

Réduction optimale d'un pavé de tris en temps optimal*

Noëlle Bleuzen-Guernalec et Alain Colmerauer[†]

31 janvier 1999

Résumé

Soit \mathbf{D} un ensemble totalement ordonné. Appelons n -pavé, tout produit cartésien de n intervalles de \mathbf{D} , fermés et éventuellement vides. Désignons par *tri* l'ensemble des $2n$ -uplets d'éléments de \mathbf{D} qui sont de la forme (x_1, \dots, x_{2n}) , où (x_{n+1}, \dots, x_{2n}) est le n -uplet obtenu en rangeant dans un ordre faiblement croissant les éléments du n -uplet (x_1, \dots, x_n) .

Nous présentons et justifions un algorithme de complexité $\mathcal{O}(n \log n)$ qui, étant donné un $2n$ -pavé a , calcule un $2n$ -pavé qui, au sens de l'inclusion, est le plus petit pavé contenant l'ensemble $tri \cap a$. Nous montrons que cette complexité est optimale.

1 Introduction

Etant donné un $2n$ -uplet (x_1, \dots, x_{2n}) d'éléments d'un ensemble totalement ordonné, la contrainte de tri

$$(x_1, \dots, x_n, x_{n+1}, \dots, x_{2n}) \in tri$$

exprime que le n -uplet (x_{n+1}, \dots, x_{2n}) est égal au n -uplet obtenu en triant les éléments du n -uplet (x_1, \dots, x_n) dans un ordre faiblement croissant. W.J. Older, F. Swinkels et M. van Emden ont introduit cette contrainte pour formuler élégamment et résoudre des problèmes de type ordonnancement d'atelier [4]. Plus récemment Jianyang Zhou [5] a résolu des problèmes d'ordonnancement d'atelier réputés difficiles en introduisant une contrainte de tri avec $3n$ variables, les n variables entières supplémentaires servant à expliciter une permutation liant les x_i .

Nous nous intéressons à *réduire optimalement* une contrainte dite *de base* de tri et qui est de la forme :

$$(x_1, \dots, x_{2n}) \in tri \wedge x_1 \in a_1 \wedge \dots \wedge x_{2n} \in a_{2n}, \quad (1)$$

où les a_i sont des intervalles fermés éventuellement vides du domaine des x_i . Par « réduire optimalement » nous entendons : substituer aux intervalles a_i les intervalles b_i qui, pour l'inclusion, sont les plus petits intervalles qui ne changent pas l'ensemble des solutions de la contrainte, c'est-à-dire qui sont tels que

$$tri \cap a_1 \times \dots \times a_{2n} = tri \cap b_1 \times \dots \times b_{2n}$$

Il est à noter que pour déterminer si la contrainte de base de tri admet au moins une solution, il suffit de vérifier qu'après réduction, aucun des nouveaux intervalles n'est vide.

Ainsi que nous l'a enseigné BNR-Prolog, cette réduction s'applique à toutes sortes de contraintes de base et permet de progresser dans la résolution d'une conjonction de contraintes de base en réduisant tour à tour chaque contrainte jusqu'à ce que les intervalles attachés aux différentes variables atteignent un point fixe [3, 1].

Si on pose $a = a_1 \times \dots \times a_{2n}$, le problème de réduire au mieux la contrainte (1) revient à calculer le $2n$ -pavé b qui pour l'inclusion est le plus petit pavé qui respecte l'égalité $tri \cap a = tri \cap b$. On montre que ce pavé b est aussi le plus petit pavé qui contient l'ensemble $tri \cap a$. L'objet de cet article

*A paraître en anglais dans la revue *Constraints* au cours de l'année 1999.

[†]Laboratoire d'Informatique de Marseille, ESA 6077, CNRS et Universités de Provence et de la Méditerranée

est de présenter et de justifier un algorithme de complexité $\mathcal{O}(n \log n)$ pour calculer un tel $2n$ -pavé b . Le fait que cette complexité soit optimale découle d'une simple remarque. L'article s'organise en 11 sections :

- 1 Introduction
- 2 Situation du problème
- 3 Pavé normalisé
- 4 Pavé de type I
- 5 Pavé de type II
- 6 Pavé de type III
- 7 Décomposition d'un pavé de type III en pavés de type I
- 8 Algorithme
- 9 Correction de l'algorithme
- 10 Exemple
- 11 Conclusion

La section 2 introduit la terminologie et le fond du problème : calculer les extrémités des $2n$ projections d'un ensemble de $2n$ -uplets lié à la relation *tri*, en $\mathcal{O}(n \log n)$ instructions élémentaires. Les sections 3,4,5,6,7 établissent quatre théorèmes pour obtenir les plus petits éléments des n premières projections et les plus grands éléments des n dernières projections. La section 8 présente un argument de symétrie pour obtenir les autres extrémités et détaille un algorithme pour réduire au mieux une contrainte de base de tri en $\mathcal{O}(n \log n)$ instructions élémentaires. Les quatre théorèmes précédents permettent de justifier l'algorithme dans la section 9 et le déroulement complet d'un exemple illustre l'algorithme dans la section 10. Nous concluons par des compléments d'information sur cet algorithme.

2 Situation du problème

2.1 Relation globale

Soit \mathbf{D} un ensemble quelconque. On désigne par \mathbf{D}^* l'ensemble de toutes les suites finies d'éléments de \mathbf{D} , y compris la suite vide. Celle-ci est notée ε , sa longueur est nulle et les autres suites finies de longueurs n non nulles sont notées sous forme de n -uplets (d_1, \dots, d_n) , avec les d_i pris dans \mathbf{D} . On désigne par \mathbf{D}^n l'ensemble des suites de \mathbf{D}^* qui sont de longueur n . Si d est une suite de \mathbf{D}^* , on note $set(d)$ l'ensemble des éléments de \mathbf{D} qui apparaissent dans d . On note par un point l'opération de concaténation entre les éléments de \mathbf{D}^* . On a notamment $d \cdot \varepsilon = \varepsilon \cdot d = d$, pour tout d pris dans \mathbf{D}^* .

Alors qu'une *relation n -aire sur \mathbf{D}* est un sous-ensemble de \mathbf{D}^n , une *relation globale sur \mathbf{D}* est un sous-ensemble r de \mathbf{D}^* . Pour tout entier strictement positif i , la *i^e projection* de r est le sous-ensemble éventuellement vide de \mathbf{D} :

$$\pi_i(r) = \{e \in \mathbf{D} \mid \text{il existe un élément de } r \text{ de la forme } (d_1, \dots, d_n) \text{ avec } n \geq i \text{ et } e = d_i\}.$$

2.2 Intervalles et pavés

D'une façon générale, si (\mathbf{D}, \preceq) est un ensemble ordonné, formé d'un ensemble \mathbf{D} d'au moins deux éléments et d'une relation d'ordre total \preceq sur \mathbf{D} , nous appelons *intervalle de (\mathbf{D}, \preceq)* tout sous-ensemble éventuellement vide de \mathbf{D} , qui est de la forme

$$\{d \in \mathbf{D} \mid e \preceq d \text{ et } d \preceq e'\},$$

où e et e' sont des éléments quelconques de \mathbf{D} . Nous le notons $[e, e']$. Le fait que nous n'imposons pas $e \preceq e'$ et que l'ensemble \mathbf{D} ait plus d'un élément assure que l'ensemble vide fait partie des intervalles. Dans le cas particulier où l'ensemble ordonné est l'ensemble (\mathbf{N}, \leq) des entiers naturels, nous écrivons $e..e'$ au lieu de $[e, e']$.

Un *pavé de dimension n* , ou *n -pavé*, sur (\mathbf{D}, \preceq) est un sous-ensemble a de \mathbf{D}^n de la forme $a_1 \times \dots \times a_n$, avec $n \geq 1$, où chaque a_i est un intervalle de (\mathbf{D}, \preceq) . Du fait que les a_i peuvent être

vides, l'ensemble vide \emptyset est un pavé de dimension n pour tout n . Il est à noter que le n -uplet $(a_1, \dots, a_n) = (\pi_1(a), \dots, \pi_n(a))$ est tel que $a = a_1 \times \dots \times a_n$ et, si a n'est pas vide, les intervalles a_i tels que $a = a_1 \times \dots \times a_n$ sont uniques.

Soit a un sous-ensemble de \mathbf{D} . Nous désignons par \underline{a} et \bar{a} , lorsqu'ils existent, le plus petit élément et le plus grand élément de a dans (\mathbf{D}, \preceq) . Nous gardons les notations classiques $\min a$ et $\max a$ pour l'ensemble ordonné particulier (\mathbf{N}, \leq) . Nous désignons par $\text{apx}(a)$ le plus petit intervalle de (\mathbf{D}, \preceq) qui contient a , s'il existe. La notion de plus petit est prise au sens de l'inclusion. Bien entendu si \underline{a} et \bar{a} existent, $\text{apx}(a)$ existe et est égal à $[\underline{a}, \bar{a}]$. Nous généralisons la notation $\text{apx}(a)$ comme suit :

Définition 1 Si r un sous-ensemble de \mathbf{D}^n , nous désignons par $\text{apx}(r)$ le plus petit n -pavé sur (\mathbf{D}, \preceq) qui contient r , s'il existe.

Ici aussi la notion de plus petit est prise au sens de l'inclusion. Si s est un sous-ensemble de \mathbf{D}^n , on montre facilement que :

Propriété 1 Soit les ensembles $\text{apx}(s)$ et $\text{apx}(\pi_1(s)) \times \dots \times \text{apx}(\pi_n(s))$ existent et sont égaux, soit ils n'existent pas.

Enfin nous disons qu'un n -uplet (d_1, \dots, d_n) est *croissant pour \preceq* si pour chaque $i \in 1..(n-1)$ on a $d_i \preceq d_{i+1}$.

2.3 Formulation du problème

Donnons nous une bonne fois pour toutes un ensemble totalement ordonné (\mathbf{D}, \preceq) . En particulier (\mathbf{D}, \preceq) peut être l'ensemble ordonné (\mathbf{R}, \leq) des réels, l'ensemble ordonné (\mathbf{Z}, \leq) des entiers relatifs, l'ensemble ordonné (\mathbf{N}, \leq) des entiers naturels ou n'importe quel sous-ensemble de \mathbf{R} ordonné par \leq . Naturellement, nous utilisons le signe $<$ pour la relation \preceq privée de ses couples d'éléments égaux. Désignons par *tri* la relation globale sur \mathbf{D} définie par :

$$d \in \text{tri} \iff \left(\begin{array}{l} d \text{ est de la forme } (d_1, \dots, d_{2n}), \\ \text{avec } n \geq 1, \text{ et} \\ (d_{n+1}, \dots, d_{2n}) \text{ est croissant pour } \preceq \text{ et} \\ \text{il existe une applications bijective } f \\ \text{de } 1..n \text{ dans } (n+1)..(2n) \text{ telle que} \\ (d_1, \dots, d_n) = (d_{f(1)}, \dots, d_{f(n)}) \end{array} \right)$$

Par exemple, dans l'ensemble ordonné (\mathbf{Z}, \leq) , on a

$$(5, 12, 10, 5, 12, 3, 3, 5, 5, 10, 12, 12) \in \text{tri}.$$

Le problème que nous posons est donc, étant donné un $2n$ -pavé a sur (\mathbf{D}, \preceq) , de montrer que l'ensemble $\text{apx}(\text{tri} \cap a)$ existe toujours et de le calculer par un algorithme de complexité optimale $\mathcal{O}(f(n))$.

Bien entendu, par un algorithme de complexité $\mathcal{O}(f(n))$, nous entendons qu'il existe un réel strictement positif c , indépendant de n , tel que pour tout n , le nombre d'instruction élémentaires exécutées soit inférieur ou égal à $cf(n)$. Par un algorithme de complexité *optimale* $\mathcal{O}(f(n))$ nous exigeons en plus qu'il existe un réel strictement positif c' , indépendant de n , tel que pour tout n , tout autre algorithme, qui calcule $\text{apx}(\text{tri} \cap a)$, exécute un nombre d'instructions élémentaires supérieur à $c'f(n)$ pour au moins une valeur de a .

Les opérations élémentaires sont celles d'une machine fictive qui dispose d'une infinité de cellules, chacune pouvant contenir un élément quelconque de $\mathbf{D} \cup \mathbf{N}$. Sans entrer dans les détails, ces opérations élémentaires comprennent : le transfert du contenu d'une cellule dans une autre, la comparaison de deux éléments de \mathbf{D} pour \preceq et la comparaison de deux éléments de \mathbf{N} pour \leq .

Si, pour a , on prend un $2n$ -pavé de la forme $[e_1, e_1] \times \dots \times [e_n, e_n] \times [d_1, d_2] \times \dots \times [d_1, d_2]$, avec $d_1 = \{e_1, \dots, e_n\}$ et $d_2 = \overline{\{e_1, \dots, e_n\}}$, le calcul de $\text{apx}(\text{tri} \cap a)$ revient à trier le n -uplet (d_1, \dots, d_n) . Un tel tri nécessite au moins $c'n \log n$ comparaisons des d_i pour traiter le pire des cas. (Le lecteur

pourra consulter la section 5.3.1 du livre [2] de D.E. Knuth à ce sujet.) Il s'ensuit qu'un algorithme pour calculer $\text{apx}(tri \cap)$, avec une complexité en $\mathcal{O}(n \log n)$, aura une complexité optimale.

Pour calculer $\text{apx}(tri \cap a)$ nous nous servons de la propriété 1. En posant $s = tri \cap a$, ce calcul se ramène alors au calcul des $2n$ intervalles $\text{apx}(\pi_i(s))$, qui lui même se ramène au calcul de leurs extrémités $\underline{\pi_i(s)}$ et $\overline{\pi_i(s)}$.

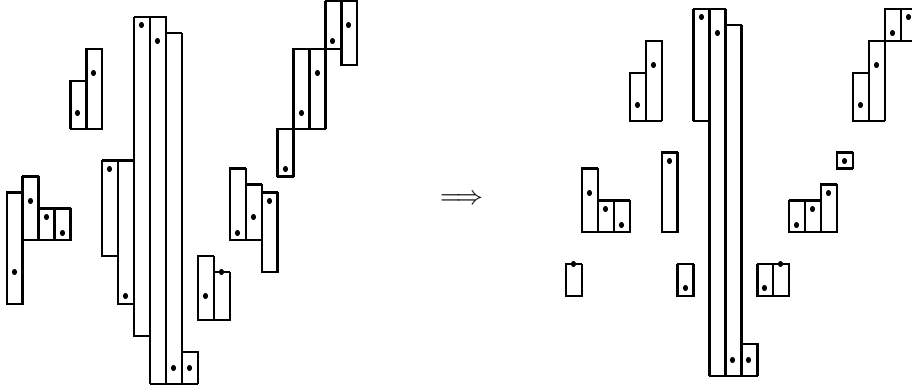
Exemple 1 Terminons par un exemple. Si dans (\mathbf{Z}, \leq) on considère le 22-pavé

$$a = [10, 24] \times [18, 26] \times [18, 22] \times [18, 22] \times [32, 38] \times [32, 42] \times [16, 28] \times [10, 28] \times [6, 46] \times [0, 46] \times [0, 44] \times [0, 4] \times [8, 16] \times [8, 14] \times [18, 27] \times [18, 25] \times [14, 24] \times [26, 32] \times [32, 42] \times [32, 42] \times [42, 48] \times [40, 48],$$

il faut obtenir

$$\text{apx}(tri \cap a) = [10, 14] \times [18, 26] \times [18, 22] \times [18, 22] \times [32, 38] \times [32, 42] \times [16, 28] \times [10, 14] \times [32, 46] \times [0, 46] \times [0, 44] \times [0, 4] \times [10, 14] \times [10, 14] \times [18, 22] \times [18, 22] \times [18, 24] \times [26, 28] \times [32, 38] \times [32, 42] \times [42, 46] \times [42, 46].$$

La transformation $a \implies \text{apx}(tri \cap a)$ se visualise bien par le schéma :



Les intervalles sont représentés par des rectangles et les points à l'intérieur de ceux-ci représentent un élément de $tri \cap a$ choisi au hasard, ici le 22-uplet

$$(14, 23, 21, 19, 34, 39, 27, 11, 45, 43, 2) \cdot (2, 11, 14, 19, 21, 23, 27, 34, 39, 43, 45).$$

3 Pavé normalisé

Nous montrons ici une comment transformer un pavé a sur (\mathbf{D}, \preceq) en un pavé b normalisé ou vide qui est tel que $tri \cap a = tri \cap b$. Nous étudions aussi les deux propriétés essentielles d'un pavé normalisé.

