ce qui est appelé niveau d'approximation. Nous donnons ci-après une liste des principaux niveaux mentionnés dans la littérature du domaine :

Algorithme à rapport constant : le rapport d'approximation est une constante ne dépendant pas de I, *i.e.*, ne dépendant d'aucun paramètre de l'instance du problème ;

Schéma d'approximation polynomial : une suite d'algorithmes indicée par $\epsilon > 0$ garantissant le rapport $1-\epsilon$; chaque algorithme est polynomial ;

Schéma complet d'approximation polynomial : schéma d'approximation dont la complexité est polynomiale en $|I|$ et en $1/\epsilon$;

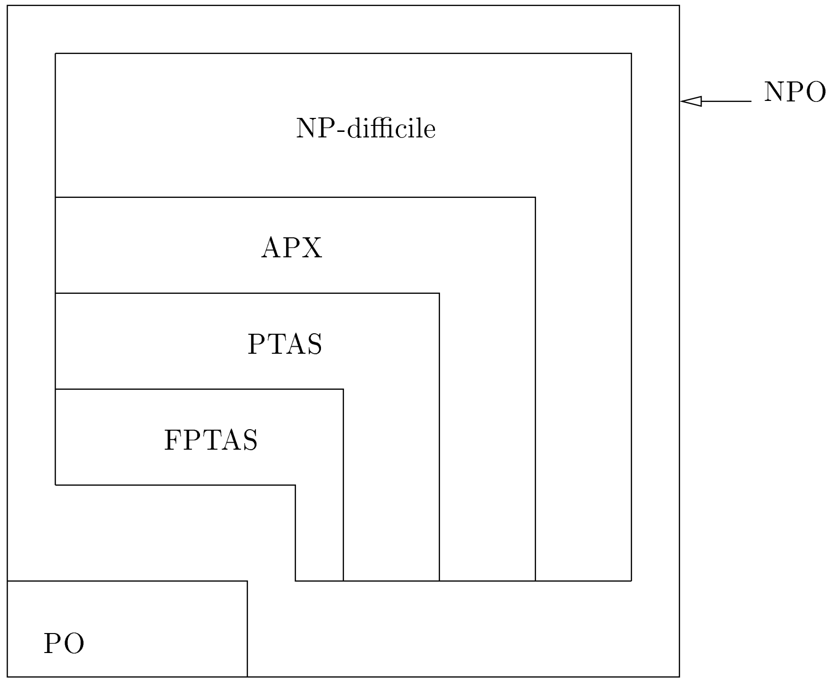

FIG. 1 – Le monde de l'approximabilité sous la conjecture P $\neq$ NP

Algorithmes à rapport logarithmique : le rapport d'approximation est supérieur ou égal à $O(1/\log|I|)$;

Algorithmes à rapport $O(n^{\epsilon-1})$: le cas où $\exists\epsilon\in]0,1[$ tel que le rapport d'approximation est supérieur ou égal à $O(n^{\epsilon-1})$.

La classification considérée pour les algorithmes approchés peut être étendue aux problèmes si on considère que le rapport d'approximation d'un problème Π est le rapport du meilleur algorithme résolvant Π. Ainsi, on peut considérer une structure pour NPO par rapport à l'approximabilité de ses problèmes. La figure 1 représente les classes d'approximabilité les plus usuelles, évidemment sous la conjecture P $\neq$ NP. Par analogie, nous notons PO la classe des problèmes d'optimisation polynomiaux.

Les principales classes d'approximabilité rencontrées dans la littérature sont les suivantes :

APX[ρ], APX[δ] : la classe des problèmes NPO admettant un algorithme à rapport classique ou différentiel constant ;

PTAS[ρ], PTAS[δ] : la classe des problèmes NPO admettant un schéma d'approximation polynomial classique ou différentiel ;

FPTAS[ρ], FPTAS[δ] : la classe des problèmes NPO admettant un schéma complet d'approximation polynomial classique ou différentiel ;

Log-APX[ρ], Log-APX[δ] : la classe des problèmes NPO admettant un algorithme approché à rapport, classique ou différentiel, logarithmique en I ;

Poly-APX[ρ], Poly-APX[δ] : la classe des problèmes NPO admettant un algorithme approché garantissant un rapport, classique ou différentiel, qui est un polynôme en la taille de l'instance.

Notons que, puisque la valeur d'un rapport d'approximation appartient à $\mathbb{R}^+$, il existe tout un *continuum* de classes d'approximabilité. Pour l'approximation classique, nous avons les relations d'inclusion stricte (à moins que P = NP) :

$$\text{FPTAS}[\rho] \subset \text{PTAS}[\rho] \subset \text{APX}[\rho] \subset \text{Log-APX}[\rho] \subset \text{Poly-APX}[\rho]$$

Pour l'approximation différentielle, nous avons, à moins toujours que P = NP, les relations d'inclusion stricte suivantes :

$$\text{FPTAS}[\delta] \subset \text{PTAS}[\delta] \subset \text{APX}[\delta] \subset \text{Poly-APX}[\delta]$$

Par contre, nous ne connaissons pas pour l'instant des problèmes à rapport logarithmique.

Le contexte général de l'approximation polynomiale étant brièvement présenté, nous remarquons que le but de cet article n'est pas de chercher les meilleurs résultats possibles d'approximation différentielle pour les problèmes traités mais de montrer que, pour une large classe de problèmes bien connus de NPO contenue dans APX[δ], toutes les solutions optimales localement sont « proches » des solutions (globalement) optimales. Plus précisément, nous définissons une classe d'approximabilité qu'on appelle GLO[δ] contenant les problèmes vérifiant la propriété suivante : *le rapport différentiel de toute solution optimale localement (par rapport à un voisinage proprement défini plus loin) est borné inférieurement par une constante qui ne dépend d'aucun paramètre de l'instance du problème que l'on considère.* Notons que l'équivalent de ce travail pour le rapport classique (*i.e.* la définition de la classe GLO[ρ]) présenté dans [5, 6] est étendu dans [3]. C'est à notre connaissance la première fois que les thématiques de la recherche et de l'optimisation locale sont systématiquement traitées en approximation classique. Notre travail est l'extension de l'étude des articles [5, 6] dans le cadre du rapport différentiel. Les deux cadres, malgré leur objet d'étude commun, la théorie de l'approximation polynomiale, diffèrent radicalement, tant par rapport à leur logique et aux méthodologies combinatoires impliquées qu'aux résultats obtenus dans chacun d'eux. Par conséquent, la définition et l'étude de la classe GLO[δ] induisent leurs propres analyses et résultats qui diffèrent, comme le lecteur peut le constater, de ceux proposés dans [5, 6].

Le plan de l'article est le suivant. Dans la section 2, nous donnons les bases de l'optimisation locale. Les sections 3 et 4 étudient le comportement de l'approximabilité de quelques problèmes de base de l'optimisation combinatoire vis-à-vis de l'optimisation locale. Plus exactement, pour chacun des problèmes Π étudiés, nous répondons à la question : *garantit-il le plus mauvais optimum local (vis-à-vis d'un type de voisinage) de* Π *un rapport d'approximation différentielle constant ?* De ce point de vue, l'objectif de cet article est d'établir une première cartographie des problèmes en fonction de leur bonne approximation différentielle ou non (*i.e.*, à rapport constant ou non constant) avec des algorithmes de recherche locale. Ainsi, si l'on note GLO[δ] la classe des problèmes où la question précédente est oui, la section 3 expose des résultats d'appartenance à cette classe tandis que la section 4 propose des résultats de non appartenance. L'ensemble de ces résultats est illustré par la figure 2. Finalement, la section 5 positionne la classe GLO[δ] dans le paysage des classes d'approximation et la section 6 donne des perspectives de recherches futures afin de mieux appréhender cette classe et d'en dessiner les contours plus finement.

Nous avons déjà mentionné qu'il n'est nullement question ici de donner *les meilleurs résultats* d'approximation différentielle possible par l'utilisation d'algorithme de recherche locale. De tels résultats existent pour le rapport classique (*cf.* [14, 9, 8]) et constituent une voie de recherche future pour les travaux sur la mesure différentielle.

2 La recherche et l'optimisation locale en quelques mots

Avant de parler d'optima locaux, il faut définir une notion de proximité : les fonctions voisinage nous permettent de le faire. Informellement, soit Π un problème d'optimisation, un voisinage $\mathcal{V}$ pour Π est une fonction qui, à toute solution s de toute instance I de Π, associe un

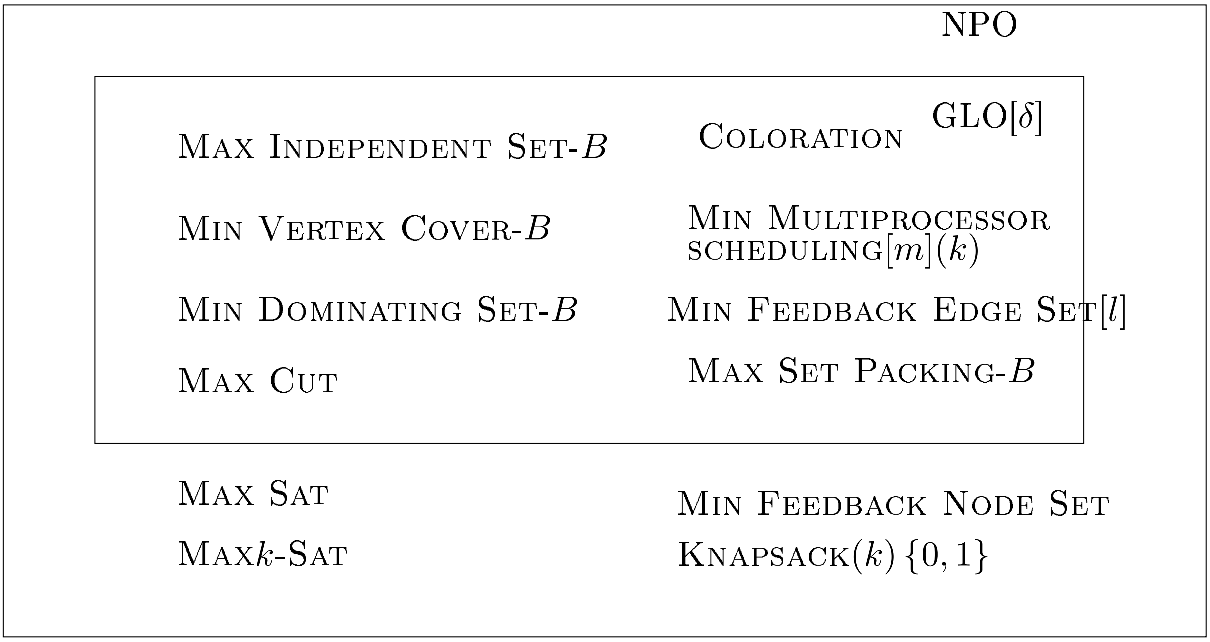

FIG. 2 – Un résumé des résultats principaux de l'article

sous-ensemble $\mathcal{V}(I,s)$ de solutions autour de s ; les éléments de $\mathcal{V}(I,s)\setminus\{s\}$ sont appelés solutions voisines de s. Étant donné un voisinage $\mathcal{V}$ sur Π, un optimum local d'une instance I n'est autre qu'une solution au moins aussi bonne que ses voisins. On ne cherche pas une solution optimale sur un sous-ensemble statique mais une solution optimale sur un ensemble *centré autour de cette solution* : il s'agit donc d'un sous-ensemble évolutif en fonction de la solution courante et l'optimalité locale ne se définit pas à partir d'un sous-ensemble défini *a priori* mais à partir de la solution considérée et des voisins qu'elle désigne pour une structure de voisinage considérée. Afin d'éviter d'avoir à préciser les sens d'optimisation des problèmes manipulés, nous utiliserons par la suite les signes $\succeq$ (resp. $\succ$) pour signifier qu'une solution est au moins aussi bonne (resp. strictement meilleure) qu'une autre[1].

Definition 1 *La solution $\tilde{s}$ est optimum local de I relativement à $\mathcal{V}$ si et seulement si, $\forall s \in \mathcal{V}(I,\tilde{s})$, $m_\Pi(I,\tilde{s}) \succeq m_\Pi(I,s)$.*

Disposant d'un problème et d'un voisinage, il est facile de déterminer un optimum local. Partant d'une solution, on explore son voisinage à la recherche d'une éventuelle meilleure solution, et ainsi de suite jusqu'à ce que la solution courante n'ait pas de meilleur voisin : c'est un optimum local. Cette recherche est formalisée sous le nom de `LSA` (*Local Search Algorithm*) : algorithme de recherche locale. Étant donné un problème Π à résoudre, un voisinage $\mathcal{V}$ pour Π et un algorithme polynomial approché $\mathtt{A}_\Pi$ pour Π, un `LSA` se déroule comme suit :

1. poser $s_1 = \mathtt{A}_\Pi(I)$; $s = s_1$; ok = faux ;
2. tant que $\neg$ok faire :
   (a) déterminer $\mathcal{V}(I,s)$;
   (b) s'il existe $s' \in \mathcal{V}(I,s)$ telle que $m_\Pi(I,s') \succ m_\Pi(I,s)$, alors $s = s'$, sinon ok = vrai ;

[1] $\mathrm{opt}_\Pi = \min \Rightarrow \left\{\begin{array}{ccc} \succeq & = & \leqslant \\ \succ" & = & < \end{array}\right.$ $\mathrm{opt}_\Pi = \max \Rightarrow \left\{\begin{array}{ccc} \succeq & = & \geqslant \\ \succ & = & > \end{array}\right.$

3. retourner s.

