



ELSEVIER

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

SCIENCE @ DIRECT®

C. R. Acad. Sci. Paris, Ser. I 336 (2003) 185–190



Probabilités/Statistique

## Un algorithme probabiliste de calcul d'approximations polynômiales sur un hypercube

### A Monte Carlo computation of polynomial approximations on a hypercube

Sylvain Maire

*ISITV, Université de Toulon et du Var, avenue G. Pompidou, BP 56, 83262 La Valette du Var cedex, France*

Reçu le 27 mars 2002 ; accepté après révision le 16 décembre 2002

Présenté par Marc Yor

---

#### Résumé

On décrit une méthode de Monte Carlo permettant un calcul itératif de l'approximation quadratique d'une fonction sur une base orthonormée quelconque. On l'applique à l'approximation de fonctions régulières sur un hypercube à l'aide de bases de polynômes orthogonaux multidimensionnels contenant peu d'éléments. L'algorithme constitue à la fois un outil d'approximation et d'intégration numérique. *Pour citer cet article : S. Maire, C. R. Acad. Sci. Paris, Ser. I 336 (2003).*

© 2003 Académie des sciences/Éditions scientifiques et médicales Elsevier SAS. Tous droits réservés.

#### Abstract

We describe a Monte Carlo method which enables an iterative computation of the  $L^2$  approximation of a function on any orthonormal basis. We use it for the approximation of smooth functions on an hypercube with the help of multidimensional orthogonal polynomial basis containing only few terms. The algorithm is both a tool for approximation and numerical integration. *To cite this article: S. Maire, C. R. Acad. Sci. Paris, Ser. I 336 (2003).*

© 2003 Académie des sciences/Éditions scientifiques et médicales Elsevier SAS. All rights reserved.

---

#### Abridged English version

We have developed in a previous work [6] a Monte Carlo method which enables an iterative computation of  $L^2$  approximations. In Section 1, we make a short description of this algorithm and we give its intrinsic properties in Lemmas 1.1 and 1.2. We have studied especially in this work polynomial approximations of monodimensional regular functions, for which a decay as  $C/k^L$  of the coefficients  $a_k$  on the approximation basis holds [2]. We have obtained Monte Carlo estimators for numerical integration with increased convergence rate.

---

Adresse e-mail : [mair@isitv.univ-tln.fr](mailto:mair@isitv.univ-tln.fr) (S. Maire).

In Section 2, we make a generalisation of this work in the multidimensional case by computing the approximations on polynomial basis which coefficients belong to a Korobov-like space [4]. This enables us to reduce the dimensional effect as with the use of lattice rules formulas. In Theorem 2.2, we obtain the same kind of error estimation as with these formulas for numerical integration, with an additional term due to the iterative method. There are two main advantages for our algorithm. First, no artificial constants due to periodisation [4] appears in our estimations. Second, we obtain not only an approximation of the integral of the function but an approximation of the function itself.

In Section 3, we give two numerical examples to show the efficiency of this algorithm. Example 1 shows that the method is comparable to an optimal one [1] for the numerical integration of regular functions (Fig. 1). Example 2 emphasizes the approximation properties of the algorithm and especially the diminishing of the dimensional effect (Table 1). More examples and applications can be found in [5].

**1. Introduction**

Nous avons développé dans un travail antérieur [6] une méthode de Monte Carlo permettant le calcul d’approximations quadratiques sur une base orthonormée quelconque dont nous rappelons brièvement le principe. On cherche à calculer sur  $D = [0, 1]^n$  les coefficients  $a_k = \langle f, e_k \rangle$  de l’approximation de  $f$  sur une base orthonormée de  $L^2(D)$  dont les éléments  $e_k$  sont bornés. On écrit

$$f(x) = \sum_{k=1}^p a_k e_k(x) + r(x)$$

avec  $a_k = \langle f, e_k \rangle$  et  $\langle r, e_k \rangle = 0$ . Leur approximation ainsi que celle de  $f$  sont obtenus à l’aide d’un échantillon  $X_i$  uniforme de taille  $N$  par

$$a_k^{(1)} = \frac{1}{N} \sum_{i=1}^N f(X_i) e_k(X_i), \quad f^{(1)}(x) = \sum_{k=1}^p a_k^{(1)} e_k(x).$$