Définition 2 Un $2n$ -pavé normalisé est un $2n$ -pavé non vide sur (\mathbf{D}, \preceq) qui est de la forme $a_1 \times \dots \times a_{2n}$, avec les n -uplets $(\underline{a_{n+1}}, \dots, \underline{a_{2n}})$ et $(\overline{a_{n+1}}, \dots, \overline{a_{2n}})$ croissants pour \preceq .

3.1 Normalisation

Si (d_1, \dots, d_n) est un élément de \mathbf{D}^n , on définit les n -uplets $(e_1, \dots, e_n) = \uparrow(d_1, \dots, d_n)$ et $(e'_1, \dots, e'_n) = \downarrow(d_1, \dots, d_n)$ comme suit :

$$e_1 = d_1 \quad \text{et} \quad e_{i+1} = \overline{\{e_i, d_{i+1}\}}, \text{ pour chaque } i \text{ de } 1..(n-1),$$

$$e'_n = d_n \quad \text{et} \quad e'_{i-1} = \underline{\{e'_i, d_{i-1}\}}, \text{ pour chaque } i \text{ de } 2..n.$$

De la définition de la relation globale tri , on déduit immédiatement :

Propriété 2 Soit $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé non vide sur (\mathbf{D}, \preceq) et soit le $2n$ -pavé

$$b = a_1 \times \dots \times a_n \times [e_1, e'_1] \times \dots \times [e_n, e'_n], \quad \text{avec} \quad \begin{aligned} (e_1, \dots, e_n) &= \uparrow(\underline{a_{n+1}}, \dots, \underline{a_{2n}}), \\ (e'_1, \dots, e'_n) &= \downarrow(\overline{a_{n+1}}, \dots, \overline{a_{2n}}). \end{aligned}$$

On a alors $tri \cap a = tri \cap b$ et le pavé b est soit vide soit normalisé.

Exemple 2 Si dans (\mathbf{R}, \leq) on prend pour $2n$ -pavé

$$a = [0, 13] \times [6, 10] \times [10, 11] \times [4, 16] \times [4, 6] \times [1, 3] \times [5, 10] \times [6, 9] \times [11, 17] \times [10, 15],$$

on a

$$\begin{aligned} \uparrow(1, 5, 6, 11, 10) &= (1, 5, 6, 11, 11), \\ \downarrow(3, 10, 9, 17, 15) &= (3, 9, 9, 15, 15) \end{aligned}$$

et donc le pavé

$$b = [0, 13] \times [6, 10] \times [10, 11] \times [4, 16] \times [4, 6] \times [1, 3] \times [5, 9] \times [6, 9] \times [11, 15] \times [11, 15].$$

est tel que $tri \cap a = tri \cap b$ et est normalisé du fait qu'il n'est pas vide.

Si pour $2n$ -pavé on prend maintenant

$$a = [3, 3] \times [1, 3] \times [1, 9] \times [2, 7] \times [5, 6] \times [6, 6] \times [5, 6] \times [5, 10] \times [3, 11] \times [9, 11] \times [9, 9] \times [1, 4] \times [1, 2] \times [2, 4] \times [4, 7] \times [7, 8] \times [5, 7] \times [4, 6] \times [5, 8] \times [8, 10] \times [7, 11] \times [9, 11],$$

on a

$$\begin{aligned} \uparrow(1, 1, 2, 4, 7, 5, 4, 5, 8, 7, 9) &= (1, 1, 2, 4, 7, 7, 7, 7, 8, 8, 9), \\ \downarrow(4, 2, 4, 7, 8, 7, 6, 8, 10, 11, 11) &= (2, 2, 4, 6, 6, 6, 6, 8, 10, 11, 11) \end{aligned}$$

et donc le pavé

$$b = [3, 3] \times [1, 3] \times [1, 9] \times [2, 7] \times [5, 6] \times [6, 6] \times [5, 6] \times [5, 10] \times [3, 11] \times [9, 11] \times [9, 9] \times [1, 2] \times [1, 2] \times [2, 4] \times [4, 6] \times [7, 6] \times [7, 6] \times [7, 6] \times [7, 8] \times [8, 10] \times [8, 11] \times [9, 11]$$

est tel que $tri \cap a = tri \cap b$. Du fait de l'occurrence de l'intervalle vide $[7, 6]$, on a $b = \emptyset$ et donc $tri \cap a = \emptyset$

Complexité 1 Bien entendu, étant donné un $2n$ -pavé a , le calcul du pavé vide ou normalisé b qui est tel que $tri \cap a = tri \cap b$ peut se faire en $\mathcal{O}(n)$ instructions élémentaires.

3.2 Ensembles d'indices pertinents

Soit a un $2n$ -pavé sur (\mathbf{D}, \preceq) . L'ensemble des *indices pertinents de a pour i* pris dans $1..n$ est

$$pert(a, i) = \{j \in (n+1)..(2n) \mid a_i \cap a_j \neq \emptyset\}$$

et l'ensemble des *indices pertinents de a pour j* pris dans $(n+1)..(2n)$ est

$$pert(a, j) = \{i \in 1..n \mid a_i \cap a_j \neq \emptyset\}$$

En remarquant que, pour des intervalles non vides, on a $a_i \cap a_j \neq \emptyset$ si et seulement si on a $\underline{a_i} \preceq \overline{a_j}$ et $\underline{a_j} \preceq \overline{a_i}$, on vérifie que :

Propriété 3 Si $a = a_1 \times \dots \times a_{2n}$ est un $2n$ -pavé normalisé et i un élément de $1..n$, l'ensemble d'indices $pert(a, i)$ est un intervalle de (\mathbf{N}, \leq) et, si cet intervalle n'est pas vide, on a

$$\begin{aligned} \min(pert(a, i)) &= \min\{j \in (n+1)..(2n) \mid \underline{a_i} \preceq \overline{a_j}\}, \\ \max(pert(a, i)) &= \max\{j \in (n+1)..(2n) \mid \underline{a_j} \preceq \overline{a_i}\}. \end{aligned}$$

3.3 Premières projections de $tri \cap a$

Les pavés normalisés ont une autre propriété essentielle :

Propriété 4 Si $a = a_1 \times \dots \times a_{2n}$ est un $2n$ -pavé normalisé et si Φ désigne l'ensemble des applications bijectives f de $1..n$ dans $(n+1)..(2n)$ telles que $a_1 \times \dots \times a_n \cap a_{f(1)} \times \dots \times a_{f(n)} \neq \emptyset$ alors, pour tout $i \in 1..n$, on a

$$\pi_i(tri \cap a) = \bigcup_{f \in \Phi} a_i \cap a_{f(i)}. \quad (2)$$

Preuve Soit e un élément de \mathbf{D} . D'après la définition de la relation globale tri , il va de soi que si e appartient au membre gauche de (2) il appartient aussi au membre droit de (2). Supposons que e appartienne au membre droit de (2) et montrons qu'il appartient au membre gauche de (2). Il existe donc une application bijective f de $1..n$ dans $(n+1)..(2n)$ telle que l'ensemble $b = a_1 \times \dots \times a_n \cap a_{f(1)} \times \dots \times a_{f(n)}$

ne soit pas vide et telle que $e \in a_i \cap a_{f(i)}$. Il existe alors un n -uplet $(d_1, \dots, d_n) \in b$ tel que $e = d_i$. Soit d le $2n$ -uplet (d_1, \dots, d_{2n}) obtenu en posant $(d_{f(1)}, \dots, d_{f(n)}) = (d_1, \dots, d_n)$. On a donc $d \in a$. Si (j, j') est un couple d'indices pris dans $(n+1)..(2n)$, tels que $j < j'$ et $d_{j'} \prec d_j$, du fait que a est normalisé, le $2n$ -uplet obtenu en échangeant d_j et $d_{j'}$ dans d appartient toujours à a . Par un nombre fini de tels échanges, on peut donc ranger dans l'ordre croissant les n derniers termes du $2n$ -uplet d , et ainsi obtenir un élément de l'ensemble $tri \cap a$. Du fait que l'on a toujours $e = d_i$, l'élément e appartient bien au membre gauche de (2). Ceci termine la preuve.

4 Pavé de type I

La propriété 4 va nous permettre, étant donné un $2n$ -pavé normalisé dit de type I, de calculer les plus petits éléments des n premières projections de $tri \cap a$ en $\mathcal{O}(n \log n)$ instructions élémentaires. Mais la transformation d'un pavé normalisé en un pavé de type I n'est pas évidente. Ainsi que nous le verrons par la suite, il faut d'abord passer à un pavé dit de type II, puis à un pavé dit de type III, puis décomposer ce pavé de type III en pavés de type I.

Définition 3 Un $2n$ -pavé de type I est un pavé normalisé de la forme $a_1 \times \dots \times a_{2n}$ pour lequel on a $a_1 \times \dots \times a_n \cap a_{n+1} \times \dots \times a_{2n} \neq \emptyset$ et $a_1 \times \dots \times a_{n-1} \cap a_{n+2} \times \dots \times a_{2n} \neq \emptyset$.

4.1 Calcul des plus petits éléments des n premières projections de $tri \cap a$

Théorème 1 Si $a = a_1 \times \dots \times a_{2n}$ est un $2n$ -pavé de type I alors, pour tout $i \in 1..n$, l'élément $\pi_i(tri \cap a)$ existe et est égal à $\underline{a_i \cap a_j}$, avec $j = \min(\text{pert}(a, i))$.

Preuve Soit $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé de type I, soit i un élément de $1..n$ et soit Φ l'ensemble des applications bijectives f de $1..n$ dans $(n+1)..(2n)$ telles que $a_1 \times \dots \times a_n \cap a_{f(1)} \times \dots \times a_{f(n)} \neq \emptyset$. Du fait que $a_i \cap a_{i+n} \neq \emptyset$, l'indice $j = \min(\text{pert}(a, i))$ et donc l'élément $\underline{a_i \cap a_j}$ existent.

Montrons que $\underline{a_i \cap a_j} \in \pi_i(tri \cap a)$. Le pavé a étant de type I, l'application de $1..n$ dans $(n+1)..(2n)$ définie par

$$i' \mapsto \begin{cases} i' + n, & \text{si } i' \notin (j-n)..i, \\ i' + n + 1, & \text{si } i' \in (j-n)..(i-1), \\ j, & \text{si } i' = i \end{cases}$$

appartient à Φ . D'après la propriété 4, on a alors $a_i \cap a_j \subseteq \pi_i(tri \cap a)$ et donc $\underline{a_i \cap a_j} \in \pi_i(tri \cap a)$.

Montrons que si $e \in \pi_i(tri \cap a)$ alors $\underline{a_i \cap a_j} \preceq e$. Si $e \in \pi_i(tri \cap a)$, d'après la définition de la relation tri , il existe $f \in \Phi$ telle que $e \in a_i \cap a_{f(i)}$ et donc il existe $j' \in (n+1)..(2n)$ tel que $a_i \cap a_{j'} \neq \emptyset$ et $\underline{a_i \cap a_{j'}} \preceq e$. Le pavé a étant normalisé, on a $\underline{a_i \cap a_j} \preceq e$. Ceci termine la preuve.

Exemple 3 Dans (\mathbf{R}, \leq) le $2n$ -pavé

$$a = [0, 13] \times [10, 11] \times [4, 16] \times [1, 3] \times [11, 11] \times [11, 15]$$

est de type I. D'après le théorème 1, on obtient

$$\begin{aligned} \pi_1(tri \cap a) &= [0, 13] \cap [1, 3] = 1, \\ \pi_2(tri \cap a) &= [10, 11] \cap [11, 11] = 11, \\ \pi_3(tri \cap a) &= [4, 16] \cap [11, 11] = 11. \end{aligned}$$

Complexité 2 Le calcul de l'ensemble des $\pi_i(tri \cap a)$, avec $i \in 1..n$, peut se faire en $\mathcal{O}(n \log n)$ instructions élémentaires. En effet, d'après la propriété 3, il s'agit d'associer à chaque $i \in 1..n$ l'indice $j_i \in (n+1)..(2n)$ tel que

$$j_i = \min \{j' \in (n+1)..(2n) \mid \underline{a_i} \preceq \overline{a_{j'}}\}.$$

On trie d'abord le n -uplet de couples $((1, a_1), \dots, (n, a_n))$ dans l'ordre croissant (pour \preceq) des $\underline{a_i}$. On obtient en $\mathcal{O}(n \log n)$ instructions élémentaires un n -uplet de la forme $((h_1, a_{h_1}), \dots, (h_n, a_{h_n}))$.

Le calcul des j_i se fait alors en $\mathcal{O}(n)$ instructions élémentaires au cours d'un parcours simultané de gauche à droite des n -uplets $((h_1, a_{h_1}), \dots, (h_n, a_{h_n}))$ et $((n+1, a_{n+1}), \dots, (2n, a_{2n}))$.

5 Pavé de type II

Nous étudions ici la façon de déterminer si, étant donné un $2n$ -pavé normalisé a , l'ensemble $\text{tri} \cap a$ est ou n'est pas vide et de calculer les plus grands éléments de ses n dernières projections. Pour simplifier, nous supposons que a est d'une forme particulière, dite de type II, à laquelle on se ramène facilement.

Définition 4 *Un $2n$ -pavé de type II est un pavé normalisé de la forme $a_1 \times \dots \times a_{2n}$ avec le n -uplet $(\overline{a_1}, \dots, \overline{a_n})$ croissant pour \preceq .*

5.1 Résultats préliminaires

Propriété 5 *Soient $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé et φ une bijection de $1..n$ dans $(n+1)..(2n)$. Les trois propriétés suivantes sont équivalentes :*

$$\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1))), \quad \text{pour tout } i \in 1..n, \quad (3)$$

$$\varphi^{-1}(j) = \min(\text{pert}(a, j) - \varphi^{-1}((n+1)..(j-1))), \quad \text{pour tout } j \in (n+1)..(2n), \quad (4)$$

$$\begin{aligned} a_i \cap a_{\varphi(i)} \neq \emptyset, \quad \text{pour tout } i \in 1..n, \quad \text{et} \\ a_i \cap a_{\varphi(i')} \neq \emptyset, \quad i < i', \quad \varphi(i) > \varphi(i'), \quad \text{pour aucun couple } (i, i') \text{ d'éléments de } 1..n. \end{aligned} \quad (5)$$

Bien entendu $\varphi(1..(i-1))$ désigne ici l'ensemble $\{\varphi(1), \dots, \varphi(i-1)\}$ et est vide lorsque $i = 1$. De même pour $\varphi^{-1}((n+1)..(j-1))$.

