

Rédigé par : Hervé de Milleville

Ref : PRO-DYN

Revu par : Jean-Paul Forest et Yannick Le Nir

Approuvé par :

1 Rappels

La programmation dynamique est basée sur le principe de Bellmann : toute sous solution d'une solution optimale est elle-même optimale.

En fait, ce principe n'est vérifié que si la fonction objectif est décomposable au sens suivant :

$f(x_1, \dots, x_n) = h(g(x_1, \dots, x_{n-1}), x_n)$ où h est une fonction de \mathbb{R}^2 dans \mathbb{R} ayant la propriété suivante :

$\forall \beta, \text{ la fonction définie par } \alpha \longrightarrow h_\beta(\alpha) = h(\alpha, \beta) \text{ est croissante}$

Si ce principe est vérifié alors : $\text{Max } f(x_1, \dots, x_n) = h(\text{Max } g(x_1, \dots, x_{n-1}), x_n)$

2 Exercices

2.1 Isopérimétrie

L'extrait qui suit provient de l'article wikipedia [Isopérimétrie](#).

*En géométrie euclidienne, l'**isopérimétrie** est initialement l'étude des propriétés des formes géométriques du plan qui partagent le même périmètre, ce qui se généralise ensuite dans les autres espaces euclidiens.*

En particulier, le problème le plus classique consiste à déterminer la forme géométrique plane qui maximise son aire avec un périmètre fixé. La réponse est intuitive, c'est le disque, mais malgré son apparence anodine ce problème fait appel à des théories sophistiquées pour obtenir une démonstration rigoureuse (particularité qu'il partage par exemple avec le théorème de Jordan qui indique qu'une boucle tracée sans croisement divise le plan en deux parties). Pour cette raison on simplifie parfois le problème isopérimétrique en limitant les surfaces autorisées, par exemple en se restreignant aux seuls quadrilatères ou triangles, ce qui donne alors respectivement le carré et le triangle équilatéral.

De manière générale, le polygone à n côtés ayant la plus grande surface, à périmètre donné, est celui qui se rapproche le plus du cercle : c'est le polygone régulier.

2.1.1 Enoncé :

Sur la base de cet article, nous allons résoudre un problème équivalent.

On donne une corde de longueur l . On forme avec cette corde un polygone de n côtés. On cherche parmi tous ces polygones celui qui maximise le produit des côtés.

2.1.2 Questions

- Modéliser ce problème
- Montrer qu'il est dans la classe des problèmes de programmation dynamique
- Résoudre ce problème en appliquant le principe de Bellmann

2.1.3 Pour finir, un petit problème de réflexion

On considère un polygone convexe régulier à n côtés. Chaque côté est de longueur a . On peut démontrer que la surface et le périmètre sont donnés par les formules qui suivent :

$$S = \frac{na^2}{4 \tan\left(\frac{\pi}{n}\right)} \quad \text{et} \quad P = na$$

- Montrer que si on prend deux polygones réguliers de même périmètre alors celui de plus grande surface est celui qui a le plus grand nombre de côtés.

TD DE PROGRAMMATION DYNAMIQUE

- En déduire que parmi les figures régulières d'un périmètre donné, celle qui a la plus grande surface est un cercle.

2.2 Choix d'investissement

Une entreprise peut investir dans 4 projets. Elle dispose d'un budget maximum de 7 M€. Elle peut répartir son budget entre ces différents projets. Pour simplifier le problème, on suppose que chaque montant investi est un nombre entier de millions d'euros.

Montant investi en M€	Rendement			
	I1	I2	I3	I4
0.00	0.00	0.00	0.00	0.00
1.00	0.56	0.50	0.30	0.50
2.00	0.90	0.82	0.50	0.66
3.00	1.30	1.10	0.80	0.84
4.00	1.56	1.30	1.00	0.96
5.00	1.80	1.50	1.24	1.06
6.00	2.04	1.60	1.46	1.12
7.00	2.26	1.70	1.64	1.16

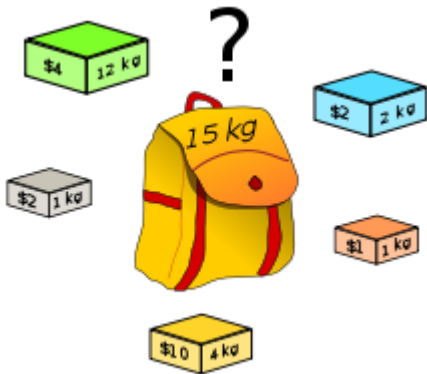
Vous devez aider cette entreprise à investir afin d'optimiser la somme des rendements.

2.2.1 Questions

- Modéliser ce problème
- Montrer qu'il est dans la classe des problèmes de programmation dynamique
- Résoudre ce problème en appliquant la méthode des tableaux

2.3 Problème du sac à dos

En algorithmique, le **problème du sac à dos**, noté également **KP** (en anglais, *Knapsack Problem*) est un problème d'optimisation combinatoire. Il modélise une situation analogue au remplissage d'un sac à dos, ne pouvant supporter plus d'un certain poids, avec tout ou partie d'un ensemble donné d'objets ayant chacun un poids et une valeur. Les objets mis dans le sac à dos doivent maximiser la valeur totale, sans dépasser le poids maximum.



Le problème du sac à dos : quelles boîtes choisir afin de maximiser la somme emportée tout en ne dépassant pas les 15 kg autorisés ?

L'article et l'image et empruntée sur le site wikipédia sur le thème du [problème du sac à dos](#)

2.3.1 Questions

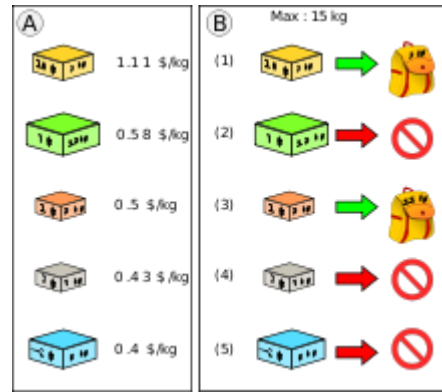
- Modéliser ce problème

TD DE PROGRAMMATION DYNAMIQUE

- Commenter l'algorithme glouton qui suit :

```
trier les objets par ordre décroissant  
d'efficacité  
w_conso := 0
```

```
pour i de 1 à n  
  si w[i] + w_conso ≤ W alors  
    x[i] := 1  
    w_conso := w_conso + w[i]  
  sinon  
    x[i] := 0  
  fin si  
fin pour
```



- Montrer qu'il est dans la classe des problèmes de programmation dynamique
- Résoudre ce problème en appliquant la méthode des tableaux
- Que cela change-t-il dans la modélisation du problème si on propose plusieurs boites de chaque type ?

2.4 Programmer la méthode des tableaux

2.4.1 Enoncé

On vous demande d'écrire un programme pour résoudre des problèmes de programmation dynamique. On se restreindra au cas suivants :

- La fonction objectif est de la forme $f(x_1, \dots, x_n) = f_1(x_1) + \dots + f_n(x_n)$
- Chaque variable x_i ne prend que des valeurs entières.
- Il n'y a qu'une seule contrainte de la forme $\sum_{i=1}^n c_i \cdot x_i \leq \lambda$