

CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
(Université Paris-Dauphine)
Unité de Recherche Associée au CNRS ESA 7024

APPROXIMATION POLYNOMIALE : NOTIONS DE DIFFICULTÉ ET LEUR IMPACT POUR ÉTUDIER LA STRUCTURE DE NP

CAHIER N° 173
juin 2000

Marc DEMANGE ¹
Vangelis PASCHOS ¹

reçu : octobre 1999.

¹ LAMSADE, Université Paris-Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France ({demange,paschos}@lamsade.dauphine.fr).

Table des matières

ABSTRACT	ii
RÉSUMÉ	ii
1 INTRODUCTION	1
2 NIVEAUX DE RESOLUTION D'UN PROBLEME	4
2.1 Algorithmes et chaînes d'approximation	4
2.2 Niveaux d'approximation	5
2.3 Exemples	5
3 OUTILS SUPPLEMENTAIRES POUR CLASSIFIER LES PROBLEMES	6
3.1 Comparaison de problèmes	6
3.2 Exemples	7
3.3 Classes d'approximabilité	9
4 DIFFICULTE DES INSTANCES D'UN PROBLEME	9
4.1 Classes d'instances: le point de vue des sous-problèmes	9
4.2 La notion d'ordre de difficulté	11
4.2.1 Exemples	11
4.2.2 Notions asymptotiques	13
4.3 Famille critique d'instances	14
5 CONCLUSION	15

POLYNOMIAL APPROXIMATION: NOTIONS OF HARDNESS AND THEIR IMPACT IN STUDYING THE STRUCTURE OF NP

Abstract

We define several notions related to the “hardness of approximately solving a problem” and describe many aspects of this theory through its foundations, its objectives, through the answers it produces and the ones it poses, or, even, through its own formalism. Among the questions tackled central are the one of “good approximability” which enables us to associate a problem with a “hardness level”, this association drawing hierarchies which describe NP’s internal structure. Another question tackled is the bringing to the fore of a notion of “hard instances” for a given problem. Such a notion can illustrate which are the structural characteristics of the instances where an algorithm attains its worst case approximation ratio and, on the other hand, provides hints enabling improvement of the approximability of a given problem. The different aspects of polynomial approximation theory treated in this paper are tackled from both epistemological and technical points of view. Our motivations are simultaneously theoretical and operational.

Keywords: Complexity, approximation, NP, reduction, instance

APPROXIMATION POLYNOMIALE : NOTIONS DE DIFFICULTÉ ET LEUR IMPACT POUR ÉTUDIER LA STRUCTURE DE NP

Résumé

Nous présentons différentes notions de difficulté intervenant en approximation polynomiale et décrivons de nombreux aspects de cette théorie à travers ses fondements, ses objectifs, les réponses qu’elle a apporté et celles qu’elle pose. Nous envisageons d’abord différentes notions de « bonne résolution » permettant d’associer chaque problème à un « niveau de difficulté ». On définit ainsi une hiérarchie qui décrit une sorte de structure interne de la classe des problèmes NP-complets. Une autre question abordée est la mise en évidence d’une notion « d’instances difficiles » qui illustre d’une part, ce qui rend un problème difficile (en approximation) et donne, d’autre part, des pistes pour une meilleure résolution (approchée). Précisons enfin que ces différentes questions peuvent être abordées tant du point de vue épistémologique et général que dans l’optique spécifique de l’approximation polynomiale. De même, nos propos sont autant motivés par les aspects théoriques de ses fondements que par leur intérêt appliqué et opérationnel.

Mots-clé : Complexité, approximation, NP, réduction, instance

1 INTRODUCTION

L'approximation polynomiale a pour objet la conception et l'analyse d'algorithmes polynomiaux avec garanties de performance pour des problèmes difficiles n'admettant pas d'algorithmes polynomiaux exacts. Nous avons présenté, dans l'article [11], quelques réflexions à caractère tant mathématique qu'épistémologique s'inscrivant dans ce cadre. Nous avons discuté la notion de *valeurs extrémales* d'un problème, ses enjeux théoriques et pratiques, notamment en ce qui concerne l'évaluation de la qualité d'un algorithme approché ainsi que les différentes problématiques qui lui sont associées.

Le travail qui suit poursuit la présentation de l'approximation polynomiale en insistant notamment sur son objet, son cadre, ses outils, ses enjeux et ses résultats. Nous nous intéressons ici à différentes notions de *difficulté de résolution* qui sont au cœur même du domaine. L'approximation polynomiale est une réponse au problème de la complexité mais nous verrons qu'inversement une étape essentielle de sa démarche consiste à enrichir la théorie classique de la complexité par de nouvelles notions de difficulté afin d'affiner la connaissance sur la nature des problèmes **NP**-complets. Ces notions sont donc non seulement un point de départ mais aussi un moteur et un sujet d'étude primordial dans ce cadre. Cette réflexion nous donnera l'occasion de décrire de nombreux aspects de l'approximation à travers ses fondements, ses objectifs, les réponses qu'elle a apporté et celles qu'elle pose ou encore sa formalisation¹.

Parmi les questions que nous abordons, citons la définition d'un *problème*, différentes notions de *bonne résolution* qui permettent d'associer chaque problème à un *niveau de difficulté* ainsi que la *structure* interne de la classe **NP** qui en résulte. Pour chaque problème, nous proposons d'étudier la *difficulté des instances* qui illustre d'une part ce qui rend le problème difficile et donne, d'autre part, des pistes pour une meilleure résolution.

Précisons enfin que, comme dans l'article [11], ces différentes questions peuvent être abordées du point de vue général ou dans l'optique spécifique de l'approximation polynomiale. De même, nos propos sont autant motivés par les aspects théoriques de ses fondements que par leur intérêt appliqué et opérationnel.

La classe **NP** désigne l'ensemble des problèmes résolubles par une machine de **TURING** polynomiale non déterministe; il s'agit, au moins dans la théorie classique de la complexité, de l'univers de travail. La classe **P** (des *problèmes polynomiaux*) est l'ensemble des problèmes de **NP** résolus par une machine de **TURING** polynomiale déterministe; ils sont considérés comme bien résolus. En effet, dans ce type d'étude, la complexité algorithmique polynomiale correspond à ce qui est communément considéré comme une *bonne résolution*. A l'opposé, la classe **NP** contient les problèmes **NP**-complets (classe **NP-C**) qui sont, dans un certain sens, les problèmes les plus difficiles de **NP**: la résolution de l'un d'entre eux par une machine de **TURING** polynomiale déterministe permettrait la résolution de n'importe quel problème de **NP** par une machine de ce type. Parmi les problèmes connus et entrant dans le cadre de nombreux modèles industriels ou en sciences humaines, relativement peu sont polynomiaux: les principaux problèmes de **P** sont les problèmes de tris, de parcours d'arbres et de graphes, certains problèmes de flots, d'affectation, des problèmes de cheminement optimaux (avec notamment, comme application, les problèmes d'ordonnancement les plus simples et, plus généralement, l'optimisation séquentielle), le problème de couplage maximum d'un graphe, la programmation linéaire en variables continues. Ces problèmes suscitent un intérêt particulier non seulement parce qu'ils ont de très nombreuses applications, mais aussi parce que en tant que *seuls problèmes bien résolus*², ils constituent les outils de base de l'algorithmique polynomiale. Les problèmes polynomiaux étant relativement

1. Comme nous l'avions déjà mentionné dans l'article [11], ce point suscite actuellement de nombreux travaux qui nécessitent et justifient en particulier de tels débats et réflexions de fond.

2. Toujours par rapport au choix de la résolution polynomiale comme référence de bonne résolution.

rares. de très nombreux modèles réels conduisent à des problèmes **NP**-complets. Parmi eux, citons bien sûr le fameux problème de satisfiabilité qui fut le premier problème **NP**-complet à être mis en évidence par COOK ([4]). Mentionnons aussi le problème du voyageur de commerce, du stable maximum d'un graphe, de l'arbre de STEINER, de la coloration minimum d'un graphe, les principaux problèmes de partage, de découpe et de couverture minimale, la plupart des problèmes d'ordonnancement, le problème du sac à dos et, plus généralement, la programmation linéaire en variables entières. La découverte, depuis les travaux de COOK et de KARP [16], d'une multitude de problèmes **NP**-complets issus d'applications réelles a mis en évidence d'une part les limites de la résolution informatique et, d'autre part, la nécessité de développer des stratégies pour leur résolution, au moins partielle. Dans cette optique, le recours à des routines polynomiales connues et l'identification de sous problèmes polynomiaux d'un modèle complexe s'avèrent particulièrement utiles.