Preuve En tenant compte du fait que φ est une bijection, la propriété (3) s'écrit aussi :

$$\begin{aligned} \text{pour tout } i \in 1..n, \\ \varphi(i) \in \text{pert}(a, i) \quad \text{et,} \quad \text{pour tout } j \in \text{pert}(a, i) - \varphi(1..(i-1)), \quad \varphi(i) \leq j, \end{aligned}$$

ou encore, en introduisant l'entier $i' = \varphi^{-1}(j)$,

$$\begin{aligned} \text{pour tout } i \in 1..n, \\ \varphi(i) \in \text{pert}(a, i) \quad \text{et,} \quad \text{pour tout } i' \in i..n, \quad \varphi(i') \in \text{pert}(a, i) \implies \varphi(i) \leq \varphi(i'). \end{aligned}$$

ce qui n'est rien d'autre que la propriété (5). De même la propriété (4) s'écrit :

$$\begin{aligned} \text{pour tout } j \in (n+1)..(2n), \\ \varphi^{-1}(j) \in \text{pert}(a, j) \quad \text{et,} \quad \text{pour tout } i' \in \text{pert}(a, j) - \varphi^{-1}((n+1)..(j-1)), \quad \varphi^{-1}(j) \leq i', \end{aligned}$$

ou encore, en introduisant l'entier $i = \varphi^{-1}(j)$,

$$\begin{aligned} \text{pour tout } i \in 1..n, \\ i \in \text{pert}(a, \varphi(i)) \quad \text{et,} \quad \text{pour tout } i' \in \text{pert}(a, \varphi(i)), \quad \varphi(i') \in \varphi(i)..2n \implies i \leq i', \end{aligned}$$

ce qui n'est rien d'autre que la propriété (5). Ceci termine la preuve.

Lemme 1 *Si $a = a_1 \times \dots \times a_{2n}$ est un $2n$ -pavé de type II et s'il est possible de définir une application bijective φ de $1..n$ dans $(n+1)..(2n)$ telle que, pour tout $i \in 1..(n-1)$,*

$$\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1))), \quad (6)$$

alors le $2n$ -uplet $e = (e_1, \dots, e_{2n})$ avec, pour chaque $i \in 1..n$, $e_{\varphi(i)} = e_i$ et $e_i = \overline{a_i \cap a_{\varphi(i)}}$, appartient à l'ensemble $\text{tri} \cap a$.

Preuve Supposons que la propriété ne soit pas vraie et montrons que l'on aboutit à une contradiction. Par construction, on a $e \in a$. Il existe donc un couple (i, i') d'éléments de $1..n$ tel que $\varphi(i) < \varphi(i')$ et $e_{i'} \prec e_i$. Du fait que a est normalisé, on a $\overline{a_{\varphi(i)}} \preceq \overline{a_{\varphi(i'')}}$ et donc $\overline{a_{i'}} \prec \overline{a_i}$. Du fait que a est de type II, on a alors $i > i'$. D'après la propriété **3**, l'égalité (6) du lemme peut être remplacée par

$$a_i \cap a_{\varphi(i)} \neq \emptyset \quad \text{et} \quad \varphi(i) = \min\{j \in (n+1)..(2n) \mid \underline{a_i} \preceq \overline{a_j} \text{ et } j \notin \varphi(1..(i-1))\}.$$

On a $i > i'$ et $\varphi(i) < \varphi(i')$. Il s'ensuit que $\overline{a_{i'}} \prec \overline{a_{\varphi(i)}}$. Du fait que a est normalisé et que $\varphi(i) < \varphi(i')$ on a aussi $\overline{a_{\varphi(i)}} \preceq \overline{a_{\varphi(i')}}$. Par transitivité, on a donc $\overline{a_{i'}} \prec \overline{a_{\varphi(i')}}$, ce qui contredit le fait que $a_{i'} \cap a_{\varphi(i')} \neq \emptyset$. Ceci termine la preuve.

Lemme 2 Soit $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé de type II tel que l'ensemble $\text{tri} \cap a$ ait au moins un élément (d_1, \dots, d_{2n}) . Il est alors possible de définir une application bijective φ de $1..n$ dans $(n+1)..(2n)$ telle qu'à la fois

$$\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1))), \quad \text{pour tout } i \in 1..n, \quad (7)$$

$$d_{\varphi(i)} \preceq \overline{a_i \cap a_{\varphi(i)}}, \quad \text{pour tout } i \in 1..n. \quad (8)$$

Preuve D'après la propriété 5, la propriété (7) s'écrit comme la conjonction des deux propriétés :

$$a_i \cap a_{\varphi(i)} \neq \emptyset, \quad \text{pour tout élément } i \text{ de } 1..n, \quad (9)$$

$$a_i \cap a_{\varphi(i')} \neq \emptyset \text{ et } i < i' \text{ et } \varphi(i) > \varphi(i'), \quad \text{pour aucun couple } (i, i') \text{ d'éléments de } 1..n. \quad (10)$$

Du fait que $(d_1, \dots, d_{2n}) \in \text{tri} \cap a$, il existe une application bijective φ de $1..n$ dans $(n+1)..(2n)$ qui a les propriétés (9) et (8). Modifions autant de fois que possible cette application par la transformation suivante : choisir dans $1..n$ un couple (i, i') d'indices tel que

$$a_i \cap a_{\varphi(i')} \neq \emptyset \text{ et } i < i' \text{ et } \varphi(i) > \varphi(i') \quad (11)$$

et remplacer φ par l'application bijective φ' de $1..n$ dans $(n+1)..(2n)$ qui ne diffère de φ que par le fait que $(\varphi'(i), \varphi'(i')) = (\varphi(i'), \varphi(i))$. Du fait qu'après chaque transformation, l'entier $\sum_{i=1}^n i \times \varphi(i)$ croît strictement tout en restant inférieur ou égal à n^3 , ce processus s'arrête et on obtient une application finale φ qui a la propriété (10). Il reste à montrer que cette application finale a les propriétés (9) et (8). L'application initiale φ ayant ces propriétés, il suffit de montrer qu'elles sont préservées après chaque transformation de φ .

Soit donc φ une bijection de $1..n$ sur $(n+1)..(2n)$ qui a les propriétés (9) et (8), et soit (i, i') un couple d'éléments de $1..n$ qui a la propriété (11). Il faut montrer que

$$\varphi(i') \in \text{pert}(a, i) \text{ et } \varphi(i) \in \text{pert}(a, i') \text{ et } d_{\varphi(i')} \preceq \overline{a_i \cap a_{\varphi(i')}} \text{ et } d_{\varphi(i)} \preceq \overline{a_{i'} \cap a_{\varphi(i)}}$$

Par la propriété (11) on a déjà $\varphi(i') \in \text{pert}(a, i)$. Des propriétés (9) et (11), du fait que a est de type II et de la propriété 3, on déduit que

$$\min(\text{pert}(a, i')) \leq \varphi(i') < \varphi(i) \leq \max(\text{pert}(a, i)) \leq \max(\text{pert}(a, i')).$$

L'ensemble $\text{pert}(a, i')$ étant un intervalle, on a donc $\varphi(i) \in \text{pert}(a, i')$. Du fait que $(d_1, \dots, d_{2n}) \in \text{tri}$, du fait des propriétés (8) et (11) et du fait que a est de type II, on a

$$d_{\varphi(i')} \preceq d_{\varphi(i)} \preceq \overline{a_i} \preceq \overline{a_{i'}}.$$

De ceci et de la propriété (8) on déduit que $d_{\varphi(i')} \preceq \overline{a_i \cap a_{\varphi(i')}}$ et $d_{\varphi(i)} \preceq \overline{a_{i'} \cap a_{\varphi(i)}}$. Ceci termine la preuve.

5.2 Calcul des plus grands éléments des n dernières projections de $\text{tri} \cap a$

Les lemmes 1 et 2 mettent en évidence une bijection particulière φ , qui est construite en associant successivement, à chaque élément i de $1..n$, le plus petit indice pertinent encore disponible pour i . Le théorème qui suit montre l'importance de cette bijection φ qui, de par sa construction, est unique si elle existe.

Théorème 2 Soit $a = a_1 \times \dots \times a_{2n}$ un pavé de type II et, s'il est possible de la définir, soit φ une application bijective de $1..n$ dans $(n+1)..(2n)$ telle que, pour tout $i \in 1..n$,

$$\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1))). \quad (12)$$

Alors

- (1) il existe au plus une telle application φ ,
- (2) si l'ensemble $\text{tri} \cap a$ n'est pas vide, φ existe,
- (3) si φ existe, l'ensemble $\text{tri} \cap a$ n'est pas vide et pour chaque $j \in (n+1)..(2n)$, l'élément $\overline{\pi_j(\text{tri} \cap a)}$ existe et est égal à $\overline{a_j \cap a_{\varphi^{-1}(j)}}$.

Preuve Du fait que $\varphi(1) = \min(\text{pert}(a, 1))$ et que, pour tout $i \in 2..(n-1)$, la valeur de $\varphi(i)$ ne dépend que de celles de $\varphi(1), \dots, \varphi(i-1)$, il existe au plus une application φ .

Si l'ensemble $\text{tri} \cap a$ n'est pas vide, il a au moins un élément (d_1, \dots, d_{2n}) et, d'après le lemme 2, l'application φ existe.

Si l'application φ existe, d'après le lemme 1, le $2n$ -uplet (e_1, \dots, e_{2n}) , avec, pour chaque $i \in 1..n$, $e_i = e_{\varphi(i)} = \overline{a_i \cap a_{\varphi(i)}}$ appartient à $\text{tri} \cap a$. De plus, d'après le lemme 2, si (d_1, \dots, d_{2n}) est un élément de $\text{tri} \cap a$, pour chaque $i \in 1..n$, on a $d_{\varphi(i)} \preceq \overline{a_i \cap a_{\varphi(i)}}$, c'est-à-dire $d_{\varphi(i)} \preceq e_i$. L'élément $\pi_{\varphi(i)}(\text{tri} \cap a)$ existe donc et est égal à $\overline{a_i \cap a_{\varphi(i)}}$. En posant $j = \varphi(i)$, il s'ensuit que, pour tout $j \in (n+1)..(2n)$, l'élément $\pi_j(\text{tri} \cap a)$ existe et est égal à $\overline{a_j \cap a_{\varphi^{-1}(j)}}$. Ceci termine la preuve.

Exemple 4 Pour $n = 5$ soit le $2n$ -pavé sur (\mathbf{Z}, \leq)

$$a = [4, 6] \times [6, 10] \times [10, 11] \times [7, 13] \times [4, 16] \times [1, 3] \times [5, 9] \times [6, 9] \times [11, 15] \times [11, 15].$$

Ce pavé est bien de type II. On construit de gauche à droite la suite la plus longue $\varphi(1), \varphi(2), \dots, \varphi(m)$ vérifiant, pour chaque $i \in 1..m$:

$$\varphi(i) = \min \{j \in \text{pert}(a, i) \mid j \notin \varphi(1..(i-1))\}.$$

et donc en particulier

$$\varphi(1) = \min(\text{pert}(a, 1)).$$

On obtient

$$\varphi(1), \dots, \varphi(4) = 7, 8, 9, 10$$

On a donc $m = 4$. Du fait que $m < n$, l'application φ n'est pas définie sur $1..n$ et donc $\text{tri} \cap a = \emptyset$. Si, par contre, pour pavé on prend

$$a = [4, 6] \times [6, 10] \times [10, 11] \times [0, 13] \times [4, 16] \times [1, 3] \times [5, 9] \times [6, 9] \times [11, 15] \times [11, 15],$$

la même construction donne

$$\varphi(1), \dots, \varphi(5) = 7, 8, 9, 6, 10.$$

La suite étant de longueur n , on a construit ainsi l'application bijective φ qui respecte la condition (12). On sait donc que l'ensemble $r = \text{tri} \cap a$ n'est pas vide. On a alors

$$\varphi^{-1} = \{6 \mapsto 4, 7 \mapsto 1, 8 \mapsto 2, 9 \mapsto 3, 10 \mapsto 5\}$$

et

$$(\overline{\pi_6(r)}, \dots, \overline{\pi_{10}(r)}) = (3, 6, 9, 11, 15).$$

Complexité 3 Il est possible de déterminer si l'application φ existe et, dans l'affirmative, de la calculer, en $\mathcal{O}(n \log n)$ instructions élémentaires. En fait, il faut construire la plus longue suite $\varphi(1), \varphi(2), \dots, \varphi(m)$ qui, pour chaque $i \in 1..m$, satisfait à la condition (12). Comme dans la démonstration du lemme 1, la condition (12) peut être remplacée par

$$a_i \cap a_{\varphi(i)} \neq \emptyset \text{ et } \varphi(i) = \min \{j \in (n+1)..(2n) \mid \underline{a_i} \preceq \underline{a_j} \text{ et } j \notin \varphi(1..(i-1))\}.$$

Etant donné l'ensemble $c = (n+1)..(2n)$ et sachant que la suite $(\overline{a_{n+1}}, \overline{a_{n+2}}, \dots, \overline{a_{2n}})$ est croissante pour \preceq , il faut donc successivement donner les valeurs $1, 2, \dots, n$ à i et chaque fois, en $\mathcal{O}(\log n)$ instructions élémentaires,

- déterminer s'il existe un élément j de c tel que $\overline{a_i} \preceq \underline{a_j}$ et, si c'est le cas, calculer le plus petit élément $\varphi(i)$ de c ,
- vérifier si l'ensemble $a_i \cap a_{\varphi(i)}$ n'est pas vide,
- enlever $\varphi(i)$ de c .

Ceci est possible en représentant l'ensemble initial c par un arbre binaire équilibré de n nœuds p dont les étiquettes $\text{étiq}(p)$ sont les éléments de c . La relation $<$ est codée en imposant que, pour tout couple (p, q) de nœuds, si q est accessible à partir du fils gauche de p , alors $\text{étiq}(q) < \text{étiq}(p)$ et si q est accessible à partir du fils droit de p , alors $\text{étiq}(p) < \text{étiq}(q)$.

En complétant raisonnablement le jeu d'opérations élémentaires de notre machine fictive, on sait que la construction d'un tel arbre peut se faire en $\mathcal{O}(n)$ instructions élémentaires, et que la recherche ou la suppression d'un nœud d'un tel arbre peut se faire en $\mathcal{O}(\log n)$ instructions élémentaires. Bien entendu la vérification de $a_i \cap a_{\varphi(i)} \neq \emptyset$ se fait en un nombre constant d'opérations élémentaires.

