Programmation Linéaire en Nombres Entiers
II. Procédures de séparation et évaluation et Programmation dynamique

Sandra U. Ngueveu

INP-ENSEEIHT / LAAS-CNRS
sandra.ngueveu@toulouse-inp.fr - ngueveu@laas.fr

2020/2021

Procédures de séparation et évaluation pour PLNE

Principe :

- Séparer progressivement le problème en sous-problèmes traitables (aussi appelés "sondables")

Objectif

- Enumérer intelligemment l'espace des solutions
- Hiérarchiser les sous-problèmes sous forme d'arbre
- "tuer" des branches/nœuds au plus tôt lors de l'exploration de l'arbre

3 composantes :

- Evaluation/Tests de Sondabilité
- Règle de Séparation / Principe de séparation
- Stratégie d'exploration

Tests de Sondabilité

Un nœud est sondable si on peut répondre "OUI" à un des 3 tests suivants :

TA : Test d'admissibilité = montrer que (mq) il n'y a pas de solution admissible

- ex1 : si contrainte du type $\geq b_i$, alors évaluer un majorant G_i^k du membre de gauche. Si $G_i^k < b_i$ alors même dans le meilleur des cas on ne pourra pas valider cette contrainte, donc inutile de continuer sur cette branche, "tuer" le nœud correspondant
- ex2 : raisonnement inverse si contrainte du type $\leq b_i$
- ex3 : mq une relaxation du problème est déjà infaisable

TO : Test d'optimalité = mq la sol optimale ne peut pas être au sein de ce nœud

- mq les coûts des sols du nœud seront toujours pires que celui d'une solution déjà connue
- ex : calcul de BINF (E_i) à comparer avec le coût d'une solution déjà connue (préalablement obtenue par la même PSE ou un autre algorithme)

TR : Test de résolution = mq le sous-problème se résoud de manière triviale

- sous-problème à une seule variable
- sous-problème polynomial

Tests de Sondabilité

Actions

- Un nœud sondable peut être éliminé.
- Un nœud non sondable doit être séparé en plusieurs ss-pbs suivant la règle de séparation prévue.

Une PSE efficace trouve un compromis entre la qualité des résultats fournis par les tests et l'investissement accepté en temps et volumes de calculs, car ces calculs doivent être faits pour chaque nœud exploré.

Principe de Séparation

- Aucune solution ne doit être perdue ou ajoutée
  - l'union des sous-problèmes reforme le problème de départ
- Souvent un choix heuristique validé par des tests numériques
  - choix glouton (ex : l'arc libre de plus petit coût dans un graphe)
  - choix le plus informant (ex : la variable présente dans le plus grand nombre de contraintes)
  - ...
- Le but est :
  - soit de précipiter la découverte d'une nouvelle solution
  - soit de détecter au plus vite des contradictions prouvant qu'un nœud est vide
- Une séparation poussée à l'extrême $\Leftrightarrow$ fragmenter en sous-ensembles à une solution $\Rightarrow$ nbre exponentiel de nœuds explorés (énumération complète).
  - Pour éviter cela, on évite de séparer le nœud-père en "trop" de nœuds fils et on compte sur les tests TA,TO,TR pour réduire l'arborescence en identifiant au plus tôt les nœuds qui ne valent pas la peine d'être séparés à nouveau.

Par convention, les nœuds sont souvent numérotés dans l'ordre de leur création.

Stratégie d'Exploration

Il s'agit de définir l'ordre suivant lequel les nœuds seront étudiés

PSEP

- Priorité au nœud qui donne la BINF de plus petite valeur
- Avantage : Les sol réalisables trouvées sont tt de suite de très bonne qualité
- Inconvénient : Requiert plus de mémoire (liste des nœuds non séparés)

