Principe de la méthode des A-splines récurrentes pour l'interpolation et la compression des signaux

A Method for the Development of Recurrent A-Splines for Interpolation and Compression of Signals

Avraam SEREDINSKI

55 rue Servan
75011 Paris

A. Seredinski est ingénieur en Radiotechnique de l'Institut Electrotechnique de Saint Pétersbourg et Docteur ès sciences techniques de l'Institut Central de Recherches scientifiques de Télévision de St Pétersbourg.
Actuellement en année sabatique, il exerce comme professeur invité au Département IMAGES de Télécom-Paris.

RÉSUMÉ

Un algorithme est développé permettant de réaliser le calcul des coefficients de la fonction spline cubique « en ligne » (en temps réel). Les conditions de stabilité de l'algorithme sont déterminées. On montre comment les splines récurrentes (« A-splines ») peuvent être appliquées pour l'interpolation de données discrètes.

En ajoutant une procédure non linéaire (une comparaison avec un seuil) on peut obtenir une compression importante des signaux de provenance arbitraire.

Les A-splines récurrentes peuvent être utilisées pour le traitement, le transfert et la mémorisation de différents signaux (vidéo, parole, télédétection, etc...).

MOTS CLÉS

Spline, traitement récurrent, « traitement en ligne », temps réel, interpolation, compression.

ABSTRACT

An algorithm is developed for on-line calculation of the coefficients of the cubic spline-function. The stability conditions of this algorithm are established. It is shown how the recurrent A-splines can be applied for the interpolation of discrete data.

With the addition of a non-linear procedure (a comparison with a threshold), a significant compression of signals can be realized for signals of arbitrary nature.

The recurrent A-splines can be used for the processing, transfer and storage of different signals (video, speech, radar etc.).

KEY WORDS

Spline, recurrent processing, « on-line », real-time, interpolation, compression.

Pour de nombreux problèmes de traitement et de transmission de l'information, il arrive qu'il soit nécessaire d'éliminer la redondance statistique puis ensuite de la reconstituer afin de recréer le message original. De plus, le signal initial peut être connu par ses propriétés statistiques (c'est par exemple le mélange d'un signal et de bruit). Nous exposons ici une des variantes possibles de la résolution de ces problèmes, fondée sur l'exploitation de splines spéciales (nous les nommerons A-splines) qui se distinguent des autres splines connues — les B-splines [1], Bêta-splines [2], splines classiques [3], etc... — par la possibilité de calculer leurs coefficients de manière récurrente.

1. Notations et définitions

Nous allons supposer que le signal est représenté par une séquence d'échantillons discrets issus d'un signal initial analogique à des intervalles fixés (par exemple aux intervalles de Nyquist-Kotelnikoff), bien que l'égalité des intervalles ne soit pas nécessaire pour la suite des développements.

Par spline (définie classiquement en [3]), nous entendons une fonction polynômiale par morceaux $S(x)$ d'une variable x ; la spline est représentée sur chaque intervalle i par un polynôme de degré m :

$$S_i(x) = \sum_{k=0}^{m} a_k^{(i)} . (x - x_i)^k . \quad (1)$$

Le nombre des intervalles est déterminé par une échelle H de la variable x

$$H : x_0 < x_1 < \cdots < x_N \quad (2)$$

et les conditions de continuité de la spline et de ses dérivées jusqu'à l'ordre $(m - D)$ sont satisfaites aux extrémités des intervalles (D est le défaut de la spline).

$$\frac{d^j}{dx^j} S_{i-1}(x_i) = \frac{d^j}{dx^j} S_i(x_i) \quad (3)$$

$i = 1, 2, \ldots, N - 1$

$j = 0, 1, \ldots, m - D$.