La résolution effective de problèmes difficiles fut l'une des principales motivations de la théorie de l'approximation polynomiale. Le principe est de se limiter à l'emploi de *méthodes polynomiales* en exigeant de plus des *garanties théoriques absolues* sur la qualité des solutions obtenues. Ce choix quant aux garanties de complexité et de qualité des solutions obtenues distingue l'approximation polynomiale de l'étude, plus classique, d'heuristiques. D'un point de vue opérationnel, la pertinence de ces approches dépend des besoins spécifiques de chaque application. Dans la pratique, le recours à des méthodes heuristiques reste majoritaire, ce qui peut être attribué non seulement à leur bon comportement en pratique mais aussi à une relative méconnaissance des particularités de l'approximation polynomiale. L'autre raison est le pessimisme de certains de ses résultats qui provient essentiellement des restrictions que ce cadre impose. Soulignons néanmoins que les méthodes polynomiales peuvent avoir, dans la pratique, des comportements bien meilleurs que les bornes qu'elles garantissent.

Plusieurs travaux sur les fondements de l'approximation visent à unifier les résultats existants. Ils sont l'occasion d'une réflexion de fond sur la pertinence des notions de *bonne résolution* et des résultats qui leur sont associés. Cet article se situe justement dans cet esprit. Pour des raisons de clarté, nous rappelons succinctement certaines notions de base sur lesquelles repose la suite de ce travail.

L'approximation polynomiale restreint son champs d'étude à certains problèmes pouvant s'exprimer en termes d'optimisation. En effet, les problèmes **NP** sont définis à l'origine ([14]) comme des problèmes de décision exprimés sous forme d'une question ; la *résolution* du problème consiste alors à apporter une réponse (oui ou non). Pour de tels problèmes, il est difficile de concevoir une notion de résolution approchée. Par contre, pour certains d'entre eux, il est possible d'associer une *version optimisation* exprimée par un programme mathématique (optimisation d'une fonction réelle sous contraintes). A chaque instance, on associe une question portant sur l'existence ou non d'une solution meilleure qu'un seuil fixé. C'est par ce biais qu'on peut faire le lien entre les problèmes d'optimisation et les problèmes de décision de la classe **NP**. Plus précisément, nous rappelons ici la définition formelle des problèmes d'optimisation.

DEFINITION 1. Un problème élémentaire ou « instance » (notée I) est un programme mathématique de la forme

$$I : \begin{cases} \text{opt} & v(\vec{x}) \\ & \vec{x} \in C \\ & x_i \in \{0, 1\} \end{cases}$$

Un « problème d'optimisation combinatoire **NP** (resp., **NP**-complet) » est un ensemble d'instances tel que l'ensemble correspondant des questions « pour un M donné, existe-t-il $\vec{x} \in C$ tel que $v(\vec{x}) \theta M$? » (où θ vaut $\geq$ (resp., $\leq$) si opt vaut $\max$ (resp., $\min$)) est exactement l'ensemble des instances d'un problème **NP** (resp., **NP**-complet) au sens usuel de la théorie des langages ([14, 17]). ■

Parmi les problèmes **NP**-complets admettant une version optimisation, citons par exemple le problème de stable maximum, de coloration minimum, de voyageur de commerce ou encore de programmation linéaire en variables entières. En fait, de très nombreux problèmes **NP** et plus particulièrement **NP**-complets peuvent être associés à un problème d'optimisation qui est au moins aussi difficile que le problème de décision associé. En effet, la résolution de la version optimisation permet de résoudre la version décisionnelle. Ces problèmes, au moins aussi difficiles que des problèmes **NP**-complets, sont appelés **NP**-difficiles. Le principal intérêt d'exprimer un problème en termes d'optimisation est d'avoir à sa disposition non seulement une notion de solution acceptable³ (ou réalisable) du problème mais aussi une notion de qualité d'une telle solution réalisable par référence à la valeur de l'objectif.

Ainsi, pour un tel problème, on appelle *algorithme approché* un algorithme polynomial par rapport à la taille des instances fournissant, pour toute instance I , une solution réalisable de valeur $\lambda(I)$. Un résultat d'approximation permet de caractériser la qualité de la solution approchée à l'aide d'une *mesure d'approximation* définie pour chaque instance. La mesure la plus utilisée est le rapport

$$\gamma(I) = \min \left\{ \frac{\lambda(I)}{\beta(I)}, \frac{\beta(I)}{\lambda(I)} \right\}$$

où $\beta(I)$ désigne la valeur optimale de l'instance I . D'autres mesures d'approximation peuvent être employées, notamment le *rapport différentiel* que nous avons défini, justifié et discuté dans [7, 10, 11]. Pour une instance I , ce rapport s'exprime par

$$\delta(I) = \frac{\lambda(I) - \omega(I)}{\beta(I) - \omega(I)}$$

où $\omega(I)$ désigne la *pire valeur* de l'instance I .

Etant donnée une mesure d'approximation⁴ $\rho(I)$, un résultat d'approximation s'énonce comme une borne valable pour toute instance I . Comme pour la plupart des travaux de théorie de la complexité, on mesure donc le comportement d'un algorithme par rapport au pire cas, ce qui permet d'obtenir des garanties absolues valables pour toutes les instances. Précisons enfin que non seulement les résultats d'approximation mais aussi les méthodes déployées diffèrent en fonction du rapport. Ces différentes mesures apparaissent ainsi comme complémentaires.

Les résultats d'approximation mettent en évidence des différences de difficulté entre les problèmes **NP**-complets qui sont, par définition, de difficulté équivalente quant à leur résolution exacte mais se prêtent par contre plus ou moins bien à l'approximation. Certains problèmes d'optimisation **NP**-complets sont très bien approchés par un algorithme polynomial; le problème du sac à dos est l'exemple le plus favorable puisqu'il admet un *schéma d'approximation polynomial*, c'est-à-dire, une famille (indicée par $\epsilon \in]0, 1[$) d'algorithmes polynomiaux $\mathcal{A}_\epsilon$ telle que, pour tout ϵ , l'algorithme $\mathcal{A}_\epsilon$ garantit un rapport γ supérieur à $1 - \epsilon$. Par contre, pour d'autres problèmes, par exemple pour le problème de stable maximum d'un graphe, aucun algorithme polynomial ne peut garantir un rapport γ borné inférieurement par une constante $c > 0$. Les résultats positifs garantissent un certain niveau d'approximabilité et les résultats négatifs expriment l'impossibilité, sous une hypothèse fortement probable du type $P \neq NP$, de tel ou tel type d'approximation. L'association des deux permet d'établir une hiérarchie des problèmes d'optimisation **NP**-complets par rapport à l'approximation. De ce point de vue, l'approximation polynomiale devient un outil fondamental pour la compréhension de la classe **NP** et l'étude de sa structure.

3. Cette notion comprend, par le biais des contraintes, ce qu'on désire imposer à n'importe quelle solution envisageable.

4. Ici, nous prendrons la convention d'une mesure dans l'intervalle $[0, 1]$ avec la valeur 1 pour la résolution exacte.

L'un des aspects de cette théorie est donc l'étude de la difficulté intrinsèque des problèmes en offrant, par rapport à la théorie de la complexité, plusieurs niveaux de difficulté. La définition de la **NP**-complétude offre deux niveaux de difficulté, les classes **P** et **NP-C**. Par contre, en approximation, l'ensemble des problèmes d'optimisation **NP**-complets se décompose en couches d'équi-approximabilité. Pour ces deux théories, la notion de *difficulté* correspond à l'aptitude d'un problème à être résolu (de façon exacte ou approchée) par un algorithme polynomial.

A l'occasion de cette réflexion sur les notions de difficulté, nous abordons certains aspects d'un formalisme nouveau et nous citons des exemples d'outils pour mesurer ou comparer des notions de difficulté. La pertinence de certaines notions relève directement de cette réflexion ; ce sera en particulier le cas pour les notions asymptotiques. Enfin, nous présentons des outils ou méthodes qui exploitent des notions de difficulté. Notre objet n'est pas de développer les détails de ce formalisme mais plutôt d'en retirer la logique générale, sa pertinence et ses conséquences pratiques. De même, les résultats d'approximation cités dans ce document ont valeur d'exemples pour illustrer ces notions. C'est pourquoi ils sont donnés sans preuve et parfois seulement mentionnés dans le texte. Par souci de lisibilité et de continuité du document, nous avons sélectionné quelques problèmes combinatoires (notamment les problèmes de stable et de bin-packing) auxquels se rapportent la plupart des exemples présentés.

