

JEAN BOVET

## **Représentation de matrices euclidiennes en vue de leur génération aléatoire**

*RAIRO. Recherche opérationnelle*, tome 19, n° 4 (1985),  
p. 375-379

[http://www.numdam.org/item?id=RO\\_1985\\_\\_19\\_4\\_375\\_0](http://www.numdam.org/item?id=RO_1985__19_4_375_0)

© AFCET, 1985, tous droits réservés.

L'accès aux archives de la revue « RAIRO. Recherche opérationnelle » implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/conditions>). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

Article numérisé dans le cadre du programme  
Numérisation de documents anciens mathématiques  
<http://www.numdam.org/>

## REPRÉSENTATION DE MATRICES EUCLIDIENNES EN VUE DE LEUR GÉNÉRATION ALÉATOIRE (\*)

par Jean BOVET <sup>(1)</sup>

---

**Résumé.** — *Pour tester de nombreuses méthodes permettant de résoudre des problèmes combinatoires sur des réseaux (routiers, par exemple), il est nécessaire de disposer d'un grand nombre d'exemples de réseaux aléatoires. Une méthode simple est présentée permettant de générer aléatoirement toute matrice dont les termes sont égaux à une certaine distance (symétrique) entre chaque paire de points. On montre qu'il n'est pas possible d'utiliser n'importe quelle distance métrique; finalement, on propose une telle distance admissible.*

**Mots clés :** Réseaux routiers; voyageur de commerce.

**Abstract.** — *In order to test some methods solving various combinatorial problems on networks (e.g. road-networks), it is necessary to have many randomly generated networks. A method is given for generating any matrix with entries being equal to a certain (symmetric) distance between every pair of nodes. It is shown that it is not possible to use any metric distance; finally, a feasible distance function is proposed.*

**Keywords:** Road networks; travelling salesman.

### 0. INTRODUCTION

Lorsque l'on développe des méthodes de calcul de cheminement dans des réseaux (plus courts chemins, tournées, etc., voir par exemple [2 à 5]) et que l'on désire mesurer les performances de telles méthodes, il est nécessaire de disposer d'un ensemble d'exemples-test assez grand pour pouvoir tirer des conclusions fiables des résultats obtenus. Souvent, on n'a pas accès à un nombre suffisant de réseaux pour effectuer ces tests. Il est alors utile de générer des exemples aléatoires.

Un réseau (routier, par exemple) peut être donné par un ensemble de points (carrefours) et par un ensemble de routes caractérisées par une longueur non négative. A partir de ces données, on peut, avec une procédure de calcul de plus courts chemins, obtenir une matrice des longueurs des plus courts chemins entre les paires de nœuds du réseau. Si le réseau utilisé est symétrique, on obtient une matrice  $A$  dont chaque terme  $a_{ij}$  est la longueur du plus court

---

(\*) Reçu octobre 1984.

(<sup>1</sup>) Département de Mathématiques, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Suisse.

chemin entre les nœuds  $i$  et  $j$ . De telles matrices, que nous appellerons *matrices euclidiennes*, vérifient les inégalités triangulaires :

$$a_{ij} \leq a_{ik} + a_{kj}, \quad \forall i, j, k.$$

A notre connaissance, peu d'articles ont été publiés traitant du problème de la génération aléatoire de telles matrices. Dans [1], par exemple, le problème des distances ne se pose pas, les réseaux générés n'étant pas sensés être des réseaux « réels ». Il est évident qu'une méthode pourrait consister à générer n'importe quelle matrice symétrique  $A'$  de la bonne dimension, puis d'appliquer une méthode de recherche de plus courts chemins entre toutes les paires de nœuds à partir des longueurs données dans la matrice  $A'$ . La matrice  $A$  sera alors la matrice (euclidienne) des plus courts chemins entre chaque paire de nœuds du réseau. Un tel procédé serait toutefois très coûteux en temps de calcul.

On se propose d'étudier une méthode plus simple permettant de générer (aléatoirement) des matrices euclidiennes, en s'inspirant d'une méthode qui est donnée par les étapes (a) et (b) suivantes :

#### Méthode (G) (réseau à $n$ sommets)

(a) Générer aléatoirement  $n$  points dans  $R^m$  :  $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^n$ .