2. Algorithme récurrent de calcul de la spline interpolante

En développant les conditions (3) pour $m = 3$ et $D = 1$, on aboutit au système d'équations bien connues :

$$\begin{aligned} &a)\ a_0^{(i-1)} + a_1^{(i-1)} h_{i-1} + a_2^{(i-1)} h_{i-1}^2 + a_3^{(i-1)} h_{i-1}^3 = a_0^{(i)} \\ &b)\ a_1^{(i-1)} + 2\, a_2^{(i-1)} h_{i-1} + 3\, a_3^{(i-1)} h_{i-1}^2 = a_1^{(i)} \qquad (4) \\ &c)\ a_2^{(i-1)} + 3\, a_3^{(i-1)} h_{i-1} = a_2^{(i)} \end{aligned}$$

$$i = 1, 2, \ldots, N - 1 ; \quad h_i = x_i - x_{i-1}$$

dont la résolution donne simultanément tous les coefficients dans tous les intervalles pour un ensemble de données $\{f(x_i)\}$ $(i = 0, \ldots, N)$ quand deux conditions limites sont appliquées aux extrémités de l'échelle H.

Mais cette méthode du calcul des coefficients de la spline ne convient pas à ce cas-là où seules les valeurs précédentes sont connues à chaque instant et toutes les valeurs futures sont encore inconnues (comme, par exemple, en transmission d'information).

C'est pourquoi il est logique de se poser la question suivante : comment peut-on calculer les coefficients de la spline sur l'intervalle courant i si on ne connaît que la valeur future $f(x_{i+1})$ et toutes les valeurs précédentes (c'est-à-dire tous les coefficients de la spline sur tous les intervalles précédents).

A première vue, la solution est assez simple. En effet, comme la spline interpolante doit prendre les valeurs $f(x_i)$ aux points x_i $(i = 0, \ldots, N)$ l'équation (1) donne immédiatement :

$$a_0^{(i)} = f(x_i) \quad (i = 0, 1, \ldots, N) . \quad (5)$$

En supposant que la spline a pour défaut $D = 1$ (c'est-à-dire que ses première et deuxième dérivées sont continues aux points de conjugaison des intervalles), on peut calculer les coefficients $a_1^{(i)}$ et $a_2^{(i)}$ d'après les équations (4 b) et (4 c). Ainsi il reste à trouver $a_3^{(i)}$ de manière à ce que la spline passe par le point $f(x_{i+1})$. Pour cela il faut résoudre l'équation (4 a) d'inconnue $a_3^{(i)}$. On obtient ainsi :

$$a_3^{(i)} = (1/h_i^3)(a_0^{(i+1)} - a_0^{(i)} - a_1^{(i)} h_i - a_2^{(i)} h_i^2) . \quad (6)$$

Mais cet algorithme qui semble très simple n'est pas stable, c'est-à-dire que la spline peut prendre des valeurs infinies entre les points de données bien qu'elle passe par tous ces points et qu'il garde la continuité de la spline et de ses dérivées jusqu'à l'ordre 2.

On peut montrer facilement et rigoureusement que cette procédure de calcul des coefficients de la spline pour $m = 3$ et $D = 1$ n'est pas stable. On utilise pour cela la théorie de la transformation en Z [4] en substituant au système (4) sa transformée en Z pour $h_i = h =$ constante :

$$\begin{aligned} &a)\ b_1 = b_1 z^{-1} + 2\, b_2 h z^{-1} + 3\, b_3 h^2 z^{-1} \\ &b)\ b_2 = b_2 z^{-1} + 3\, b_3 h z^{-1} \qquad (7) \\ &c)\ b_3 = (b_0 z - b_0 - b_1 h - b_2 h^2)/h^3 \end{aligned}$$

De ce système, en éliminant b_1 et b_2 dans l'équation (7 c) on tire :

$$b_3 h^3 (1 + 4 z^{-1} + z^{-2}) = b_0 (z - 1)(1 - z^{-1})^2 \quad (8)$$

l'une des racines de l'équation $1 + 4 z^{-1} + z^{-2} = 0$ est supérieure à 1 (en valeur absolue), d'où l'instabilité de cet algorithme.