2 NIVEAUX DE RESOLUTION D'UN PROBLEME

Toute idée de difficulté fait référence à une notion de résolution qui, de toute façon, est déjà implicite dans la définition même du problème. On ne peut en effet définir un problème ou, plus généralement, une question que par rapport au concept de réponse qu'on adopte. D'un point de vue historique d'ailleurs, le moteur (mais aussi le frein) des travaux sur la décidabilité, la complexité ou plus généralement de toute démarche visant à étudier les limites de l'algorithmique a été la modélisation de la notion d'algorithme⁵ représentant une forme de résolution sur laquelle repose la définition des classes de complexité. De même, dans le cadre de l'approximation polynomiale, la notion de problème d'optimisation ne s'impose qu'au terme d'une réflexion sur les buts de cette théorie, notamment à propos de la signification à donner à la *résolution approchée*. Dans l'article [11], nous avons insisté sur le choix du *tout dans la formulation* qui consiste à intégrer dans le problème même les caractéristiques de sa résolution. De ce point de vue, nous avons montré à quel point non seulement les notions de fonction objective et de contraintes mais également celles de valeurs extrémales sont indispensables pour clarifier le concept de *bonne résolution* en vue d'une évaluation pertinente des algorithmes.

2.1 Algorithmes et chaînes d'approximation

Etant donné un problème d'optimisation **NP**-complet, nous avons déjà précisé dans l'introduction la notion d'*algorithme approché* qui, pour toute instance I de taille n , construit en temps polynomial par rapport à n une solution réalisable. La valeur objective $\lambda(I)$ de cette solution ainsi que les valeurs optimale ($\beta(I)$) et (éventuellement) pire ($\omega(I)$) de I sont essentielles pour l'évaluation (par le biais d'un rapport d'approximation) de la qualité de l'algorithme. Parmi les rapports utilisés, nous avons déjà mentionné $\gamma(I)$ et $\delta(I)$ mais l'étude qui suit est applicable à n'importe quelle mesure d'évaluation. Notre seule restriction est de considérer une mesure $\rho : I \mapsto \rho(I)$ à valeurs entre 0 et 1 d'autant plus proche de 1 que I est bien résolue. Ainsi, un résultat d'approximation correspond à garantir, pour toute instance du problème, un minorant de $\rho(I)$.

5. Notamment par l'outil des machines de TURING.

Pour pouvoir envisager les développements de plus en plus fins des études algorithmiques, nous introduisons la notion de *chaîne algorithmique* qui généralise celle d'algorithme approché. Le principe est d'associer un paramètre aux algorithmes en vue d'étudier le comportement non plus d'un algorithme mais d'une famille d'algorithmes.

DEFINITION 2. Soit Π un problème d'optimisation combinatoire (définition 1) ; on note $\mathcal{I}$ l'ensemble des instances de Π , soit $f : \mathcal{I} \times \mathbb{N} \rightarrow]0, 1[; (I, k) \mapsto f(I, k)$ une application croissante en k . Une chaîne (algorithmique) d'approximation polynomiale⁶ de rapport f pour Π est une suite d'algorithmes $(\mathcal{A}_k)_{k \in \mathbb{N}}$ telle que, pour tout $k \in \mathbb{N}$, $\mathcal{A}_k$ est un algorithme approché polynomial pour Π garantissant le rapport d'approximation $\rho_k(I) \geq f(I, k)$, $\forall I \in \mathcal{I}$. ■

2.2 Niveaux d'approximation

L'un des enjeux principaux de l'approximation polynomiale étant de mettre en évidence, au sein de la classe NP-complet, une structure fondée sur l'approximabilité des problèmes, nous considérons la notion de *niveau d'approximation*.

Etant donné un problème Π d'ensemble d'instances $\mathcal{I}$, on note $\mathcal{F}_\Pi$ (ou seulement $\mathcal{F}$ lorsqu'aucune confusion n'est possible) l'ensemble des rapports possibles pour les chaînes d'approximation polynomiale pour Π , i.e.,

$$\mathcal{F}_\Pi = \{f : \mathcal{I} \times \mathbb{N} \rightarrow]0, 1[; (I, k) \mapsto f(I, k)\}.$$

Nous utilisons également la notation $\mathcal{F}'_\Pi$ pour désigner les rapports indépendants de k , i.e.,

$$\mathcal{F}'_\Pi = \{f : \mathcal{I} \rightarrow]0, 1[; I \mapsto f(I)\}.$$

A chaque famille de rapports $F \subset \mathcal{F}_\Pi$ correspond un *niveau d'approximation*, toutes les chaînes d'approximation polynomiale garantissant un rapport dans F appartiennent au même niveau. Les exemples suivant interprètent, sous ce point de vue, plusieurs résultats classiques et montrent que ce cadre permet d'exprimer des résultats récents plus fins pour lesquels les notions classiques ne suffisent plus.

2.3 Exemples

Lorsque f est constante par rapport à k , i.e.,

$$\exists \tilde{f} : \mathcal{I} \rightarrow]0, 1[, \forall I \in \mathcal{I}, \forall k \in \mathbb{N}, f(I, k) = \tilde{f}(I)$$

(f peut alors être assimilée à un élément de $\mathcal{F}'_\Pi$), on peut, sans perte de généralité, restreindre la chaîne d'approximation à un unique algorithme $\mathcal{A}$ en sélectionnant l'un des $\mathcal{A}_k$. On retrouve dans ce cas la notion classique d'algorithme approché à rapport $\tilde{f}$ qui constitue un premier niveau d'approximation pour Π . On peut alors affiner ce niveau en fonction de la dépendance de $\tilde{f}$ par rapport au paramètre I : si $\tilde{f}$ est une constante universelle, on retrouve le niveau des algorithmes à rapport constant (citons par exemple l'algorithme de couplage pour la couverture de sommets qui garantit le rapport $\gamma = 1/2$) ; ce niveau peut lui-même être divisé en sous-niveaux en fonction des valeurs de $\tilde{f}$. Par contre, si $\tilde{f}$ est une fonction non-constante de l'instance (et en particulier si $\inf_{I \in \mathcal{I}} \tilde{f}(I) = 0$), on retrouve la notion de rapport d'approximation dépendant de l'instance. Citons à ce titre tous les algorithmes approchés pour le problème de stable maximum dans un graphe ou encore de couverture d'ensembles. Les résultats négatifs qui établissent la non approximabilité à rapport constant de problèmes comme, par exemple, le stable maximum ([1]), la couverture

6. Ou simplement *chaîne* lorsqu'aucune confusion ne sera possible.

d'ensembles minimum ou la coloration minimum ([18]) justifient qu'on doive prendre en compte la possibilité de rapports d'approximation dépendant de l'instance.

L'ensemble des fonctions $f \in \mathcal{F}_\Pi$ constantes en I , i.e., des fonctions pour lesquelles il existe une suite $(x_k)_{k \in \mathbb{N}}$ de $]0, 1[$ telle que $\forall (I, k) \in \mathcal{I} \times \mathbb{N}, f(I, k) = x_k$, définit le niveau des *chaînes indépendantes de l'instance*. Remarquons que le cas où (x_k) est une suite convergeant vers 1 correspond à un schéma d'approximation polynomial.

À titre d'exemple, intéressons-nous au problème S de stable maximum qui consiste à déterminer, dans un graphe, un ensemble stable⁷ de cardinal maximum. De nombreux travaux traitent de l'approximation de S et donnent lieu à des résultats de plus en plus difficiles à exprimer. Citons par exemple les résultats de [15] ; l'inadéquation du cadre classique pour exprimer ces résultats contraint les auteurs à n'en présenter qu'une forme partielle ; cet exemple justifie le besoin d'enrichir les concepts d'approximation sans quoi le manque de formalisme risquerait d'être un frein à de nouveaux résultats. En termes de chaînes d'approximation polynomiale, le principal résultat de [15] ainsi que ses améliorations présentées dans l'article [12] s'expriment de la façon suivante (dans ce qui suit, nous notons par Δ le degré maximum des sommets d'un graphe G et par μ le degré moyen des sommets de G ; $\mu \leq \Delta$).

PROPOSITION 1.

- Il existe une chaîne d'approximation polynomiale de complexité $O(n^k)$ pour le stable garantissant le rapport d'approximation $(6/\Delta)(1 - 1/k) + o(1/\Delta)$ ([15]).
- Il existe une chaîne d'approximation polynomiale de complexité $O(n^{k/2})$ pour S garantissant le rapport d'approximation $k/\Delta + o(1/\Delta)$ ([12]).
- Il existe une chaîne d'approximation polynomiale de complexité $O(n^k)$ pour S garantissant le rapport d'approximation $k/\mu + o(1/\Delta)$ ([12]).

3 OUTILS SUPPLEMENTAIRES POUR CLASSIFIER LES PROBLEMES

Les notions d'approximation que nous venons de présenter définissent une échelle de qualité ainsi qu'une notion de difficulté. Tous les problèmes **NP**-complets équivalents du point de vue de la résolution exacte ne le sont plus dans le cadre d'une résolution approchée, si bien que cette notion de difficulté par rapport à l'approximation, qui représente la possibilité d'établir tel ou tel résultat d'approximation, conduit à envisager une classification des problèmes d'optimisation **NP**-complets.