La solution initiale s_1 est obtenue en temps $p(|I|)$. La boucle « tant que » (item 2) qui consiste en la construction du voisinage, nécessite un temps de l'ordre de $|\mathcal{V}(I,s)| \times t_{\mathcal{V}}(I)$ si $t_{\mathcal{V}}(I)$ représente le temps maximum de passage d'une solution à une solution voisine pour le voisinage $\mathcal{V}$ sur l'instance I. L'évaluation des voisins demande un temps au plus $|\mathcal{V}(I,s)| \times t_m(I)$ si $t_m(I)$ est le temps d'évaluation d'une solution sur l'instance I, supposé polynomial par définition de NPO. Enfin, la boucle « tant que » est itérée au plus $|\omega_\Pi(I) - \beta_\Pi(I)|$ fois : les coefficients de la fonction objectif ainsi que les variables étant à valeurs entières, la valeur de la solution augmente si $\mathrm{opt}_\Pi = \max$, diminue sinon d'au moins une unité à chaque itération.

Quelques mots sur la complexité de l'algorithme `LSA`. Il est nécessaire, pour le parcours du voisinage de la solution courante, que celui-ci soit de taille polynomiale ; il faut de plus que la construction de tout voisin à partir de la solution courante soit elle-même polynomiale. On traduit ces exigences par les deux conditions nécessaires : (i) il existe un polynôme p_1 tel que, $\forall I$, $\forall s \in \mathrm{Sol}_\Pi(I)$, $|\mathcal{V}(I,s)| \leqslant p_1(|I|)$ et (ii) il existe un polynôme p_2 tel que, $\forall I$, $t_{\mathcal{V}}(I) \leqslant p_2(|I|)$. Quant au nombre d'itérations de la boucle « tant que » de l'algorithme `LSA`, on ne peut s'assurer de sa polynomialité qu'en bornant le nombre d'étapes que nécessiterait le passage d'une solution de valeur $m_\Pi(I, \mathtt{A}_\Pi(I))$ à un optimum de valeur $\beta_\Pi(I)$. Une condition suffisante pour que l'item 2 soit polynomial est de n'avoir qu'un nombre polynomial de valeurs possibles pour les solutions du problème entre les valeurs $m_\Pi(I, \mathtt{A}_\Pi(I))$ et $\beta_\Pi(I)$, c.-à-d. qu'il existe un polynôme p qui vérifie, pour toute instance I, la relation suivante : si $\alpha_I = m_\Pi(I, s_1)$ et $n(\alpha_I) = |\{m_\Pi(I,s) \succeq \alpha_I, s \in \mathrm{Sol}_\Pi(I)\}|$, alors $n(\alpha_I) \leqslant p(|I|)$. Malheureusement, cette condition n'est pas vérifiable *a priori* en temps polynomial.

Nous distinguons deux façons de la vérifier : soit le problème lui-même n'admet qu'un nombre polynomial de valeurs distinctes pour toute instance (c'est notamment vrai si le problème est polynomialement borné), soit c'est à nous qu'il revient de se rapprocher de la valeur optimale en partant d'une bonne solution $\mathtt{A}_\Pi(I)$. Effectivement, si la valeur d'une solution optimale est bornée par un polynôme, i.e., $\forall I \in I_\Pi$, $|\beta_\Pi(I)| \leqslant p_\beta(|I|)$, il suffirait alors, pour rendre la recherche locale polynomiale, de partir d'une solution initiale « *assez proche de l'optimum* », soit de disposer d'un algorithme $\mathtt{A}_\Pi$ approché à rapport constant r pour le rapport classique dans le sens où, si $\mathrm{opt}_\Pi = \max$ (le cas $\mathrm{opt}_\Pi = \min$ se traite exactement de la même façon), alors $\lambda_{\mathtt{A}}(I) \geqslant r\beta_\Pi(I)$, ce qui implique $|\lambda_{\mathtt{A}}(I) - \beta_\Pi(I)| \leqslant |1-r|\beta_\Pi(I) \leqslant |1-r|p_\beta(|I|)$. Cette discussion nous amène aux deux conditions suffisantes suivantes concernant la polynomialité d'un algorithme `LSA` : (i) il existe un polynôme p tel que, $\forall I$, le nombre de solutions réalisables de I est inférieur ou égal à $p(|I|)$ et (ii) $\Pi \in \mathrm{APX}[\rho]$ et $\beta_\Pi(I)$ est borné par un polynôme.

La conclusion est qu'on ne peut assurer *a priori* l'efficacité en temps d'un algorithme de recherche locale : cela dépend fortement du problème traité (ou de la restriction d'un problème général à une famille d'instances particulières) ainsi que de la définition de voisinage considérée. Pour plus de renseignements sur la difficulté d'obtenir des optima locaux en temps polynomial pour des problèmes qui sont même en dehors de la classe NPO-PB, cf. [28, 22, 20].

On l'a vu, le voisinage est la notion centrale de la recherche locale ; or, un voisinage n'est autre qu'un ensemble de points autour d'une solution donnée. Cependant, notre but étant de construire, à partir de ces voisinages, des algorithmes polynômiaux, nous ne nous intéresserons qu'à des voisinages polynômiaux, c'est-à-dire qu'une solution, quel que soit le problème traité, aura un *nombre polynomial de voisins* pour assurer de manière simple et systématique l'utilisation de procédures polynomiales en temps. Notons néanmoins que de grands efforts ont été entrepris par différents auteurs pour élargir les voisinages à des structures moins triviales. Citons par exemple, la mise en évidence de la classe PLS ([25, 28]), des problèmes « étoilés » ([13]) et, plus récemment, l'existence de voisinages de taille exponentielle qui sont pourtant examinables en temps polynomial ([16, 24, 1]). Cela nous amène à la définition suivante du voisinage.

Definition 1. Soit Π un problème de NPO. Un voisinage sur Π est une fonction $\mathcal{V} : I_\Pi \times \mathrm{Sol}_\Pi \rightarrow \mathcal{P}(\mathrm{Sol}_\Pi)$ où $\mathcal{P}(X)$ désigne l'ensemble des parties de X qui, à toute solution $s \in \mathrm{Sol}_\Pi(I)$ de toute instance $I \in I_\Pi$, associe un sous-ensemble $\mathcal{V}(I,s) \subseteq \mathrm{Sol}_\Pi(I)$ de solutions de I contenant s, de taille au plus $p_\mathcal{V}(|I|)$, où $p_\mathcal{V}$ est un polynôme. ∎

Par ailleurs, nous souhaitons définir une famille de voisinages qui permette de reconnaître une classe de problèmes dont la spécificité serait d'avoir de bons optima locaux relativement à un rapport d'approximation donné. Si Π est dans la classe APX[r] où $r \in \{\rho, \delta\}$, pour un certain algorithme `A`, la fonction qui, à toute instance I et toute solution s associe l'ensemble $V(I,s) = \{s, \mathtt{A}(I)\}$, est bien un voisinage conformément à la définition que nous venons d'en donner ; or, tout optimum local pour ce voisinage faisant au moins aussi bien que la solution $\mathtt{A}(I)$ garantit le même rapport constant d'approximation que `A` et cela quel que soit le rapport considéré.

Une telle définition nous amènerait à considérer simplement les classes APX[r] alors que l'on souhaiterait justement, par l'étude des optima locaux, révéler des propriétés structurelles de ces classes ! Il faut donc trouver une définition pertinente pour la *reconnaissance* de problèmes dont les optima locaux auraient un *bon* ou un *mauvais comportement* vis-à-vis des rapports d'approximation. Enfin, il faut également, pour être en mesure de construire ces voisinages, savoir quelle est, d'un point donné, la visibilité que l'on s'offre de l'ensemble des solutions.

Nous avons vu que la mise en œuvre d'un algorithme `LSA` polynomial était conditionnée par le nombre de voisins d'une solution mais aussi par le temps de construction d'un voisin à partir de toute solution. Les voisinages manipulés pour la résolution des problèmes, quel que soit le cadre de recherche (métaheuristiques comme approximation polynomiale), suivent souvent le principe qui consiste à obtenir les voisins d'une solution en changeant un nombre constant de composantes de celle-ci (ajout/suppression de sommets pour des problèmes de couverture, échange d'arêtes pour des problèmes de tournées ...). Ou dit autrement, à partir d'une solution donnée, on borne la visibilité de l'ensemble des solutions à une distance d'un certain nombre, constant, de transformations de la solution courante.

Ces voisinages peuvent être regroupés sous le formalisme des *voisinages h-bornés* qui offrent une réponse à ces exigences et que nous définissons à présent. Mais, avant de manipuler la notion de borne, il faut définir une *distance* entre les solutions des instances des problèmes de NPO.

Definition 2. $\forall \Pi \in$ NPO, $\forall I \in I_\Pi$, $\forall s,t \in \mathrm{Sol}_\Pi(I)$, $d(s,t) = \|s-t\|_1 = \sum_{i=1}^{|I|} |s_i - t_i|$. Soit h une constante ; le voisinage $\mathcal{V}_\Pi^h$ h-borné pour Π est défini en toute solution s comme l'ensemble des solutions au plus h-distantes de s : $\forall h$, $\forall I \in I_\Pi$, $\forall s \in \mathrm{Sol}_\Pi(I)$, $\mathcal{V}_\Pi^h(I,s) = \{t \in \mathrm{Sol}_\Pi(I) : d(s,t) \leqslant h\}$. ∎

Par abus de langage, on qualifiera de h-borné tout voisinage $\mathcal{V}$ contenu dans un voisinage h-borné, soit toute fonction $\mathcal{V}$ vérifiant : $\forall I \in I_\Pi$, $\forall s \in \mathrm{Sol}_\Pi(I)$, $\mathcal{V}(I,s) \subseteq \mathcal{V}_\Pi^h(I,s)$.

Lorsque l'on parlera, pour une constante h et un problème Π donnés, *du* voisinage h-borné sans autre précision, on se référera à tout $\mathcal{V}_\Pi^h$. Pour une constante universelle h et un problème Π de NPO, soit I une instance de Π, le nombre de solutions h-distantes d'une solution quelconque $s \in \mathrm{Sol}_\Pi(I)$ est majoré par un polynôme en la taille de l'instance : $|\{t \in \mathrm{Sol}_\Pi(I) : d(s,t) \leqslant h\}| \leqslant \sum_{i=1}^{h} C_i^{p(|I|)} \leqslant p(|I|)^{h+1}$. Enfin, la construction d'une solution voisine à toute solution de toute instance demande un temps constant : les voisinages h-bornés remplissent les conditions nécessaires de déroulement polynomial d'un algorithme de recherche locale présentées ci-dessus.

3 La classe GLO[δ]

Dans cette section, on s'intéresse exclusivement aux structures de voisinages h-bornés. Nous définissons d'abord la notion d'optimum local garanti (pour des voisinages h-bornés).

Definition 3. Un problème Π garantit pour δ la valeur de ses optima locaux s'il existe, pour une constante $r \in]0,1]$ et un entier h, un voisinage h-borné $\mathcal{V}$ tel que tout optimum local $\tilde{s}$ vis-à-vis de $\mathcal{V}$ de toute instance I réalise un rapport d'approximation $\delta_{\Pi}(I, \tilde{s})$ d'au moins r. Un problème Π est dans GLO[δ] s'il garantit la qualité de ses optima locaux et si on sait, pour le voisinage qui permet d'établir cette garantie, déterminer un optimum local en temps polynomial. ∎

Dans la classe GLO[δ] (il en est de même pour GLO[ρ]), il faut distinguer deux choses : les optima locaux de qualité garantie (ce qu'on pourrait qualifier de « Stricte-GLO ») et la garantie d'obtention d'un optimum local en temps polynomial ; le but restant de trouver des solutions approchées en temps polynomial, nous nous restreindrons toujours, pour la classe GLO[δ], aux instances des problèmes pour lesquelles on sait trouver un optimum local en temps polynomial. C'est dans cet esprit que les auteurs de [6] ont placé GLO[ρ] dans NPO-PB car cette restriction est suffisante à la déduction d'algorithmes polynômiaux à rapport constant à partir du bon comportement global des optima locaux du problème général considéré.

3.1 Les premiers problèmes de GLO[δ]

Dans la proposition qui suit, nous montrons que divers problèmes bien connus font partie de la classe GLO[δ] pour des voisinages 1-bornés (*i.e.* étant donné une solution s, les solutions voisines consistent en celles résultant de s qui sont à distance 1 de s). Plus particulièrement, nous considérons dans un premier temps les problèmes MAX CUT, MAX INDEPENDENT SET-B, MIN DOMINATING SET-B, MIN VERTEX COVER-B et MAX SET PACKING-B. Considérons un graphe $G(V,E)$ et, pour deux sous-ensembles X et Y de V, notons par $\langle X, Y \rangle = \{xy \in E : x \in X \text{ et } y \in Y\}$.

Le problème MAX CUT consiste à déterminer un ensemble U de sommets qui maximise $|\langle U, V \backslash U \rangle|$. Ici, la pire solution est l'ensemble vide (de valeur 0).

