Approche hiérarchique pour l'optimisation de la précision des systèmes de traitement du signal utilisant l'arithmétique virgule fixe

Karthick Parashar, Olivier Sentieys, Daniel Ménard

Approche hiérarchique pour l'optimisation de la précision des systèmes de traitement du signal utilisant l'arithmétique virgule fixe

Karthick PARASHAR, Olivier SENTIEYS, Daniel MENARD

IRISA/INRIA - Université de Rennes 1
6 rue de kerampont, BP 80518, 22305, Lannion, France
sentieys@irisa.fr

Résumé – La conversion en virgule fixe est souvent traitée comme un problème d'optimisation recherchant un compromis entre la précision et le coût de l'implémentation. La complexité du problème croit de façon exponentielle avec le nombre de variables à optimiser. Cet article propose une approche hiérarchique basée sur le principe de diviser pour régner. Une approche mixte combinant la simulation et une technique analytique est utilisée pour déterminer les performances du système. Le modèle de source de bruit unique pour caractériser le comportement en précision finie d'un sous-système est proposé.

Abstract – Fixed-point conversion is often solved as an optimization problem where the precision is traded to gain in the implementation cost. The complexity of the problem is known to grow exponentially with more optimizable variables. This paper proposes a divide and conquer technique to solve the growing size of the problem. This paper introduces the single noise source model based on which the proposed technique is built. A mixed approach for error propagation is also explained keeping in view of the elements in the circuit that cannot be handled analytically.

1 Introduction

Une implémentation optimisée est un facteur important de différenciation entre le succès ou l'échec des systèmes embarqués et des appareils électroniques modernes. L'arithmétique virgule fixe est donc souvent préférée à la virgule flottante pour des raisons évidentes de coût d'implémentation en termes de consommation d'énergie, de coût ou de performance. Il en découle par contre un problème complexe d'optimisation dans lequel un compromis entre la perte maximale de précision permettant de conserver les performances de l'application dans une limite acceptable, et une réduction du coût du système est recherché. Cependant, l'utilisation de la virgule fixe vient avec une pénalité sur le temps de conception et des outils pour automatiser ce processus sont donc requis.

L'objectif d'une méthodologie de conversion en virgule fixe est d'optimiser les largeurs des opérations $\mathbf{W_D}$ (et des variables qui en découlent) par une minimisation du coût d'implémentation $C(\mathbf{W_D})$ tout en garantissant que les performances du système $\lambda(\mathbf{W_D})$ correspondent à un critère applicatif minimum λ_{obj}.

$$\min\left(C\left(\mathbf{W_D}\right)\right) \quad \text{such as} \quad \lambda\left(\mathbf{W_D}\right) \geq \lambda_{obj} \tag{1}$$

Ce processus de détermination des largeurs est itératif et nécessite l'évaluation du coût d'implémentation et des performances de l'application à chaque itération. Par conséquent, le challenge principal de ce problème d'optimisation consiste en une évaluation, rapide et précise, des performances du système λ en fonction de la largeur des opérations $\mathbf{W_D}$.

Cette évaluation des performance est habituellement accomplie selon deux approches : par simulation ou de façon analytique. La méthode classique consiste à utiliser des simulations « au bit près » *(bit-true)* en virgule fixe [1, 2]. Ces simulations peuvent être effectuées au niveau système avec des environnements tels que Matlab-Simulink (Mathworks) ou des bibliothèques telles que des classes *SystemC* [3] ou d'autres classes C++ [4]. Si les méthodes basées sur la simulation peuvent être appliquées à n'importe quel type de système, leur lenteur est un frein important à leur utilisation dans le processus d'optimisation de systèmes complexes.

Comme alternative à ces temps de simulation, les approches analytiques permettent de déterminer une expression mathématique pour calculer une métrique (souvent un rapport signal à bruit de quantification ou SQNR) utilisée pour estimer les performances du système. Ces expressions, une fois déterminées, sont évaluées très rapidement tout au long du processus d'optimisation ce qui permet d'explorer beaucoup plus de solutions. Ces méthodes analytiques se basent sur un modèle de bruit [5] qui suppose que le bruit de quantification est petit par rapport au signal et qu'il n'est pas corrélé avec le signal. On parle aussi de théorie de la perturbation où le bruit est une petite déviation autour du signal en précision infinie. Dans [6], un développement de Taylor au premier ordre est utilisé pour éliminer les produits croisés de bruits, et les bruits en sortie du système sont ramenés à des sommes pondérées des différents bruits de calcul présents en entrée du système ou générés en interne. L'état de l'art actuel permet donc de traiter tous les systèmes basés sur des opérations arithmétiques y compris les systèmes non linéaires et variant dans le temps [6]. Seuls les opérateurs

à modèle de bruit non linéaire *(unsmooth)*, comme les opérateurs de seuillage ou de décision, ne peuvent pas être traités par les approches analytiques. Cependant, pour des systèmes complexes et hiérarchiques, ces approches deviennent rapidement complexes car i) basées sur une analyse statique du code et ii) devant chercher à exprimer le bruit en sortie en fonction de toutes les sources de bruit internes au système.