Ces outils permettent de positionner chaque problème sur une échelle d'approximabilité et sont en particulier parfaitement adaptés pour comparer différents résultats relatifs à un même problème. Par contre, il est moins aisé et en tout cas moins pertinent de comparer directement des résultats d'approximation pour des problèmes différents. En vue d'établir une classification des problèmes, nous avons besoin d'un outil spécifique permettant de comparer la difficulté d'approximation de différents problèmes. C'est l'objet des *réductions* que nous définissons ci-dessous. Ce type d'outil est d'ailleurs à la base de toutes les démarches visant à envisager une notion de difficulté d'un problème, aussi bien dans le cadre de la théorie de la décidabilité que dans celui de la complexité. En particulier, ce type d'outil est à la base des démonstrations de non-décidabilité ou de **NP**-complétude ([14]).

3.1 Comparaison de problèmes

Le principe est de définir, sur un ensemble de problèmes, un préordre représentant une certaine notion de difficulté. Un problème Π' est considéré comme plus difficile qu'un problème Π si la

7. Etant donné un graphe $G = (V, E)$, un ensemble stable est un ensemble $V' \subseteq V$ de sommets tel que, $\forall (i, j) \in V' \times V', ij \notin E$.

résolution de Π' permet, via une transformation appelée *réduction de Π à Π'* , la résolution de Π . Un tel schéma permet alors de faire correspondre, à chaque forme de résolution, une notion de difficulté. Ainsi par exemple la notion de non-décidabilité correspond à la résolution « acceptation d'un langage par une machine de TURING déterministe » ; celle de complexité consiste à prendre en compte la complexité algorithmique de cette machine.

Chaque classe d'approximation est associée à un type de réduction préservant ce niveau. Parmi les premiers travaux sur ce type d'outil, citons ceux de SIMON ([28]) sur les réductions dites *continues* qui permettent de transformer un algorithme à rapport d'approximation constant pour Π' en un algorithme du même type pour Π . Dans le même ordre d'idée, la notion de *L-réduction* de [24] est fondée sur la résolution par un schéma d'approximation polynomial. De très nombreuses notions ont été introduites et utilisées, chacune adaptée à une certaine classe d'approximation ([6, 5, 22, 23]). Pour une étude complète du sujet, on pourra se référer à [26]. Les réductions permettent de transférer des résultats positifs ou négatifs d'un problème à un autre et contribuent ainsi à enrichir la connaissance d'une certaine classe d'approximabilité mais elles donnent aussi accès à autant de notions de complétude qui définissent de nouvelles classes.

La définition suivante élargit cet outil au cas de différentes notions de mesure d'approximation et d'une plus large gamme de niveaux d'approximation. Elle permet également d'unifier la plupart des notions de réduction connues.

DEFINITION 3. Soit Π et Π' deux problèmes d'optimisation et g une application de $\mathcal{F}_{\Pi'} \rightarrow \mathcal{F}_{\Pi}$. Une réduction en approximation $\alpha_{\Pi, \Pi'}$ de Π à Π' d'expansion g est un algorithme polynomial qui permet de transformer toute chaîne d'approximation Π' , de rapport $f \in \mathcal{F}_{\Pi'}$, en une chaîne d'approximation pour Π de rapport $g(f)$. ■

L'expansion représente l'effet de la transformation sur la qualité d'approximation. Une réduction permet de comparer les niveaux d'approximation des problèmes Π et Π' . Pour deux niveaux $F \subset \mathcal{F}_{\Pi}$ et $F' \subset \mathcal{F}_{\Pi'}$:

- si $g(F') \subset F$, alors on dit que la réduction $\alpha_{\Pi, \Pi'}$ transforme le niveau F' en F ;
- si $g(F') \subseteq F'$ et si les deux problèmes ont le même ensemble d'instances ou lorsque les définitions de F et F' sont indépendantes des instances, on dira que $\alpha_{\Pi, \Pi'}$ préserve le niveau F' .

Dans certains cas, on ne parvient à transformer polynomialement que certaines chaînes d'approximation de Π' en une chaîne d'approximation de Π , on considère alors une notion de *réduction partielle* dont l'usage est restreint à deux niveaux F et F' .

Faisons alors les remarques suivantes concernant la définition 3 :

- la *L-réduction* ([24]) correspond à la notion de réduction partielle préservant les schémas d'approximation polynomiale, i.e., les chaînes convergeant vers 1 ;
- la *réduction faiblement continue* étudiée par SIMON dans [28] devient un cas particulier de réduction partielle préservant les algorithmes (considérés comme cas particuliers de chaînes) à rapport constant ; remarquons qu'alors une telle réduction d'expansion ζ (ζ est une constante) de Π à Π' permet de transformer polynomialement toute chaîne d'approximation à rapport $(x_k)_{k \in \mathbb{N}}$ indépendant de l'instance en une chaîne d'approximation à rapport $(\zeta x_k)_{k \in \mathbb{N}}$; il suffit d'appliquer la réduction à chaque algorithme de la chaîne ; il s'agit donc de réductions partielles préservant les chaînes à rapport indépendant de l'instance et d'expansion l'homothétie de rapport ζ .

3.2 Exemples

Les exemples de réductions ci-dessous permettent de transférer les résultats d'approximation de la proposition 1 à de nouveaux problèmes.

Considérons d'abord la version pondérée WS du problème de stable pour laquelle les sommets du graphe instance sont affectés de poids. Le problème consiste alors à déterminer un ensemble stable de poids maximum.

Dans le cas de poids positifs entiers, une construction proposée par SIMON [28] permet de formuler le problème de stable de poids maximum dans un graphe G comme la recherche d'un stable de cardinal maximum dans un graphe construit à partir de (G, w) noté G_w . L'idée est la suivante : chaque sommet v_i de poids w_i dans G donne lieu à w_i sommets $v_{i,l}$, $l = 1, \dots, w_i$ dans G_w qui forment un stable. Une arête $v_i v_j$ de G donne lieu, dans G_w , à toutes les arêtes $v_{i,l} v_{j,k}$ avec $l = 1, \dots, w_i$, $k = 1, \dots, w_j$, c'est-à-dire à un graphe biparti complet. Tous les sommets $v_{i,l}$, $l = 1, \dots, w_i$ issus d'un même v_i ont donc le même voisinage dans G_w de sorte qu'un stable maximal de ce graphe contient toutes les copies de v_i ou aucune. Par conséquent, tout stable S de G de poids $w(S)$ correspond à un stable de cardinal $w(S)$ dans G_w composé de toutes les copies des sommets de S . Réciproquement, tout stable maximal S_w de G_w se projette en un stable de poids $|S_w|$ dans G . Ainsi, toute solution réalisable (resp., optimale) pour le problème WS dans G correspond à une solution réalisable (resp., optimale) pour S dans G_w et réciproquement. Etant donné la définition des problèmes WS et S, la valeur objective est conservée par cette transformation.

La seule restriction à laquelle il faut rester attentif est que la transformation de (G, w) en G_w n'est polynomiale que si les poids de l'instance (G, w) de WS sont bornés par un polynôme de la taille de G . En effet, l'ordre de G_w est exactement la somme des poids des sommets de G . Enfin, notons que le degré maximum des sommets de G_w est majoré par $w_{\max} \Delta$. La discussion ci-dessus donne lieu au résultat suivant.

PROPOSITION 2. *Il existe une réduction partielle $\propto_{\text{PWS}, S}$ de PWS (le problème du stable pondéré où tous les poids sont bornés supérieurement par un polynôme en n) à S transformant toute chaîne de rapport $f(\Delta, k)$ (f étant croissant en k et décroissant en Δ) pour S en une chaîne de rapport $f(w_{\max} \Delta, k)$ pour PWS.*

PROPOSITION 3. *Il existe une réduction partielle $\propto_{\text{BWS}, S}$ de BWS (le problème du stable pondéré où tous les poids sont bornés supérieurement par une constante) à S transformant toute chaîne de rapport $f(\Delta, k)$ (f étant croissant en k et décroissant en Δ) pour S en une chaîne de rapport $f(\Delta, k)$ pour BWS.*

Cette réduction préserve en particulier les niveaux d'approximations associés aux rapports de la forme k/δ où δ peut désigner le degré maximum de G , son degré minimum, son degré moyen ou encore des versions pondérées de ces notions. Ceci permet de déduire de la proposition 1 le résultat d'approximation pour BWS suivant.