Le problème MAX INDEPENDENT SET consiste à déterminer le plus grand sous-ensemble de sommets $U \subseteq V$ (ensemble stable) tel que $\langle U, U \rangle = \emptyset$. Ici aussi, la pire solution est l'ensemble vide. MAX INDEPENDENT SET-B est la variante où le graphe est à degré maximal borné par B.

Le problème MIN DOMINATING SET consiste à déterminer le plus petit sous-ensemble de sommets $U \subseteq V$ (ensemble dominant) vérifiant, pour tout sommet v de V : $v \in U$ ou $\langle \{v\}, U \rangle \neq \emptyset$. Ici, la pire solution est l'ensemble V (tous les sommets du graphe) de valeur $|V|$. MIN DOMINATING SET-B est la variante où le graphe est à degré maximal borné par B.

Le problème MIN VERTEX COVER consiste à déterminer le plus petit sous-ensemble de sommets $U \subseteq V$ (couverture de sommets) tel que $E = \langle U, V \rangle$. Ici aussi, la pire solution est l'ensemble V. MIN VERTEX COVER-B est la variante où le graphe est à degré maximal borné par B.

Considérons maintenant une collection $\mathcal{C}$ de sous-ensembles $C_1, C_2, \ldots, C_n$ d'un ensemble D. Le problème MAX SET PACKING consiste à extraire un nombre maximal de sous-ensembles $C_j \in \mathcal{C}$ d'intersection vide, soit à maximiser la quantité $p = |\mathcal{C}'|$ où, $\mathcal{C}' = \left\{C_{j_1}, C_{j_2}, \ldots, C_{j_p}\right\} \subseteq \mathcal{C}$ tel que, $\forall k < l \in \{1, \ldots, p\}$, $C_{j_k} \cap C_{j_\ell} = \emptyset$. Pour ce problème, la pire solution est une famille vide de sous-ensembles de D (de valeur 0). MAX SET PACKING-B est la variante de MAX SET PACKING où le cardinal du plus grand ensemble de la famille $\mathcal{C}$ est borné supérieurement par B.

Lemma 1. *Considérons un graphe $G(V,E)$ connexe, posons $|V| = n$ et notons par Δ son degré maximal. Alors,*

1. *les problèmes* MAX INDEPENDENT SET *et* MIN VERTEX COVER *sont équi-approximables pour le rapport différentiel ; cela reste vrai également pour* MAX INDEPENDENT SET-B *et* MIN VERTEX COVER-B *;*

2. *si U est un ensemble dominant de G, alors $|U| \geqslant |V|/(\Delta+1)$; supposons maintenant que U est aussi minimal pour l'inclusion ; alors, $\overline{U} = V \setminus U$ est aussi un ensemble dominant de G et, de plus, $|U| \leqslant \Delta|V|/(\Delta+1)$.*

Preuve du point 1. Il est bien connu ([7]) que U est un stable de G, si et seulement si $V \setminus U$ est une couverture de sommets. Par conséquent, pour tout stable U de cardinal $|U|$, il existe une couverture de sommets de cardinal $n - |U|$. Par ailleurs, le rapport différentiel étant stable pour toute transformation affine de la fonction objectif ([11]), la preuve du point 1 du lemme est complète.

Preuve du point 2. Considérons un ensemble dominant U. La définition d'un tel ensemble implique $|\overline{U}| \leqslant |\langle\, U, \overline{U}\,\rangle| = \sum_{u \in U} |\langle\, \{u\}\,, \overline{U}\,\rangle| \leqslant \Delta|U|$. L'expression précédente avec $|\overline{U}| = |V| - |U|$ conclut $|U| \geqslant |V|/(\Delta+1)$.

Supposons maintenant que U soit minimal pour l'inclusion de surcroît. Démontrons d'abord que $\overline{U}$ est également un ensemble dominant. En effet, le graphe G étant supposé connexe, tout sommet $y \in U$ est relié à l'ensemble U ou son complémentaire $\overline{U}$. Or, si y n'est relié qu'à des sommets de U, $U \setminus \{y\}$ est toujours dominant, ce qui contredirait la minimalité de U ; y est donc relié à au moins un sommet de $\overline{U}$, autrement dit, « $\forall u \in U$, $|\langle\, \{u\}\,, \overline{U}\,\rangle| \geqslant 1$ », c.q.f.d.

Puisque maintenant $\overline{U}$ est un ensemble dominant, nous avons, par la première partie du point 2, $|\overline{U}| \geqslant |V|/(\Delta+1)$ qui implique $|U| \leqslant \Delta|V|/(\Delta+1)$. Cela termine la preuve du point 2 et du lemme.

Dans la proposition ci-dessous, nous considérons uniquement des optima locaux pour les voisinages 1-bornés. Pour les problèmes étudiés, cela reviendra à l'étude des solutions maximales ou minimales pour l'inclusion. Les analyses présentées sont les plus fines possibles puisque nous montrons que les ratios d'approximations résultants sont atteints pour certaines instances.

Proposition 1. *Pour les voisinages 1-bornés :*

1. Max Cut $\in$ GLO$[\delta]$;
2. Min Dominating Set-B $\in$ GLO$[\delta]$;
3. Max Independent Set-B, Min Vertex Cover-B $\in$ GLO$[\delta]$;
4. Max Set Packing-B $\in$ GLO$[\delta]$.

Preuve du point 1. Soit $G(V, E)$ un graphe et soit U un sous-ensemble dominant de sommets. L'ensemble U est un optimum local si et seulement si le changement d'affectation d'un sommet u de U à $\overline{U}$ ou de $\overline{U}$ à U n'améliore pas la solution, soit si U vérifie : $\forall u \in \overline{U}$, $|\langle\, \{u\}\,, \overline{U}\,\rangle| \leqslant |\langle\, \{u\}\,, U\,\rangle|$ et, $\forall u \in U$, $|\langle\, \{u\}\,, U\,\rangle| \leqslant |\langle\, \{u\}\,, \overline{U}\,\rangle|$. Les relations précédentes impliquent :

$$\sum_{u \in \overline{U}} |\langle\, \{u\}\,, \overline{U}\,\rangle| + \sum_{u \in U} |\langle\, \{u\}\,, U\,\rangle| \leqslant \sum_{u \in \overline{U}} |\langle\, \{u\}\,, U\,\rangle| + \sum_{u \in U} |\langle\, \{u\}\,, \overline{U}\,\rangle|$$

ce qui donne $2|\langle\, \overline{U}, \overline{U}\,\rangle| + 2|\langle\, U, U\,\rangle| \leqslant 2|\langle\, \overline{U}, U\,\rangle|$; cela implique $|E| - m(G,U) \leqslant m(G,U)$, *i.e.* $m(G,U) \geqslant |E|/2$. Avec $\beta(G) \leqslant |E|$ et $\omega(G) = 0$ (la pire solution pour Max Cut consiste à prendre $U = V$), on obtient $\delta \geqslant 1/2$.

Considérons maintenant l'instance résultant d'un C_4 (*i.e.* un cycle de taille 4) ; un optimum local (le moins bon) pour les voisinages 1-bornés est donné par deux sommets consécutifs du cycle et a une valeur de 2 tandis qu'un optimum global est donné par deux sommets alternés du cycle et a une valeur de 4. Cela conclut la preuve du point 1 de la proposition.

Preuve du point 2. Par la première partie du point 2 du lemme 1, $\beta(G) \geqslant |V|/(B+1)$ (l'optimum est lui-même dominant) et, par la deuxième, $m(U,G) \leqslant B|V|/(B+1)$ puisque l'on a $\Delta \leqslant B$. Par ailleurs, $\omega(G) = |V|$ (la pire solution consiste à prendre tous les sommets). On obtient donc le rapport $\delta(G,U) = (n - |U|)/(n - \beta(G)) \geqslant 1/B$.

Considérons maintenant l'instance résultant d'un graphe biparti complet $K_{1,B}$. Un optimum local pour les voisinages 1-bornés est déterminé par l'ensemble stable de taille B issu de la bipartition tandis que l'optimum global est donné par l'autre ensemble stable issu de la bipartition et vaut 1 et que la pire solution est donnée par les $B+1$ sommets. Cela termine la preuve du point 2 de la proposition.

Preuve du point 3. Par le point 1 du lemme 1, Max Independent Set-B et Min Vertex Cover-B sont équi-approximables pour le rapport différentiel. Par ailleurs, si on considère que pour Max Independent Set, $\omega(G) = 0$ et que pour Min Vertex Cover $\omega(G) = n$, les rapports classique et différentiel pour les deux problèmes coïncident pour tout graphe G. Cela complète la preuve du point 1 du lemme. Etant donné enfin que Max Independent Set-B$\in$ GLO[ρ] ([6]), en considérant les voisinages 1-bornés, la preuve du point 3 est conclue.

Preuve du point 4. Considérons une instance $I(D, \mathcal{C})$ de Max Set Packing et construisons une instance $f(I) = G(V, E)$ de Max Independent Set en associant, à chaque sous-ensemble C_j de la famille $\mathcal{C}$, un sommet v_j et en créant une arête $v_j v_{j'}$ à chaque fois que deux ensembles C_j et $C_{j'}$ correspondants comporteront des éléments communs : $f : I = (D, \{C_1, \ldots, C_n\}) \mapsto f(I) = G(V, E)$ avec $V = \{v_1, \ldots, v_n\}$ et $E = \{v_j v_{j'} : j \neq j', C_j \cap C_{j'} \neq \emptyset\}$. La construction de l'instance $f(I)$ prend un temps au plus $n^2 B^2$ pour considérer les couples d'ensembles $(C_j, C_{j'})$ et comparer leurs éléments. Les deux instances I et $f(I)$ ont le même ensemble de solutions $\{0, 1\}^n$ et une telle solution s sera interprétée de la façon suivante :

- $s_i = 1 \Leftrightarrow C_i \in \mathcal{C}'$ pour l'instance I de Max Set Packing ;
- $s_i = 1 \Leftrightarrow v_i \in U$ pour l'instance $f(I)$ de Max Independent Set.

Ainsi, la fonction g est la fonction identité : $g = \mathrm{Id} : \{0, 1\}^n \to \{0, 1\}^n$, $s \mapsto g(s) = s$. Par construction, la relation « s stable dans $G \Leftrightarrow s$ solution de Max Set Packing dans $\mathcal{C}$ » est vérifiée : « s stable dans G » équivaut « $\forall j < j'$, $s_j \times s_{j'} = 1$ implique $v_j v_{j'} \notin E$ », ce qui équivaut « $\forall j < j'$, $s_j \times s_{j'} = 1$ implique $C_j \cap C_{j'} = \emptyset$ », ce qui à son tour est équivalent à « s solution de Max Set Packing dans $(\mathcal{C}, D)$ ».

Les deux instances ont donc le même ensemble de solutions réalisables. De plus, toute solution s a même valeur sur les instances I et $f(I)$, *i.e.* $\forall s \in \{0, 1\}^n$,

$$m(G, s) = m(\mathcal{C}, s) = \sum_{i=1}^{n} s_i$$

Par conséquent, Max Set Packing et Max Independent Set sont équi-approximables pour le rapport différentiel (et pour le rapport classique). Ainsi, il est facile de voir que, par la réduction que nous venons de décrire, si le cardinal du plus grand ensemble de la famille $\mathcal{C}$ est borné par B, alors le degré maximal de G est borné par B également. Enfin, soit k une constante ; les voisins k-distants d'une solution s sont les mêmes sur I et $f(I)$ et, en proposant pour tout voisinage $\mathcal{V}'$ considéré pour Max Independent Set le voisinage $g(\mathcal{V}')$ pour Max Set Packing, on conclut aisément que si Max Independent Set-$B \in$ GLO[δ], alors Max Set Packing-$B \in$ GLO[δ]. Le point 3 suffit alors pour conclure la preuve du point 4 et de la proposition.

Si B n'est plus seulement une borne mais également le nombre exact de sommets adjacents à tout sommet (graphes B-réguliers), le rapport d'approximation différentiel de tout optimum 1-local des problèmes de Max Independent Set-B et Min Vertex Cover-B est porté, pour des graphes connexes, à $2/(B+1)$. Effectivement, on remarque alors que, pour la couverture de sommets, toute solution d'une instance I doit intégrer au moins m/B sommets pour couvrir tout E, chaque sommet permettant de couvrir au plus B arêtes ; or, dans le cadre de graphes B-réguliers, les nombres m et n d'arêtes et de sommets sont liés par la relation : $2 \times m = 2|E| = \sum_{j=1}^{n} d(v_j) = B \times n$ où $d(v_j)$ est le degré du sommet v_j dans le graphe. Ainsi, de $\beta(I) \geqslant m/B$, on déduit $\beta(I) \geqslant n/2$ et le rapport différentiel d'approximation d'une couverture minimale U, qui est toujours de taille au plus $n \times B/(B+1)$ par la dominance de $\overline{U}$, réalise un rapport différentiel de : $\delta(I,U) = (n - |U|)/(n - \beta(I)) \geqslant 2/(B+1)$.