PSES

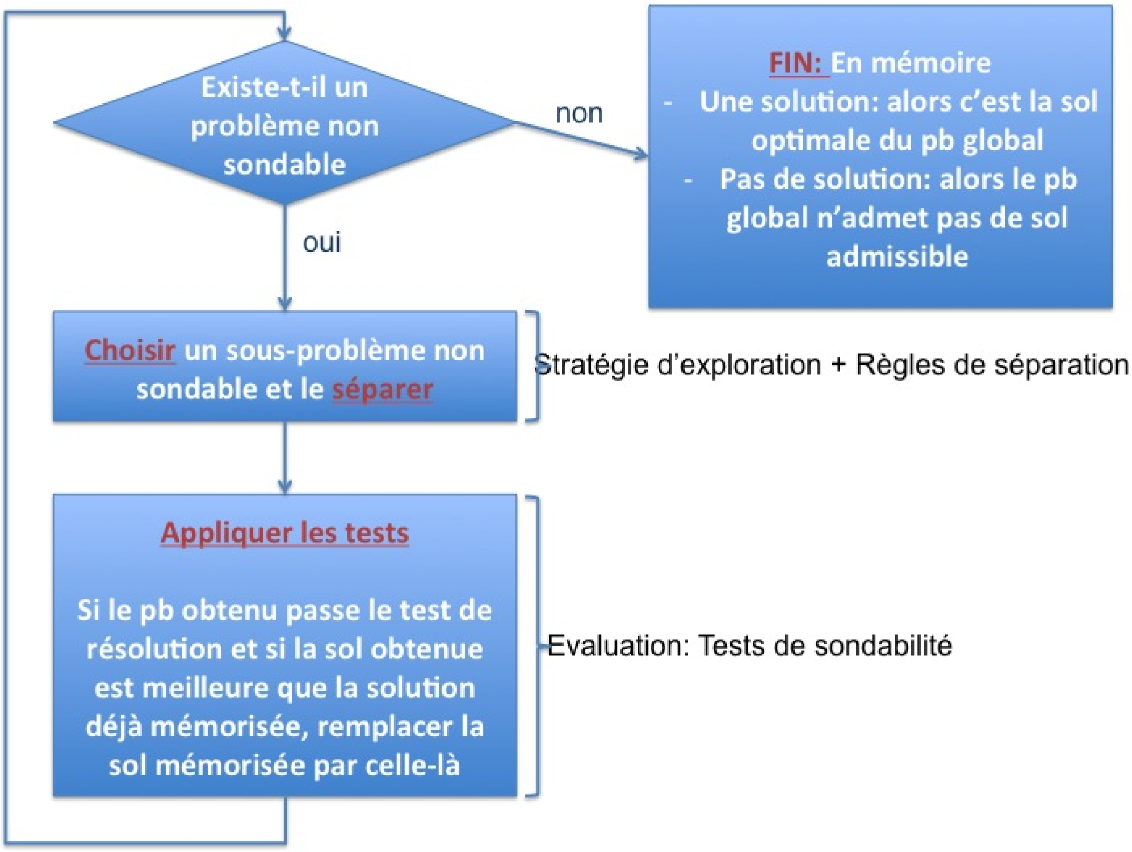
- Priorité toujours "à gauche" (1er des noeuds-fils nouvellement créés)
- Avantage : Requiert moins d'espace mémoire, Trouve plus vite une solution réalisable
- Inconvénient : Globalement moins rapide que PSEP

Autres

- Probabilité sur le choix de la prochaine branche à explorer
- ...

Les tests numériques et les contraintes technologiques (espace mémoire, ...) permettent de valider/invalider le choix d'une stratégie d'exploration par rapport à une autre.

Procédures de séparation et évaluation pour PLNE



Exemples d'applications

Application 1 : Méthode arborescente de Dakin

$$(\mathrm{P})\ \max f(x) = 4x_1 + 3x_2 \tag{1}$$

s.c.

$$3x_1 + 4x_2 \leq 12 \tag{2}$$
$$4x_1 + 2x_2 \leq 9x_i \in \{0,1\} \quad \mathrm{i} = 1,2 \tag{3}$$

Résoudre (P) par PSE à l'aide des paramètres suivants :

1. Critère de séparation :
   - variable la plus fractionnaire
2. Tests de Sondabilité
   - TA : réussi si la relaxation linéaire (RL) n'a pas de solution admissible
   - TO : réussi si la solution de la RL est pire que la meilleure solution connue
   - TR : réussi si la solution de la RL est entière
3. Stratégie d'Exploration
   - PSEP : priorité au sous-problème non sondable de meilleure borne

L'arborescence résultante est visible sur la figure 2.4 en page 41 du livre

Application 2

$$(\mathrm{P})\ \min f(x) = 5x_1 + 7x_2 + 10x_3 + 3x_4 + x_5 \tag{4}$$

s.c.