PROPOSITION 4. *Il existe des chaînes d'approximation polynomiales pour BWS de rapports respectifs $k/\Delta + o(1/\Delta)$ et $k/\mu + o(1/\Delta)$.*

Certaines réductions peuvent être établies entre différents niveaux d'approximation pour un même problème. Mentionnons dans ce cadre le résultat classique ([14]) selon lequel tout algorithme à rapport constant pour le stable peut être transformé (via une composition de graphe) en un schéma d'approximation. Cette réduction du problème de stable à lui-même est le point de départ des résultats négatifs pour ce problème. De même, dans [8], nous établissons la réduction entre les approximations classique et différentielle pour le bin-packing, ce qui nous permet d'établir un schéma d'approximation différentiel pour ce problème, niveau d'approximation qu'on ne peut garantir pour le rapport classique.

3.3 Classes d'approximabilité

La complémentarité entre des résultats positifs et négatifs permet de hiérarchiser la classe **NP**. Ils représentent respectivement l'appartenance d'un problème à une classe et la différentiation entre deux classes. La notion de niveau d'approximabilité donne lieu à une classification absolue (i.e., sans référence à un problème particulier) : le concept de réduction enrichit cette classification en créant des liens entre des problèmes différents. Il s'agit d'un outil fondamental pour élaborer des résultats positifs ou négatifs et pour concevoir des algorithmes approchés à partir d'algorithmes établis pour d'autres problèmes.

Les réductions permettent aussi d'identifier le noyau des problèmes les plus difficiles de chaque classe grâce à la notion de complétude. La complétude par rapport aux réductions polynomiales préservant les solutions exactes permet de définir la classe **NP-C**. Ce type de construction est également possible avec des réductions préservant l'approximation. Etant donné un problème Π et deux niveaux d'approximation $\mathcal{F} \subset \mathcal{F}'$ pour Π , Π est *complet par rapport aux niveaux $\mathcal{F}$ et $\mathcal{F}'$* si $\Pi \in \mathcal{F}'$ et si, pour tout $\Pi' \in \mathcal{F}'$, il existe une réduction $\alpha_{\Pi', \Pi}$ préservant le niveau $\mathcal{F}$. Plusieurs résultats de complétude en approximation ont déjà été obtenus sous des formalismes similaires ([2]).

La notion de réduction que nous venons de discuter permet d'étudier la structure de la classe **NP**, de mettre en évidence des liens entre les problèmes et enfin, pour un problème donné, d'exprimer les résultats d'approximation (connus ou possibles) le concernant. Les outils présentés sont des raffinements des notions existantes qui permettent d'enrichir la gamme des résultats possibles et la classification des problèmes de **NP**. Une autre problématique naturelle (mais moins étudiée) consiste, à problème fixé, étudier la structure et la difficulté de ses instances. Ce point de vue que nous nous proposons d'aborder maintenant s'avère particulièrement pertinent, aussi bien dans le cadre de la formalisation de l'approximation polynomiale que pour l'étude de l'approximation de problèmes spécifiques.

4 DIFFICULTE DES INSTANCES D'UN PROBLEME

La question de la difficulté intrinsèque des instances d'un problème donné s'inscrit naturellement dans notre démarche en vue de comprendre ou, au moins, illustrer la difficulté du problème. Comme le reste de notre étude, cette question peut être envisagée sous différents angles :

- des points de vue épistémologique et mathématique d'abord, tenter de caractériser la difficulté des instances d'un problème est une voie vers une meilleure perception de schémas qui rendent un problème accessible ou non à une résolution algorithmique ;
- d'un point de vue opérationnel maintenant, mettre en évidence, parmi les instances d'un problème, les structures donnant lieu à une difficulté de résolution ou qui mettent en défaut un algorithme précis est une étape essentielle pour l'analyser et rechercher de meilleures heuristiques ;
- enfin, pour la formalisation de l'approximation polynomiale, la question d'une mesure de la difficulté des instances d'un problème se pose pour définir certaines notions d'approximation asymptotique.

4.1 Classes d'instances : le point de vue des sous-problèmes

La première idée pour appréhender la difficulté d'instances d'un problème est de se ramener à l'étude de la difficulté d'un problème en exploitant les notions que nous venons de présenter. Il s'agit d'interpréter une partie des instances d'un problème Π comme l'ensemble des instances d'un problème Π' appelé sous-problème. Parfois le sous-problème se prête à une approximation alors

que ce n'est pas le cas du problème général. Dans ce cas, les instances sélectionnées peuvent être considérées comme plus faciles (du point de vue de l'approximation) qu'une instance générale. Pour le problème du stable maximum par exemple, la famille des graphes de degré borné est intéressante car elle admet, contrairement au cas général, un algorithme polynomial garantissant un rapport d'approximation γ constant. La proposition 4 implique le même type de résultat pour le cas pondéré par des poids bornés. Ce point de vue est très courant dès qu'on cherche à exploiter certaines caractéristiques des instances effectives intervenant dans une application dans le but de garantir leur bonne résolution.

Au contraire, une réduction du problème général au problème restreint désigne un noyau d'instances difficiles. Bien entendu, la réduction inverse, du sous-problème au problème général, est immédiate, si bien que les deux problèmes peuvent alors être considérés comme équivalents en approximation. Un tel résultat montre un sous-problème sur lequel focaliser son attention. On peut essentiellement exploiter sa structure spécifique pour établir des résultats positifs qui s'étendront, par réduction, au problème général.

A titre d'exemple, considérons le problème de stable de poids (positifs) maximum. Parmi les stratégies développées pour le résoudre efficacement, le recours à la programmation linéaire et, en particulier, l'analyse des liens entre le problème exprimé comme programme linéaire en variables bivalentes (noté WS) et son relaxé (noté WSr) a souvent été exploité. Bien entendu, le problème WSr est polynomial comme programme linéaire en variables continues. Ces deux programmes linéaires sont définis de la façon suivante pour une instance composée d'un graphe $G = (V, E)$ et d'un système de poids $\vec{w} \in \mathbb{Q}^{|V|}$:

$$\text{WS} = \begin{cases} \max & \vec{w} \cdot \vec{x} \\ & A \cdot \vec{x} \leq \vec{1}_{|E|} \\ & \vec{x} \in \{0, 1\}^{|V|} \end{cases}$$

où $\vec{1}_{|E|}$ est le vecteur colonne de $\mathbb{Q}^{|E|}$ dont toutes les coordonnées valent 1.

$$\text{WSr} = \begin{cases} \max & \vec{w} \cdot \vec{x} \\ & A \cdot \vec{x} \leq \vec{1}_{|E|} \\ & \vec{x} \in \mathbb{Q}^{|V|} \end{cases}$$

Parmi les résultats intéressants dans ce cadre, mentionnons celui de la solution semi-intégrale.

PROPOSITION 5. *Il existe un algorithme polynomial permettant, pour toute instance de WS, de partitionner V en trois ensembles V_1 , $V_{1/2}$ et V_0 de telle sorte que*

- 1° *il existe une solution optimale $\hat{S}$ de WS telle que $V_1 \subset \hat{S}$, $V_0 \cap \hat{S} = \emptyset$ ([21]);*
- 2° *l'unique solution optimale de WSr sur le graphe induit par $V_{1/2}$ consiste à assigner la valeur 1/2 à tout sommet ([25]);*
- 3° *V_1 et $V_{1/2}$ n'ont pas de sommets adjacents ([21]).*

Ce résultat montre l'intérêt de la classe $\mathcal{C}_{1/2}$ des instances pour lesquelles la solution assignant 1/2 à tout sommet est l'unique solution optimale du programme WSr qui correspond, en quelque sorte, aux graphes pour lesquels la solution du programme relaxé ne fournit aucune information aidant à la détermination d'une solution du problème de stable de poids maximum. Plus précisément, $\mathcal{C}_{1/2}$ peut être interprétée comme un noyau d'instances difficiles.

PROPOSITION 6. *Le problème général du stable pondéré se réduit au problème restreint à la classe $\mathcal{C}_{1/2}$ avec une expansion préservant tout niveau d'approximation (relativement au rapport d'approximation γ).*

La réduction de la proposition 6 consiste à construire les ensembles V_1 et $V_{1/2}$ et à compléter toute solution approchée sur le graphe induit par $V_{1/2}$ par l'ensemble V_1 . V_1 et $V_{1/2}$ n'ayant aucun

sommet adjacent, la solution est bien réalisable. Par ailleurs, en notant respectivement $\lambda_{1/2}(\lambda)$ et $\beta_{1/2}(\beta)$ les valeurs approchées et optimales correspondant au graphe induit par $V_{1/2}$, on a

$$\begin{aligned}\lambda &= w(V_1) + \lambda_{1/2} \\ \beta &= w(V_1) + \beta_{1/2}\end{aligned}$$

de sorte que le rapport correspondant vérifie

$$\frac{\lambda}{\beta} \geq \frac{\lambda_{1/2}}{\beta_{1/2}}.$$

4.2 La notion d'ordre de difficulté

Dans certains cas, en particulier pour définir des notions asymptotiques, on a besoin d'une analyse plus précise permettant de quantifier la difficulté des instances. C'est dans cette optique que nous sommes amenés à la notion d'ordre de difficulté.