6 Pavé de type III

L'application φ permet donc de calculer les plus grands éléments des n dernières projections de $\text{tri} \cap a$ pour un $2n$ -pavé a . Elle permet aussi de transformer un pavé de type II en un pavé dit de type III, qu'il est alors possible de décomposer en $2m$ -pavés b de type I. Rappelons que, dans ce cas, nous savons calculer les plus petits éléments des m premières projections de $\text{tri} \cap b$. Avant de pouvoir passer à la décomposition proprement dite d'un $2n$ -pavé a de type III, nous établirons dans cette section deux résultats : l'un concernant le tri des éléments de $(n+1)..(2n)$ dans un ordre ad hoc \leq_a , l'autre concernant l'utilisation de ce tri pour partitionner l'ensemble $1..2n$ en sous-ensembles d'indices dits autonomes.

Définition 5 Un $2n$ -pavé de type III est un $2n$ -pavé normalisé de la forme $a_1 \times \cdots \times a_{2n}$, pour lequel $a_1 \times \cdots \times a_n \cap a_{n+1} \times \cdots \times a_{2n} \neq \emptyset$ et pour lequel il n'existe aucun couple (j, j') d'éléments de $(n+1)..(2n)$ tels que $j < j'$ et $\overline{a_{j'-n}} < \overline{a_{j-n}}$ et $a_{j'-n} \cap a_j \neq \emptyset$.

6.1 Passage d'un pavé de type II à un pavé de type III

Propriété 6 Soit $a = a_1 \times \cdots \times a_{2n}$ un pavé de type II pour lequel il existe une application bijective φ de $1..n$ dans $(n+1)..(2n)$ telle que, pour chaque $i \in 1..n$, on ait

$$\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1))).$$

Le pavé

$$a_{\varphi^{-1}(n+1)} \times \cdots \times a_{\varphi^{-1}(2n)} \times [\overline{a_{n+1}}, \overline{a_{n+1} \cap a_{\varphi^{-1}(n+1)}}] \times \cdots \times [\overline{a_{2n}}, \overline{a_{2n} \cap a_{\varphi^{-1}(2n)}}] \quad (13)$$

est un pavé de type III.

Preuve Du lemme 1 on déduit que le pavé (13) est normalisé. De la définition de φ , on déduit alors qu'il est de type III. Ceci termine la preuve.

Exemple 5 Reprenons le $2n$ -pavé de type II de l'exemple 4, avec

$$\begin{aligned} a &= [4, 6] \times [6, 10] \times [10, 11] \times [0, 13] \times [4, 16] \times [1, 3] \times [5, 9] \times [6, 9] \times [11, 15] \times [11, 15], \\ \varphi^{-1} &= \{6 \mapsto 4, 7 \mapsto 1, 8 \mapsto 2, 9 \mapsto 3, 10 \mapsto 5\}, \\ (\overline{\pi_6(r)}, \dots, \overline{\pi_{10}(r)}) &= (\overline{a_6 \cap a_{\varphi^{-1}(6)}}, \dots, \overline{a_{10} \cap a_{\varphi^{-1}(10)}}) = (3, 6, 9, 11, 15). \end{aligned}$$

Le pavé

$$b = [0, 13] \times [4, 6] \times [6, 10] \times [10, 11] \times [4, 16] \times [1, 3] \times [5, 6] \times [6, 9] \times [11, 11] \times [11, 15]$$

est donc de type III.

6.2 Tri des n derniers indices dans l'ordre croissant de \leq_a

Soit $a = a_1 \times \cdots \times a_{2n}$ un $2n$ -pavé normalisé. On définit les relations binaires \triangleright_a et \leq_a entre les éléments de $(n+1)..(2n)$ par les équivalences suivantes

$$i \triangleright_a j \iff a_{i-n} \cap a_j \neq \emptyset$$

$$i \leq_a j \iff \begin{pmatrix} \text{soit } i \leq j & \text{et} & \text{il existe } k_1, \dots, k_h \text{ avec } i = k_1, h \geq 1, k_h = j, \\ & & k_i < k_{i+1}, k_i \triangleright_a k_{i+1}, \text{ pour chaque } i \in 1..(h-1), \\ \text{soit } i > j & \text{et} & \text{on n'a pas } j \leq_a i. \end{pmatrix}$$

On établit tout d'abord deux propriétés simples :

Propriété 7 Soient a un $2n$ -pavé de type III et i, j, k des éléments de $(n+1)..(2n)$ tels que $i < j$ et $j < k$. Si $j \triangleright_a i$ et $i \triangleright_a k$ alors $j \triangleright_a k$.

Preuve On remarque tout d'abord que, du fait que a est de type III, on a $i \in \text{pert}(a, i-n)$ et $j \in \text{pert}(a, j-n)$ et donc qu'aucun de ces ensembles d'indices pertinents n'est vide.

Si $j \triangleright_a i$ alors $a_{j-n} \cap a_i \neq \emptyset$. Du fait que $i < j$ et que a est de type III, on déduit que $\overline{a_{i-n}} \preceq \overline{a_{j-n}}$ et, d'après la propriété **3**, que $\max(\text{pert}(a, i-n)) \leq \max(\text{pert}(a, j-n))$. Si de plus, on a $i \triangleright_a k$, on a aussi $k \in \text{pert}(a, i-n)$ et donc

$$\min(\text{pert}(a, j-n)) \leq j < k \leq \max(\text{pert}(a, i-n)) \leq \max(\text{pert}(a, j-n))$$

Il s'ensuit que $k \in \text{pert}(a, j-n)$, c'est-à-dire $j \triangleright_a k$. Ceci termine la preuve.

Propriété 8 Soient a un $2n$ -pavé normalisé et i, j, k des éléments de $(n+1)..(2n)$ tels que $i < j$ et $j < k$. Si $i \leq_a k$, alors $i \leq_a j$.

Preuve Si $i < j < k$ et $i \leq_a k$, il existe un m -uplet (k_1, \dots, k_m) d'entiers croissant pour $<$ tel que $i = k_1$, $k = k_m$ et $k_{h+1} \in \text{pert}(a, k_h - n)$ pour chaque $h \in 1..(m-1)$. Il existe donc $h \in 1..(m-1)$ tel que $j \in k_h..k_{h+1}$. Du fait que $k_h \in \text{pert}(a, k_h - n)$, que $k_{h+1} \in \text{pert}(a, k_h - n)$ et que $\text{pert}(a, k_h - n)$ est un intervalle, on a $j \in \text{pert}(a, k_h - n)$ et donc $i \leq_a j$. Ceci termine la preuve.

De cette dernière propriété on conclut que :

Propriété 9 Si a est un $2n$ -pavé normalisé, la relation \leq_a est une relation d'ordre total sur l'ensemble $(n+1)..(2n)$.

Preuve Il va de soi que si i, j sont des éléments de $(n+1)..(2n)$ on a $i \leq_a j$ ou $j \leq_a i$. Il va aussi de soi que la relation \leq_a est antisymétrique et réflexive. Reste à montrer qu'elle est transitive. Soient i, j, k des éléments de $(n+1)..(2n)$ tels que $i \leq_a j$ et $j \leq_a k$. Six cas se présentent :

Si $i \leq j$ et $k < i$, on ne peut avoir $k \leq_a i$ car, d'après le premier point de la définition de \leq_a , on aurait $k \leq_a j$, ce qui contredirait $j \leq_a k$. Donc $i \leq_a k$.

Si $i \leq j$ et $k \in i..j$, d'après la propriété **8** on a $i \leq_a k$.

Si $i \leq j$ et $j < k$, d'après le premier point de la définition de \leq_a on a $i \leq_a k$.

Si $j < i$ et $k < j$, on ne peut avoir $k \leq_a i$ car, d'après la propriété **8** on aurait $k \leq_a j$, ce qui contredirait $j \leq_a k$. Donc $i \leq_a k$.

Si $j < i$ et $k \in j..i$, on ne peut avoir $k \leq_a i$ car, d'après le premier point de la définition de \leq_a , on aurait $j \leq_a i$, ce qui contredirait $i \leq_a j$. Donc $i \leq_a k$.

Si $j < i$ et $i < k$, on a $j \leq_a k$ et, d'après la propriété **8**, $j \leq_a i$ ce qui contredit $i \leq_a j$. Ce cas ne peut donc se présenter. Ceci termine la preuve.

On peut maintenant passer au résultat principal de cette sous-section.

Théorème 3 Soit a un $2n$ -pavé de type III. Si, sur le triplet $(\varepsilon, (n+1, \dots, 2n), \varepsilon)$, on applique tant que cela est possible l'une des deux transformations décrites ci-dessous, on obtient en $2n$ applications un triplet de la forme $(\varepsilon, \varepsilon, w)$, où w est le n -uplet $(n+1, \dots, 2n)$ trié dans l'ordre croissant de \leq_a .

Transformations :

$$1. \quad (u, (k) \cdot v, w) \longrightarrow (u \cdot (k), v, w), \quad \text{si } u = \varepsilon \text{ ou à la fois } u \neq \varepsilon \text{ et } u \triangleright_a k$$

$$2. \quad (u \cdot (k), v, w) \longrightarrow (u, v, (k) \cdot w), \quad \text{si } v = \varepsilon \text{ ou à la fois } v \neq \varepsilon \text{ et } k \not\triangleright_a v.$$

où k est un entier et u, v des suites éventuellement vides d'entiers et où \triangleright désigne le dernier terme de u et $\dot{\triangleright}$ le premier terme de v , quand u et v ne sont pas vides.

Preuve Soit (u', v', w') un triplet courant obtenu par transformations du triplet initial. Si u' et v' sont vides, aucune transformation ne s'applique sur le triplet et, si l'une au moins des suites u', v' n'est pas vide, une et une seule transformation s'applique. Si à cela on ajoute le fait que la transformation 1 fait passer un élément de la suite v' dans la suite u' et que la transformation 2 fait passer un élément de la suite u' dans la suite w' , on conclut que l'on aboutit d'une façon déterministe et en $2n$ transformations à une configuration finale de la forme $(\varepsilon, \varepsilon, w')$, où w' est la suite $(n+1, n+2, \dots, 2n)$ arrangée dans un

ordre différent. Pour montrer que la suite w' du triplet final est croissante pour \leq_a nous montrerons une propriété plus forte, à savoir que le triplet courant (u', v', w') est tel que

- (1) la suite $u' \cdot v'$ est croissante pour $<$,
- (2) pour chaque $i \in \text{set}(v')$ et chaque $j \in \text{set}(w')$ on a $i > j$,
- (3) pour chaque $i \in \text{set}(v')$ et chaque $j \in \text{set}(w')$ on a $i \leq_a j$,
- (4) la suite $u' \cdot w'$ est croissante pour \leq_a .

Les propriétés (1),(2),(3) et (4) sont vraies pour le triplet initial. Il suffit donc de montrer qu'elles sont préservées après chaque transformation du triplet courant (u', v', w') . Le seul point non évident est de montrer que la propriété (3) est préservée lorsque le triplet (u', v', w') est de la forme

$$(u \cdot (k), v, w), \text{ avec } v \neq \varepsilon \text{ et qu'on a pas } k \triangleright_a v$$

et qu'on applique la transformation 2 pour obtenir

$$(u, v, (k) \cdot w)$$

Supposons que dans ce cas la propriété (3) ne soit pas préservée. Il existe donc $l \in \text{set}(v)$ tel que $k \leq_a l$. Du fait que la suite $u \cdot (k) \cdot v$ est croissante pour $<$ et d'après la propriété **8**, on a $k \leq_a v$. Du fait que, pour tout $i \in \text{set}(v)$ et tout $j \in \text{set}(w)$, on a $i \leq_a j$, il n'existe pas $k' \in (n+1)..(2n)$ tel que $k < k' < v$ et $k \leq_a k' \leq_a v$. On conclut que $k \triangleright_a v$ et on aboutit à une contradiction. La propriété (3) est donc aussi préservée dans ce cas. Ceci termine la preuve.

Exemple 6 Dans (\mathbf{Z}, \leq) le $2n$ -pavé

$$a = [0, 13] \times [4, 6] \times [6, 10] \times [10, 11] \times [4, 16] \times [1, 3] \times [5, 6] \times [6, 9] \times [11, 11] \times [11, 15].$$

avec $n = 5$ est bien de type III. Le tri du 5-uplet $(6, 7, 8, 9, 10)$ dans l'ordre croissant pour \leq_a se fait par les $2n$ transformations

$$\begin{aligned} &(\varepsilon, (6, 7, 8, 9, 10), \varepsilon) \\ &\longrightarrow ((6), (7, 8, 9, 10), \varepsilon) \\ &\longrightarrow ((6, 7), (8, 9, 10), \varepsilon) \\ &\longrightarrow ((6, 7, 8), (9, 10), \varepsilon) \\ &\longrightarrow ((6, 7), (9, 10), (8)) \\ &\longrightarrow ((6), (9, 10), (7, 8)) \\ &\longrightarrow ((6, 9), (10), (7, 8)) \\ &\longrightarrow ((6, 9, 10), \varepsilon, (7, 8)) \\ &\longrightarrow ((6, 9), \varepsilon, (10, 7, 8)) \\ &\longrightarrow ((6), \varepsilon, (9, 10, 7, 8)) \\ &\longrightarrow (\varepsilon, \varepsilon, (6, 9, 10, 7, 8)) \end{aligned}$$

et on obtient le 5-uplet $(6, 9, 10, 7, 8)$.

Complexité 4 Etant donnée la nature des transformations du théorème **3** et étant donné le fait que $2n$ transformations sont appliquées, le tri du n -uplet $(n+1, \dots, 2n)$, dans l'ordre croissant de \leq_a , se fait en $\mathcal{O}(n)$ instructions élémentaires.

6.3 Sous-ensembles autonomes d'indices

Soit c un ensemble d'entiers et n un entier positif. On dit que c est n -équilibré si c est un sous-ensemble de $1..2n$ et si les ensembles $c \cap 1..n$ et $c \cap (n+1)..(2n)$ ont le même nombre d'éléments. Si a est un $2n$ -pavé sur (\mathbf{D}, \preceq) , on dit que c est a -autonome si c est un sous-ensemble n -équilibré de $1..2n$ et si, pour toute application bijective f de $1..n$ sur $(n+1)..(2n)$, on a l'implication :

$$a_1 \times \dots \times a_n \cap a_{f(1)} \times \dots \times a_{f(n)} \neq \emptyset \implies f(c \cap 1..n) = c \cap (n+1)..(2n),$$

avec bien entendu $a_i = \pi_i(a)$.

On remarque que, si c et d sont a -autonomes, les sous-ensembles $c \cap d$, $c \cup d$ et $c - d$ le sont aussi.

Propriété 10 Soit $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé non vide sur (\mathbf{D}, \preceq) et soit $P = \{c_1, \dots, c_k\}$ une partition de $1..2n$ en k classes n -équilibrées. Une condition suffisante pour que les classes de P soient a -autonomes est que, pour tout $i \in 1..n$ et $j \in (n+1)..(2n)$, on ait l'implication

$$\text{rang}(i, P) > \text{rang}(j, P) \implies a_i \cap a_j = \emptyset, \tag{14}$$

où d'une façon générale, $\text{rang}(i, P)$ désigne l'indice h tel que $i \in c_h$.