3.2 Problèmes de partitionnement héréditaire et GLO[δ]

Une propriété P est *héréditaire* si elle vérifie, $\forall X$ et $\forall Y \subseteq X$, $P(X) \Rightarrow P(Y)$. De plus, la propriété est dite non triviale si elle est vraie pour une infinité d'ensembles X et fausse pour une autre infinité. Le problème MAX INDEPENDENT SET ou celui de la clique de taille maximale sont des problèmes définis sur des propriétés héréditaires. Soit P une propriété héréditaire et X un ensemble. Une P-partition de X est un ensemble $\mathcal{S} = \{V_1, \dots, V_q\}$ de sous-ensembles de X qui vérifie les trois propriétés suivantes :

- $\cup_{i=1}^{q} V_i = X$;
- $\forall i, j = 1, \dots, q,\ i \neq j \Rightarrow V_i \cap V_j = \emptyset$;
- $\forall i = 1, \dots, q,\ P(V_i)$.

Soit Π un problème de NPO dont les instances sont la donnée d'un ensemble X et éventuellement d'une valuation $p : X \rightarrow \mathbb{N}$ des éléments de X ; Π est un problème de partitionnement héréditaire (MIN HEREDITARY PARTITIONING) s'il existe une propriété héréditaire P telle que toute instance I de Π revient à résoudre un problème du type :

$$\beta_{\Pi}(I) = \min \left\{ \sum_{i=1}^{q} \alpha\left(V_i\right) : \mathcal{S} = \{V_1, \dots, V_q\} \text{ est une } P\text{-partition de } V \right\}$$

où α est une fonction d'évaluation des ensembles de type fonction indicatrice notée $\mathbb{1}$, poids maximal, poids moyen, etc. La pire solution pour ces problèmes est une P-partition de V qui maximise la somme des $\alpha(V_i)$.

Le *bin-packing*, la coloration, les problèmes de partition en cliques, en sous-graphes à degré borné, en sous-graphes planaires, etc., sont tous des problèmes de partitionnement héréditaire. Il s'agit, pour tous ces problèmes, de partager un ensemble (d'éléments, de sommets) en un nombre minimal de sous-ensembles vérifiant une certaine propriété. Ainsi, le problème de coloration partage les sommets en un minimum de stables quand celui de la partition en cliques les partage en un minimum de cliques. Pour le problème de partition en sous-graphes à degré borné, le sous-graphe induit par chaque sous-ensemble de la partition doit être de degré borné par une constante ; pour la partition en sous-graphes planaires, les sous-graphes induits doivent être planaires. Pour tous ces problèmes, la pire solution est l'ensemble V des sommets du graphe d'entrée.

Enfin, le *bin-packing* relève du rangement d'éléments dans un nombre minimal de boîtes de sorte que la somme des volumes des objets dans une boîte n'excède pas le volume de la boîte. Ici, la pire solution consiste à placer un élément par boîte.

La stabilité, la clique, la majoration (sur les degrés dans les graphes ou sur le poids d'un ensemble d'éléments), la planarité sont toutes des propriétés héréditaires.

Chacun de ces problèmes a fait l'objet de nombreuses études dans son propre cadre et ont donné lieu à différents algorithmes de résolution approchée ; ce sont tous des problèmes-clefs de la recherche opérationnelle. Le problème de coloration, par exemple, a de nombreuses applications dans la conception d'emplois du temps (de tournois, d'examens, etc. [26, 27]) ; celui de la partition en sous-graphes planaires (appelé aussi *vertex thickness*) intervient notamment dans la conception de circuits imprimés (cf. [23] pour un tour d'horizon des études liées à ce problème).

Proposition 2. *Si Π est un problème de partitionnement héréditaire d'évaluation $\alpha = \mathbb{1}$, alors $\Pi \in GLO[\delta]$.*

Proof. Nous allons montrer que tout optimum local pour le voisinage 2-borné réalise un rapport différentiel de $1/2$ pour les problèmes de partitionnement héréditaire et que ce rapport est atteint asymptotiquement pour une propriété héréditaire quelconque non triviale.

Soit Π un problème de partitionnement héréditaire et $X = \{x_1, \ldots, x_n\}$ un ensemble à n éléments. Considérant une partition comme une répartition des éléments de X sur au plus n ensembles, on peut représenter toute solution $V_1, V_2, \ldots, V_q$ par un vecteur $s \in \{0,1\}^{n^2}$ qu'il faut interpréter selon : « $s^i_j = 1$ » si et seulement si « l'élément x_i est dans le sous-ensemble V_j » et qui vérifie : $\forall i = 1, \ldots, n$, $\sum_{j=1}^{n} s^i_j = 1$ si s est une partition et, $\forall j = q+1, \ldots, n$, $\sum_{i=1}^{n} s^i_j = 0$ si seuls les q premiers indices sont utilisés.

La valeur d'une solution pour $\alpha =1\!\!1$ est le nombre de sous-ensembles formant la partition et comme ce nombre ne peut excéder le cardinal n de l'instance, il s'ensuit que Π est évidemment polynomialement borné : $\forall X \in I_\Pi$, $\omega_\Pi(X) = |X| = n$.

Considérons le voisinage $\mathcal{V}$ qui consiste à passer d'une solution à une solution voisine en changeant l'affectation de deux composantes au plus du vecteur $s \in \{0,1\}^{n^2}$ représentant la solution. En d'autres termes, deux solutions sont voisines si, en partant de l'une d'entre elles, on peut arriver à l'autre en faisant passer un élément d'un ensemble à un autre. Notons que ce type de voisinage ne permet l'amélioration de la valeur de la solution que dans un cadre restreint. Plus précisément, soit $\{V_1, \ldots, V_q\}$ un optimum local relativement à ce voisinage. L'optimalité de la partition $\{V_1, \ldots, V_q\}$ signifie qu'aucune affectation réalisable d'un élément à un autre sous-ensemble n'améliorerait la solution actuelle : en d'autres termes, il n'est pas dans $\{V_1, \ldots, V_q\}$ de singleton que l'on puisse éliminer. Si la solution comporte k singletons, on suppose qu'il s'agit des k premiers sous-ensembles $V_1 = \{v_1\}, \ldots, V_k = \{v_k\}$ et l'on remarque que P étant héréditaire, les sommets $v_1, \ldots, v_k$ seront toujours dans k ensembles distincts ; en d'autres termes : $\forall (i \neq j) \in \{1, \ldots, k\}$, si $\neg P(\{v_i, v_j\})$, alors, $\forall s \in \mathrm{Sol}_\Pi(X)$, $m_\Pi(X, s) \geqslant k$ et, par conséquent, $\beta_\Pi(X) \geqslant k$.

Par ailleurs, les autres sous-ensembles $V_{k+1}, \ldots, V_q$, contenant chacun au moins deux éléments, vérifient $\sum_{i=k+1}^{q} |V_i| = n - k \geqslant 2(q-k)$ si et seulement si $q \leqslant (n+k)/2$. Donc,

$$\delta_\Pi\left(X, \{V_1, \ldots, V_q\}\right) = \frac{\omega_\Pi(X) - q}{\omega_\Pi(X) - \beta_\Pi(X)} \geqslant n - \frac{n+k}{2} n - k = \frac{1}{2}$$

Démontrons que ce rapport est atteint. Soit une propriété héréditaire P non triviale et X_n une famille d'ensembles strictement croissante pour l'inclusion vérifiant P. Considérons la suite d'instances $I_n = (X_n, P)$ du problème de partitionnement héréditaire Π. On a $\beta_\Pi(X_n) = 1$, $\omega_\Pi(X_n) = |X_n|$ et un optimum local pour les voisinages 2-bornés partitionne X_n en sous-ensembles de taille deux plus, peut-être, un sous-ensemble de taille trois et a pour valeur $\lfloor |X_n|/2 \rfloor$, c.q.f.d.

Ce résultat, par l'apparente naïveté de la solution approchée, n'est cependant pas si anodin puisqu'il égale le rapport obtenu pour le problème de coloration minimale dans [10] (monté à $289/360$ depuis [14]), mais il laisse surtout augurer de meilleures approximations à l'aide d'une analyse un peu plus fine avec des voisinages un peu plus grands.

3.3 Couverture d'ensembles

Une instance $I(C, S)$ du problème de couverture minimale d'ensembles (Min Set Cover) est la donnée d'un ensemble $C = \{c_1, c_2, \ldots, c_m\}$ d'éléments à couvrir et d'une famille $S = \{S_1, S_2, \ldots, S_n\} \subseteq 2^C$ de sous-ensembles d'éléments de C d'union C. Le but est de trouver un ensemble $\tilde{S} \subseteq S$ de cardinal minimal qui recouvre C. La pire solution est représentée par la famille S (de valeur n). On se restreint ici aux instances B-bornées, c'est-à-dire aux instances $I(C, S)$ dont tous les sous-ensembles S_i de la famille S vérifient $|S_i| \leqslant B$. Nous notons cette variante de Min Set Cover par Min Set Cover-B.

Proposition 3. MIN SET COVER-$B \in \text{GLO}[\delta]$.

Proof. Nous allons montrer que tout optimum local pour le voisinage 1-borné réalise un rapport différentiel de $1/(B+1)$.

La valeur d'une solution $\tilde{S}$ est donnée par $m_{\text{SC}}(I, \tilde{S}) = |\tilde{S}|$. Il s'agit d'un problème polynomialement borné puisque la pire solution est de valeur n. Le voisinage 1-borné désigne ici, comme voisine d'une solution $\tilde{S}$, toute sélection $\tilde{S}'$ dont au plus un sous-ensemble diffère de $\tilde{S}$; les optima locaux pour ce voisinage sont simplement des solutions minimales.

Considérons une couverture $\tilde{S} = \{S_1, S_2, \ldots, S_p\}$ de taille p ; $\tilde{S}$ est une couverture minimale si elle vérifie les deux conditions suivantes : (i) $\forall i = 1, \ldots, m$, $\exists j \in \{1, \ldots, p\}$ tel que $c_i \in S_j$ et (ii) $\forall j = 1, \ldots, p$, $\exists i_j \in \{1, \ldots, m\}$ tel que $c_{i_j} \in S_j$ et, $\forall j' \neq j \in \{1, \ldots, p\}$, $c_{i_j} \notin S_{j'}$. La condition (i) traduit la réalisabilité de la solution $\tilde{S}$, la condition (ii) sa minimalité. Par (ii), on sait qu'on peut construire un ensemble $\tilde{C} = \left\{c_{i_1}, \ldots, c_{i_p}\right\}$ de p éléments distincts (un élément par sous-ensemble S_j de la couverture) qui vérifient : $\forall c_{i'} \in \tilde{C}$, $\exists! j \in \{1, \ldots, p\}$ tel que $c_{i'} \in S_j$. Si l'instance initiale est telle que tout élément $c_i \in C$ apparaît dans au moins deux sous-ensembles S_j, alors c'est en particulier vrai des éléments $c'_1, \ldots, c'_{p'}$; ceux-ci ne pouvant, par construction, appartenir à un second sous-ensemble de $\tilde{S}$, ils apparaissent donc dans les sous-ensembles $S_{p+1}, \ldots, S_n$, ce qui permet d'établir l'inclusion $\tilde{C} \subseteq \cup_{j=p+1}^{n} S_j$.

Les sous-ensembles S_j de $S \backslash \tilde{S}$ étant de taille bornée par B, on en déduit sur p la relation : $p \leqslant B(n-p) \Leftrightarrow p \leqslant Bn/(B+1)$, ce qui nous amène au rapport de performance : $\delta(I, \tilde{S}) = (\omega(I) - p)/(\omega(I) - \beta(I)) \geqslant (n - (Bn/(B+1)))/n = (n/(B+1))/n = 1/(B+1)$.

Si un certain élément à couvrir c_i n'apparaît que dans un sous-ensemble S_j, ce sous-ensemble sera contenu dans toute solution. Aussi, pour se ramener au cas précédent, suffit-il d'appliquer préalablement à l'instance I le processus suivant qui permet d'isoler de tels sous-ensembles et de se ramener ainsi au cas précédent :

- poser $S_0 = \emptyset$; $C' = C$; $S' = S$; stop = faux ;
- tant que $\neg$stop faire : s'il existe $c \in C'$ vérifiant $\exists! j : (S_j \in S') \wedge (c \in S_j)$,
  - alors poser $S_0 = S_0 \cup \{S_j\}$; $C' \leftarrow C' \backslash S_j$; $S' \leftarrow S' \backslash \{S_j\}$; pour tout $S_k \in S'$ faire : $S_k \leftarrow S_k \backslash S_j$;
  - sinon poser stop = vrai ;
- retourner $[I' = (C', S'), S_0]$.

Le prétraitement ci-dessus renvoie un ensemble S_0 de sous-ensembles compris dans toute solution réalisable et une instance I' de MIN SET COVER-B pour laquelle tout élément c'_i est contenu dans au moins deux ensembles S'_j. Les instances I et I' sont étroitement liées par les relations suivantes : $\tilde{S}' \in \text{Sol}(I')$ si et seulement si $\tilde{S}' \cup S_0 \in \text{Sol}(I)$ et $m(I, \tilde{S}' \cup S_0) = m(I', \tilde{S}') + |S_0|$. Elles impliquent $\delta(I, \tilde{S}' \cup S_0) = \delta(I', \tilde{S}')$ et la preuve de la proposition est complète.