3. Principe des A-splines interpolantes

Pour garantir la stabilité de l'algorithme, il faut appliquer une condition locale sur la conduite de la A-spline à l'intérieur de chaque intervalle d'interpolation. Par exemple, on peut imposer la minimisation d'une fonctionnelle :

$$K_i = \int_{x_i}^{x_{i+1}} [A_i''(x)]^2 \, dx \Rightarrow \min (K_i) ,$$

$$i = 0, 1, \ldots, N - 1 . \quad (9)$$

Cette exigence [5] est équivalente à la faculté de minimisation d'énergie potentielle sur chaque intervalle et ceci concorde avec le théorème de Holladay [6] qui exprime la même caractéristique de la spline classique construite sur l'échelle entière.

En développant l'expression (9) on obtient :

$$\mathrm{K}_i = \int_{x_i}^{x_{i+1}} [2\, a_2^{(i)} + 6\, a_3^{(i)}(x - x_i)]^2\, dx$$
$$= 4\, h_i [(a_2^{(i)})^2 + 3\, a_2^{(i)} a_3^{(i)} h_i + 3(a_3^{(i)})^2 h_i^2]$$
$$\Rightarrow \min\ (\mathrm{K}_i)\,. \qquad (10)$$

Désormais, on a à notre disposition 4 conditions indépendantes (le système (4) et l'équation (10)) pour 3 coefficients ($a_1^{(i)}$, $a_2^{(i)}$, $a_3^{(i)}$), la solution exacte dans le cas général n'existe pas ; c'est pourquoi il faut écarter l'une des conditions. Naturellement, les conditions de continuité de la spline elle-même et de sa première dérivée sont obligatoires, et la condition (9) remplace la condition (4 c) exprimant la continuité de la dérivée seconde.

L'équation (4 b) est gardée pour le coefficient $a_1^{(i)}$, et pour trouver $a_2^{(i)}$ et $a_3^{(i)}$ il est nécessaire de résoudre simultanément les équations (6) et (10). Si on note la première différence relative :

$$f(x_i, x_{i+1}) = [f(x_{i+1}) - f(x_i)]/h_i \qquad (11)$$

on obtient :

$$\frac{d\mathrm{K}_i}{da_2^{(i)}} = 4\, \{3[a_1^{(i)} - f(x_i, x_{i+1})] + 2\, a_2^{(1)} h_i\} = 0 \quad (12)$$

d'où :

$$a_2^{(i)} = \frac{3}{2\, h_i}\, [f(x_i, x_{i+1}) - a_1^{(i)}] \qquad (13)$$

et après une substitution en (6) on obtient :

$$a_3^{(i)} = \frac{-1}{2\, h_i^2}\, [f(x_i, x_{i+1}) - a_1^{(i)}]\,. \qquad (14)$$

La deuxième dérivée partielle est :

$$\frac{d}{da_2^{(i)}} \frac{d\mathrm{K}_i}{da_2^{(i)}} = 8\, h_i > 0 \qquad (15)$$

c'est-à-dire que $a_2^{(i)}$ et $a_3^{(i)}$ calculés selon les expressions (13) et (14) garantissennt le minimum de l'intégrale K_i ; la possibilité de calcul de spline récurrente satisfaisant une condition locale est alors démontrée. Nous appellerons les splines de ce type les A-splines.

Il faut cependant justifier la stabilité de cet algorithme. Nous employons à nouveau la transformation en Z pour les 3 expressions du calcul des coefficients $a_1^{(i)}$, $a_2^{(i)}$, $a_3^{(i)}$ (à savoir les formules (4 b), (13) et (14)). En supposant $h_i = h =$ constante on obtient le système des images dans le plan Z :

$$\begin{aligned} b_1 &= b_1 z^{-1} + 2\, b_2 h z^{-1} + 3\, b_3 h^2 z^{-1} \\ b_2 &= (3/2\, h)\, [(b_0 z - b_0)/h - b_1] \\ b_3 &= -\,(1/2\, h^2)\, [(b_0 z - b_0)/h - b_1] \end{aligned} \qquad (16)$$

d'où :

$$b_1 h (1 + z^{-1}/2) = (3\, b_0/2)\, (1 - z^{-1})\,. \qquad (17)$$

La racine unique de l'équation :

$$1 + z^{-1}/2 = 0 \qquad (18)$$

est $z = -\,1/2$, le module de cette racine est inférieure à 1, la stabilité est démontrée.