DEFINITION 4. Soit un problème Π d'ensemble d'instances $\mathcal{I}$ et un niveau d'approximation $\mathcal{F}$ pour Π tel que Π n'appartient pas à ce niveau (i.e., il n'existe pas, sous une hypothèse du type $\mathbf{P} \neq \mathbf{NP}$, de chaîne polynomiale pour Π garantissant un rapport dans $\mathcal{F}$). Alors une application $d: \mathcal{I} \rightarrow \mathbb{N}$ est un ordre de difficulté pour Π par rapport au niveau $\mathcal{F}$ si, $\forall M \in \mathbb{N}$, la restriction de Π aux instances $I \in \mathcal{I}, d(I) \leq M$ appartient au niveau $\mathcal{F}$. ■

Sous les hypothèses de la définition 4, pour toute constante k , le problème général se réduit au sous-problème $\Pi_k = \{I, d(I) \geq k\}$ par une réduction préservant le niveau $\mathcal{F}$. La seule restriction est qu'éventuellement les instances de Π_k ne sont pas reconnaissables en temps polynomial (Π_k n'est alors pas dans le cadre que nous avons présenté).

4.2.1 Exemples

Dans [10], nous avons discuté et adapté une notion de *structure des instances d'un problème* introduite dans [3] en vue de définir une notion de rapport différentiel asymptotique. L'un des éléments de la structure d'une instance I est son *support*, noté $\sigma(I)$, défini comme le nombre de valeurs réalisables de I . Cette notion est très naturelle dans l'optique de capter la difficulté d'une instance et, en particulier, si on cherche à étudier la capacité qu'a une instance à être bien résolue par un algorithme glouton ou encore par programmation dynamique. Si on s'intéresse à la classe des algorithmes gloutons dont le principe est de construire une solution en itérant une opération élémentaire à partir d'une solution arbitraire de départ (par exemple une pire solution), les instances de support élevé s'interprètent comme les cas les plus défavorables. En effet, la construction d'une bonne solution à partir de la solution de base nécessite alors (dans le pire des cas) de nombreuses étapes gloutonnes. Cet argument peut être précisé grâce au formalisme que nous venons d'introduire; pour cela, il suffit de se restreindre à une classe de problèmes adaptés à la résolution par un algorithme glouton. Cette restriction est relativement naturelle dans la mesure où ces algorithmes sont les plus répandus et où la classe (problèmes radiaux) que nous introduisons est très vaste. Il s'agit d'une généralisation de la notion de problème héréditaire⁸.

Considérons un problème d'optimisation Π dont chaque instance s'exprime par :

$$\begin{cases} \text{opt } v(\vec{x}) \\ \vec{x} \in C \\ x_i \in \{0, 1\}, \quad i = 1, \dots, t \end{cases}$$

8. Problèmes pour lesquels toute partie d'une solution réalisable est réalisable.

où t est une dimension (polynomiale en n), $v(\cdot)$ est une fonction affine sur $\mathbb{R}^t$ et C est un polytope de $\mathbb{R}^t$.

DEFINITION 5. Un problème Π est *radial* s'il existe trois algorithmes polynomiaux (en n) ξ , ψ et φ tels que pour toute instance I de Π de taille n :

- 1° ξ construit une solution réalisable $\bar{x}^{(0)}$.
- 2° pour toute solution réalisable $\bar{x}$ de I strictement meilleure (au sens de l'objectif) que $\bar{x}^{(0)}$, l'algorithme φ construit une solution réalisable $\varphi(\bar{x})$ qui vérifie $v(\varphi(\bar{x})) < (>)v(\bar{x})$ si Π est respectivement un problème de maximisation (minimisation),
- 3° pour tout vecteur réalisable $\bar{x}$ de I de valeur objective strictement meilleure que $v(\bar{x}^{(0)})$, il existe un entier k tel que $\varphi^k(\bar{x}) = \bar{x}^{(0)}$ (φ^k désignant l'itéré k -fois de φ),
- 4° pour $\bar{x}^{(0)}$ ainsi que pour toute solution réalisable $\bar{x}$ de I de valeur objective strictement meilleure que $v(\bar{x}^{(0)})$, $\psi(\bar{x}) = \varphi^{-1}(\{\bar{x}\}) = \{\bar{y}, \varphi(\bar{y}) = \bar{x}\}$. ■

Géométriquement, cette définition correspond au fait que toutes les solutions réalisables peuvent être construites à partir de la solution centrale $\bar{x}_0$ par une certaine exécution d'un procédé glouton (en d'autres termes, il s'agit de problèmes résolubles par une machine de TURING polynomiale non déterministe de type glouton). Il s'agit donc bien de problèmes adaptés à une résolution gloutonne qui n'exclue a priori aucune solution réalisable.

Les problèmes héréditaires sont l'exemple le plus classique de cette situation : la solution de départ $\bar{x}^{(0)}$ est la solution vide correspondant au vecteur $\vec{0}$ et l'opération élémentaire représentée par l'algorithme φ est le retrait d'un élément. Une démarche gloutonne consiste alors à construire une solution par sélections successives. Le problème de stable maximum d'un graphe appartient à cette classe mais de très nombreux autres problèmes non héréditaires entrent dans le cadre de la définition 5. Citons notamment les problèmes de couverture minimum d'ensembles ou de sommets, le *bin-packing* ou encore la coloration minimum d'un graphe. Par contre, le problème de la coloration minimum d'un graphe 4-colorable⁹ n'est pas radial.

Si on désigne par $\sigma(I)$ le nombre de valeurs réalisables d'une instance I d'un problème donné, un argument fondé sur un parcours d'arbre permet d'établir le résultat suivant.

PROPOSITION 7. $\sigma(I)$ est un ordre de difficulté pour la résolution exacte pour les problèmes radiaux **NP-complets**.

Cette discussion prolonge une étude menée dans [2]. Si on note $\omega(I)$ la pire valeur (au sens de l'objectif) de l'instance I et $\beta(I)$ sa valeur optimale, la notion de problèmes simples¹⁰ ([2]) correspond aux problèmes pour lesquels $d(I) = |\beta(I) - \omega(I)|$ est un ordre de difficulté pour la résolution exacte (i.e., pour le niveau d'approximation correspondant au rapport constant égal à 1). Le problème de stable maximum est un exemple de problème simple. A l'inverse, le problème *max-weighted-sat* ([2]) n'est pas simple.

Si on se limite pour $d(I)$ à la quantité $\beta(I)$ (on retrouve alors la notion de problèmes simples au sens de [27]), d reste un ordre de difficulté pour le problème de stable maximum mais ne l'est plus pour le problème de coloration, la k -colorabilité, pour $k \geq 3$ étant **NP-complet** ([14]).

Les ordres de difficulté que nous venons de citer ne sont pas évaluable en temps polynomial ce qui, pour certaines applications, peut s'avérer particulièrement gênant. Pour les exemples qui suivent, au contraire, la valeur de l'ordre de difficulté est évaluable en temps polynomial par rapport à la taille de l'instance.

Pour tout problème **NP-complet**, la taille de l'instance constitue un ordre de difficulté pour le niveau de la résolution exacte.

9. Une instance de ce problème étant la donnée d'un graphe G et d'une 4-coloration de G , les colorations réalisables ayant au moins 4 couleurs.

10. Un problème **NPO** est *simple* si, pour toute constante k , sa restriction aux instances dont la valeur optimale est bornée par k est polynomiale.

Envisageons maintenant la notion de problème pseudo-polynomial définie dans [14] à partir d'une fonction $\max : I \rightarrow (\max(I))$ représentant la valeur du plus grand nombre intervenant dans I ; par exemple, dans le cas de problèmes pondérés, $\max$ est la valeur du plus grand poids. Les problèmes NP-complets pseudo-polynomiaux s'interprètent comme les problèmes pour lesquels $\log(|\max|)$ est un ordre de difficulté pour la résolution exacte.