Si, pour $j = 1, \ldots, n$, la taille des sous-ensembles S_j est bornée par B, une solution optimale devra prendre, pour couvrir les m éléments de C, au moins m/B sous-ensembles. Par ailleurs, si tout élément apparaît dans au plus Δ sous-ensembles, la famille S ne peut disposer de plus de $\Delta \times m$ sous-ensembles. En conséquence, la valeur $\beta(I)$ d'une solution optimale sera d'au moins $m/B \geqslant n/(B\Delta)$, ce qui conduit pour tout optimal 1-local à :

$$\delta\left(I, \tilde{S}\right) \geqslant \frac{n - \frac{B}{B+1} n}{n - \frac{n}{B\Delta}} = \frac{1}{B+1} \times \frac{B\Delta}{B\Delta - 1} \tag{2}$$

Par ailleurs, puisque $\omega(I) \leqslant B\Delta\beta(I)$ pour toute instance I, on constate, d'après l'équivalence donnée en [1], que

$$\begin{aligned} m\left(I, \tilde{S}\right) &\leqslant \delta\left(I, \tilde{S}\right)\beta(I) + \left(1 - \delta\left(I, \tilde{S}\right)\right)\omega(I) \\ &\leqslant \left(\delta\left(I, \tilde{S}\right) + B\Delta\left(1 - \delta\left(I, \tilde{S}\right)\right)\right)\beta(I) \end{aligned}$$

d'où $1/\rho(I, \tilde{S}) \leqslant \delta(I, \tilde{S}) + (1 - \delta(I, \tilde{S}))B\Delta$. En utilisant pour $\delta(I, \tilde{S})$ l'expression [2], on obtient : $1/\rho(I, \tilde{S}) \leqslant B^2\Delta/B + 1$. La discussion ci-dessus introduit le corollaire suivant.

Corollary 1. *Considérons la variante de* Min Set Cover-B *où le nombre de sous-ensembles contenant un élément donné est borné par une constante Δ. Alors toute solution minimale garantit un rapport différentiel d'approximation minoré par $B\Delta/((B+1)(B\Delta-1))$. Cette variante est également élément de* GLO$[\rho]$ *avec un rapport classique d'approximation de toute solution minimale de $(B+1)/(B^2\Delta)$.*

Considérons à présent la variante Min Weigted(K) Set Cover-B de Min Set Cover-B pour laquelle les sous-ensembles S_j de la famille S sont pondérés par des entiers non nuls $p_j \leqslant K$ où K est une constante. La pire solution pour cette variante est la même que pour la variante non-pondérée ; la valeur de cette solution est la somme des poids des membres de la famille S. Nous montrons que les preuves faites précédemment permettent en réalité d'établir la qualité des optima 1-locaux, même dans ce cas plus général : cela signifie qu'un optimum local du problème non pondéré est une bonne solution du problème pondéré, la minimalité des solutions qui désigne les optima 1-locaux étant indépendante d'une quelconque pondération.

Proposition 4. Min Weighted*(K)* Set Cover-$B \in$ GLO$[\delta]$.

Proof. Considérons une instance I de Min Weigted(K) Set Cover-B et notons par I' la même instance vue comme instance de Min Set Cover-B, c.-à-d. où les poids sont ignorés. Evidemment, I et I' ont les mêmes ensembles de solutions et les mêmes optima locaux. La différence en valeur sur I entre une pire solution S et un optimum local $\widetilde{S}$ est donnée par la somme des poids des sous-ensembles de $S\backslash\widetilde{S}$ et les poids de ces sous-ensembles étant au moins de 1, on écrit naturellement :

$$\omega(I) - m\left(I, \widetilde{S}\right) \geqslant \omega(I') - m\left(I', \widetilde{S}\right) \tag{3}$$

De même, la différence en valeur, toujours dans le cas pondéré, entre une pire solution S et un optimum global $S^\star$ est donnée par la somme des poids des sous-ensembles de $S\backslash S^\star$ et les poids de ces sous-ensembles étant d'au plus K, on déduit simplement :

$$\omega(I) - \beta(I) \leqslant K\left(\omega\left(I'\right) - \beta\left(I'\right)\right) \tag{4}$$

Par [3] et [4], $\delta(I, \tilde{S}) \geqslant \delta(I', \tilde{S})/K$, c.q.f.d.

On peut trivialement généraliser le résultat de la proposition 4 en différentiel comme en classique à tout problème ensembliste Π dont la version non pondérée évalue ses solutions par leur cardinal. Soit Π un tel problème ; on note WΠ sa version pondérée ; si l'on introduit des poids compris dans un intervalle $[k, K]$, on a toujours $|\omega_{\mathrm{W}\Pi}(I) - m_{\mathrm{W}\Pi}(I, s)| \geqslant k|\omega_{\Pi}(I) - m_{\Pi}(I, s)|$, pour toute solution réalisable s, et $|\omega_{\mathrm{W}\Pi}(I) - \beta_{\mathrm{W}\Pi}(I)| \leqslant K|\omega_{\Pi}(I) - \beta_{\Pi}(I)|$, ce qui nous assure la préservation du rapport différentiel au facteur k/K près ; pour le rapport classique, il suffit de considérer $m_{\mathrm{K}\Pi}(I, s) \leqslant K \times m_{\Pi}(I, s)$ et $\beta_{\mathrm{W}\Pi}(I) \geqslant k \times \beta_{\Pi}(I)$ si Π est un problème de *minimisation* ($m_{\mathrm{W}\Pi}(I, s) \geqslant k \times m_{\Pi}(I, s)$ et $\beta_{\mathrm{W}\Pi}(I) \leqslant K \times \beta_{\Pi}(I)$ si Π est un problème de *maximisation*) pour s'assurer de préserver le rapport d'approximation, au même facteur k/K près.

Soit maintenant C et S définis comme ci-dessus. Un ensemble transversal (*Hitting Set*) est une sélection $\tilde{C}$ d'éléments de C qui parcourt tout S dans le sens où tout sous-ensemble $S_j \in S$ doit avoir au moins un élément dans $\tilde{C}$: $\forall j = 1, \dots, n$, $S_j \cap \tilde{C} \neq \emptyset$. Nous notons par Min Hitting Set-B la variante où le nombre d'ensembles contenant un élément donné est majoré par B. On peut voir aisément que si on intervertit les rôles des éléments et des sous-ensembles, le problème Min Hitting Set devient Min Set Cover et le corollaire suivant est immédiatement déduit.

Corollary 2. Min Hitting Set-$B \in$ GLO$[\delta]$.

3.4 Ensemble minimal d'arêtes retour

Soit $G(V, E)$ un graphe simple ; le problème d'ensemble minimal d'arêtes retour pour les cycles de tailles au plus l avec $l \geqslant 3$ (*Feedback Edge Set*), noté MIN FEEDBACK EDGE SET[l] dans ce qui suit, consiste à déterminer un sous-ensemble d'arêtes $F \subseteq E$ de taille minimale telle que tout cycle de taille au plus l emprunte au moins une arête de F. La pire solution de ce problème est l'ensemble E. Pour tout $l \geqslant 3$ constant, MIN FEEDBACK EDGE SET[l] est NP-difficile ([15]) tandis que, par exemple, lorsque $l = |V|$, ce problème est polynomial puisqu'il revient à chercher $E \backslash F$ le plus grand possible tel que le graphe partiel engendré par $E \backslash F$ soit sans cycle.

Proposition 5. *Pour tout $l \geqslant 3$,* MIN FEEDBACK EDGE SET[l] $\in$ GLO[δ].

Proof. Nous allons montrer que tout optimum local pour les voisinages 1-bornés réalise un rapport différentiel de $1/2$ et que ce rapport est atteint asymptotiquement pour $l = 3$.

De façon évidente, on a pour tout graphe $G(V, E)$ $\omega(G) = |E|$ et $\beta(G) \geqslant 0$: c'est un problème polynomialement borné. Considérons une fois de plus les solutions minimales, optima locaux pour les voisinages 1-bornés, qui consiste à accepter comme solutions voisines d'une solution F tous les ensembles d'arêtes obtenus par l'ajout ou le retrait d'une arête à l'ensemble F. Si $\Gamma = \{\gamma_1, \ldots, \gamma_r\}$ désigne l'ensemble des cycles élémentaires sur G, un ensemble minimal $F = \{e_1, e_2, \ldots, e_p\}$ d'arêtes retour vérifie : (i) $\forall i = 1, \ldots, r$, $\exists j \in \{1, \ldots, p\}$ tel que $e_j \in \gamma_i$ (F ensemble d'arêtes retour) et (ii) $\forall j = 1, \ldots, p$, $\exists i_j \in \{1, \ldots, r\}$ tel que $\gamma_{i_j} \cap F = \{e_j\}$ (F minimal).

Considérons la famille $\left\{\gamma_{i_1}, \ldots, \gamma_{i_j}, \ldots, \gamma_{i_p}\right\}$ des cycles qui, selon (ii), passent chacun par une unique arête $e_j = u_j v_j$ de F ; tout cycle γ_{i_j} contenant au moins trois arêtes, il emprunte deux arêtes distinctes $t_j u_j$ et $v_j w_j$ (éventuellement $t_j = w_j$) de $E \backslash F$ incidentes à e_j. Soit $F' = \cup_{j=1}^{p} \{t_j u_j, v_j w_j\}$ l'ensemble de ces arêtes ; comme l'ensemble $V(F) = \cup_{j=1}^{p} \{u_j, v_j\}$ des sommets engendrés par les arêtes de F est constitué d'au moins p sommets distincts, l'ensemble F', en tant qu'ensemble d'arêtes dont les extrémités sont les sommets de $V(F)$, contient également au moins p arêtes distinctes, d'où : $|F'| \geqslant p$ et $F' \subseteq E \backslash F \Rightarrow |E \backslash F| \geqslant p \Rightarrow |E| - p \geqslant p \Leftrightarrow \ p \leqslant |E|/2$ dont le résultat nous permet de déduire et de conclure : $\delta(G, F) = (\omega_{\text{FES}}(G) - p)/(\omega(G) - \beta(G)) \geqslant (|E| - (|E|/2))/|E| = 1/2$.

Considérons maintenant la suite de graphes $G_n(V_n, E_n)$ pour MIN FEEDBACK EDGE SET[3] où $V_n = \{v_1, \ldots, v_{n+3}\}$ et $E_n = \{v_1 v_2\} \cup \{v_1 v_{i+2}, v_2 v_{i+2} :\ 1 \leqslant i \leqslant n + 1\}$. Un optimum global est donné par $v_1 v_2$ et vaut 1 puisque seuls les cycles $\{v_1 v_2, v_1 v_{i+2}, v_2 v_{i+2}\}$ sont de taille 3, la pire solution est donnée par E_n et vaut $2n + 3$, tandis que le moins bon des optima locaux pour les voisinages 1-bornés est donné par l'ensemble $\{v_1 v_{i+2} :\ 1 \leqslant i \leqslant n + 1\}$ et vaut $n + 1$, c.q.f.d.

3.5 Ensemble minimal de sommets retour

Soit $G(V, E)$ un graphe quelconque ; une solution du problème d'ensemble minimal de sommets retour (*Feedback Node Set*), noté MIN FEEDBACK NODE SET, est un sous-ensemble $U \subseteq V$ de sommets tel que tout cycle de G emprunte au moins un sommet de U (par rapport à MIN FEEDBACK EDGE SET, il ne s'agit plus comme pour le problème MIN FEEDBACK EDGE SET de couvrir les cycles de G par les arêtes mais par les sommets). La pire solution ici est l'ensemble V. Comme d'habitude, nous notons par MIN FEEDBACK NODE SET-B la variante du problème général restreinte aux graphes à degré maximal majoré par B.

Proposition 6. MIN FEEDBACK NODE SET-$B \in$ GLO[δ].

Proof. La preuve est semblable à celle qui nous a permis d'établir l'appartenance de MIN FEEDBACK EDGE SET à GLO[δ] pour les voisinages 1-bornés. Considérons dans un graphe $G(V, E)$ un sous-ensemble U de sommets incident à tout cycle de G de taille minimale ; cela signifie : $\forall x \in U$,

$\exists\gamma$ cycle élémentaire tel que $\gamma \cap U = \{x\}$. Si U est constitué de p sommets $u_1, \ldots, u_p$, alors on peut, à chacun d'entre-eux, associer un cycle élémentaire γ_i qui n'intersecte U qu'en le sommet u_i. Notons a_i et b_i les sommets adjacents à u_i dans γ_i, a_i et b_i sont bien distincts (tout cycle étant de taille au moins deux) et appartiennent (par minimalité) au complémentaire $\overline{U}$ de U. Un sommet ne pouvant être adjacent à plus de B sommets, il y a, parmi l'ensemble $\{a_1, \ldots, a_p, b_1, \ldots, b_p\}$, au moins $2p/B$ sommets distincts. Par conséquent, $|\overline{U}| \geqslant 2|U|/B$ implique $|U|((2/B)+1) \leqslant n$ qui équivaut à $|U| \leqslant (B/(B+2))\omega(G) \leqslant (B/(B+2))\omega(G) + (2/(B+2))\beta(G)$. En d'autres termes, un rapport différentiel d'au moins $2/(B+2)$ vient d'être mis en évidence, ce qui conclut la preuve.