Preuve Supposons qu'au moins une classe de P ne soit pas a -autonome et montrons qu'il existe des indices $i \in 1..n$ et $j \in (n+1)..(2n)$ qui ne vérifient pas la condition (14). Soit h le plus grand indice tel que la classe c_h de P ne soit pas a -autonome. Il existe donc une application bijective f de $1..n$ dans $(n+1)..(2n)$ telle que $a_1 \times \dots \times a_n \cap a_{f(1)} \times \dots \times a_{f(n)} \neq \emptyset$ et $f(c_h \cap 1..n) \neq c_h \cap (n+1)..(2n)$. Du fait que c_h est n -équilibré, il existe alors un indice $i \in c_h$ tel que $f(i) \notin c_h$. La classe de $f(i)$ dans P n'est donc pas a -autonome et est distincte de c_h . Du fait que h est le plus grand indice tel que c_h soit une classe non a -autonome de P , on a $h > \text{rang}(f(i), P)$. En remarquant que $\text{rang}(i, P) = h$ et en posant $j = f(i)$, on a $i \in 1..n$ et $j \in (n+1)..(2n)$ avec $\text{rang}(i, P) > \text{rang}(j, P)$ et $a_i \cap a_j \neq \emptyset$. Ceci termine la preuve.

On peut maintenant passer au résultat principal de cette sous-section.

Lemme 3 Soit a un $2n$ -pavé de type III, soit w le $2n$ -uplet $(n+1, \dots, 2n)$ trié dans l'ordre croissant de \leq_a et soient w_1, \dots, w_k les n_i -uplets, croissants pour $<$, de longueurs maximales et tels que $w = (w_1 - - - w_k)$. L'ensemble $P = \{c_1, \dots, c_k\}$ avec $c_i = \text{set}(w_i) \cup \{j-n \mid j \in \text{set}(w_i)\}$ est une partition de $1..2n$ en sous-ensembles a -autonomes.

Preuve Soient i, j des éléments quelconques de $1..k$, avec $j < i$. D'après la propriété (10) il suffit de montrer que pour aucun $i' \in \text{set}(w_i)$ et aucun $j' \in \text{set}(w_j)$ on n'a $i' \triangleright_a j'$. Si $i' < j'$, du fait que $j' \leq_a i'$ on n'a pas $i' \triangleright_a j'$. Si $j' < i'$, soit \hat{w}_j le dernier terme de w_j et w_{j+1} le premier terme de w_{j+1} . On a alors $i' < \hat{w}_j$. Sinon, de $w_{j+1} < \hat{w}_j < i'$ et $w_{j+1} \leq_a i'$, par la propriété 8, on déduirait $w_{j+1} \leq_a \hat{w}_j$, ce qui est faux. Donc $i' \in (j'+1)..(\hat{w}_j-1)$.

Soit j'' le terme qui précède i' et k le terme qui suit i' dans w_j . Si on avait $i' \triangleright_a j'$, d'après la propriété 3, on aurait $i' \triangleright_a j''$ et, du fait que $j'' \triangleright_a k$, par la propriété 7, on aurait $i' \triangleright_a k$, ce qui contredirait le fait que $k \leq_a i'$ et $i' < k$. On n'a donc pas $i' \triangleright_a j'$. Ceci termine la preuve.

Exemple 7 Pour le $2n$ -pavé

$$a = [0, 13] \times [4, 6] \times [6, 10] \times [10, 11] \times [4, 16] \times [1, 3] \times [5, 6] \times [6, 9] \times [11, 11] \times [11, 15].$$

de type III de l'exemple précédent, $w = (6, 9, 10, 7, 8)$. Pour partition de $1..2n$ en sous-ensembles autonomes, on obtient donc

$$P = \{\{6, 9, 10, 1, 4, 5\}, \{7, 8, 2, 3\}\}.$$

7 Décomposition d'un pavé de type III en pavés de type I

Nous pouvons maintenant formuler un théorème pour décomposer un pavé de type III en pavés de type I. L'énoncé du théorème se fera en utilisant un formalisme de contraintes que nous introduirons préalablement. Ce même formalisme sera utilisé pour décrire l'algorithme du calcul de $\text{apx}(tri \cap a)$ dans la section suivante.

7.1 Contrainte de base et contrainte composée

On se donne un ensemble infini \mathbf{V} de variables. On appelle *contrainte de base* une formule p de la forme

$$(x_1, \dots, x_m) \in r \wedge x_1 \in a_1 \wedge \dots \wedge x_m \in a_m \quad (15)$$

où les x_i sont des éléments distincts de \mathbf{V} , où les a_i sont des intervalles de (\mathbf{D}, \preceq) et où r est une relation globale, c'est-à-dire un sous ensemble de \mathbf{D}^* . On note *pavé*(p) le m -pavé $a_1 \times \dots \times a_m$. On appelle *contrainte composée* une formule de la forme

$$p_1 \wedge \dots \wedge p_k, \quad (16)$$

où les p_i sont des contraintes de base n'ayant aucune variable commune. On ne distingue pas deux contraintes composées qui peuvent être rendues égales en changeant l'association ou en commutant

les opérandes du connecteur \wedge . On désigne par $\text{var}(q)$ l'ensemble des éléments de \mathbf{V} qui apparaissent dans la contrainte composée q . Si $x \in \text{var}(q)$, on désigne par $\text{dom}(x, q)$ l'intervalle b de \mathbf{D} tel que la formule $x \in b$ figure dans q .

Bien entendu, une *solution* σ d'une contrainte de base de la forme (15) est une application de \mathbf{V} dans \mathbf{D} telle que $(\sigma(x_1), \dots, \sigma(x_m))$ appartienne à r et telle que chaque $\sigma(x_i)$ appartienne au a_i correspondant. Une *solution* d'une contrainte composée q de la forme (16) est une application σ de \mathbf{V} dans \mathbf{D} qui est solution de chaque contrainte de base p_i qui la compose. On désigne par $\text{sol}(q)$ l'ensemble des solutions de q . Deux contraintes composées q_1 et q_2 sont *équivalentes* et on écrit $q_1 \equiv q_2$, ssi $\text{sol}(q_1) = \text{sol}(q_2)$.

On remarque que les solutions d'une contrainte de base de la forme (15) sont les applications σ de \mathbf{V} dans \mathbf{D} qui sont telles que le m -uplet $(\sigma(x_1), \dots, \sigma(x_m))$ appartienne à l'ensemble $r \cap \text{pavé}(p)$. Si s est un ensemble d'applications de \mathbf{V} dans \mathbf{D} , la *projection de s sur la variable x* est le sous-ensemble de \mathbf{D}

$$\pi_x(s) = \{e \in \mathbf{D} \mid \text{il existe } \sigma \in s \text{ avec } e = \sigma(x)\}$$

Les contraintes de base d'une contrainte composée étant deux à deux sans variable commune, on a :

Propriété 11 *Soit q une contrainte composée dont toutes les contraintes de base p sont telles que $\text{sol}(p) \neq \emptyset$. Si $x \in \text{var}(q)$, alors $\pi_x(\text{sol}(q)) = \pi_x(\text{sol}(p_x))$, où p_x est la contrainte de base de q dans laquelle x apparaît.*

7.2 Décomposition d'un pavé

Théorème 4 *Soit $a = a_1 \times \dots \times a_{2n}$ un $2n$ -pavé de type III et soit $w = w_1 \cdot \dots \cdot w_k$ le n -uplet $(n+1, \dots, 2n)$ trié dans l'ordre croissant de \leq_a , où les w_i sont des suites non vides croissantes pour $<$ et de longueurs n_i maximales. Pour chaque $i \in 1..k$ on introduit le $2n_i$ -uplet*

$$(h_{(i,1)}, \dots, h_{(i,2n_i)}), \quad \text{avec} \quad (n+h_{(i,1)}, \dots, n+h_{(i,n_i)}) = (h_{(i,n_i+1)}, \dots, h_{(i,2n_i)}) = w_i.$$

Chaque $2n_i$ -pavé $b_i = a_{h_{(i,1)}} \times \dots \times a_{h_{(i,2n_i)}}$ est de type I et on a l'équivalence de contraintes

$$\left[\begin{array}{l} (x_1, \dots, x_{2n}) \in \text{tri} \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_{2n} \in a_{2n} \end{array} \right] \equiv \left[\begin{array}{l} (x_{h_{(1,1)}}, \dots, x_{h_{(1,2n_1)}}) \in \text{tri} \\ \wedge x_{h_{(1,1)}} \in a_{h_{(1,1)}} \\ \dots \\ \wedge x_{h_{(1,2n_1)}} \in a_{h_{(1,2n_1)}} \end{array} \right] \wedge \dots \wedge \left[\begin{array}{l} (x_{h_{(k,1)}}, \dots, x_{h_{(k,2n_k)}}) \in \text{tri} \\ \wedge x_{h_{(k,1)}} \in a_{h_{(k,1)}} \\ \dots \\ \wedge x_{h_{(k,2n_k)}} \in a_{h_{(k,2n_k)}} \end{array} \right] \quad (17)$$

où les x_i sont des éléments distincts appartenant à \mathbf{V} .

Preuve Soit i un élément de $1..k$. Du fait que a est de type III, le pavé b_i est normalisé et on a $a_{h_{(i,1)}} \times \dots \times a_{h_{(i,n_i)}} \cap a_{h_{(i,n_i+1)}} \times \dots \times a_{h_{(i,2n_i)}} \neq \emptyset$. Du fait que w est le n -uplet $(n+1, \dots, 2n)$ trié dans l'ordre croissant de \leq_a , pour aucun $j \in (n_i+1)..(2n_i-1)$ il n'existe $l \in (n+1)..(2n)$ tel que $h_{(i,j)} \leq_a l \leq_a h_{(i,j+1)}$ et $h_{(i,j)} < l < h_{(i,j+1)}$. Il s'ensuit que, pour tout $j \in (n_i+1)..(2n_i-1)$, on a $h_{(i,j)} \triangleright_a h_{(i,j+1)}$, soit $a_{h_{(i,j)}} \supset_a a_{h_{(i,j+1)}} \neq \emptyset$. Donc on a aussi $a_{h_{(i,1)}} \times \dots \times a_{h_{(i,n_i-1)}} \cap a_{h_{(i,n_i+2)}} \times a_{h_{(i,2n_i)}} \neq \emptyset$ et le pavé b_i est bien de type I.

Pour démontrer l'équivalence (17), remarquons tout d'abord que, d'après le lemme 3, l'ensemble $P = \{c_1, \dots, c_k\}$ avec $c_i = \text{set}(w_i) \cup \{j - n \mid j \in \text{set}(w_i)\}$ est une partition de $1..2n$ en sous-ensembles a -autonomes. Si j est un élément quelconque de $(n+1)..(2n)$, désignons par *classe* (j, P) l'ensemble c_i tel que $j \in c_i$.

Soit σ une application de \mathbf{V} dans \mathbf{D} . Posons $(d_1, \dots, d_{2n}) = (\sigma(x_1), \dots, \sigma(x_{2n}))$. Il est clair que, si σ est solution de la première contrainte de (17), alors σ est solution de la deuxième. Supposons que σ soit solution de la deuxième et pas de la première et montrons que l'on aboutit à une contradiction. Du fait que σ est solution de la deuxième contrainte, pour chaque $i \in 1..(2n)$, on a $d_i \in a_i$, il existe une application bijective f de $1..n$ dans $(n+1)..(2n)$ telle que $(d_1, \dots, d_m) = (d_{f(1)}, \dots, d_{f(m)})$ et, pour tout couple (j, j') d'éléments de $(n+1)..(2n)$ tel que $\text{classe}(j, P) = \text{classe}(j', P)$ et $j < j'$, on a $d_j \preceq d_{j'}$. Du fait que σ n'est pas solution de la première contrainte, il existe un couple (j, j') d'éléments de $(n+1)..(2n)$ avec $\text{classe}(j, P) \neq \text{classe}(j', P)$ et $j < j'$ et $d_{j'} \preceq d_j$. Du fait que $d_j \in a_j$ et $d_{j'} \in a_{j'}$ et du fait que a est normalisé on a $d_j \in a_{j'}$ et $d_{j'} \in a_j$ et donc le $2n$ -uplet obtenu en échangeant dans (d_1, \dots, d_{2n}) le terme d_j avec le terme $d_{j'}$ appartient à a . Soit (i, i') le couple $(f^{-1}(j), f^{-1}(j'))$. L'application g qui ne diffère de

f que par le fait que $(g(i), g(i')) = (f(i'), f(i))$ est alors telle que $a_1 \times \dots \times a_n \cap a_{g(1)} \times \dots \times a_{g(n)} \neq \emptyset$ et $classe(i, P) \neq classe(g(i), P)$, ce qui contredit le fait que les classes de P soient a -autonomes. Ceci termine la preuve.

Exemple 8 Reprenons le $2n$ -pavé de type III de l'exemple 6.

$$a = a_1 \times \dots \times a_{2n} = [0, 13] \times [4, 6] \times [6, 10] \times [10, 11] \times [4, 16] \times [1, 3] \times [5, 6] \times [6, 9] \times [11, 11] \times [11, 15].$$

avec $n = 5$ et $w = (6, 9, 10, 7, 8)$. On a $k = 2$ et $w = w_1 \cdot w_2$ avec $w_1 = (6, 9, 10)$ et $w_2 = (7, 8)$. On obtient

$$\begin{aligned} (h_{(1,1)}, \dots, h_{(1,6)}) &= (1, 4, 5, 6, 9, 10), \\ (h_{(2,1)}, \dots, h_{(2,4)}) &= (2, 3, 7, 8). \end{aligned}$$

On a donc l'équivalence de contraintes

$$\left[\begin{array}{l} (x_1, \dots, x_{10}) \in tri \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_{10} \in a_{10} \end{array} \right] \equiv \left[\begin{array}{l} (x_1, x_4, x_5, x_6, x_9, x_{10}) \in tri \\ \wedge x_1 \in a_1 \\ \wedge x_4 \in a_4 \\ \wedge x_5 \in a_5 \\ \wedge x_6 \in a_6 \\ \wedge x_9 \in a_9 \\ \wedge x_{10} \in a_{10} \end{array} \right] \wedge \left[\begin{array}{l} (x_2, x_3, x_7, x_8) \in tri \\ \wedge x_2 \in a_2 \\ \wedge x_3 \in a_3 \\ \wedge x_7 \in a_7 \\ \wedge x_8 \in a_8 \end{array} \right].$$

De plus, les pavés $a_1 \times a_4 \times a_5 \times a_6 \times a_9 \times a_{10}$ et $a_2 \times a_3 \times a_7 \times a_8$ sont de type I.

8 Algorithmes

Etant donné un $2m$ -pavé b , nous disposons maintenant d'un ensemble de résultats pour décider si l'ensemble $tri \cap b$ n'est pas vide et, dans l'affirmative, calculer les plus petits éléments de ses m premières projections et les plus grands éléments de ses m dernières projections. Nous introduirons un principe de dualité pour calculer symétriquement les extrémités manquantes des projections de $tri \cap b$. Nous passerons ensuite aux détails d'un algorithme pour calculer en $\mathcal{O}(m \log m)$ instructions élémentaires un $2m$ -pavé égal à $apx(tri \cap b)$. Remarquons que l'existence d'un tel algorithme montre que l'ensemble $apx(tri \cap b)$ est toujours défini.

8.1 Principe de dualité

Généralisons nos notations. Soit ρ une relation d'ordre total quelconque sur \mathbf{D} , soit a un sous-ensemble de \mathbf{D} , soit d un élément de \mathbf{D}^n et e, e' des éléments de \mathbf{D} . Définissons la relation globale tri^ρ , les éléments $\underline{a}_\rho, \bar{a}^\rho, \downarrow_\rho d, \uparrow^\rho d$ de \mathbf{D} et le sous-ensemble $[e, e']^\rho$ de \mathbf{D} de la même façon que $tri, \underline{a}, \bar{a}, \downarrow d, \uparrow d, [e, e']$, mais en remplaçant \preceq par ρ . De la même façon introduisons les notions de pavé ρ -normalisé, de type ρ -I, de type ρ -II et de type ρ -III.