$$x_1 - 3x_2 + 5x_3 + x_4 - 4x_5 + 2 \geq 0 \tag{5}$$
$$-2x_1 + 6x_2 - 3x_3 - 2x_4 + 2x_5 \geq 0 \tag{6}$$
$$-x_2 + 2x_3 - x_4 - 1 \geq 0 \tag{7}$$
$$x_i \in \{0,1\} \quad \mathrm{i} = 1, \ldots, 5 \tag{8}$$

Résoudre (P) par PSE à l'aide des paramètres suivants :

1. Critère de séparation :
   - variable de plus grand coefficient dans $f(x)$
   - 2 sous-problèmes : var = 1 ou var = 0
2. Tests de Sondabilité
   - TA : éval des contraintes en majorant le membre de gauche ($= G_i^k$)
   - TO : éval d'une borne ($= E_k$) de la fction-obj, comparaison avec sol si connue
   - TR : si le sous-problème obtenu est trivial à résoudre
3. Stratégie d'Exploration
   - PSEP : priorité au sous-problème non sondable de meilleure borne
   - PSES : priorité au sous-problème "à gauche"

* **Résolution par P.S.E.P. :**

Mémorisation :

$x_3=1$ $x_2=1$ $x_1=0$ $x_4=0$ $x_5=0$ $f=17$

0

Min $f(x)=5x_1+7x_2+10x_3+3x_4+x_5$ $E_0 = 0$

$x_1-3x_2+5x_3+x_4-4x_5+2 \geq 0$ $G_1^{\,0} = 1+5+1+2 = 9$

$-2x_1+6x_2-3x_3-2x_4+2x_5 \geq 0$ $G_2^{\,0} = 6+2 = 8$

$-x_2+2x_3-x_4-1 \geq 0$ $G_3^{\,0} = 2-1 = 1$

$x_i \in \{0,1\}$ $i = 1,\ldots,5$

$x_3=1$ → **1**

Min $f(x)=5x_1+7x_2+3x_4+x_5+10$ $E_1 = 10$

$x_1-3x_2+x_4-4x_5+7 \geq 0$ $G_1^{\,1} = 2+7 = 9$

$-2x_1+6x_2-2x_4+2x_5-3 \geq 0$ $G_2^{\,1} = 8-3 = 5$

$-x_2-x_4+1 \geq 0$ $G_3^{\,0} = 0+1 = 1$

$x_i \in \{0,1\}$

$x_3=0$ → **2**

Min $f(x)=5x_1+7x_2+3x_4+x_5$ $E_2 = 0$

$x_1-3x_2+x_4-4x_5+2 \geq 0$ $G_1^{\,2} = 2+2 = 4$

$-2x_1+6x_2-2x_4+2x_5 \geq 0$ $G_2^{\,2} = 8$

$-x_2-x_4-1 \geq 0$ $G_3^{\,2} = 0-1 = -1$

$x_i \in \{0,1\}$

S → TA

$x_2=1$ → **3**

Min $f(x)=5x_1+3x_4+x_5+17$ $E_3 = 17$

$x_1+x_4-4x_5+4 \geq 0$ $G_1^{\,3} = 2+4 = 6$

$-2x_1-2x_4+2x_5+3 \geq 0$ $G_2^{\,3} = 5$

$-x_4 \geq 0$ $G_3^{\,3} = 0$

$x_i \in \{0,1\}$

$x_2=0$ → **4**

Min $f(x)=5x_1+3x_4+x_5+10$ $E_4 = 10$

$x_1+x_4-4x_5+7 \geq 0$ $G_1^{\,4} = 2+7 = 9$

$-2x_1-2x_4+2x_5-3 \geq 0$ $G_2^{\,4} = 2-3 = -1$

$-x_4+1 \geq 0$ $G_3^{\,4} = 0+1 = 1$

$x_i \in \{0,1\}$

S → TA

$x_1=1$ → **5**

Min $f(x)=+3x_4+x_5+22$ $E_5 = 22$

$+x_4-4x_5+5 \geq 0$ $G_1^{\,5} = 1+5 = 6$

$-2x_4+2x_5+1 \geq 0$ $G_2^{\,5} = 2+1 = 3$

$-x_4 \geq 0$ $G_3^{\,5} = 0$

$x_i \in \{0,1\}$

S → TO

$x_1=0$ → **6**

Min $f(x)=+3x_4+x_5+17$ $E_6 = 17$

$+x_4-4x_5+4 \geq 0$ $G_1^{\,6} = 5$

$-2x_4+2x_5+3 \geq 0$ $G_2^{\,6} = 5$

$-x_4 \geq 0$ $G_3^{\,6} = 0$

$x_i \in \{0,1\}$

$x_4=0$ → **8**

Min $f(x)=+x_5+17$ $E_8 = 17$

$-4x_5+4 \geq 0$ $G_1^{\,8} = 4$

$+2x_5+3 \geq 0$ $G_2^{\,8} = 5$

$0 \geq 0$ $G_3^{\,8} = 0$

$x_i \in \{0,1\}$

S → TR

$(x_5=0)$

$x_4=1$ → **7**

Min $f(x)=+x_5+20$ $E_7 = 20$

$-4x_5+5 \geq 0$ $G_1^{\,7} = 5$

$+2x_5-1 \geq 0$ $G_2^{\,7} = 3$

$-1 \geq 0$ $G_3^{\,7} = -1$

$x_i \in \{0,1\}$

17

S → TA

* Résolution par P.S.E.S. :

Mémorisation :