(b) Calculer  $a_{ij} = d(\mathbf{x}^i, \mathbf{x}^j)$ , où la distance  $d$  vérifie :

(i)  $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ ;

(ii)  $d(\mathbf{x}, \mathbf{x}) = 0$ ;

(iii)  $d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y})$ .

Une telle matrice est alors évidemment euclidienne. On verra pourtant dans le chapitre suivant que la méthode (G) ne permet pas d'engendrer *toutes* les matrices euclidiennes avec n'importe quelle distance  $d$  vérifiant (i)-(iii).

#### 1. UNE « MAUVAISE » DISTANCE

Soit :

$$d_1(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}| = \left( \sum_{k=1}^m (x_k - y_k)^2 \right)^{1/2},$$

la distance euclidienne usuelle.

La méthode (G) ne permet pas de générer *toutes* les matrices euclidiennes à l'aide de la distance  $d_1$ , comme le montre le contre-exemple suivant :

Soit la matrice :

$$A = \begin{pmatrix} - & 2 & 3 & 4 \\ 2 & - & 5 & 4 \\ 3 & 5 & - & 2 \\ 4 & 4 & 2 & - \end{pmatrix}.$$

$a_{21} + a_{13} = a_{23}$  implique que les points 1, 2 et 3 sont alignés. Sans restriction de la généralité, on peut donc, dans  $R^2$ , définir les points :

$$\mathbf{x}^1 = (2, 0),$$

$$\mathbf{x}^2 = (0, 0),$$

et :

$$\mathbf{x}^3 = (5, 0).$$

De plus,  $a_{14} = a_{24} = 4$  implique :

$$\mathbf{x}^4 \simeq (1, \pm 3.9),$$

mais :

$$d_1(\mathbf{x}^3, \mathbf{x}^4) \simeq 5.6 \neq a_{34} = 2,$$

ce qui montre qu'aucun choix des coordonnées des points ne permet de générer la matrice donnée au départ.

## 2. UNE « BONNE » DISTANCE POUR LA MÉTHODE (G)

PROPOSITION : La méthode (G), appliquée avec la distance  $d_2$  donnée par :

$$d_2(x, y) = \max_i |x_i - y_i|,$$

permet de générer toute matrice euclidienne.

Preuve : Définissons  $\mathbf{x}^k \in R^{n-1}$  ( $k = 1, \dots, n$ ) par :

$$x_i^k = \begin{cases} a_{ik}, & \text{si } i < k, \\ 0, & \text{si } i = k, \\ \max(0, \max_{j>i} (a_{ij} - a_{kj})) & \text{si } i > k. \end{cases}$$

Nous allons montrer qu'on a bien :

$$\forall k, l \in [1, n], \quad d_2(\mathbf{x}^k, \mathbf{x}^l) = a_{kl}.$$

c'est-à-dire :

$$|x_i^k - x_i^l| \leq a_{kl}, \quad \forall i = 1, \dots, n-1,$$

avec l'égalité pour un certain  $i$ .

Notons  $y_{ikl} = x_i^k - x_i^l$  et supposons  $k < l$  :

Nous avons cinq cas à distinguer :

- (1)  $i < k < l$ ;
- (2)  $i = k < l$ ;
- (3)  $k < i < l$ ;
- (4)  $k < i = l$ ;
- (5)  $k < l < i$ .

Nous donnerons la démonstration des cas (1), (2) et (5), les autres cas se traitant de manière semblable.

- (1)  $i < k < l$  :

$$y_{ikl} = a_{ik} - a_{il},$$

comme  $a_{ik} \leq a_{il} + a_{lk}$ , il vient  $y_{ikl} \leq a_{lk}$  et comme  $a_{il} \leq a_{ik} + a_{kl}$ , il vient  $-y_{ikl} \leq a_{kl}$  donc  $|y_{ikl}| \leq a_{kl}$ ;

- (2)  $i = k < l$  :

$$y_{kkl} = 0 - a_{kl},$$

donc  $|y_{kkl}| = a_{kl}$  : égalité!