3.6 Un problème d'ordonnancement

Nous considérons les problèmes simples d'ordonnancement de tâches sur m multiprocesseurs, noté Min Multiprocessor Scheduling$[m]$ (*Minimum Multiprocessor Scheduling*). Une instance I de Min Multiprocessor Scheduling$[m]$ consiste en la donnée de n tâches $J_1, \ldots, J_n$, de durées d'exécution $p_1, \ldots, p_n$, respectivement (on suppose que la durée d'exécution d'une tâche est indépendante de la machine sur laquelle elle sera effectuée). Une solution consiste en la répartition des tâches sur m machines $M_1, \ldots, M_m$ de sorte à minimiser le temps d'exécution de la machine la plus sollicitée (il n'y a pas de contraintes de précédence). Une solution sera représentée par le vecteur $x \in \{0,1\}^{nm}$ dont les composantes x_j^i non nulles correspondent à l'affectation de la tâche j à la machine i. On se restreint, pour assurer la convergence en temps raisonnable des algorithmes de recherche d'optima locaux, à des poids polynomialement bornés par n^k où k est une constante absolue (ne dépendant ni de n ni de m). Nous notons par Min Multiprocessor Scheduling$[m](k)$ cette variante.

Proposition 7. Min Multiprocessor Scheduling$[m](k) \in$ GLO$[\delta]$.

Proof. Nous allons montrer que tout optimum local pour le voisinage 1-borné réalise un rapport différentiel de $m/(m+1)$ où m désigne le nombre de machines et que ce rapport est atteint.

Considérons une instance I de Min Multiprocessor Scheduling$[m](k)$. La pire solution revient naturellement à placer toutes les tâches sur une même machine, de valeur $\omega(I) = \sum_{j=1}^{n} p_j$, quand une meilleure solution ne pourra jamais faire moins que la durée maximale d'exécution d'une tâche, soit $\beta(I) \geqslant p_1$. Ainsi, pour des durées bornées par n^k (pour une constante universelle k), la quantité $\omega(I) - \beta(I)$ est bien bornée par un polynôme $(n-1)n^k$ en la taille n de l'instance. Considérons un optimum local x relativement aux voisinages 1-bornés qui accepte, comme solution voisine d'une affectation des tâches aux machines, toute répartition dont au plus l'affectation d'une tâche diffère et supposons que la première machine soit la plus chargée, soit qu'elle vérifie, si $p(M_i)$ désigne la charge de la machine M_i, $p(M_1) = \sum_{j=1}^{n} x_j^1 \times p_j = \max_{i=1,\ldots,m}\left\{p(M_i) = \sum_{j=i}^{n} x_j^i \times p_j\right\}$. On note alors $1, \ldots, l$ les indices des tâches placées dans M_1. Si l'affectation x est un optimum 1-local, alors elle vérifie $\forall i = 2, \ldots, m$, $\forall j = 1, \ldots, l$:

$$\begin{aligned} p_j + p\left(M_i\right) &\geqslant p\left(M_1\right) && (5)\\ p\left(M_i\right) &\geqslant \frac{1}{2} p\left(M_1\right) && (6)\\ \omega(I) = \sum_{j=1}^{n} p_j &\geqslant \frac{m+1}{2} p\left(M_1\right) && (7)\\ p\left(M_1\right) &\leqslant \frac{m}{m+1}\beta(I) + \frac{1}{m+1}\omega(I) && (8) \end{aligned}$$

où [6] est le résultat de l'addition des [5] pour $j = 1, \ldots, l$ et, du fait que l'on peut supposer $l \geqslant 2$ (sinon l'optimum local est global) et [7] est le résultat de l'addition des [6], pour $i = 2, \ldots, m$, et de $p(M_1)$. Finalement, [8] est impliquée par [7] puisque la meilleure solution est nécessairement supérieure ou égale à la durée moyenne de l'ensemble des tâches, d'où le résultat. Considérons maintenant l'instance suivante $I = (J, p)$ où $J = \{j_1, \ldots, j_{m(m-1)+2}\}$ et $p_1 = p_2 = m$ et, pour tout $i \geqslant 3$, $p_i = 1$. Une solution optimale est déterminée en mettant une tâche de taille m et une tâche de taille 1 sur les deux premières machines et $(m+1)$ tâches de taille 1 sur toutes les autres machines et a pour valeur $\beta(I) = m + 1$, un optimum 1-local est déterminé en mettant les deux tâches de taille m sur la première machine et m tâches de taille 1 sur toutes les autres machines et a pour valeur $m(I, x) = 2m$ tandis que la pire solution a pour valeur $\omega(I) = m^2 + m$ et consiste à mettre toutes les tâches sur la première machine, c.q.f.d.

De la relation $\beta(I) \geqslant (1/m) \times \sum_{j=1}^{n} p_j$, on déduit $\omega(I) \leqslant m \times \beta(I)$, ce qui permet de conclure avec la preuve de la proposition 7 : $m(I, x) \leqslant 2(1 - 1/(m+1))\beta(I)$ et d'obtenir le corollaire suivant.

Corollary 3. Min Multiprocessor Scheduling$[m](k) \in \mathrm{GLO}[\rho]$.

Le résultat du corollaire 3 est plus intéressant qu'il n'y paraît. En effet, nous venons de mettre en évidence le premier problème de GLO$[\rho]$ et GLO$[\delta]$ dont nous savons qu'il admet un schéma d'approximation polynomial pour l'approximation classique ([17]).

4 Résultats négatifs

Nous proposons maintenant des cas de non appartenance à la classe GLO$[\delta]$. Il faut cependant considérer ces résultats avec prudence, la notion de voisinage h-borné pouvant être fortement dépendante du codage choisi des solutions. Nous pensons notamment à l'intégration, dans le voisinage, des solutions complémentaires. En termes de distances, pour le codage naturel qui considère sur une instance à n variables un vecteur binaire de taille n, deux affectations complémentaires T et $\overline{T}$ sont effectivement n-distantes et ne peuvent ainsi être considérées comme voisines dans le cadre de voisinages h-bornés ; cependant, il suffit d'ajouter un bit supplémentaire seulement au vecteur T (ce qui demeure un codage raisonnable) pour rendre cela possible : il s'agit d'interpréter ce bit comme le sens de lecture des n bits suivants. Alors, le passage d'une solution à son complémentaire ne réclame plus n mais un unique changement et les solutions T et $\overline{T}$ se retrouvent voisines pour tout voisinage h-borné. Pour lever l'ambiguïté afférente au codage des solutions, il faudra comprendre les résultats négatifs proposés comme étant édictés pour une interprétation « physique » et non plus mathématique des voisinages h-bornés.

4.1 Satisfaisabilité maximale

Une instance du problème de satisfaisabilité maximale (Max Sat) est la donnée d'un ensemble X de variables bivalentes et d'une famille $C = \{c_1, \ldots, c_m\}$ de clauses disjonctives sur l'ensemble des littéraux $\{x_1, \ldots, x_n\} \cup \{\bar{x}_1, \ldots, \bar{x}_n\}$; une solution réalisable est une affectation des valeurs de vérité et la solution optimale est une affectation qui rend vrai un nombre maximal de clauses. La pire solution est une affectation des valeurs de vérité qui rend vrai un nombre minimum de clauses ; une telle solution n'est rien d'autre qu'une solution optimale du problème Max Sat qui, lui aussi, est NP-difficile ([21]).

La preuve de l'appartenance du problème Max Sat à la classe GLO$[\rho]$ a été faite dans [6] pour les voisinages 1-bornés avec un rapport $1/2$; nous montrons qu'il n'en est pas de même pour le rapport différentiel.

Theorem 1. Max Sat $\notin \mathrm{GLO}[\delta]$.

Proof. On montre que, pour tout h, MAX SAT $\notin$ GLO[δ] pour des voisinages h-bornés.

Considérons la famille d'instances $(I_k)_{k\geqslant 1}$, $I_k = (X_k, C_k)$ $\forall k$, définie par :

– $X_k = \{x_1, x_2, \ldots, x_k\}$ (*i.e.* $L_k = \{x_1, x_2, \ldots, x_k\} \cup \{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_k\}$) ;

– $C_k = \mathcal{P}'_k(L_k) \backslash \{(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_k)\}$.

où $\mathcal{P}'_k(L_k)$ désigne l'ensemble des parties de taille k de l'ensemble L_k des littéraux qui ne sont pas des tautologies. Toutes les clauses de taille k, sauf la clause composée des k littéraux négatifs $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_k)$, sont donc générées. Toute clause contenant au moins un littéral positif, l'affectation $(1, 1, \ldots, 1)$ satisfait les $|C_k|$ clauses ; d'autre part, toute autre affectation satisfait $|C_k| - 1$ clauses. Effectivement, soit t une affectation ; on note t^0 l'ensemble des indices des composantes nulles de t : $t^0 = \{i_1, \ldots, i_p\} = \{i \in \{1, \ldots, n\} : t_i = 0\}$; t avère toutes les clauses sauf la clause c^0 suivante :

$$c^0 = \big(\bar{x}_1, \ldots, \bar{x}_{i_1-1}, x_{i_1}, \bar{x}_{i_1+1}, \ldots, \bar{x}_{i_j-1}, x_{i_j}, \\ \bar{x}_{i_j+1}, \ldots, \bar{x}_{i_p-1}, x_{i_p}, \bar{x}_{i_p+1}, \ldots, \bar{x}_n\big) .$$

dans laquelle les variables d'indice dans t^0 apparaissent sous forme positive et les autres variables sous forme négative ; c^0 existe toujours dans C_k si $t \neq (1, 1, \ldots, 1)$. On a ainsi, pour k quelconque $\omega(I_k) = |C_k| - 1$ et $\beta(I_k) = |C_k|$: quels que soient la constante h et le voisinage h-borné associé, il existe, avec C_k pour $k > h$, une instance du problème MAX SAT pour laquelle l'affectation $(0, 0, \ldots, 0)$ est à la fois optimum local et pire solution (puisqu'il faudrait changer k valeurs pour augmenter la performance, soit plus de h changements) et, de fait, cette affectation constitue un optimum local de rapport différentiel nul. On en conclut la non appartenance de MAX SAT à la classe GLO[δ], c.q.f.d.

La démonstration précédente nous permet de déduire une information importante sur les fameuses sous-classes MAXk-SAT des instances de satisfaisabilité formées de clauses d'au plus k littéraux.

Corollary 4. *$\forall k$, $\forall h < k$,* MAXk-SAT $\notin$ GLO[δ] *pour des voisinages h-bornés.*

4.2 Ensemble minimal de sommets retour (le cas général)

Malgré leur grande similarité, MIN FEEDBACK NODE SET a un comportement fortement divergent quant à l'approximation par des optima locaux par rapport à MIN FEEDBACK EDGE SET puisque, $\forall h > 0$, il n'assure pas, dans le cas général, la qualité de ses optima locaux relativement aux voisinages h-bornés.

Proposition 8. MIN FEEDBACK NODE SET $\notin$ GLO[δ].

Proof. Considérons la famille des graphes bipartis complets $G_p(X_p \cup Y_p, E_p)$ définis pour tout entier $p \geqslant 1$ par : $X_p = \{x_1, \ldots, x_p\}$, $Y_p = \big\{y_1, \ldots, y_{p^2}\big\}$ et $E_p = X_p \times Y_p$. Les cycles élémentaires de G étant de la forme :

$$\big(x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, \ldots, x_{i_j}, y_{i_j}, \ldots, x_{i_s}, y_{i_s}, x_{i_1}\big)$$

une solution U sera réalisable si et seulement si elle contient, à un sommet près, tout X_p ou tout Y_p. En d'autres termes, $\forall U \subseteq X_p \cup Y_p$, U est une solution réalisable si et seulement si $\min\{|X_p \backslash U|, |Y_p \backslash U|\} \leqslant 1$; sinon, $\exists \{x_{i_1} \neq x_{i_2}\} \in X_p \backslash U$, $\exists \{y_{j_1} \neq y_{j_2}\} \in Y_p \backslash U$ tels que $(x_{i_1}, y_{j_1}, x_{i_2}, y_{j_2}, x_{i_1}) \cap U = \emptyset$.

Alors les solutions $U^*_p = X_p \backslash \{x_1\}$ et $U_p = Y_p \backslash \{y_1\}$ sont non seulement réalisables mais aussi remarquables : la première est un optimum global de valeur $p - 1$, la seconde un optimum local de valeur $p^2 - 1$ pour tout voisinage h-borné, $h < 2p - 1$, puisqu'il faudrait intégrer à la solution $p-1$ sommets de X_p puis retirer p sommets de $U_p \cap Y_p$ pour commencer à l'améliorer strictement. La pire solution consiste, quant à elle, à sélectionner tous les sommets $X_p \cup Y_p$.

La solution U_p, pour tout entier $p \geqslant 1$, constitue donc un optimum local réalisant le rapport différentiel : $\delta(G_p, U_p) = (p + p^2 - (p^2 - 1))/(p + p^2 - (p-1)) = (p+1)/(p^2+1) \to 0$ quand $p \to +\infty$.

Ainsi, quelle que soit la constante h, on peut créer la famille d'instances $(G_p)_{p>h}$ pour lesquelles une suite d'optima locaux $(U_p)_{p \geqslant 1}$, relativement à tout voisinage h-borné, réalise un rapport de performance asymptotiquement nul, c.q.f.d.