On calcule ensuite une correction

$$b_k^{(1)} = \frac{1}{N} \sum_{i=1}^N (f(Y_i) - f^{(1)}(Y_i)) e_k(Y_i)$$

sur ces coefficients à l’aide de tirages  $Y_i$  indépendants des  $X_i$  et on obtient une nouvelle approximation des  $a_k$  et de  $f$  à l’étape 2 par

$$a_k^{(2)} = a_k^{(1)} + b_k^{(1)}, \quad f^{(2)}(x) = \sum_{k=1}^p a_k^{(2)} e_k(x).$$

En remplaçant  $f$  par son expression et en effectuant  $M$  étapes de l’algorithme précédent on obtient

$$a_k^{(M)} = \sum_{j=1}^p Q_{k,j}^{(M)} a_j + T_k^{(M)}, \quad f^{(M)}(x) = \sum_{k=1}^p a_k^{(M)} e_k(x).$$

Les expressions de  $Q_{k,j}^{(2)}$  et  $T_k^{(2)}$  sont données par

$$Q_{k,j}^{(2)} = \alpha_{k,j} + \beta_{k,j} - \sum_{s=1}^p \alpha_{s,j} \beta_{k,s}$$

et

$$T_k^{(2)} = R_k^{(1)}(1 - \beta_{k,k}) - \sum_{j \neq k} R_j^{(1)} \beta_{k,j} + R_k^{(2)}$$

avec

$$\alpha_{k,j} = \frac{1}{N} \sum_{i=1}^N e_k(X_i) e_j(X_i), \quad \beta_{k,j} = \frac{1}{N} \sum_{i=1}^N e_k(Y_i) e_j(Y_i)$$

et

$$R_k^{(1)} = \frac{1}{N} \sum_{i=1}^N e_k(X_i) r(X_i), \quad R_k^{(2)} = \frac{1}{N} \sum_{i=1}^N e_k(Y_i) r(Y_i).$$

Le terme  $\sum_{j=1}^p Q_{k,j}^{(M)} a_j$  représente l'estimation de  $a_k$  si  $r(x) \equiv 0$ . Le lemme suivant montre qu'il est sans biais et que la variance de ses composantes se réduit géométriquement si on prend suffisamment de tirages par étape.

**Lemme 1.1.** *Il existe des constantes  $C(p)$  et  $K(p) \leq p^2$ , ne dépendant que de la base d'approximation telles que*

$$E(Q_{k,j}^{(M)}) = \delta_{k,j}, \quad \text{Var}(Q_{k,j}^{(M)}) \leq \frac{K(p)^{M-1}}{N^M} C(p)^M.$$

Le terme  $T_k^{(M)}$  représente le terme lié à l'erreur de troncature. Le lemme suivant en donne un contrôle en fonction justement de  $r(x)$ .

**Lemme 1.2.** *Il existe des constantes  $\gamma(p)$  et  $\gamma_1(p)$  telles que*

$$\text{Var}(T_k^{(2)}) \leq 2 \left[ \frac{\gamma(p)}{N} + \frac{p^2}{N^2} \gamma_1(p) \right] \int_D r^2(x) dx.$$

Nous avons alors appliqué l'algorithme en dimension un pour des fonctions de classe  $C^L$ . Le cadre naturel est une décroissance en  $C/k^L$  des  $a_k$  qui est vérifiée pour la base de Fourier périodisée [4], les bases de polynômes de Legendre et de Tchebychev [2]. En effectuant une étude précise des constantes  $C(p)$  et  $K(p)$ , en utilisant cette décroissance et les deux lemmes précédents nous avons montré la convergence de l'algorithme et que la vitesse de convergence des estimateurs obtenus était d'ordre  $1/N^{L-0.5-\varepsilon}$  pour l'intégration numérique. Pour la base de Tchebychev normalisée  $\tilde{T}_j(x)$  le calcul de  $\langle f, \tilde{T}_j \rangle$  se fait sur  $[-1, 1]$  par l'approximation de Monte Carlo de  $E(\pi f(V) \tilde{T}_j(V))$  où  $V$  a pour densité  $1/\pi \sqrt{1-v^2} 1_{[-1,1]}(v)$ . Cette approche a donné les meilleurs résultats numériques en termes de précision et de minimisation du nombre d'étapes.