- (5)  $k < l < i$  :

$$y_{ikl} = \max(0, \max_{j>i} (a_{ij} - a_{kj})) - \max(0, \max_{j>i} (a_{ij} - a_{lj})).$$

Il faut ici encore distinguer trois sous-cas :

1<sup>er</sup> cas :  $x_i^k = 0$ , donc  $y_{ikl} \leq 0 \leq a_{kl}$ ;

2<sup>e</sup> cas :  $x_i^l = 0$ , donc  $a_{ij} - a_{lj} \leq 0, \forall j > i$ , or :

$$a_{ij} \leq a_{lj} \leq a_{lk} + a_{kj},$$

d'où :

$$x_i^k = \max(0, \max_{j>i} (a_{ij} - a_{kj})) \leq a_{lk};$$

3<sup>e</sup> cas :  $x_i^k > 0$  et  $x_i^l > 0$ , alors il existe  $s, t > i$ , tels que :

$$a_{is} - a_{ks} = x_i^k > 0 \quad \text{et} \quad a_{it} - a_{lt} = x_i^l > 0$$

et par définition de  $x_i^l$ , on a :

$$a_{it} - a_{it} \geq a_{is} - a_{is},$$

d'où :

$$y_{ikl} = x_i^k - x_i^l = a_{is} - a_{kj} - (a_{it} - a_{it}) \leq a_{is} - a_{ks} - (a_{is} - a_{is}) = a_{is} - a_{ks} \leq a_{lk},$$

même raisonnement pour  $-y_{ikl} = y_{ilk}$ .

Exemple : la matrice :

$$A = \begin{pmatrix} - & 2 & 3 & 4 \\ 2 & - & 5 & 4 \\ 3 & 5 & - & 2 \\ 4 & 4 & 2 & - \end{pmatrix}$$

peut être obtenue avec la distance  $d_2$  en définissant les points  $x^1, x^2, x^3$  et  $x^4 \in R^3$  de la manière suivante :

$$x^1 = (0, 2, 0), \quad x^2 = (2, 0, 0), \quad x^3 = (3, 5, 0), \quad x^4 = (4, 4, 2)$$

et on a bien :

$$a_{12} = d_2(x^1, x^2) = \max(2, 2, 0) = 2,$$

$$a_{13} = d_2(x^1, x^3) = \max(3, 3, 0) = 3,$$

$$a_{14} = d_2(x^1, x^4) = \max(4, 2, 2) = 4,$$

$$a_{23} = d_2(x^2, x^3) = \max(1, 5, 0) = 5,$$

$$a_{24} = d_2(x^2, x^4) = \max(2, 4, 2) = 4,$$

$$a_{34} = d_2(x^3, x^4) = \max(1, 1, 2) = 2.$$

Nous avons ainsi apporté la preuve que l'utilisation de la distance  $d_2$  nous permet de générer toute matrice euclidienne en utilisant la méthode (G), ce qui n'est pas vrai pour la distance euclidienne usuelle  $d_1$ .

## BIBLIOGRAPHIE

1. M. BRANDENBERGER, *Generalized Network Generator*, Institut pour l'Automation et la Recherche Opérationnelle, CH-1700 Fribourg, 1983.
2. N. CHRISTOFIDES, *The Travelling Salesman Problem*, Report 77-11, Dept. of Management Science, Imperial College, London, 1977.
3. F. MATTHÄUS, *Tourenplanung-Verfahren zur Einsatzdisposition von Fuhrparks*, S. Toeche-Mittler Verlag, Darmstadt, 1978.
4. D. J. ROSENKRANTZ, R. E. STEARNS et P. M. LEWIS, *Approximate Algorithms for the T.S.P.*, Proc. 15th I.E.E.E. Symp. on Switching and Automata Theory, 1974, p. 33.
5. M. H. J. WEBB, *Some Methods of Producing Approximate Solutions to Travelling Salesman Problems with Hundreds or Thousands of Cities*, Operational Research Quarterly, vol. 22, n° 1, 1971, p. 49-66.