4.3 Le sac à dos

Considérons un ensemble X d'objets à valeur dans $\mathbb{N}^{|X|}$, pour chaque objet x un volume unitaire $a_x \in \mathbb{N}$, une utilité unitaire $c_x \in \mathbb{N}$, éventuellement une borne $b_x \in \mathbb{N}$ et une capacité $b \in \mathbb{N}$ (capacité du sac). Un solution réalisable du problème de *sac à dos*, KNAPSACK, est un sous-ensemble $X' \subseteq X$ tel que $\sum_{x \in X'} a_x \leqslant b$; la solution optimale X^* est un sous-ensemble de X qui maximise l'utilité $\sum_{x \in X^*} c_x$. La pire solution pour ce problème est de ne prendre aucun objet ; la valeur d'une telle solution est 0.

Il est facile à voir que les instances de KNAPSACK forment une sous-classe de problèmes de programmation linéaire en nombres entiers pour lesquels la matrice des contraintes se réduit à un seul vecteur.

Nous nous limiterons ici à la version restreinte notée KNAPSACK(k), qui regroupe les instances pour lesquelles la plus grande valeur numérique en valeur absolue est bornée par la quantité n^k où n désigne la dimension de l'espace de l'instance ; le problème de sac à dos, limité à cette famille d'instances, est polynomial puisque le problème global est pseudopolynomial ([15]). Or, nous montrons qu'un cas particulier de cette famille de problèmes polynômiaux, celui du sac à dos bivalent qui consiste à considérer les variables à valeur dans $\{0,1\}$ et que l'on note KNAPSACK(k) $\{0,1\}$, n'est même pas dans GLO[δ].

Proposition 9. KNAPSACK(k) $\{0,1\} \notin$ GLO[δ].

Proof. Soit une instance I du problème de sac à dos bivalent polynomialement borné :

$$I = \begin{cases} \max & c \cdot x \\ & a \cdot x \leqslant b \\ & x \in \{0,1\}^n \end{cases}$$

avec $c \in (\mathbb{N}^\star)^n$ et $\max\{c_i\} \leqslant n^k$, $a \in (\mathbb{N}^\star)^n$ et $\max\{a_i\} \leqslant n^k$, $b \in \mathbb{N}^\star$. Puisque $\omega(I) = 0$ pour $x = \overrightarrow{0}$ et $\beta(I) \leqslant \sum_{i=1}^n c_i \leqslant n^{k+1}$ pour $x = \overrightarrow{1}$, la valeur de l'instance I est bornée par le polynôme n^{k+1}.

Considérons maintenant la famille d'instances $(I_{h,n})_{n \geqslant 0}$ définie, $\forall n \geqslant 0$, par : $a = (n-1, 1, 1, \ldots, 1)$, $c = (h-1, 1, 1, \ldots, 1)$ et $b = n-1$. Le vecteur optimal $x_n^* = (0, 1, \ldots, 1)$, qui consiste à prendre tous les objets sauf le premier, apporte une utilité de $n-1$ quand la solution complémentaire $x_n = (1, 0, 0, \ldots, 0)$ n'apporte qu'une utilité de $h-1$; or, c'est un optimum local pour tout voisinage h'-borné, $h' \leqslant h$, puisqu'il faudrait, pour améliorer la composition du sac à dos x_n, ôter le premier objet d'utilité $h-1$ puis ajouter au moins h objets d'utilité unitaire. Les optima locaux x_n des instances $I_{h,n}$ réalisent ainsi un rapport de performance asymptotiquement nul : $\forall n$, $\delta(I_{h,n}, x_n) = (h-1)/(n-1) \to 0$ quand $n \to +\infty$. Ces instances $I_{h,n}$ montrent que le problème général non borné ne garantit pas la qualité de ses optima locaux vis-à-vis de voisinages h-bornés, et ce quelle que soit la constante h, c.q.f.d.

En fait, il faudrait borner les poids a_i comme les utilités c_i pour obtenir quelque chose : si, pour tout i, $a_i \leqslant A$ et $c_i \leqslant C$, alors un optimum ne fera jamais mieux que $b \times C$ (si tout élément

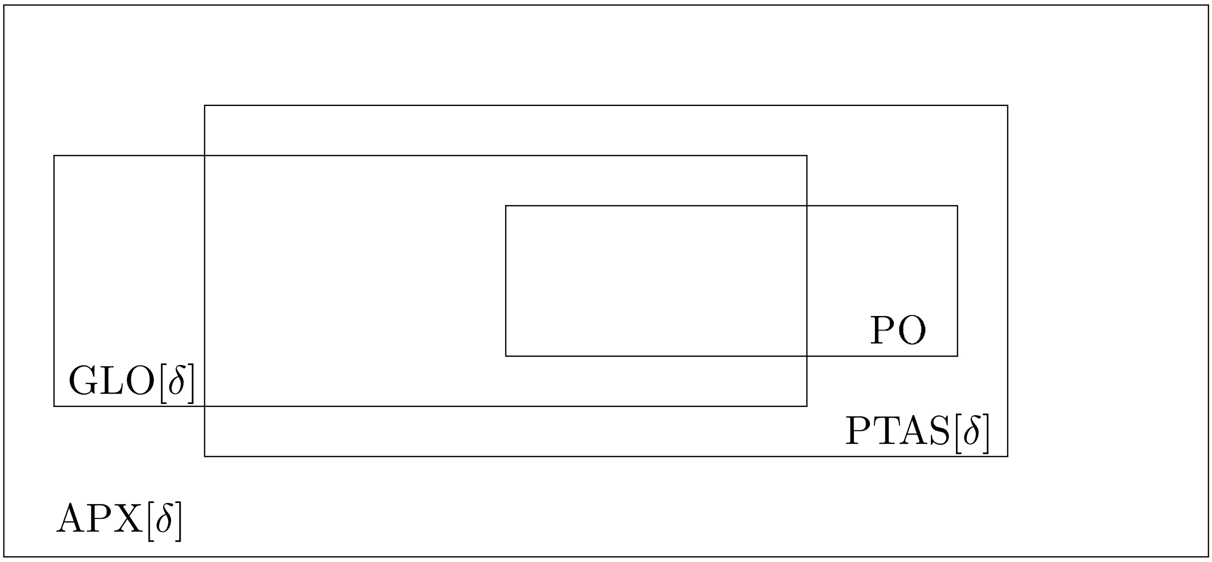

FIG. 3 – Les classes GLO[δ] et les classes d'approximation

est de poids 1 et d'utilité C) et un optimum local, une solution maximale simplement, jamais moins que $\lfloor b/A \rfloor \geqslant (b-A+1)/A$ (si tout élément est de poids A et d'utilité 1), assurant un rapport d'approximation (en considérant $b \geqslant A$) : $\delta = (b-A+1)/AbC = (b/AbC) - (A-1)/AbC \geqslant (1/AC) - (A-1)/A^2C = 1/A^2C$. Ce résultat n'est cependant pas satisfaisant ; une autre piste serait de se reporter sur un sous-problème tel MAX SUBSET SUM (*Maximum Subset Sum*) pour lequel les vecteurs a et c coïncident : il s'agit, étant donné un entier b et n entiers $c_1, \ldots, c_n$ de $\mathbb{N}^n$, de trouver le meilleur minorant de b possible par une combinaison d'éléments c_i. Il se trouve malheureusement que, de nouveau sur ce problème, on peut exhiber, pour toute constante h, une famille $J_{h,n}$ d'instances qui mette à défaut la qualité des optima h-locaux : pour tout n, on définit $c = (n^2, n^2, 1, \ldots, 1)$, vecteur de $\mathbb{N}^n$, et $b = n^2$; la solution $x^\star = (1, 0, \ldots, 0)$ est optimum global de valeur n^2 et la solution $x = (0, 0, 1, \ldots, 1)$, de valeur $n-2$, est optimum h-local pour tout $h \leqslant n-2$ puisqu'il faudrait, pour l'améliorer, retirer tous les $n-2$ éléments contenus dans x avant de pouvoir éventuellement mettre x_1 ou x_2 à 1. Or, le rapport réalisé par x, de $(n-2)/n^2$, tend indubitablement vers 0 quand n tend vers l'infini.

Corollary 5. MAX SUBSET SUM $\notin$ GLO[δ].

5 GLO[δ] et les classes d'approximation

Comme nous venons de l'illustrer avec KNAPSACK(k), un problème peut être « facile » au sens de la résolution en temps polynomial et admettre de mauvais optima locaux : P $\nsubseteq$ GLO[δ]. Notons cependant que certains problèmes « faciles » ont de bons optima locaux comme nous l'avons vu avec MIN FEEDBACK EDGE SET$|V|$: P $\cap$ GLO[δ] $\neq \emptyset$.

Tout optimum local garantit un rapport constant et, par définition, GLO[δ] renferme des problèmes sur lesquels les algorithmes de recherche locale obtiennent un optimum local en temps polynomial : ainsi, les algorithmes `LSA` sont, sur les problèmes des classes GLO[δ], des algorithmes d'approximation à rapport d'approximation différentiel constant. En revanche, puisque tout problème polynomial (qui est *a fortiori* APX[δ]) n'est pas nécessairement GLO[δ], il s'agit d'une inclusion stricte : GLO[δ] $\subset$ APX[δ].

Par ailleurs, puisque le problème de *bin packing* est dans PTAS[δ] ([12]), et dans GLO[δ]

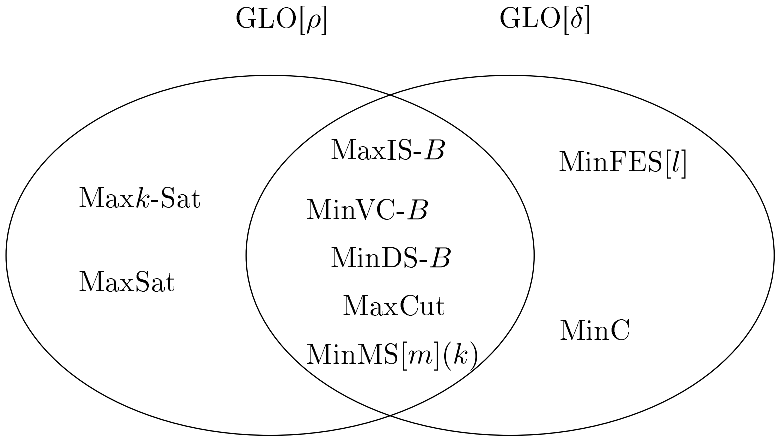

FIG. 4 – GLO[ρ] et GLO[δ]

(paragraphe 3.2), la classe GLO[δ] intersecte la classe PTAS[δ] : $\exists \Pi \in$ GLO[δ] tel que $\Pi \in$ PTAS[δ]. La figure 3 offre une illustration des relations qu'entretiennent, à notre connaissance actuelle, la classe GLO[δ] avec les classes d'approximation.

En guise de conclusion de cette section, nous proposons, dans la figure 4, la position relative des classes GLO[ρ] et GLO[δ] par quelques-uns des problèmes analysés relativement à ces classes. Pour des raisons de lisibilité de la figure, nous avons abrégé les noms des problèmes : max$k$-Sat pour MAXk-SAT, MaxSat pour MAX SAT, MaxIS-B pour MAX INDEPENDENT SET-B, MinVC-B pour MIN VERTEX COVER-B, MinDS-B pour MIN DOMINATING SET-B, MinMS[m](k) pour MIN MULTIPROCESSOR SCHEDULING[m](k), maxCut pour MAX CUT, MinFES[l] pour MIN FEEDBACK EDGE SET[l] et MinC pour coloration.

6 Remarques finales

Il nous paraît pertinent de mettre en évidence certaines conditions d'appartenance à GLO[δ] selon l'expression du problème, de sa structure. On pourrait par exemple tenter d'isoler des familles de problèmes bien approximables comme nous l'avons fait avec les problèmes de partitionnement héréditaire. Nous pensons notamment, en restant dans le même ordre d'idées, aux problèmes de partitionnement héréditaire valorisant leur solution par une autre fonction de coût (poids maximum, poids moyen…) que la fonction indicatrice ainsi qu'aux problèmes de partitionnement 1-héréditaire. Une propriété π sur un ensemble X est dite 1-héréditaire si elle n'est avérée sur un sous-ensemble S qu'à condition de vérifier, pour un élément x_0 du sous-ensemble S : $\forall S' \subseteq S$, $\pi(S' \cup \{x_0\})$. D'autres caractérisations des problèmes se situant *dans* ou *hors* GLO[δ] devraient cependant être envisagées comme nous avons essayé de le faire par l'exploitation de la structure radiale des ensembles de solutions.

Par ailleurs, plutôt que d'exhiber un trait caractéristique de tel ou tel problème, par exemple par sa formulation logique ou par la propriété de son ensemble de solutions, il est souvent intéressant de lier les problèmes entre eux de manière à dessiner des sous-classes de problèmes relativement à une réduction proprement définie, éventuellement les hiérarchiser (toujours par réduction) et, pourquoi pas, à parvenir par ce biais à tracer les limites de la classe GLO[δ] en établissant la complétude de certains problèmes.