4. Algorithme généralisé des A-splines interpolantes

Une condition locale appliquée à la conduite de la spine en sauvegardant sa stabilité ne doit pas toujours être limitée à celle décrite par l'expression (9). On peut par exemple s'imposer la minimisation de la fonctionnelle suivante [7] :

$$\mathrm{L}_i = \int_{x_i}^{x_{i+1}} [\mathrm{A}_i'(x)]^2\, dx \Rightarrow \min\ (\mathrm{L}_i)\,. \qquad (19)$$

En menant des calculs identiques à ceux que nous avons exposés précédemment, on arrive à :

$$a_2^{(i)} = \frac{9}{4\, h_i}\, [f(x_i, x_{i+1}) - a_1^{(i)}] \qquad (20)$$

$$a_3^{(i)} = \frac{-5}{4\, h_i^2}\, [f(x_i, x_{i+1}) - a_1^{(i)}]\,. \qquad (21)$$

Le passage par la méthode de la transformée en Z montre que cet algorithme est également stable.

On peut remarquer que les coefficients indépendants $a_2^{(i)}$ et $a_3^{(i)}$ des équations (13), (14) et (20), (21) respectivement ne se distinguent que par des facteurs numériques multiplicatifs. Il est alors logique de se poser la question suivante : dans quelles limites peut-on modifier ces facteurs numériques de sorte que l'algorithme de calcul des coefficients de la A-spline reste encore stable ? La réponse à cette question permettrait d'obtenir un ensemble d'algorithmes indépendemment de la condition locale concrère exprimée en (9) et (19).

Pour trouver cette réponse, il faut rappeler que la A-spline doit avoir la valeur $a_0^{(i+1)}$ au point x_{i+1} :

$$A_i(x_{i+1}) = a_0^{(i+1)} \qquad (22)$$

où

$$a_0^{(i+1)} - a_0^{(i)} = a_1^{(i)} h_i + a_2^{(i)} h_i^2 + a_3^{(i)} h_i^3 \qquad (23)$$

soit, si nous utilisons la notation (11) :

$$f(x_i, x_{i+1}) - a_1^{(i)} = a_2^{(i)} h_i + a_3^{(i)} h_i^2\,. \qquad (24)$$

Cette égalité doit être strictement réalisée à gauche de chaque valeur de départ. Nous cherchons $a_2^{(i)}$ et $a_3^{(i)}$

vérifiant cette égalité, et en introduisant une constante réelle C, on peut écrire :

$$(1 - C + C)\,[f(x_i, x_{i+1}) - a_1^{(i)}] = a_2^{(i)} h_i + a_3^{(i)} h_i^2 . \quad (25)$$

Dans ce cas, la résolution conduit à :

$$a_2^{(i)} = \frac{C}{h_i}\,[f(x_i, x_{i+1}) - a_1^{(i)}] \quad (26)$$

$$a_3^{(i)} = \frac{(1 - C)}{h_i^2}\,[f(x_i, x_{i+1}) - a_1^{(i)}] . \quad (27)$$

Pour trouver l'ensemble des valeurs C pour lesquelles l'algorithme reste stable, il faut encore se reférer à la transformation en Z. En laissant passer les calculs intermédiaires, on montre que l'équation dont la racine détermine la stabilité de l'algorithme est de la forme :

$$z + 2 - C = 0 . \quad (28)$$

Il est alors évident que l'éventail qui garantit la stabilité est :

$$1 < C < 3 . \quad (29)$$

Le point qui se trouve au milieu de l'intervalle de stabilité (C = 2) possède une propriété remarquable : la A-spline devient locale. En effet, si nous calculons dans ce cas le coefficient $a_1^{(i)}$ d'après l'expression (4 *b*), on trouve en y reportant les équations (26) et (27) :

$$a_1^{(i)} = f(x_{i-1}, x_i) . \quad (30)$$

Le coefficient $a_1^{(i)}$ ne se détermine donc que par la différence divisée de l'intervalle précédent.