Désignons par ρ^T la relation définie par $e\rho^T e'$ ssi $e'\rho e$. L'ensemble ordonné (\mathbf{D}, ρ^T) sera dit *dual* de (\mathbf{D}, ρ) . On constate que $(\rho^T)^T = \rho$ et que :

Propriété 12 Soit a un sous-ensemble de \mathbf{D} et (d_1, \dots, d_{2n}) un élément de \mathbf{D}^{2n} . On a les propriétés suivantes :

- (1) a est un intervalle de (\mathbf{D}, ρ) si et seulement si a est un intervalle de (\mathbf{D}, ρ^T) ,
- (2) soit les éléments \underline{a}_ρ et \bar{a}^{ρ^T} existent et sont égaux, soit ils n'existent pas,
- (3) soit les éléments \bar{a}^ρ et \underline{a}_{ρ^T} existent et sont égaux, soit ils n'existent pas,
- (4) $(d_1, \dots, d_{2n}) \in tri^\rho$ si et seulement si $(d_1, \dots, d_n, d_{2n}, \dots, d_{n+1}) \in tri^{\rho^T}$.

8.2 Description de l'algorithme

Algorithme Etant donné un $2m$ -pavé b dans (\mathbf{D}, \preceq) , qui est de la forme $b_1 \times \dots \times b_{2m}$, on s'intéresse à calculer $\text{apx}(\text{tri} \cap b)$. L'algorithme consiste à calculer une suite q_1, \dots, q_{12} de douze contraintes composées. Chacune de ces contraintes fait intervenir le même ensemble $\{y_1, \dots, y_{2m}\}$ de $2m$ variables, uniquement la relation tri ou la relation tri^{\preceq^T} , et est définie par :

$$q_1 = (y_1, \dots, y_{2m}) \in \text{tri}^\rho \wedge y_1 \in b_1 \wedge \dots \wedge y_{2m} \in b_{2m},$$

$$q_{i+1} = \begin{cases} q_i, & \text{s'il existe } j \in 1..m \text{ tel que } \text{dom}(y_j, q_i) = \emptyset, \\ \mathcal{T}_i(p_{i1}) \wedge \dots \wedge \mathcal{T}_i(p_{ik_i}), & \text{avec } q_i = p_{i1} \wedge \dots \wedge p_{ik_i}, \text{ sinon,} \end{cases}$$

où i est pris dans $1..11$, où chaque p_{ij} est une contrainte de base et où $\mathcal{T}_i(p_{ij})$ est la contrainte composée définie ci-dessous. On a alors

$$\text{apx}(\text{tri} \cap b) = \text{dom}(y_1, q_{12}) \times \dots \times \text{dom}(y_{2m}, q_{12}).$$

Définition 6 (de \mathcal{T}_i) Pour toute contrainte de base p de la forme

$$\left[\begin{array}{l} (x_1, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_{2n} \in a_{2n} \end{array} \right],$$

où ρ est \preceq ou \preceq^T , la contrainte composée $\mathcal{T}_i(p)$, avec i pris dans $1..11$, est définie comme suit au paragraphe (i).

(1) **Passage à un pavé ρ -normalisé**

$$\mathcal{T}_1(p) = \left[\begin{array}{l} (x_1, \dots, x_n, x_{n+1}, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_n \in a_n \\ \wedge x_{n+1} \in [e_1, e'_1]^\rho \\ \dots \\ \wedge x_{2n} \in [e_n, e'_n]^\rho \end{array} \right]$$

avec $(e_1, \dots, e_n) = \uparrow^\rho(\underline{a_{n+1}}_\rho, \dots, \underline{a_{2n}}_\rho)$ et $(e'_1, \dots, e'_n) = \downarrow_\rho(\overline{a_{n+1}}^\rho, \dots, \overline{a_{2n}}^\rho)$.

(2) **Passage à un pavé de type ρ -II**

$$\mathcal{T}_2(p) = \left[\begin{array}{l} (x_{\nu(1)}, \dots, x_{\nu(n)}, x_{n+1}, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_{2n} \in a_{2n} \end{array} \right]$$

où ν est une bijection de $1..n$ sur $(n+1)..(2n)$, telle que $(\overline{a_{\nu(1)}}^\rho, \dots, \overline{a_{\nu(n)}}^\rho)$ soit croissant pour ρ .

(3) **Ecrêtement supérieur de la fin du pavé et passage à un pavé de type ρ -III**

$$\mathcal{T}_3(p) = \begin{cases} \left[\begin{array}{l} (x_1, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in \emptyset \\ \dots \\ \wedge x_{2n} \in \emptyset \end{array} \right], & \text{si } \varphi \text{ n'existe pas} \\ \left[\begin{array}{l} (x_{\varphi^{-1}(n+1)}, \dots, x_{\varphi^{-1}(2n)}, x_{n+1}, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_n \in a_n \\ \wedge x_{n+1} \in [\underline{a_{n+1}}_\rho, \overline{a_{n+1} \cap a_{\varphi^{-1}(n+1)}}^\rho]^\rho \\ \dots \\ \wedge x_{2n} \in [\underline{a_{2n}}_\rho, \overline{a_{2n} \cap a_{\varphi^{-1}(2n)}}^\rho]^\rho \end{array} \right], & \text{si } \varphi \text{ existe} \end{cases}$$

où φ est la bijection de $1..n$ dans $(n+1)..(2n)$ définie par

$$\begin{aligned}\varphi(1) &= \min(\text{pert}(a, 1)), \\ \varphi(i) &= \min(\text{pert}(a, i) - \varphi(1..(i-1))), \quad \text{si } i \in 2..n,\end{aligned}$$

avec, on le rappelle,

$$\text{pert}(a, i) = \min\{j \in (n+1)..(2n) \mid a_i \cap a_j \neq \emptyset\}.$$

(4) Décomposition en pavés de type ρ -I

$$\mathcal{T}_4(p) = \left[\begin{array}{l} (x_{h(1,1)}, \dots, x_{h(1,2n_1)}) \in \text{tri}^\rho \\ \wedge x_{h(1,1)} \in a_{h(1,1)} \\ \dots \\ \wedge x_{h(1,2n_1)} \in a_{h(1,2n_1)} \end{array} \right] \wedge \dots \wedge \left[\begin{array}{l} (x_{h(k,1)}, \dots, x_{h(k,2n_k)}) \in \text{tri}^\rho \\ \wedge x_{h(k,1)} \in a_{h(k,1)} \\ \dots \\ \wedge x_{h(k,2n_k)} \in a_{h(k,2n_k)} \end{array} \right]$$

où k , les n_i et les $h(i,j)$, avec $i \in 1..k$ et $j \in 1..(2n_i)$, sont calculés comme suit. A l'aide de l'algorithme décrit dans le théorème **3**, on trie le n -uplet $(n+1, \dots, 2n)$ dans l'ordre croissant de \leq_a avec $a = a_1 \times \dots \times a_{2n}$. On met le n -uplet ainsi obtenu sous la forme $w_1 \cdot - \dots \cdot w_k$ où les n_i -uplets w_i sont croissants pour \leq et de longueurs n_i maximales. Pour chaque $i \in 1..k$, le $2n_i$ -uplet $(h(i,1), \dots, h(i,2n_i))$ est défini par

$$(n+h(i,1), \dots, n+h(i,2n_i)) = (h(i, n_i+1), \dots, h(i, 2n_i)) = w_i.$$

(5) Ecrêtement inférieur du début du pavé

$$\mathcal{T}_5(p) = \left[\begin{array}{l} (x_1, \dots, x_{2n}) \in \text{tri}^\rho \\ \wedge x_1 \in [\underline{a_1 \cap a_{f(1)}}_\rho, \overline{a_1}^\rho]^\rho \\ \dots \\ \wedge x_n \in [\underline{a_n \cap a_{f(n)}}_\rho, \overline{a_n}^\rho]^\rho \\ \wedge x_{n+1} \in a_{n+1} \\ \dots \\ \wedge x_{2n} \in a_{2n} \end{array} \right]$$

avec $f(i) = \min\{j \in (n+1)..(2n) \mid \underline{a_i}_\rho \cap \overline{a_j}^\rho\}$, pour $i \in 1..n$.

(6) Passage à un pavé ρ^T -normalisé

$$\mathcal{T}_6(p) = \left[\begin{array}{l} (x_1, \dots, x_n, x_{2n}, \dots, x_{n+1}) \in \text{tri}^{\rho^T} \\ \wedge x_1 \in a_1 \\ \dots \\ \wedge x_{2n} \in a_{2n} \end{array} \right]$$

(7) à (11)

La suite $\mathcal{T}_7(p), \mathcal{T}_8(p), \mathcal{T}_9(p), \mathcal{T}_{10}(p), \mathcal{T}_{11}(p)$ est égale à la suite $\mathcal{T}_2(p), \mathcal{T}_3(p), \mathcal{T}_4(p), \mathcal{T}_5(p), \mathcal{T}_6(p)$.

8.3 Complexité de l'algorithme

Dans les sections précédentes, nous avons vu comment effectuer chaque calcul de $\mathcal{T}_i(p)$ en $\mathcal{O}(n)$ ou $\mathcal{O}(n \log n)$ instructions élémentaires. Si m, m_1, \dots, m_k sont des entiers positifs tels que $\sum_{j=1}^k m_j = m$, on a $\sum_{j=1}^k m_j \log m_j \leq m \log m$. Il s'ensuit, compte tenu du mode de définition des contraintes q_1, \dots, q_{12} , que le calcul de $\text{apx}(\text{tri} \cap b)$, avec b de dimension $2m$, se fait en $\mathcal{O}(m \log m)$ instructions élémentaires.

9 Correction de l'algorithme

La correction de l'algorithme découle de propriétés et de théorèmes que nous avons démontrés dans les sections précédentes. Pour démontrer en détails cette correction, nous introduisons un tableau qui pour chaque $i \in 1..11$ énonce et justifie un certain nombre de propriétés de q_i et q_{i+1} , vérifiées uniquement lorsque $dom(y_1, q_i) \times \dots \times dom(y_{2m}, q_i) \neq \emptyset$.

i	Propriété de chaque contrainte de base p de q_i	Forme de $pavé(p)$ pour une contrainte de base p de q_{i+1}	Propriétés de chaque contrainte de base p de q_{i+1}	Justification
1	$\mathcal{T}_i(p) \equiv p$	normalisé		propriété 2
2	$\mathcal{T}_i(p) \equiv p$	de type II		propriété d'un tri
3	$\mathcal{T}_i(p) \equiv p$	de type III	$sol(p) \neq \emptyset$, pour tout $x \in varfin(p)$, $\overline{\pi_x(sol(p))} = \overline{dom(x, p)}$	théorème 2 et propriété 6
4	$\mathcal{T}_i(p) \equiv p$	de type I	mêmes prop. que pour $i-1$	théorèmes 3 et 4
5	$\mathcal{T}_i(p) \equiv p$	normalisé	en plus des prop. pour $i-1$, pour tout $x \in vardébut(p)$, $\overline{\pi_x(sol(p))} = \overline{dom(x, p)}$	théorème 1
6	$\mathcal{T}_i(p) \equiv p$	\preceq^T -normalisé	mêmes prop. que pour $i-1$	propriété 12
7	$\mathcal{T}_i(p) \equiv p$	de type \preceq^T -II	mêmes prop. que pour $i-1$	propriété d'un tri
8	$\mathcal{T}_i(p) \equiv p$	de type \preceq^T -III	en plus des prop. pour $i-1$, pour tout $x \in varfin(p)$, $\overline{\pi_x(sol(p))} = \overline{dom(x, p)}$	théorème 2 et propriété 6
9	$\mathcal{T}_i(p) \equiv p$	de type \preceq^T -I	même prop. que pour $i-1$	théorèmes 3 et 4
10	$\mathcal{T}_i(p) \equiv p$	\preceq^T -normalisé	en plus des prop. pour $i-1$, pour tout $x \in vardébut(p)$, $\overline{\pi_x(sol(p))} = \overline{dom(x, p)}$	théorème 1
11	$\mathcal{T}_i(p) \equiv p$	normalisé	mêmes prop. que pour $i-1$	propriété 12

Ici $vardébut(p)$ et $varfin(p)$ désignent les sous-ensembles de variables $\{x_1, \dots, x_n\}$ et $\{x_{n+1}, \dots, x_{2n}\}$, avec $\{x_1, \dots, x_{2n}\} = var(p)$.

De la deuxième colonne du tableau et de la façon dont q_{i+1} est construit à partir de q_i , on déduit que $q_1 \equiv q_{12}$. Si $dom(y_1, q_{12}) \times \dots \times dom(y_{2m}, q_{12}) \neq \emptyset$ on a donc $dom(y_1, q_{11}) \times \dots \times dom(y_{2m}, q_{11}) \neq \emptyset$ et les propriétés de q_{12} dans la dernière ligne du tableau sont vérifiées. Pour chaque contrainte de base p de q_{12} , on a

$$sol(p) \neq \emptyset \text{ et } \overline{\pi_x(sol(p))} = \overline{dom(x, p)} \text{ et } \overline{\pi_x(sol(p))} = \overline{dom(x, p)}, \text{ pour tout } x \in var(p),$$

c'est-à-dire,

$$sol(p) \neq \emptyset \text{ et } apx(\pi_x(sol(p))) \text{ existe et est égal à } dom(x, p), \text{ pour tout } x \in var(p)$$

et donc, d'après la propriété **11**,

$$apx(\pi_{y_j}(sol(q_{12}))) \text{ existe et est égal à } dom(y_j, q_{12}), \text{ pour tout } j \in 1..2m.$$

Si $dom(y_1, q_{12}) \times \dots \times dom(y_{2m}, q_{12}) = \emptyset$, pour chaque $j \in 1..2m$, l'ensemble $apx(\pi_{y_j}(sol(q_{12})))$ existe aussi et est égal à \emptyset . Il s'ensuit que dans tous les cas de figure, les ensembles $apx(\pi_{y_j}(sol(q_{12})))$ existent et

$$apx(\pi_{y_1}(sol(q_{12}))) \times \dots \times apx(\pi_{y_{2m}}(sol(q_{12}))) = dom(y_1, q_{12}) \times \dots \times dom(y_{2m}, q_{12}).$$

Puisque $sol(q_1) = sol(q_{12})$, les ensembles $apx(\pi_{y_j}(sol(q_1)))$ existent et

$$apx(\pi_{y_1}(sol(q_1))) \times \cdots \times apx(\pi_{y_{2m}}(sol(q_1))) = dom(y_1, q_{12}) \times \cdots \times dom(y_{2m}, q_{12}).$$

Par construction on a $\pi_{y_j}(sol(q_1)) = \pi_j(tri \cap b)$. Donc les ensembles $\pi_j(tri \cap b)$ existent et

$$apx(\pi_1(tri \cap b)) \times \cdots \times apx(\pi_{2m}(tri \cap b)) = dom(y_1, q_{12}) \times \cdots \times dom(y_{2m}, q_{12}).$$

D'après la propriété **1**, on a donc bien

$$apx(tri \cap b) = dom(y_1, q_{12}) \times \cdots \times dom(y_{2m}, q_{12}).$$

10 Exemple

Pour mieux faire comprendre l'algorithme de la section **8** nous le déroulons complètement sur un exemple. Comme ensemble ordonné (\mathbf{D}, \preceq) nous prenons (\mathbf{R}, \leq) , et comme $2m$ -pavé

$$b = \begin{array}{l} [0, 13] \times [6, 10] \times [10, 11] \times [4, 16] \times [4, 6] \times \\ [1, 3] \times [5, 10] \times [6, 9] \times [11, 17] \times [10, 15]. \end{array}$$

avec $m = 5$.