Ce papier propose une méthodologie de type diviser pour régner pour accomplir l'optimisation de la précision d'un système défini de façon hiérarchique à base de sous-systèmes tels que représentés à la figure 1. Le comportement en précision finie de chaque sous-système est modélisé par une source de bruit unique située à sa sortie qui agrège l'ensemble des bruits qui le composent. L'objectif est alors de trouver les niveaux de puissance de bruit en sortie de chaque sous-système de façon à minimiser le coût de l'implémentation du système complet, tout en maintenant les performances globales du système. Ces performances sont évaluées grâce à une approche originale mixant les approches analytiques et par simulations. Comparée aux approches existantes, notre méthode permet de réduire les temps d'optimisation et de supporter des systèmes complexes en combinant les avantages des deux mondes.

2 Approche hiérarchique

Un système B décrit de façon hiérarchique selon la figure 1 est considéré. B est constitué de N_b sous-systèmes B_i composés uniquement d'opérations arithmétiques et de N_o opérateurs « unsmooth » O_j. Les B_i affichent un comportement linéaire vis à vis des bruits de calcul et peuvent donc être traités de façon analytique. Les opérateurs O_j peuvent être des décisions ou des calculs de modulo et demandent des simulations car ile ne peuvent pas être traités de façon analytique. Ceci est dû au fait qu'une faible amplitude de bruit peut amener une déviation important par rapport à la référence du signal en précision infinie (par exemple un faible bruit à l'entrée d'un opérateur de seuil peut amener à une erreur de décision). Les hypothèses des modèles de bruit ne sont donc plus valables.

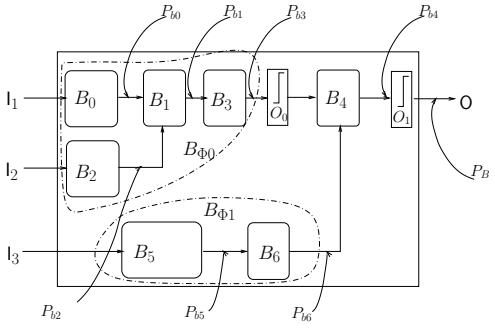

FIGURE 1 – Description flot de données d'un système sous forme hiérarchique

Le comportement en précision finie de chaque sous-système B_i est modélisé par une source unique n_i située en sortie. Le bruit de quantification n_i est défini par sa densité de probabilité (PDF) f_{n_i} et sa densité spectrale $S_{n_i}(\omega)$. La puissance P_i du bruit en sortie de B_i est utilisée comme variable d'optimisation au niveau système. Soit $\mathbf{P} = [P_0, P_1, \ldots, P_{N_b-1}]$ le vecteur regroupant les puissances des sources n_i, le problème d'optimisation donné par l'équation 1 peut être redéfini comme

$$\min\left(C\left(\mathbf{P}\right)\right) \quad \text{such as} \quad \lambda\left(\mathbf{P}\right) \geq \lambda_{obj}. \tag{2}$$

En d'autres termes, le problème d'optimisation peut être vu comme une répartition *(budgeting)* optimale des sources de bruit vers chaque sous-système, de façon à ce que le critère de performance minimale équivalent en sortie de B soit respecté. La solution à ce problème demande l'évaluation de la fonction de coût C et des performances λ conjointement à un algorithme d'optimisation efficace pour trouver une bonne solution avec un nombre minimum d'itérations.

2.1 Évaluation du coût

Au niveau du système, l'évaluation du coût global d'implémentation C est obtenue à partir des coûts $C_i\left(P_i\right)$ de chaque sous-système de B_i. Pour chaque sous-système B_i, le coût d'implémentation C_i doit être évalué pour une valeur maximale donnée de la puissance du bruit de quantification P_i en sortie du sous-système. Ce coût doit être minimisé pour une contrainte de puissance du bruit P_i. Cet objectif se rapporte à l'optimisation de la largeur des opérations et correspond au problème classique présent au sein de le processus de conversion en virgule fixe. Pour le sous-système B_i, le coût $C_i(\mathbf{W_{D_i}})$ est minimisé sous la contrainte de précision P_i. Ce problème d'optimisation peut être énoncé comme suit

$$\min(C_i(\mathbf{W_{D_i}})) \quad \text{tel que} \quad f_{P_i}(\mathbf{W_{D_i}}) < P_i \tag{3}$$

avec $f_{P_i}(\mathbf{W_{D_i}})$ l'expression analytique de la puissance du bruit de quantification du sous-système B_i. L'utilisation dans le processus de conversion en virgule fixe d'une approche analytique pour évaluer la précision des calculs permet d'obtenir des temps d'optimisation raisonnables. Il est à noter que P_i est une borne sur la puissance maximale telle que les performances de l'ensemble du système ne soient pas trop détériorées par le bruit généré au sein du sous-système B_i . La détermination de la contrainte de précision P_i fait partie du problème d'optimisation au niveau du système.