Revisitons maintenant le problème du sac à dos. Nous avons vu au paragraphe 4.3 que ni le problème de sac à dos, ni le sous-problème MAX SUBSET SUM ne garantissaient la qualité de leurs optima locaux en mettant en avant des familles d'instances qui admettaient de *mauvais* optima locaux. Devant ces échecs, peut-être ferions-nous mieux d'affiner la notion d'optima locaux de ces problèmes. Pour KNAPSACK notamment, il pourrait sembler opportun de rechercher les optima locaux vis-à-vis d'un objectif altéré qui inciterait à sélectionner les objets de meilleur rapport « qualité-prix » : considérer comme critère de sélection des éléments le choix du meilleur rapport c_i/a_i pourrait s'avérer profitable, tant pertinent semble ce critère (qui, en réalité, définit l'optimum réel), au vu des algorithmes approchés et autres schémas dédiés à ce problème qui l'utilisent également. Il s'agirait ici de tenter l'approche introduite dans [2, 19] pour les problèmes de satisfaisabilité, celle des optima locaux altérés : peut-être qu'une solution du problème, qui est optimum local relativement à une autre fonction objective dite objectif altéré, pourrait-elle garantir de meilleurs rapports qu'un optimum local relativement à la fonction initiale ? Effectivement, si on regarde de plus près les mauvais optima locaux exhibés pour le problème de sac à dos, on remarque qu'ils peuvent sembler peu pertinents puisqu'ils auraient choisi un élément de piètre rapport « qualité-prix » $(h+1)/(n-1)$ quand bien même tous les autres éléments ont un tel rapport de 1 ; aussi pourrait-il s'avérer intéressant pour approcher le problème initial

$$\max \sum_{i=1}^{n} c_i \times x_i$$
$$\begin{cases} a \cdot x \leqslant b \\ x_i \leqslant b_i \ \forall i \\ x \in \mathbb{N}^n, \end{cases}$$

d'évaluer la qualité d'optima h-locaux du problème altéré où, dans la fonction objectif, on remplace c_i par c_i/a_i (en considérant le même ensemble de contraintes.

Une autre approche, pour KNAPSACK comme pour MAX SUBSET SUM, pourrait être de chercher un optimum local dans l'ensemble d'objets complémentaires. Considérons de nouveau MIN DOMINATING SET, sans borne sur le degré des sommets. Pour les voisinages 1-bornés, les optima locaux de MIN DOMINATING SET ne garantissent aucun rapport puisque l'on peut construire une famille $(G_n)_{n\geqslant 1}$ de graphes en étoile de façon à exhiber, sur cette famille, une suite d'optima locaux U_n tels que $\rho(G_n, U_n) \to 0$ quand $n \to +\infty$ et $\delta(G_n, U_n) \to 0$ quand $n \to +\infty$. Effectivement, si, pour tout $n \in \mathbb{N}$, le graphe G_n est défini, $\forall n \geqslant 1$, par $G_n = (\{r\} \cup V_n, E_n)$ avec $V_n = \{v_1, v_2, \ldots, v_n\}$, $E_n = \{r\} \times V_n$. et que l'on considère la solution approchée $U_n = V_n$ de valeur n, alors $\beta(G) = 1$ pour $U^\star = \{r\}$ et $\omega(G) = n+1$ pour $U_W = \{r\} \cup V_n$. Cela implique $\rho(G_n, U_n) = \delta(G_n, U_n) = 1/n \to 0$ quand $n \to +\infty$. Pourtant, il suffit simplement d'élargir le voisinage 1-borné $\mathcal{V}(G, U)$ à une solution supplémentaire, la solution complémentaire $\overline{U} = V \backslash U$, pour obtenir un rapport différentiel de $1/2$: l'ensemble de voisins $\mathcal{V}'(G, U) = \mathcal{V}(G, U) \cup \{\overline{U}\}$ reste polynomial en taille et en temps de construction et tout optimum local U au sens de $\mathcal{V}'$ nous assure toujours, en tant qu'optimum 1-local, la dominance de $\overline{U}$ mais aussi $|U| \leqslant |\overline{U}| \Rightarrow |U| \leqslant |V|/2$. Ainsi, en suivant la preuve du point 2 du lemme 1, nous obtenons le résultat souhaité.

Cet exemple illustre l'intérêt que l'on pourrait avoir à considérer conjointement une solution s et sa solution complémentaire $\overrightarrow{1} - s$, à la fois proches, l'une étant le miroir de l'autre, et éloignées puisqu'elles ne coïncident sur aucune composante. La famille des voisinages du type $\mathcal{V}(I, s) \cup \mathcal{V}(I, \overrightarrow{1} - s)$ ou $\mathcal{V}(I, s) \cup \left\{\overrightarrow{1} - s\right\}$ pour un voisinage h-borné $\mathcal{V}$ est une extension du concept de voisinage h-borné qui intègre cette notion de complémentarité : on s'intéresse aux solutions distantes de h ou $p(|I|) - h$ au lieu des seules solutions h-distantes de la solution courante. De tels voisinages ont été définis et exploités dans [3] et semblent être une source d'inspiration prolifique pour la conception d'algorithmes approchés par la recherche locale.

La nouvelle démarche que nous évoquons ici consisterait à chercher deux optima locaux : le premier de façon quelconque, puis le second dans l'ensemble complémentaire d'objets de la première. Ce n'est autre qu'une façon adaptée au problème de sac à dos d'élargir le voisinage d'une solution à l'ensemble complémentaire : si, dans un problème de satisfaisabilité, deux affectations complémentaires peuvent être comparées à tout instant ou encore pour le problème d'ensemble dominant, l'ensemble complémentaire de sommets d'une solution minimale est lui-même solution admissible, ce n'est pas le cas ici (en général) puisqu'il n'y a, *a priori*, aucune raison que l'ensemble complémentaire d'une solution constitue une solution réalisable.

Quatre idées ont été déjà évoquées : les optima h-locaux adoptés dans [6] et ici, les optima h-locaux miroirs ([3]), les optima h-locaux altérés ([2, 19]) et les optima h-locaux dans deux ensembles complémentaires où il s'agirait, étant donné un optimum local, de le comparer à un optimum local de l'ensemble complémentaire de solutions et de renvoyer naturellement le meilleur de ces deux optima. Certains problèmes suggèrent une notion encore différente d'optima locaux, pas tant par la définition de l'optimalité locale que dans la définition même du problème, dans le sens de la restriction de l'ensemble de ses solutions à des solutions que l'on pourrait qualifier de non dominées. Nous pensons à des problèmes tels celui de localisation ou encore d'arbre couvrant de profondeur 2 dont la résolution fait apparaître une dépendance entre deux types de décision. Pour le premier, on dispose d'un certain nombre de sites sur lesquels peuvent être placés des concentrateurs et des terminaux à relier aux concentrateurs ; l'installation d'un concentrateur engendre un coût fixe, puis la liaison d'un terminal à un concentrateur engendre un coût généralement proportionnel à la distance les séparant, sachant qu'à un concentrateur installé on peut, dans le cas le moins restreint, raccorder autant de terminaux que l'on veut : l'enjeu du problème est alors de décider quels concentrateurs et quelles liaisons terminal-concentrateur installer de sorte que chaque terminal soit raccordé à un concentrateur, et ce à moindre coût. Le second problème cherche, dans un graphe complet arêtes-valué et étant donné un sommet r de ce graphe, à en recouvrir les sommets par un arbre de racine r et de profondeur 2 qui soit de moindre coût. En toute rigueur, deux solutions sont h-distantes si elles diffèrent pour le problème de localisation de h choix de concentrateurs et de liaisons terminal-concentrateur, pour le problème d'arbre couvrant de profondeur 2 de h choix du père des sommets. Or, cela se fait sans exploiter la structure en deux temps de ces problèmes : une fois qu'un ensemble de concentrateurs à installer est arrêté, on sait leur raccorder les terminaux à moindre coût en temps polynomial ; une fois les sommets de premier niveau (ayant pour père la racine r) déterminés, on sait leur raccorder les sommets restants, de niveau deux dans l'arbre, à moindre coût en temps polynomial. Cela traduit le fait que toutes les décisions n'ont pas la même valeur : le choix des concentrateurs pour le premier problème, celui des sommets de premier niveau pour le second suffisent à déterminer entièrement la solution. Considérant le raccordement des terminaux aux concentrateurs ou des sommets de second niveau aux sommets de niveau 1 comme des sous-problèmes du problème global, on aurait ainsi envie de définir, comme ensemble de solutions, les combinaisons de concentrateurs et de sommets de premier niveau de sorte que deux solutions sont h-distantes si elles diffèrent du choix de h concentrateurs de h sommets de premier niveau : des solutions qui auraient pu paraître éloignées au premier abord sont en réalité voisines si l'on restreint la description de l'ensemble des solutions aux choix réellement décisifs dans la constitution des solutions. Cette approche a notamment été utilisée récemment par [4] pour des problèmes de localisation et de k-median.

Références

[1] R. Ahuja, O. Ergun, J. B. Orlin, and A. Punnen [2002]. Survey of very large scale neighborhood search techniques. *Discrete Appl. Math. 123*, 75–102.

[2] P. Alimonti [1994]. New local search approximation techniques for maximum generalized satisfiability problems. *Proc. 2nd Italian Conference on Algorithms and Complexity*, 40–53.

[3] P. Alimonti [1997]. Local search and approximability of MAX SNP problems. *PhD thesis, Università degli Studi di Roma "La Sapienza"*.

[4] V. Arya, N. Garg, R. Khandekar, K. Munagala and V. Pandit[2001]. Local search heuristic for k-median et facilty location problems. *Proc. STOC'01*, 21–29.

[5] G. Ausiello, M. Protasi[1995]. Local search, reducibility and approximability of NP-optimization problems. *Inform. Process. Lett. 54*, 73–79.

[6] G. Ausiello, M. Protasi[1995]. NP optimization problems and local optima graph theory. *Combinatorics and applications. Proc. 7th Quadriennal International Conference on the Theory and Applications of Graphs 2*, 957–975.

[7] C. Berge [1973]. *Graphs and hypergraphs.* North Holland, Amsterdam.

[8] S. Bylka, A. Idzik and Z. Tuza[1999]. Maximum cuts : improvements and local algorithmic analogues of the Edwards-Erdos inequality. *Discrete Math. 194*, 39–58.

[9] B. Chandra, M. M. Halldórsson[2001]. Greedy local improvement and weighted set packing approximation. *J. Algorithms 39*, 223–240.

[10] M. Demange, P. Grisoni and V. T. Paschos[1994]. Approximation results for the minimum graph coloring problem. *Inform. Process. Lett. 50*, 19–23.

[11] M. Demange, V. T. Paschos[1996]. On an approximation measure founded on the links between optimization and polynomial approximation theory. *Theoret. Comput. Sci. 158*, 117–141.

[12] M. Demange, J. Monnot and V. T. Paschos[1999]. Bridging gap between standard and differential polynomial approximation : the case of bin-packing. *Appl. Math. Lett. 12*, 127–133.

[13] R. Duh, M. Fürer[1997]. Approximation of k-set cover by semi-local optimization. *Proc. STOC'97*, 256–265.

[14] S. T. Fischer[1995]. A note on the complexity of local search problems. *Inform. Process. Lett. 53*, 69–75.

[15] M. R. Garey, D. S. Johnson [1979]. *Computers and intractability. A guide to the theory of NP-completeness.* W. H. Freeman, San Francisco.

[16] G. Gutin[1999]. Exponential neighbourhood local search for the traveling salesman problem. *Comput. Oper. Res. 26*, 313–320.

[17] D. S. Hochbaum, D. B. Shmoys[1987]. Using dual approximation algorithms for scheduling problems : theoretical and practical results. *J. Assoc. Comput. Mach. 34*, 144-162.

[18] V. Kann[1994]. Using dual approximation algorithms for scheduling problems : theoretical and practical results. *Nordic J. Computing 1*, 317-331.

[19] S. Khanna, R. Motwani, M. Sudan and U. Vazirani[1998]. On syntactic versus computational views of approximability. *SIAM J. Comput. 28*, 164-191.

[20] H. Klauck[1996]. On the Hardness of Global and Local Approximation. *Proc. Scandinavian Workshop on Algorithm Theory, SWAT'96, Lecture Notes in Computer Science, Springer-Verlag 1097* , 88-99.

[21] R. Kohli, R. Krishnamurti and P. Mirchandani[1994]. The minimum satisfiability problem. *SIAM J. Disc. Math. 7*, 275-283.

[22] M. W. Krentel[1989]. Structure in locally optimal solutions. *Proc. STOC'89*, 216-221.

[23] P. Mutzel, T. Odenthal and M. Scharbrodt[1998]. The thickness of graphs : a survey. *Graphs and Combinatorics 14*, 59-73.

[24] J. B. Orlin, D. Sharma [2002]. The extended neighborhood : definition and characterization. *Working Paper.*

[25] A. A. Schäffer, M. Yannakakis[1991]. Simple local search problems that are hard to solve. *SIAM J. Comput. 20*, 56-87.

[26] D. de Werra[1985]. An introduction to timetabling. *European J. Oper. Res. 19*, 151-162.

[27] D. C. Wood[1969]. A technique for coloring a graph applicable to large-scale optimization problems. *The Computer Journal. 12*, 317-319.

[28] M. Yannakakis[1997]. Computational complexity. *Aarts E and Lenstra J. eds., Local search in combinatorial optimization, John Wiley and sons.*, 19-55.