4.2.2 Notions asymptotiques

De nombreux résultats d'approximation sont, pour des besoins techniques, exprimés dans un cadre *asymptotique* sans que cette notion ne soit en général clairement définie. Néanmoins, l'idée est toujours de se restreindre à une classe d'instances « intéressantes » correspondant, dans un sens ou dans un autre, à des instances « tendant vers l'infini ». Ainsi, par exemple, les résultats concernant le stable maximum d'un graphe dans [15] sont exprimés pour des graphes ayant un grand degré. Ces résultats, qualifiés d'asymptotiques par leurs auteurs, font en fait apparaître des rapports d'approximation du type $6/\Delta + o(1/\Delta)$, ce qui effectivement leur donne un intérêt lorsque le terme en $o(1/\Delta)$ est minoritaire. Il est donc naturel d'interpréter dans ce cas le terme *asymptotique* comme « à degré Δ infiniment grand ». Mais, jusqu'à présent, les notions asymptotiques ne sont pas précisément définies ni justifiées ; elles sont tout au plus suggérées au gré des besoins pour tel ou tel problème. La réflexion que nous venons de mener permet de donner un sens général à ces notions, de les justifier et de valider ou discuter a posteriori la pertinence de plusieurs résultats considérés comme classiques.

Ce cadre est bien lié à la notion d'instances difficiles. En effet, le point de vue de ne prendre en compte que les instances « tendant vers l'infini » n'est intéressant que dans la mesure où les instances considérées sont les plus significatives, c'est-à-dire, d'une certaine façon, les plus difficiles. Dans ce cas, il paraît pertinent de se restreindre à de telles instances en exploitant les simplifications techniques qu'elles procurent sans affecter la teneur du résultat. Sous réserve de donner un sens précis à ce discours informel, on reconnaît une démarche analogue au paragraphe précédent consistant à exploiter la particularité d'instances spéciales dans le cadre d'une analyse pour le problème général ; la proposition 9 illustre cette similitude. La différence, dans le cas asymptotique, est que la notion de difficulté doit être quantifiée pour donner lieu à un concept de limite. La notion d'ordre de difficulté a justement pour vocation de jouer ce rôle. Remarquons d'ailleurs que, dans la définition 4, le fait que Π n'appartient pas¹¹ au niveau $\mathcal{F}$ permet d'établir le résultat suivant.

PROPOSITION 8. *Soit un problème Π d'ensemble d'instances $\mathcal{I}$ et d'un ordre de difficulté pour Π par rapport à un niveau d'approximation $\mathcal{F}$. Alors, $\sup_{I \in \mathcal{I}} d(I) = +\infty$.*

Nous proposons de définir des notions asymptotiques en considérant les instances ayant un grand degré de difficulté. Etant donné un problème Π et un degré de difficulté d , un résultat d'approximation asymptotique se rapporte aux instances I vérifiant $d(I) \geq D$ où D est une constante arbitrairement grande. Cette notion dépend du degré de difficulté choisi mais permet une définition générique pour l'ensemble des problèmes.

La démarche consiste donc à exclure les instances dont le degré de difficulté ne dépasse pas D ; par définition, de telles instances sont plus faciles à approcher que le problème général, ce qui justifie la pertinence de ce point de vue dans le cadre de l'approximation. Intuitivement, le résultat, qualifié d'asymptotique, peut s'interpréter comme la « partie prépondérante » d'un résultat plus lourd à exprimer pour l'ensemble des instances. Sous ce point de vue, la partie « négligée », ou plutôt non spécifiée, correspond à des instances mieux approximables, ce qui justifie qu'on s'autorise, dans la formulation du résultat, à se focaliser sur les instances de grand

11. Sous une hypothèse du type $\mathbf{P} \neq \mathbf{NP}$.

degré de difficulté.

DEFINITION 6. Etant donné un ordre de difficulté (associé à un niveau d'approximation $\mathcal{F}$) $d : \mathcal{I} \rightarrow \mathbb{N}$ des instances d'un problème Π , on dit qu'une chaîne algorithmique $(\mathcal{A}_k)_{k \in \mathbb{N}}$ pour Π a un *rapport asymptotique* $f \in \mathcal{F}_\Pi$ si : $\forall \epsilon > 0, \forall k \in \mathbb{N}, \exists D$ tel que, $\forall I \in \mathcal{I}$, avec $d(I) \geq D$, $\rho_k(I) \geq f(I, k)(1 - \epsilon)$. ■

Dans le cadre plus restreint des algorithmes approchés, la définition équivalente devient $(f \in \mathcal{F}'_\Pi) : \forall \epsilon > 0, \exists D$ tel que, $\forall I \in \mathcal{I}$, avec $d(I) \geq D$, $\rho_k(I) \geq f(I)(1 - \epsilon)$, si bien que la définition 6 correspond à une suite de rapports asymptotiques.

La propriété suivante peut être interprétée comme une réduction d'un sous-problème au problème général et se déduit immédiatement des définitions.

PROPOSITION 9. *En adoptant les notations de la définition 6, si $f \in \mathcal{F}'$, il existe une chaîne à rapport dans $\mathcal{F} \cup \mathcal{F}'$.*

En particulier, comme $\Pi \notin \mathcal{F}$, nécessairement $f \notin \mathcal{F}$. Le résultat précédent est intéressant dans la mesure où le niveau $\mathcal{F}$ est considéré comme meilleur que le rapport f et en particulier lorsque, $\forall f' \in \mathcal{F}, f \leq f'$.

Afin de situer la définition 6 par rapport à la littérature, notons que la notion de rapport asymptotique la plus courante ([14]) consiste à remplacer, dans la définition précédente, $d(I)$ par la valeur optimale $\beta(I)$; un schéma d'approximation asymptotique ([20]) correspond à une chaîne à rapport asymptotique $1 - 1/k$. Pour des problèmes simples ([27]), cette définition est un cas particulier de la définition 6; par contre, dans le cas général et notamment lorsque β n'est pas un degré de difficulté, elle perd un peu de son intérêt. En tout cas, il semble qu'aucun argument ne justifie qu'on puisse, pour des problèmes non-simples ([27]), rendre compte d'un rapport du type $\rho(I)[1 - \epsilon(\beta(I))]$, avec $\lim_{x \rightarrow \infty} \epsilon(x) = 0$, par sa partie $\rho(I)$. Pour le problème de bin-packing par exemple, de nombreux résultats d'approximation s'expriment par un rapport $\gamma(I) = \rho - k/\beta(I)$ avec ρ et k constants. Nous pensons notamment à l'analyse ([14]) de l'algorithme *first-fit-decreasing* pour *bin-packing* ($\rho = 9/11$) ou encore ([13]) au schéma d'approximation asymptotique à rapport $1 - \epsilon - k/\beta(I)$. Vu que pour ce problème les instances à faible valeur optimale semblent justement les plus difficiles à approximer (en particulier aucun rapport strictement supérieur à $2/3$ ne peut être garanti pour les instances telles que $\beta(I) = 2$), la forme générale des rapports en fonction de $\beta(I)$ paraît plus représentative que leur partie constante.

Du point de vue différentiel maintenant, la première définition utilisée ([7]) fait référence à l'ordre de difficulté $\omega(I) - \beta(I)$ pour les problèmes radiaux. L'étendue de cette classe permet d'adopter $\sigma(I)$ comme ordre de difficulté pour de nombreux problèmes et l'opportunité de comparer des résultats entre des problèmes de nature différente. Remarquons d'ailleurs que la notion de support est conservée par l'équivalence ([10]) sur laquelle repose la théorie du rapport différentiel, ce qui rend cet ordre de difficulté particulièrement adapté à ce cadre.

Notons enfin que les résultats que nous avons mentionnés pour les problèmes de stabilité peuvent s'exprimer en termes asymptotiques. En effet, le degré maximum Δ constitue un ordre de difficulté pour l'approximation à rapport constant du problème de stable, si bien que la proposition 1 correspond aux rapports asymptotiques $6/\Delta(1 - 1/k)$, $6/\Delta$, k/Δ et k/μ pour le problème de stable maximum.

4.3 Famille critique d'instances

Pour conclure, évoquons des travaux en cours qui s'inscrivent dans la continuité de la problématique que nous venons d'étudier. L'idée est toujours de définir une notion de difficulté des instances en vue d'étudier des résultats d'approximation mais, contrairement aux notions discutées dans les paragraphes précédents, ce nouveau point de vue se rapporte à un algorithme ou à

une classe d'algorithmes. L'objet n'est plus de définir une difficulté intrinsèque de certaines instances mais de réfléchir, étant donné un algorithme, à la structure des instances pour lesquelles il se comporte le moins bien.