Dans le cas général ($1 < C < 3$), on peut établir les relations de récurrences de la détermination de $a_1^{(i)}$, $a_2^{(i)}$, $a_3^{(i)}$ en utilisant les équations (4 *b*), (26) et (27). De plus, il faut fournir une condition initiale sur le coefficient $a_1^{(0)}$. Ensuite, au fil des calculs, on reçoit :

$$a_1^{(i)} = (3 - C) \sum_{k=0}^{i-1} (C - 2)^{i-k-1} f(x_k, x_{k+1}) + (C - 2)^i\, a_0^{(i)} . \quad (31)$$

On voit que l'influence de la condition initiale s'amortit au fur et à mesure de l'accroissement du nombre de pas i.

5. Application des A-splines à l'approximation des signaux

On peut utiliser la méthode de calcul récurrent des coefficients des A-splines pour la représentation réduite des signaux (cette procédure peut être considérée comme une approximation ou comme une réduction de redondance statistique). On suppose pour cela que le signal d'entrée est représenté par une succession discrète d'échantillons d'un signal analogique initial recueillie à la sortie d'un échantillonneur idéal à la fréquence déterminée par le théorème de Nyquist-Kotelnikoff.

Nous posons le problème de la manière suivante : comment peut-on trouver la quantité minimale d'échantillons d'entrée à garder dans la mémoire d'un ordinateur (ou à passer par un canal de télécommunications) de telle sorte que cette quantité soit suffisante pour interpoler l'ensemble des données par une A-spline et garantisse un écart limite fixé à l'avance entre le signal reconstitué et le signal initial ?

En utilisant comme base l'algorithme de calcul des A-splines récurrentes, il est facile de créer l'algorithme de choix des échantillons de contrôle du signal d'entrée. Soit l'échantillon (i), disposé à l'extrême droite de l'intervalle ($i - 1$), qui vient d'être choisi comme point de contrôle. (C'est-à-dire que la valeur de l'argument x_i et de la fonction $f(x_i)$ sont stockés en mémoire (ou passés par le canal de télécommunication.)) Ainsi les coefficients $a_0^{(i)}$ et $a_1^{(i)}$ pour l'intervalle suivant (i) sont déjà connus d'après les conditions démontrées plus haut — la spline doit passer par le point donné (équation (5)) et la première dérivée doit être continue (équation (4 *b*)). On choisit le point ($i + 2$) comme extrémité droite de l'intervalle (i) et $f(x_{i+2})$ comme valeur à atteindre au bout de cet intervalle. On calcule alors les coefficients $a_2^{(i)}$ et $a_3^{(i)}$ selon les formules (26) et (27), puis la valeur de la A-spline au point x_i. On compare $f(x_{i+1})$ et $A(x_{i+1})$; si la différence entre ces deux valeurs ne dépasse pas la tolérance Epsilon (dans la métrique choisie : linéaire, quadratique, etc...) cela signifie que l'échantillon $f(x_{i+1})$ peut être oublié sans porter préjudice à la qualité du signal reconstitué. Là, on

franchit x_{i+2} et on va au point x_{i+3} : on calcule les coefficients $a_2^{(i)}$ et $a_3^{(i)}$ en substituant $f(x_{i+3})$ à $f(x_{i+2})$ dans les formules (26) et (27). On compare $f(x_{i+1})$, $f(x_{i+2})$ aux valeurs de la nouvelle A-spline $A(x_{i+1})$ à $A(x_{i+2})$. Si les différences ne dépassent pas la tolérance Epsilon, on va au point x_{i+4}, etc... jusqu'à la valeur de l'argument x_{i+k} où au moins l'une des différences dépasse Epsilon. Il est alors nécessaire de reculer d'un pas et de fixer $f(x_{i+k-1})$ comme échantillon de contrôle suivant. On répète la procédure pour les points suivants : on construit la A-spline au point $f(x_{i+k+1})$, etc... Finalement le nouvel ensemble d'échantillons est réduit à une quantité de points initiaux d'autant plus faible que la tolérance Epsilon est grande.