Contrainte composée initiale On a donc

$$q_1 = (y_1, \dots, y_{10}) \in tri \wedge y_1 \in b_1 \wedge \cdots \wedge y_{10} \in b_{10},$$

avec

$$(b_1, \dots, b_{10}) = ([0, 13], [6, 10], [10, 11], [4, 16], [4, 6], [1, 3], [5, 10], [6, 9], [11, 17], [10, 15]).$$

(1) Passage à un pavé normalisé On calcule

$$\downarrow(1, 5, 6, 11, 10) = (1, 5, 6, 11, 11),$$

$$\uparrow(3, 10, 9, 17, 15) = (3, 9, 9, 15, 15).$$

et on obtient

$$q_2 = (y_1, \dots, y_{10}) \in tri \wedge y_1 \in b_1 \wedge \cdots \wedge y_{10} \in b_{10},$$

avec

$$(b_1, \dots, b_{10}) = ([0, 13], [6, 10], [10, 11], [4, 16], [4, 6], [1, 3], [5, 9], [6, 9], [11, 15], [11, 15]).$$

(2) Passage à un pavé de type II On obtient ensuite

$$q_3 = (y_5, y_2, y_3, y_1, y_4, y_6, y_7, y_8, y_9, y_{10}) \in tri \wedge y_1 \in b_1 \wedge \cdots \wedge y_{10} \in b_{10},$$

avec

$$\begin{array}{l} (b_5, b_2, b_3, b_1, b_4, b_6, b_7, b_8, b_9, b_{10}) = \\ ([4, 6], [6, 10], [10, 11], [0, 13], [4, 16], [1, 3], [5, 9], [6, 9], [11, 15], [11, 15]). \end{array}$$

(3) Ecrêtement supérieur de la fin du pavé et passage à un pavé de type III Le calcul de φ donne

$$\varphi = \{1 \mapsto 7, 2 \mapsto 8, 3 \mapsto 9, 4 \mapsto 6, 5 \mapsto 10\}.$$

et donc l'ensemble $tri \cap b$ n'est pas vide et

$$q_4 = (y_1, y_5, y_2, y_3, y_4, y_6, y_7, y_8, y_9, y_{10}) \in tri \wedge y_1 \in b_1 \wedge \cdots \wedge y_{10} \in b_{10},$$

avec maintenant

$$\begin{array}{l} (b_1, b_5, b_2, b_3, b_4, b_6, b_7, b_8, b_9, b_{10}) = \\ ([0, 13], [4, 6], [6, 10], [10, 11], [4, 16], [1, 3], [5, 6], [6, 9], [11, 11], [11, 15]). \end{array}$$

(4) Décomposition et passage à des pavés de type I Le tri de $(6, \dots, 10)$ dans l'ordre croissant de \leq_b , avec maintenant $b = b_1 \times b_5 \times b_2 \times b_3 \times b_4 \times b_6 \times \dots \times b_{10}$ se fait par la suite de transformations

$$\begin{aligned}
& (\varepsilon, (6, 7, 8, 9, 10), \varepsilon) \\
& \longrightarrow ((6), (7, 8, 9, 10), \varepsilon) \\
& \longrightarrow ((6, 7), (8, 9, 10), \varepsilon) \\
& \longrightarrow ((6, 7, 8), (9, 10), \varepsilon) \\
& \longrightarrow ((6, 7), (9, 10), (8)) \\
& \longrightarrow ((6), (9, 10), (7, 8)) \\
& \longrightarrow ((6, 9), (10), (7, 8)) \\
& \longrightarrow ((6, 9, 10), \varepsilon, (7, 8)) \\
& \longrightarrow ((6, 9), \varepsilon, (10, 7, 8)) \\
& \longrightarrow ((6), \varepsilon, (9, 10, 7, 8)) \\
& \longrightarrow (\varepsilon, \varepsilon, (6, 9, 10, 7, 8))
\end{aligned}$$

On obtient $(6, 9, 10) \cdot (7, 8)$ et on a alors

$$q_5 = (y_1, y_3, y_4, y_6, y_9, y_{10}) \in tri \wedge (y_5, y_2, y_7, y_8) \in tri \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10},$$

avec

$$\begin{aligned}
(b_1, b_3, b_4, b_6, b_9, b_{10}) &= ([0, 13], [10, 11], [4, 16], [1, 3], [11, 11], [11, 15]), \\
(b_5, b_2, b_7, b_8) &= ([4, 6], [6, 10], [5, 6], [6, 9]).
\end{aligned}$$

(5) Ecrêtement inférieur des débuts des pavés On obtient

$$q_6 = (y_1, y_3, y_4, y_6, y_9, y_{10}) \in tri \wedge (y_5, y_2, y_7, y_8) \in tri \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10},$$

avec maintenant

$$\begin{aligned}
(b_1, b_3, b_4, b_6, b_9, b_{10}) &= ([1, 13], [11, 11], [11, 16], [1, 3], [11, 11], [11, 15]), \\
(b_5, b_2, b_7, b_8) &= ([5, 6], [6, 10], [5, 6], [6, 9]).
\end{aligned}$$

(6) Passage à des pavés \preceq^T -normalisés On obtient

$$q_7 = (y_1, y_3, y_4, y_{10}, y_9, y_6) \in tri^{\preceq^T} \wedge (y_5, y_2, y_8, y_7) \in tri^{\preceq^T} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10},$$

avec

$$\begin{aligned}
(b_1, b_3, b_4, b_{10}, b_9, b_6) &= ([1, 13], [11, 11], [11, 16], [11, 15], [11, 11], [1, 3]), \\
(b_5, b_2, b_8, b_7) &= ([5, 6], [6, 10], [6, 9], [5, 6]).
\end{aligned}$$

(7) Passage à des pavés de type \preceq^T -II On obtient

$$q_8 = (y_3, y_4, y_1, y_{10}, y_9, y_6) \in tri^{\preceq^T} \wedge (y_2, y_5, y_8, y_7) \in tri^{\preceq^T} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10},$$

avec

$$\begin{aligned}
(b_3, b_4, b_1, b_{10}, b_9, b_6) &= ([11, 11], [11, 16], [1, 13], [11, 15], [11, 11], [1, 3]), \\
(b_2, b_5, b_8, b_7) &= ([6, 10], [5, 6], [6, 9], [5, 6]).
\end{aligned}$$

(8) Ecrêtement inférieur de la fin des pavés et passage à des pavés de type \preceq^T -III Si φ' désigne le φ de $b_3 \times b_4 \times b_1 \times b_{10} \times b_9 \times b_6$ et φ'' le φ de $b_2 \times b_5 \times b_8 \times b_7$, on a

$$\begin{aligned}
\varphi' &= \{1 \mapsto 4, 2 \mapsto 5, 3 \mapsto 6\}, \\
\varphi'' &= \{1 \mapsto 3, 2 \mapsto 4\}.
\end{aligned}$$

On obtient donc

$$q_9 = (y_3, y_4, y_1, y_{10}, y_9, y_6) \in tri^{\preceq^T} \wedge (y_2, y_5, y_8, y_7) \in tri^{\preceq^T} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10},$$

avec

$$\begin{aligned}
(b_3, b_4, b_1, b_{10}, b_9, b_6) &= ([11, 11], [11, 16], [1, 13], [11, 15], [11, 11], [1, 3]), \\
(b_2, b_5, b_8, b_7) &= ([6, 10], [5, 6], [6, 9], [5, 6]).
\end{aligned}$$

(9) Décomposition et passage à des pavés de type \preceq^T -III Si l'on pose $b' = b_3 \times b_4 \times b_1 \times b_{10} \times b_9 \times b_6$ et $b'' = b_2 \times b_5 \times b_8 \times b_7$, le tri de $(4, 5, 6)$ dans l'ordre croissant pour $\leq_{b'}$ et le tri de $(3, 4)$ dans l'ordre croissant pour $\leq_{b''}$ se font par les deux suites de transformations

$$\begin{array}{ll}
(\varepsilon, (4, 5, 6), \varepsilon) & \\
\longrightarrow ((4), (5, 6), \varepsilon) & (\varepsilon, (3, 4), \varepsilon) \\
\longrightarrow ((4, 5), (6), \varepsilon) & \longrightarrow ((3), (4), \varepsilon) \\
\longrightarrow ((4), (6), (5)) & \text{et} \longrightarrow ((3, 4), \varepsilon, \varepsilon) \\
\longrightarrow (\varepsilon, (6), (4, 5)) & \longrightarrow ((3), \varepsilon, (4)) \\
\longrightarrow ((6), \varepsilon, (4, 5)) & \longrightarrow (\varepsilon, \varepsilon, (3, 4)) \\
\longrightarrow (\varepsilon, \varepsilon, (6, 4, 5)) &
\end{array}$$

On obtient $(6) \cdot (4, 5)$ et $(3, 4)$. On a alors

$$\begin{aligned}
q_{10} = & \\
(y_1, y_6) \in \text{tri}^{\preceq^T} \wedge (y_3, y_4, y_{10}, y_9) \in \text{tri}^{\preceq^T} \wedge (y_2, y_5, y_8, y_7) \in \text{tri}^{\preceq^T} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10}, & \\
\text{avec} &
\end{aligned}$$

$$\begin{aligned}
(b_1, b_6) &= ([1, 13], [1, 3]), \\
(b_3, b_4, b_{10}, b_9) &= ([11, 11], [11, 16], [11, 15], [11, 11]), \\
(b_2, b_5, b_8, b_7) &= ([6, 10], [5, 6], [6, 9], [5, 6]).
\end{aligned}$$

(10) Ecrêtement supérieur des débuts des pavés On obtient

$$\begin{aligned}
q_{11} = & \\
(y_1, y_6) \in \text{tri}^{\preceq^T} \wedge (y_3, y_4, y_{10}, y_9) \in \text{tri}^{\preceq^T} \wedge (y_2, y_5, y_8, y_7) \in \text{tri}^{\preceq^T} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10}, & \\
\text{avec maintenant} &
\end{aligned}$$

$$\begin{aligned}
(b_1, b_6) &= ([1, 3], [1, 3]), \\
(b_3, b_4, b_{10}, b_9) &= ([11, 11], [11, 15], [11, 15], [11, 11]), \\
(b_2, b_5, b_8, b_7) &= ([6, 9], [5, 6], [6, 9], [5, 6]).
\end{aligned}$$

(11) Passage à des pavés \preceq -normalisés On obtient

$$\begin{aligned}
q_{12} = & \\
(y_1, y_6) \in \text{tri} \wedge (y_3, y_4, y_9, y_{10}) \in \text{tri} \wedge (y_2, y_5, y_7, y_8) \in \text{tri} \wedge y_1 \in b_1 \wedge \dots \wedge y_{10} \in b_{10}, & \\
\text{avec maintenant} &
\end{aligned}$$

$$\begin{aligned}
(b_1, b_6) &= ([1, 3], [1, 3]), \\
(b_3, b_4, b_9, b_{10}) &= ([11, 11], [11, 15], [11, 11], [11, 15]), \\
(b_2, b_5, b_7, b_8) &= ([6, 9], [5, 6], [5, 6], [6, 9]).
\end{aligned}$$

Résultat final Et donc, si b désigne à nouveau le pavé initial, on a

$$\begin{aligned}
\text{apx}(\text{tri} \cap b) &= [1, 3] \times [6, 9] \times [11, 11] \times [11, 15] \times [5, 6] \times \\
& [1, 3] \times [5, 6] \times [6, 9] \times [11, 11] \times [11, 15].
\end{aligned}$$

11 Conclusion

11.1 Caractère incrémental de l'algorithme

Notre algorithme de calcul de $\text{apx}(\text{tri} \cap a)$ a une vertu : il est incrémental. Plus précisément, étant donné un couple (a, a') de $2n$ -pavés de (\mathbf{D}, \preceq) tels que $a' \subseteq a$ il permet de calculer $\text{apx}(\text{tri} \cap a')$ en profitant du calcul de $\text{apx}(\text{tri} \cap a)$.

Pour ceci, en supposant que $a = a_1 \times \dots \times a_{2n}$ et $a' = a'_1 \times \dots \times a'_{2n}$, on procède comme suit. Le calcul de $\text{apx}(\text{tri} \cap a)$ a été effectué en transformant la contrainte de base

$$(x_1, \dots, x_{2n}) \in \text{tri} \wedge x_1 \in a_1 \wedge \dots \wedge x_{2n} \in a_{2n}$$

en une contrainte composée q telle que

$$\text{apx}(\text{tri} \cap a) = \text{dom}(x_1, q) \times \dots \times \text{dom}(x_{2n}, q).$$

Cette contrainte q est de la forme $p_1 \wedge \dots \wedge p_k$ avec chaque p_i de la forme

$$(x_{h(i,1)}, \dots, x_{h(i,2n_i)}) \in tri \wedge x_{h(i,1)} \in b_{h(i,1)} \wedge \dots \wedge x_{h(i,2n_i)} \in b_{h(i,2n_i)}.$$

A chaque contrainte p_i on associe alors la contrainte p'_i obtenue en remplaçant chaque $b_{h(i,j)}$ par $b_{h(i,j)} \cap a'_{h(i,j)}$. Sur chaque contrainte p'_i qui diffère de p_i on effectue les 11 transformations de l'algorithme et on obtient une contrainte composée q'_i . Dans la contrainte composée q on substitue les q'_i aux p'_i et on obtient une contrainte composée finale q' telle que

$$apx(tri \cap b) = dom(x_1, q') \times \dots \times dom(x_{2n}, q').$$

11.2 Projections exactes de $tri \cap a$

Le calcul de $apx(tri \cap a)$, pour un $2n$ -pavé donné a , a consisté à calculer une approximation des $2n$ projections de l'ensemble $tri \cap a$. En fait le résultat fourni par l'algorithme de la section 8 contient une représentation quasi explicite des $2n$ projections exactes $\pi_k(tri \cap a)$.