Comme exprimé dans l'équation 3, chaque évaluation du coût d'implémentation $C_i\left(P_i\right)$ exige la réalisation de la conversion en virgule fixe de chaque sous-système sous la contrainte de précision P_i. Afin de réduire le nombre de conversions en virgule fixe, l'évaluation du coût C_i peut être approximée par une interpolation de la fonction de coût à partir des points disponibles P_i. La fonction de coût est généralement monotone par nature. La contrainte de bruit et le coût de l'implémentation augmentent dans le même sens.

Les performances en termes de qualité d'un système sont généralement mesurées par des métriques telles que le rapport

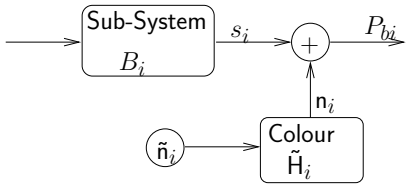

FIGURE 2 – Modèle de bruit équivalent

signal à bruit, la puissance totale du bruit ou le taux d'erreur binaire. Dans le cas d'un système composé de plusieurs sorties ou devant satisfaire plusieurs contraintes, une métrique combinant les performances associées aux différentes sorties ou aux différentes contraintes doit être mis en place. Au niveau du système, les performances sont évaluées par une approche mixte combinant la simulation en virgule fixe et une approche analytique permettant de déterminer l'expression de la puissance du bruit de quantification. L'approche basée sur la simulation est universelle et est applicable à tous les types d'opérateur. L'approche analytique est utilisée pour accélérer le processus d'évaluation des performances. Le comportement en précision finie de chaque sous-système de B_i est modélisé par une source de bruit unique.

2.2 Modèle de source de bruit unique

La figure 2 représente le modèle de bruit pouvant être utilisé comme un modèle de source de bruit équivalent. Le comportement global d'un système en virgule fixe est modélisé par une unique source de bruit localisée en sortie du système. Dans cette approche, le défi consiste à définir précisément les caractéristiques statistiques du bruit de quantification n_i. Le bruit en sortie du sous-système est défini par sa densité de probabilité f_{n_i} et sa densité spectrale de puissance $S_{n_i}(\omega)$.

Dans [7], un modèle pour caractériser la densité de probabilité du bruit de quantification est présenté. Il est démontré que la densité de probabilité du bruit à la sortie de systèmes comprenant des systèmes LTI résulte de la combinaison d'un bruit uniforme et d'un bruit suivant une loi normale. Le filtre de coloration du bruit $\tilde{H}_i$, dont les paramètres peuvent être déterminés à partir du sous-système B_i, détermine la densité spectrale de puissance du bruit additif global. Pour les systèmes LTI, le spectre est obtenu à partir de la fonction de transfert $\tilde{H}_i$.

Dans le problème d'optimisation, à chaque itération, le coût est évalué en premier, puis les performances de l'application sont mesurées. La distribution du bruit et sa densité spectrale de puissance sont déduites de la spécification virgule-fixe obtenue pour cette itération donnée.

2.3 Approche mixte pour évaluer les performances

Les performances sont évaluées à travers des simulations, mais les résultats analytiques sont utilisés pour accélérer le calcul de certaines parties du système. La propagation d'une source de bruit de quantification n_i dans un sous-système de B_j incluant uniquement des opérateurs à modèle de bruit linéaire, peut être approximée par un filtre H_j. L'objectif est de remplacer le sous-système B_j par le filtre H_j pour accélérer les simulations. Soit $S^j_{n_i}(\omega)$ la densité spectrale de puissance du sous-système B_j lorsque la source de bruit n_i est située à l'entrée sous-système. Le filtre H_j est défini tel que la relation suivante soit respectée

$$S^j_{n_i}(\omega) = S_{n_i}(\omega).H_j(\omega). \tag{4}$$

La densité spectrale de puissance $S_{n_i}(\omega)$ est obtenue à partir du processus de conversion en virgule fixe et $S^j_{n_i}(\omega)$ est calculée analytiquement à partir des caractéristiques du sous-système ou obtenue à partir de la simulation en virgule flottante.