C'est évident que cet algorithme peut être appliqué pour un signal de n'importe quelle nature : le son, l'image, la télémécanique, etc...

Un exemple de l'utilisation de cete algorithme pour la compression d'image est montré sur les figures 1, 2 et 3.

Figure 1 représente une image originale des dimensions 512×512 pixels quantifiés à 256 niveaux, c'est-à-dire 8 bit/pixel.

Les échantillons choisis par l'algorithme sont montrés sur la figure 2. Le paramètre est $C = 1,5$; celà répond à la condition (9). La tolérance est Epsilon = 7 dans la métrique linéaire. À ces conditions l'algorithme a choisi 121 350 échantillons « nœuds ». Comme la longueur des intervalles entre les « noœuds » voisins est une valeur variable, la quantité moyenne des bits sur chaque pixel est égal à 5,09 bit/pixel.

Figure 3 montre le résultat de l'interpolation des échantillons « nœuds ». La différence entre l'image originale et l'image reconstituée dans la métrique quadratique est PSNR = 38 dB.

6. Conclusion

Il est démontré que les splines récurrentes (A-splines) sont utilisables pour le traitement des signaux en temps réel. Elles peuvent donc être à la base des systèmes de stockage de données compressées et de leur transmission par le canal multiplexé de télécommunications. Le coefficient de compression peut être assez important et dépend de la tolérance fixée a priori.

On peut s'attendre à une augmentation des performances si on généralise cet algorithme au cas bidimensionnel.

REMERCIEMENTS

Je suis très reconnaissant envers le professeur Henri Maître pour la possibilité de travailler au département IMAGES de TELECOM-PARIS. Je tiens à remercier également Christophe Lepage pour sa programmation et son aide à la correction du texte en français de cet article.

Manuscrit reçu le 12 mars 1991.

BIBLIOGRAPHIE

[1] P. E. BEZIER, *Courbes et surfaces*, Mathématiques et CAO, vol. 14, Hermès Publishing, 1986.

[2] B. A. BARSKY, *Computer graphics and geometric modelling using Beta-splines*, Springer-Verlag, 1988.

[3] J. H. AHLBERG, E. N. NILSON, J. L. WALSH, *The theory of splines and their applications,* Academic Press, New York, 1967.

[4] A. V. OPPENHEIM, A. S. WILLSKY, I. T. YOUNG, *Signals and systems,* Prentice Hall International, London, 1983.

[5] L. M. GOLDENBERG, A. V. SEREDINSKI, *Discrétisation et reconstruction des signaux à base de fonctions splines* (article en russe), Vychislitelnie sredstva w technike i systémakh sviazi, Moscou, Sviaz, 1980, n° 5, pp. 25 à 31.

[6] J. C. HOLLADAY, *Smoothest curve approximation,* Math. Tables Aids Comput., 1957, pp. 233 à 243.

[7] L. M. GOLDENBERG, V. E. KLIMONTOV, V. SEREDINSKI, *Les méthodes récurrentes de construction des fonctions splines interpolantes* (article en russe), Avtomatika i télémechanika, 1979, n° 3, pp. 173 à 176.

[8] A. V. SEREDINSKI, V. A. VANDE-KIRKOV, I. A. BOUZDALINA, I. K. OKOUNEVA, M. I. ORLOVA, A. I. SOLONINA, *Application des fonctions splines récurrentes au traitement des signaux vidéo et de parole* (article en russe), Electrosviaz, 1982, n° 2, pp. 60 à 64.