## 2. Généralisation au cadre multidimensionnel

Nous allons nous placer dans le cas d'une fonction  $f \in C^L([-1, 1]^n)$ . Le développement de  $f$  sur une base de polynômes orthogonaux normalisés s'écrit pour  $x$  appartenant à  $D = [-1, 1]^n$  sous la forme

$$f(x) = \sum_{m \in \mathbb{N}^n} a_m e_m(x) = \sum_{m \in \mathbb{N}^n} a_m P_{m_1}(x_1) P_{m_2}(x_2) \cdots P_{m_n}(x_n),$$

où  $P_{m_j}(x_j)$  désigne le  $m_j$ ème polynôme orthonormé par rapport à la variable  $x_j$ . Les coefficients  $a_m$  de la projection orthogonale de  $f$  sur les polynômes orthogonaux de Legendre ou de Tchebychev vérifient

$$|a_m| \leq \frac{C_1}{(\widehat{m}_1 \widehat{m}_2 \cdots \widehat{m}_n)^L},$$

où  $C_1$  est une constante positive ne dépendant que de  $f$  et où  $\widehat{m} = \sup(1, m)$ . Ceci s'obtient en appliquant les équations différentielles vérifiées par ces polynômes [2] à chaque composante de  $e_m(x)$ . Les coefficients  $a_m$  plus grands sont donc ceux pour lesquels le produit  $\widehat{m}_1 \widehat{m}_2 \cdots \widehat{m}_n$  est le plus petit possible. On définit donc de façon analogue aux espaces de Korobov [4]

$$W_{n,d} = \{m \in \mathbf{N}^n \mid (\widehat{m}_1 \cdots \widehat{m}_n) \leq d\}, \quad S_{n,d} = \{m \in \mathbf{N}^n \mid (\widehat{m}_1 \cdots \widehat{m}_n) \leq d\}$$

et on écrit la fonction  $f$  sous la forme

$$f(t) = \sum_{m \in W_{n,d}} a_m e_m(t) + r(t).$$

La majoration de  $r(t)$  s'obtient dans le lemme suivant en remarquant que  $\text{Card}(S_{n,j}) \leq 2^n d(j)^n$  et que  $d(j)$  croît moins vite que toute puissance de  $j$  [3].

**Lemme 2.1.** *Sous les hypothèses précédentes,  $\forall \varepsilon > 0$ , il existe une constante  $C_{n,\varepsilon}$  ne dépendant que de  $n$  et de  $\varepsilon$  telle qu'on ait la majoration*

$$\int_D r(t)^2 dt \leq \frac{C_{n,\varepsilon}}{d^{2L-1-\varepsilon}}.$$

Nous pouvons maintenant donner dans le théorème suivant les performances de l'algorithme dans le calcul de l'approximation de  $f$  par  $f^{(M)}(t) = \sum_{m \in W_{n,d}} a_m^{(M)} e_m(t)$  qui comporte  $L_{n,d} = \text{Card}(W_{n,d})$  éléments. La preuve et les notations sont similaires au cas monodimensionnel [6] avec un contrôle du reste basé cette fois-ci sur le Lemme 2.1.

**Théorème 2.1.** *Sous les hypothèses précédentes et si*

- (i)  $L_{n,d}/N < 1$ ,
- (ii)  $\tau = \sup(\sup_k \int_D (1 - e_k^2(x))^2 dx, \sup_{j,k, j \neq k} \int_D e_k^2(x) e_j^2(x) dx) \leq N/4$ ,

on a

$$E(a_m^{(M)}) = a_m, \quad \text{Var}(a_m^{(M)}) \leq 2 \left( \mu_1 \frac{K(L_{n,d})^{M-1}}{N^M} C(L_{n,d})^M + \mu(L_{n,d}) \frac{1}{d^{2L-1-\varepsilon}} \right)$$

et aussi l'estimation de l'erreur quadratique entre  $f$  et  $f^{(M)}$

$$E \left( \int_D (f(x) - f^{(M)}(x))^2 dx \right) \leq 2L_{n,d} (\text{Var}(a_m^{(M)})) + \frac{\mu_2}{d^{2L-1-\varepsilon}}.$$

La majoration de  $\text{Var}(a_m^{(M)})$  comporte deux termes. Le premier décroît géométriquement si on prend suffisamment de tirages par étape. Il ne subsistera alors plus qu'une erreur en  $O(1/d^{L-0.5-\varepsilon})$  dans l'approximation de  $a_m$  par  $a_m^{(M)}$ .

### 3. Résultats numériques

De nombreux exemples et applications peuvent être consultés dans [5]. Nous allons en donner deux en utilisant des polynômes de Tchebychev pour bien montrer les performances de l'algorithme aussi bien en termes

d'intégration numérique que d'approximation. Pour calculer une valeur approchée de  $I(f) = \int_D f(x) dx$ , soit on calcule

$$\tilde{I}(f) = \sum_{k \in W_{n,d}} a_k^{(M)} \int_D e_k(x) dx,$$

soit on utilise la méthode des variables de contrôles avec  $f^{(M)}$  comme approximation de  $f$ . Cette deuxième méthode se fera avec 10 000 tirages et la valeur approchée de  $I(f)$  sera notée  $\bar{I}(f)$ .