~~$x_3=1$ $x_2=1$ $x_1=1$ $x_4=0$ $x_5=0$ f=22~~

$x_3=1$ $x_2=1$ $x_1=0$ $x_4=0$ $x_5=0$ f=17

0

Min $f(x)=5x_1+7x_2+10x_3+3x_4+x_5$ $E_0 = 0$

$x_1-3x_2+5x_3+x_4-4x_5+2 \geq 0$ $G_1^0 = 1+5+1+2 = 9$

$-2x_1+6x_2-3x_3-2x_4+2x_5 \geq 0$ $G_2^0 = 6+2 = 8$

$-x_2+2x_3-x_4-1 \geq 0$ $G_3^0 = 2-1 = 1$

$x_i \in \{0,1\}$ $i = 1,\ldots,5$

$x_3=1$ → **1** ; $x_3=0$ → **2**

1

Min $f(x)=5x_1+7x_2+3x_4+x_5+10$ $E_1 = 10$

$x_1-3x_2+x_4-4x_5+7 \geq 0$ $G_1^1 = 2+7 = 9$

$-2x_1+6x_2-2x_4+2x_5-3 \geq 0$ $G_2^1 = 8-3 = 5$

$-x_2-x_4+1 \geq 0$ $G_3^0 = 0+1 = 1$

$x_i \in \{0,1\}$

2

Min $f(x)=5x_1+7x_2+3x_4+x_5$ $E_2 = 0$

$x_1-3x_2+x_4-4x_5+2 \geq 0$ $G_1^2 = 2+2 = 4$

$-2x_1+6x_2-2x_4+2x_5 \geq 0$ $G_2^2 = 8$

$-x_2-x_4-1 \geq 0$ $G_3^2 = 0-1=-1$

$x_i \in \{0,1\}$

S → TA

$x_2=1$ → **3** ; $x_2=0$ → **4**

3

Min $f(x)=5x_1+3x_4+x_5+17$ $E_3 = 17$

$x_1+x_4-4x_5+4 \geq 0$ $G_1^3 = 2+4 = 6$

$-2x_1-2x_4+2x_5+3 \geq 0$ $G_2^3 = 5$

$-x_4 \geq 0$ $G_3^3 = 0$

$x_i \in \{0,1\}$

4

Min $f(x)=5x_1+3x_4+x_5+10$ $E_4 = 10$

$x_1+x_4-4x_5+7 \geq 0$ $G_1^4 = 2+7 = 9$

$-2x_1-2x_4+2x_5-3 \geq 0$ $G_2^4 = 2-3 = -1$

$-x_4+1 \geq 0$ $G_3^4 = 0+1 = 1$

$x_i \in \{0,1\}$

S → TA

$x_1=1$ → **5** ; $x_1=0$ → **6**

5

Min $f(x)=+3x_4+x_5+22$ $E_5 = 22$

$+x_4-4x_5+5 \geq 0$ $G_1^5 = 1+5 = 6$

$-2x_4+2x_5+1 \geq 0$ $G_2^5 = 2+1 = 3$

$-x_4 \geq 0$ $G_3^5 = 0$

$x_i \in \{0,1\}$

6

Min $f(x)=+3x_4+x_5+17$ $E_6 = 17$

$+x_4-4x_5+4 \geq 0$ $G_1^6 = 5$

$-2x_4+2x_5+3 \geq 0$ $G_2^6 = 5$

$-x_4 \geq 0$ $G_3^6 = 0$

$x_i \in \{0,1\}$

$x_4=1$ → **7** ; $x_4=0$ → **8**

$x_4=1$ → **9** ; $x_4=0$ → **10**

7

Min $f(x)=+x_5+25$ $E_7 = 25$

$-4x_5+6 \geq 0$ $G_1^7 = 6$

$+2x_5-1 \geq 0$ $G_2^7 = 1$