Supposons que $tri \cap a \neq \emptyset$ et posons $b = b_1 \times \dots \times b_{2n} = apx(tri \cap a)$. Rappelons, que pour calculer ces b_i , notre algorithme transforme une contrainte de base initiale de la forme

$$(x_1, \dots, x_{2n}) \in tri \wedge x_1 \in a_1 \wedge \dots \wedge x_{2n} \in a_{2n}$$

en une contrainte composée finale q telle que $b_i = dom(x_i, q)$. En fait cette contrainte q est de la forme $p_1 \wedge \dots \wedge p_k$, où chaque p_i est une contrainte de base de la forme

$$(x_{h(i,1)}, \dots, x_{h(i,2n_i)}) \in tri \wedge x_{h(i,1)} \in b_{h(i,1)} \wedge \dots \wedge x_{h(i,2n_i)} \in b_{h(i,2n_i)}.$$

En introduisant la partition de 1..2n,

$$P = \{c_1, \dots, c_k\}, \text{ avec } c_i = \{h(i,1), \dots, h(i,2n_i)\}, \quad (18)$$

on a les deux résultats suivants :

Théorème 5

- (1) pour tout $i \in 1..n$, $\pi_i(tri \cap a) = b_i \cap (\bigcup_{j \in c_i \cap (n+1)..(2n)} b_j)$,
- (2) pour tout $j \in (n+1)..(2n)$, $\pi_j(tri \cap a) = b_j$,

Esquisse de la preuve du point (1) Il faut d'abord montrer que l'ensemble P défini en (18) est la partition la plus fine de 1..2n en classes b -autonomes. Nous laissons de côté cette partie très technique de la démonstration.

Ensuite, d'après la propriété 11, on a $\pi_{h(i,j)}(tri \cap a) = \pi_j(tri \cap a')$, avec $a' = b_{h(i,1)} \times \dots \times b_{h(i,2n_i)}$. Si, à l'aide de l'algorithme, on calcule $b' = apx(tri \cap a')$, on obtiendra $b' = a'$ et une partition P' de 1..2n_i. Du fait que P est la partition la plus fine de 1..2n en classes b -autonomes, la partition P' aura 1..2n_i comme unique classe.

On peut donc se limiter au cas où $a = b$ et où la partition P n'a qu'une seule classe 1..2n. Il faut alors montrer que, pour tout $i \in 1..n$, on a

$$\pi_i(tri \cap a) = \bigcup_{j \in (n+1)..(2n)} a_i \cap a_j.$$

D'après la propriété 4, il suffit de montrer que, pour tout $i \in 1..n$, on a

$$\bigcup_{j \in (n+1)..(2n)} a_i \cap a_j = \bigcup_{f \in \Phi} a_i \cap a_{f(i)},$$

où Φ désigne l'ensemble des applications bijectives f de 1..n dans (n+1)..(2n) telles que

$$\text{pour tout } i' \in 1..n, \quad a_{i'} \cap a_{f(i')} \neq \emptyset. \quad (19)$$

Soit p un élément de 1..n et q un élément de (n+1)..(2n) tels que $a_p \cap a_q \neq \emptyset$. Il faut montrer qu'il existe une bijection $f \in \Phi$ telle que $f(p) = q$. Du fait de la phase (4) de l'algorithme, il existe une bijection g de 1..n dans (n+1)..(2n), telle que, pour tout $i \in 1..n$, on ait d'une part $a_i \cap a_{g(i)} \neq \emptyset$ et d'autre part $g(i) < 2n$ entraîne $a_i \cap a_{g(i)+1} \neq \emptyset$. Il existe donc une bijection g de 1..n dans (n+1)..(2n) telle que

$$\begin{aligned} &\text{pour tout } i \in 1..n, \quad a_i \cap a_{g(i)} \neq \emptyset, \\ &\text{pour tout } i \in 1..n, \quad q \leq g(i) < 2n \text{ entraîne } a_i \cap a_{g(i)+1} \neq \emptyset. \end{aligned} \quad (20)$$

Si $q \leq g(p)$, la bijection f qui suit a la propriété (19) et est telle que $f(p) = q$:

$$f(i) = \begin{cases} g(i), & \text{si } g(i) \notin q \dots g(p), \\ g(i)+1, & \text{si } g(i) \in q \dots g(p)-1, \\ q, & \text{si } g(i) = g(p). \end{cases}$$

Si $g(p) < q$, on exécute dans l'ordre les trois actions suivantes :

1. Choisir un élément p' de $1..n$ tel que $g(p) < g(p')$ et $a_{p'} \cap a_{g(p)} \neq \emptyset$. Un tel p' existe toujours car dans le cas contraire, l'ensemble $s \cup g^{-1}(s)$, avec $s = (g(p)+1)..(2n)$, serait une sous-classe a -autonome stricte de $1..2n$.
2. Si $g(p') < q$, remplacer g par la bijection g' qui ne diffère de g que par le fait que $(g'(p'), g'(p)) = (g(p), g(p'))$ et reprendre la suite des actions en 1. On note que l'on a toujours $g(p) < q$, que la nouvelle bijection g conserve la propriété (20) et que l'itération se terminera du fait que $g(q)$ croît strictement.
3. Si $q \leq g(p')$ on est dans une configuration telle que $g(p) < q \leq g(p')$, telle que $a_p \cap a_q \neq \emptyset$ et $a_{p'} \cap a_{g(p)} \neq \emptyset$ et telle que la bijection g ait la propriété (20). D'après la propriété **3**, on a aussi $a_{p'} \cap a_q \neq \emptyset$. Par construction, la bijection f qui suit a la propriété (19) et est telle que $f(p) = q$:

$$f(i) = \begin{cases} g(i), & \text{si } g(i) \notin g(p) .. g(p'), \\ q, & \text{si } g(i) = g(p), \\ g(i), & \text{si } g(i) \in g(p)+1 .. q-1, \\ g(i)+1, & \text{si } g(i) \in q .. g(p')-1, \\ g(p), & \text{si } g(i) = g(p'). \end{cases}$$

Preuve du point (2) On peut supposer que le pavé a est égal au pavé b et est de type II. On a évidemment $\pi_j(\text{tri} \cap a) \subseteq a_j$, pour tout $j \in (n+1)..(2n)$. Il reste à établir, qu'étant donnés $k \in (n+1)..(2n)$ et $e \in a_k$, on a $e \in \pi_k(\text{tri} \cap a)$. D'après le lemme **1**, le théorème **2** et la propriété **12**, il existe une bijection ψ de $1..n$ dans $(n+1)..(2n)$ telle que le $2n$ -uplet

$$(\underline{a}_{\psi(1)}, \dots, \underline{a}_{\psi(n)}, \underline{a}_{n+1}, \dots, \underline{a}_{2n}) \quad (21)$$

appartienne à $\text{tri} \cap a$. Soit k' le plus petit élément de $(n+1)..(2n)$ tel que $e \in a_{k'}$ et soit $a' = a'_1 \times \dots \times a'_{2n}$ le pavé tel que, pour tout $i \in 1..2n$, on ait $a'_i = [\underline{a}_i, e]$ si $i \in k'..k$, et $a'_i = a_i$ sinon. Du fait que le $2n$ -uplet (21) appartient aussi à $\text{tri} \cap a'$, on a $\text{tri} \cap a' \neq \emptyset$. Les ensembles $\text{tri} \cap a$ et $\text{tri} \cap a'$ étant non vides et les pavés a et a' étant de type II, d'après le théorème **2**, il existe des bijections φ, φ' , telles que, pour tout $j \in (n+1)..(2n)$

$$\overline{\pi_j(\text{tri} \cap a)} = \overline{a_j \cap a_{\varphi^{-1}(j)}},$$

$$\overline{\pi_j(\text{tri} \cap a')} = \overline{a_j \cap a_{\varphi'^{-1}(j)}}$$

et telles que pour tout $i \in 1..n$ on ait $\varphi(i) = \min(\text{pert}(a, i) - \varphi(1..(i-1)))$ et $\varphi'(i) = \min(\text{pert}(a', i) - \varphi'(1..(i-1)))$, ou encore, d'après la propriété **5**, telles que, pour tout $j \in (n+1)..(2n)$,

$$\varphi^{-1}(j) = \min(\text{pert}(a, j) - \varphi^{-1}((n+1)..(j-1))),$$

$$\varphi'^{-1}(j) = \min(\text{pert}(a', j) - \varphi'^{-1}((n+1)..(j-1)))$$

Du fait qu'on a supposé les pavés a et $a_{p'}(\text{tri} \cap a)$ égaux, pour tout $j \in (n+1)..(2n)$, on a $\overline{a_j} \preceq \overline{a_{\varphi^{-1}(j)}}$ et donc, pour tout $j \in k'..2n$, on a $e \preceq \overline{a_{\varphi^{-1}(j)}}$. Par construction, les applications φ^{-1} et φ'^{-1} coïncident sur $(n+1)..(k'-1)$. Il s'ensuit que, pour tout $j \in k'..2n$ (et en particulier pour $j = k$), on a $e \preceq \overline{a_{\varphi'^{-1}(j)}}$. Donc $\overline{\pi_k(\text{tri} \cap a')} = \overline{a_k} = e$ et $e \in \pi_k(\text{tri} \cap a)$. Ceci termine les preuves des deux points.

11.3 Résultats expérimentaux de complexité

Nous avons testé les performances de notre algorithme sur des $2n$ -pavés a construits au hasard dans (\mathbf{Z}, \preceq) . Chacun de ces pavés appartient à l'ensemble $\mathcal{B}(n, m)$ des $2n$ -pavés $a = a_1 \times \dots \times a_{2n}$ qui sont tels que

(1) chaque a_i a au plus m éléments,

(2) $\text{tri} \cap [1, 4n]^{2n} \cap a \neq \emptyset$.

La valeur $4n$ a été choisie pour permettre aux $4n$ extrémités du pavé $[1, 4n]^{2n}$ de prendre toutes les configurations possibles vis-à-vis de la relation \preceq . Le paramètre m permet de contrôler le nombre d'éléments de l'ensemble $\text{tri} \cap a$.

La construction d'un élément $a = a_1 \times \dots \times a_{2n}$ de $\mathcal{B}(n, m)$ se fait en tirant au hasard un élément d' de $[1, 4n]^{2n}$ et en le triant pour obtenir d'' . Le $2n$ -uplet $d = d'.d''$ étant un élément de $\text{tri} \cap [1, 4n]^{2n}$,

il est alors aisé de construire aléatoirement le pavé a avec $d \in a$ et chaque a_i composé d'au plus m éléments.

Ci-dessous on trouvera un tableau résumant les performances de notre algorithme de calcul de $\text{apx}(\text{tri} \cap a)$ pour un $a \in \mathcal{B}(n, m)$ avec différentes valeurs de n et m . L'algorithme est programmé en Java et exécuté sur un Pentium II-MMX à 450MHz, avec le kit de développement jdk1.2 de Sun, sous Windows NT 4.0. Dans le tableau, on désigne par k , le nombre de contraintes de base obtenues à la fin de l'algorithme, t_1 , le temps, en milli-secondes, pour trier les n éléments du n -uplet d' servant à construire a , t_2 , le temps, en milli-secondes, pour calculer $\text{apx}(\text{tri} \cap a)$.

$2n$	m	k	t_1	t_2	t_2/t_1
10^4	200	4	10	100	10,0
10^4	20	113	10	100	10,0
10^4	2	4 233	20	90	4,5
10^5	200	7	100	431	4,3
10^5	20	11 634	100	400	4,0
10^5	2	42 579	100	431	4,3
10^6	200	101	1 562	5 769	3,7
10^6	20	11 5654	1 663	4 837	2,9
10^6	2	42 4618	1 542	5 087	3,3
10^7	200	907	21 852	69 600	3,2
10^7	20	1 161 364	21 741	55 880	2,6
10^7	2	4 250 255	21 722	63 472	2,9

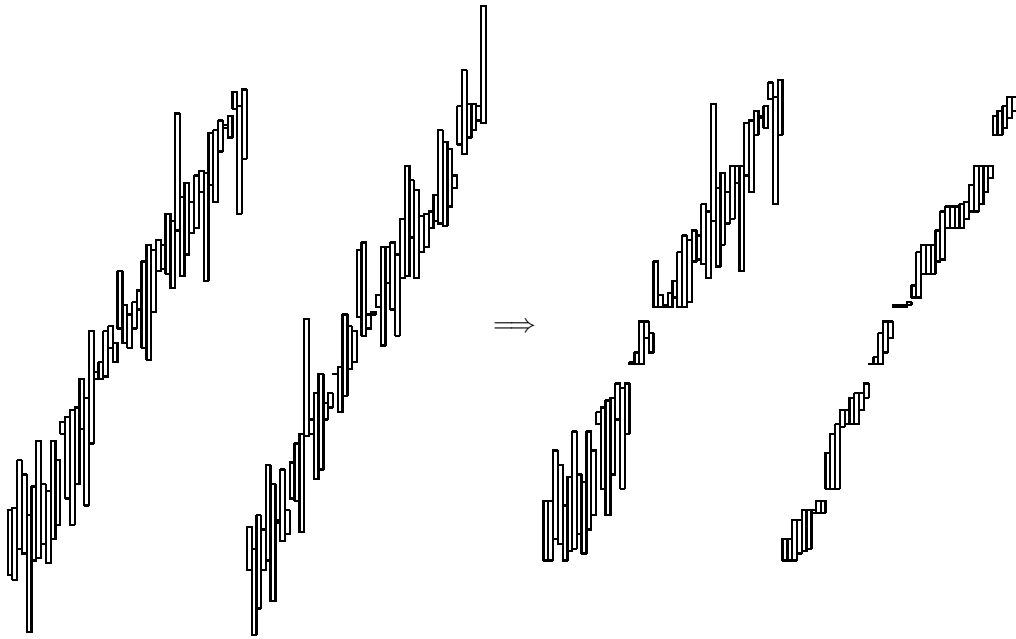
Dans notre implantation nous avons ramené tous les tris à : (1) un tri de n intervalles sur leurs plus grands éléments et (2), pour chaque classe d'intervalles créée dans la première décomposition, un tri des intervalles sur leurs plus petits éléments. L'algorithme de tri choisi est de complexité $\mathcal{O}(n \log n)$. Il consiste à trier successivement des sous-suites de longueurs $2^1, 2^2, \dots, 2^p$, avec $p = \lceil \log_2 n \rceil$. Chaque tri de 2^{i+i} éléments consiste alors en une fusion de deux listes de 2^i éléments.

L'arbre binaire équilibré de n nœuds nécessaire au calcul de φ a été codé par un vecteur de n éléments avec la technique classique de mettre en position $2i$ le fils gauche et en position $2i+1$ le fils droit du nœud en position i .

La conclusion qui s'impose est que, étant donné un $2n$ -pavé a , le temps de calcul de $\text{apx}(\text{tri} \cap a)$ est de l'ordre de 4 fois le temps nécessaire pour trier n individus.

Terminons par le dessin de la transformation $a \implies \text{apx}(\text{tri} \cap a)$ pour un pavé a de $\mathcal{B}(50, 19)$.

Pour une meilleure présentation, nous avons réordonné les 50 premiers intervalles du 100-pavé a .



Références

- [1] Benhamou F. et W.J. Older, Applying Interval Arithmetic to Real, Integer and Boolean Constraints. *Journal of Logic Programming*, 1997.
- [2] Knuth D.E., Sorting and Searching, The Art of Computer Programming, volume 3, Addison Wesley, 1973.
- [3] Older W.J. and A. Vellino, Extending Prolog with Constraint Arithmetic on Real Intervals, Proceedings of the Canadian Conference on Electrical and Computer Engineering, 1990
- [4] Older W.J., G.M. Swinkels et M.H. van Emden, Getting to the Real Problem : Experience with BNR Prolog in OR, in *Proceedings of the Third International Conference on the Practical Applications of Prolog*, (PAP'95 à Paris), Alinmead Software Ltd, ISBN 0 9 525554 0 9, avril 1995.
- [5] Zhou J., A permutation-based approach for solving the job-shop problem, *Constraints*, vol 2, no 2, 185-213, octobre 1997.