**Exemple 1.** Nous allons tracer respectivement dans res1f1 et res2f1 (Fig. 1) :

$$-\log_{10} \frac{|I - \tilde{I}|}{|I|}, \quad -\log_{10} \frac{|I - \bar{I}|}{|I|}$$

en fonction du  $\log_{10}$  du nombre de tirages pour  $f_1(x, y, z, t) = \exp(x + y + z + t)$ .

Les résultats montrent que la précision sur l'intégrale recherchée est de 7 à 8 chiffres significatifs pour un nombre de tirages de l'ordre de 40 000. Ceci est tout à fait comparable aux résultats tirés de [1] où la méthode utilisée atteint un ordre optimal. Une valeur approchée de la pente de la courbe (et donc de l'ordre de la méthode) est environ 1,4 ce qui est nettement plus que 0,5 pour la méthode de Monte Carlo classique.

**Exemple 2.** Nous prenons maintenant dans le Tableau 1 l'exemple de la fonction  $f(x, y, z, t, u, v) = \exp((x + y + z + t + u + v)/6)$  dont l'intégrale vaut environ 1,66 et la variance  $3,9 \times 10^{-2}$ .

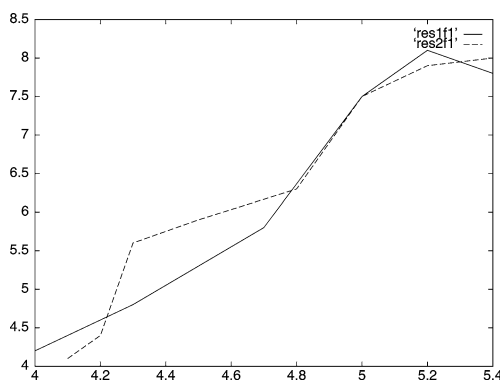


Fig. 1. res1f1, res2f1 en fonction du  $\log_{10}$  du nombre de tirages.

Fig. 1. res1f1, res2f1 as functions of  $\log_{10}$  the number of draws.

Tableau 1

Exemple 2 :  $f(x, y, z, t, u, v) = \exp((x + y + z + t + u + v)/6)$

Table 1

Example 2:  $f(x, y, z, t, u, v) = \exp((x + y + z + t + u + v)/6)$

$d$	$L_{6,d}$	$ I - \tilde{I} $	$\sigma_{f-f^{(M)}}^2$	$ I - \bar{I} $
2	256	$6,0 \times 10^{-6}$	$4,8 \times 10^{-9}$	$1,7 \times 10^{-6}$
3	448	$8,2 \times 10^{-7}$	$2,6 \times 10^{-10}$	$8,0 \times 10^{-8}$
5	1072	$6,0 \times 10^{-9}$	$6,5 \times 10^{-14}$	$2,0 \times 10^{-9}$

Ceci illustre parfaitement les qualités de l'algorithme en termes d'approximation et de réduction de variance. En ce qui concerne la réduction de l'effet dimensionnel, le choix des éléments à conserver dans  $W_{6,5}$  a permis par exemple de diminuer le nombre de fonctions de base de  $46\,656 = 6^6$  à 1072.

## Références

- [1] E.I. Atanassov, I.T. Dimov, A new optimal Monte Carlo method for calculating integral of smooth functions, *Monte Carlo Methods Appl.* 5 (2) (1999) 149–167.
- [2] C. Bernardi, Y. Maday, *Approximations spectrales de problèmes aux limites elliptiques*, Springer-Verlag, 1992.
- [3] G.H. Hardy, E.M. Wright, *An Introduction to the Theory of Numbers*, Fifth edition, Oxford University Press, 1979.
- [4] A.R. Krommer, C.W. Ueberhuber, *Computational Integration*, SIAM, 1998.
- [5] S. Maire, Réduction de variance pour l'intégration numérique et pour le calcul critique en transport neutronique, Thèse de doctorat de l'université de Toulon, 2001.
- [6] S. Maire, Reducing variance using iterated control variates, Preprint 2001 du laboratoire ANAM, accepté à *J. Statist. Comput. Simulation*, 2002.