$-1 \geq 0$ $G_3^7 = -1$

$x_i \in \{0,1\}$

S → TA

10

Min $f(x)=+x_5+17$ $E_{10} = 17$

$-4x_5+4 \geq 0$ $G_1^{10} = 4$

$+2x_5+3 \geq 0$ $G_2^{10} = 5$

$0 \geq 0$ $G_3^{10} = 0$

$x_i \in \{0,1\}$

S → TR
($x_5=0$)

8

Min $f(x)=+x_5+17$ $E_8 = 22$

$-4x_5+5 \geq 0$ $G_1^8 = 5$

$+2x_5+1 \geq 0$ $G_2^8 = 3$

$-x_4 \geq 0$ $G_3^8 = 0$

$x_i \in \{0,1\}$

S → TR
($x_5=0$)

9

Min $f(x)=+x_5+20$ $E_9= 20$

$-4x_5+5 \geq 0$ $G_1^9 = 5$

$+2x_5-1 \geq 0$ $G_2^9 = 3$

$-1 \geq 0$ $G_3^9 = -1$

$x_i \in \{0,1\}$

S → TA

Programmation dynamique

Application au calcul de plus court chemin

Application : Calcul de plus court chemin dans un graphe quelconque

Algorithme de Bellman-Ford
(déjà vu en cours de graphe ? sinon sera expliqué rapidement en cours)

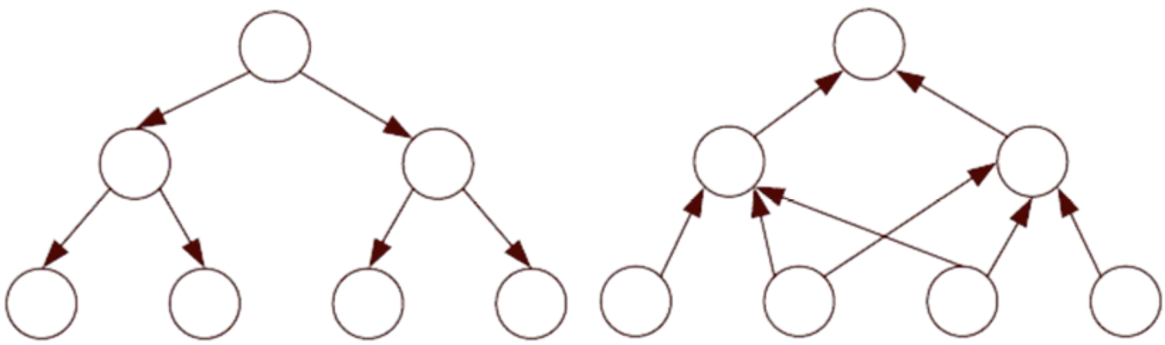
Généralités

Principe

- décompose le problème en étapes
- parcourt à rebours (de la dernière décision aux précédentes) le processus de décision séquentiel associé au problème
- chaque étape corresponds à un sous-problème résolu **optimalement** compte tenu des infos obtenues lors des étapes précédentes
- nécessite de trouver une relation de récurrence entre les valeurs des critères de deux niveaux successifs

Programmation dynamique vs PSE

Similitudes : les deux méthodes essaient de contenir l'explosion combinatoire tout en gardant la garantie d'optimalité de la solution obtenue



(a) PSE (b) Programmation dynamique

Différences :

- ordre d'exploration : "du haut vers le bas" vs "du bas vers le haut"
- en programmation dynamique classique chaque niveau ne peut être entamé qu'après avoir finalisé les niveaux qui précèdent. Donc aucune solution réalisable n'est obtenue avant le dernier niveau !