DEFINITION 7. ([19]) Etant donné un problème Π (d'ensembles d'instances $\mathcal{I}$), un algorithme approché $\mathcal{A}$ pour Π et une mesure d'approximation ρ ($\rho \in [0, 1]$ et d'autant plus proche de 1 que l'instance est bien résolue), une famille d'instances $S \subset \mathcal{I}$ est *critique* pour $\mathcal{A}$ si, $\forall I_0 \in \mathcal{I}, \exists I_1 \in S, \rho_{\mathcal{A}}(I_1) \leq \rho_{\mathcal{A}}(I_0)$. ■

Une conséquence immédiate de cette définition est que si l'algorithme $\mathcal{A}$ garantit un rapport constant ρ_0 pour les instances S , cette analyse est valable pour toute instance. Bien sûr, cette notion n'est adaptée qu'à l'analyse d'algorithmes à rapports constants. Il serait facile de l'étendre au cas d'autres niveaux d'approximation ; cependant, ce cadre restreint suffit pour montrer l'esprit de la démarche et son intérêt opérationnel. Remarquons aussi la similarité avec le paragraphe 4.1 mais la différence essentielle est qu'ici la réduction du problème général au sous-problème n'est valable que pour un algorithme. Dans la pratique, la forme particulière de l'algorithme implique une structure des instances associées beaucoup plus riche que dans le cadre de réductions générales. C'est ce qui rend cette notion opérante mais, par contre, les instances mises en évidence n'ont pas de signification particulière en termes de difficulté ; elles sont plutôt significatives du comportement de $\mathcal{A}$. Néanmoins, les deux points de vue pourraient être approchés par l'étude de notions intermédiaires correspondant à des instances critiques pour des familles d'algorithmes.

Ce cadre restreint étudié dans [19] s'est d'ores et déjà avéré particulièrement fécond. En particulier, sur l'exemple du bin-packing, nous mettons en évidence dans [9] une famille d'instances critiques pour l'algorithme *first-fit-decreasing* et la mesure différentielle. La preuve illustre comment la définition 7 intervient dans la conception de la famille S et met en évidence des propriétés vérifiées par toute famille d'instances critiques pour cet algorithme et cette mesure. Dans un second temps, une simple évaluation de la mesure sur la famille trouvée permet d'établir que l'algorithme *first-fit-decreasing* garantit le rapport différentiel $3/4$.

5 CONCLUSION

Dans ce travail, nous avons discuté différentes notions de difficulté (du point de vue de l'approximation) associées à un problème ou à une instance. Le cadre classique de la théorie de la complexité est également couvert par notre point de vue en considérant la résolution exacte comme résolution approchée avec garantie 1. De même, associer difficulté et possibilité de résolution permet d'inclure la notion d'instance critique bien qu'elle se rapporte à un algorithme précis.

Notre discussion a mis en évidence le caractère central des concepts de difficulté en théorie de la complexité : ils sont apparus aussi bien en tant que justification de définitions et outils que comme sujet même d'étude. Ils interviennent aussi sous forme d'outils avec un réel intérêt opérationnel. Cette étude a été l'occasion de présenter des problématiques, des démarches ou encore des outils d'approximation polynomiale. En particulier, nous avons rencontré la notion de famille d'algorithmes et les analyses associées permettant de mettre en évidence une gamme riche de niveaux de résolution et donc aussi de niveaux de difficulté. Cette multiplicité des types de difficulté associés aux problèmes NP-complets¹² est même l'un des atouts de l'approximation. Les notions de réduction sont particulièrement adaptées pour comparer la difficulté de l'approximation de problèmes difficiles et ont été identifiées, à ce titre, comme un outil tout à fait central. Elles ont l'intérêt non seulement de permettre de comparer et hiérarchiser les problèmes d'optimisation NP-complets mais aussi de donner lieu à de nouveaux résultats à partir d'analyses

12. Qui sont par définition de difficulté équivalente du point de vue de la résolution exacte.

pour des problèmes différents. Cette remarque est d'ailleurs générale : tous les concepts que nous avons discutés proviennent d'une réflexion de fond sur l'objet de l'approximation et permettent de l'asseoir sur un formalisme précis mais ils ont aussi chacun un intérêt opérationnel.

BIBLIOGRAPHIE

- [1] ARORA, S., LUND, C., MOTWANI, R., SUDAN, M., and SZEGEDY, M. Proof verification and intractability of approximation problems. In *Proc. FOCS'92*, pages 14–23. 1992.
- [2] AUSIELLO, G., CRESCENZI, P., and PROTASI, M. Approximate solutions of NP optimization problems. *Theoret. Comput. Sci.*, 150:pages 1–55, 1995.
- [3] AUSIELLO, G., D'ATRI, A., and PROTASI, M. Structure preserving reductions among convex optimization problems. *J. Comput. System Sci.*, 21:pages 136–153, 1980.
- [4] COOK, S. A. The complexity of theorem-proving procedures. In *Proc. STOC'71*, pages 151–158. 1971.
- [5] CRESCENZI, P. A short guide to approximation preserving reductions. In *Proc. Conference on Computational Complexity*, pages 262–273. 1997.
- [6] CRESCENZI, P. and PANCONESI, A. Completeness in approximation classes. In *Proc. Fundamentals of Computation Theory*, number 380 in Lecture Notes in Computer Science, pages 116–126. Springer Verlag, 1989.
- [7] DEMANGE, M., GRISONI, P., and PASCHOS, V. TH. Differential approximation algorithms for some combinatorial optimization problems. *Theoret. Comput. Sci.*, 209:pages 107–122, 1998.
- [8] DEMANGE, M., MONNOT, J., and PASCHOS, V. TH. Bridging gap between standard and differential polynomial approximation: the case of bin-packing. *Appl. Math. Lett.*, 12:pages 127–133, 1999.
- [9] DEMANGE, M., MONNOT, J., and PASCHOS, V. TH. Maximizing the number of unused bins by first-fit-decreasing. A paraître dans *Foundations on Comp. and Dec. Sci.*
- [10] DEMANGE, M. and PASCHOS, V. TH. On an approximation measure founded on the links between optimization and polynomial approximation theory. *Theoret. Comput. Sci.*, 158:pages 117–141, 1996.
- [11] DEMANGE, M. and PASCHOS, V. TH. Valeurs extrémales d'un problème d'optimisation combinatoire et approximation polynomiale. *Math. Inf. Sci. Humaines*, 135:pages 51–66, 1996.
- [12] DEMANGE, M. and PASCHOS, V. TH. Improved approximations for maximum independent set via approximation chains. *Appl. Math. Lett.*, 10(3):pages 105–110, 1997.
- [13] FERNANDEZ DE LA VEGA, W. and LUEKER, G. S. Bin packing can be solved within $1 + \epsilon$ in linear time. *Combinatorica*, 1(4):pages 349–355, 1981.
- [14] GAREY, M. R. and JOHNSON, D. S. *Computers and intractability. A guide to the theory of NP-completeness*. W. H. Freeman, San Francisco, 1979.
- [15] HALLDÓRSSON, M. M. and RADHAKRISHNAN, J. Improved approximations of independent sets in bounded-degree graphs via subgraph removal. *Nordic Journal of Computing*, 1(4):pages 475–492, 1994.
- [16] KARP, R. M. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of computer computations*, pages 85–103. Plenum Press, New York, 1972.
- [17] LEWIS, H. R. and PAPADIMITRIOU, C. H. *Elements of the theory of computation*. Prentice-Hall, 1981.

- [18] LUND, C. and YANNAKAKIS, M. On the hardness of approximating minimization problems. *J. Assoc. Comput. Mach.*, 41(5):pages 960–981, 1994.
- [19] MONNOT, J. Familles d’instances critiques et approximation polynomiale. Thèse de Doctorat, LAMSADE. Université Paris-Dauphine, 1998.
- [20] MOTWANI, R. *Lecture notes on approximation algorithms – Volume I*. Stanford University, 1993.
- [21] NEMHAUSER, G. L., WOLSEY, L. A., and FISCHER, M. L. An analysis of approximations for maximizing submodular set functions. *Math. Programming*, 14:pages 265–294, 1978.
- [22] ORPONEN, P. and MANNILA, H. On approximation preserving reductions: complete problems and robust measures. Technical Report C-1987-28, Dept. of Computer Science, University of Helsinki, Finland, 1987.
- [23] PANCONESI, A. and RANJAN, D. Quantifiers and approximation. In *Proc. STOC’90*, pages 446–456. 1990.
- [24] PAPADIMITRIOU, C. H. and YANNAKAKIS, M. Optimization, approximation and complexity classes. *J. Comput. System Sci.*, 43:pages 425–440, 1991.
- [25] PASCHOS, V. TH. and DEMANGE, M. A generalization of König-Egervary graphs and heuristics for maximum independent set problem with improved approximation ratios. *European J. Oper. Res.*, 97:pages 580–592, 1997.
- [26] PASCHOS, V. TH. and RENOTTE, L. Approximability preserving reductions for NP-complete problems. *Foundations of Computing and Decision Sciences*, 20(1):pages 49–71, 1995.
- [27] PAZ, A. and MORAN, S. Non deterministic polynomial optimization problems and their approximations. *Theoret. Comput. Sci.*, 15:pages 251–277, 1981.
- [28] SIMON, H. U. On approximate solutions for combinatorial optimization problems. *SIAM J. Disc. Math.*, 3(2):pages 294–310, 1990.