Le système B peut être considéré comme un graphe dont les nœuds correspondent à des sous-systèmes B_i ou les opérateurs de décision O_j. Les arcs du graphe correspondent alors au signal utilisé pour communiquer entre les sous-systèmes. Les nœuds B_m correspondant aux sources du graphe ne sont simulés qu'une seule fois et leurs sorties correspondantes s_m sont stockées. Ensuite, les sorties s_m sont directement utilisées pour la simulation. Dans le cas de l'exemple présenté à la figure 1, les sous-systèmes B_0, B_2 et B_5 sont simulés qu'une seule fois.

Des sous-graphes $B_{\Phi i}$ de B peuvent être identifiés de manière à accélérer l'évaluation des performances en utilisant des techniques analytiques si tous les nœuds de ce graphe sont des sous-systèmes intégrant uniquement des opérations à modèle de bruit linéaire. Les nœuds des sous-graphes $B_{\Phi i}$ peuvent être remplacés par des filtres appropriés comme spécifié dans l'équation 4. Dans le cas de l'exemple présenté à la figure 1, le sous-graphe $B_{\Phi 1}$ est constitué de B_0, B_1, B_2 et B_3 et le sous-graphe $B_{\Phi 2}$ est constitué de B_5 and B_6.

3 Algorithme d'optimisation *Max -δ_P dB*

Notre algorithme d'optimisation est une adaptation de la procédure *Min + 1 bit* basée sur une recherche séquentielle [8]. Les variables du problème sont les puissances de bruit P_i dérivées du modèle de source de bruit unique. Une phase d'initialisation est tout d'abord exécutée pour trouver les valeurs initiales des variables. Pour cela, les valeurs maximales des sources n_i permettant d'atteindre les performances de l'application sont recherchées :

$$\max(P_i) \quad \text{tel que} \quad \begin{array}{l} P_j = 0 \quad \forall j \neq i \\ \lambda(P_i) = \lambda_{obj} \end{array} \tag{5}$$

Pour chaque variable P_i la puissance du bruit est augmentée tant que la contrainte sur les performances n'est pas satisfaite, tout en gardant le reste des variables P_j à une valeur nulle. Ensuite, les variables P_i sont initialisées à leur valeur maximale associée.

Pour trouver la meilleure direction pour converger vers la distribution optimale de la puissance du bruit, le gain obtenu par la diminution de la puissance du bruit de chaque variable P_i par une valeur de δ_P est explorée.

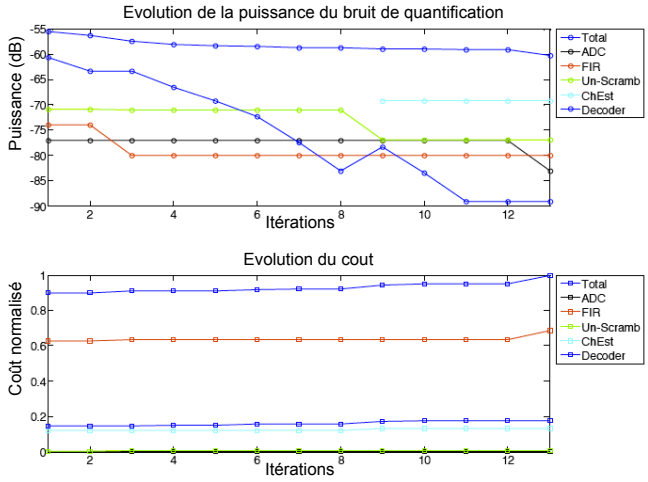

FIGURE 3 – Evolution du coût et des performances

Soit $\mathbf{P(k)}$ le vecteur des variables obtenu à l'itération k. Pour trouver la meilleure direction pour converger vers la distribution optimale de la puissance du bruit, le gain obtenu par la diminution de la puissance du bruit de chaque variable P_i par une valeur de δ_P est explorée. Soit $\delta_{\mathbf{P_i}}$ un vecteur ayant tous ses éléments nuls sauf l'élément i égal à $\mathbf{P(k)}$. Soit $\Delta\lambda_i(\mathbf{P(k)})$ la dérivée partielle de la fonction des performances au point $\mathbf{P(k)}$ et $\Delta C_i(\mathbf{P(k)})$ la dérivée partielle de la fonction de coût au point $\mathbf{P(k)}$. La métrique κ utilisée pour analyser la meilleure direction est calculée à partir de l'analyse de la dérivée partielle de chacune des variables d'optimisation (i.e. P_i) :

$$\kappa_i(\mathbf{k+1}) = \frac{\Delta\lambda_i(\mathbf{P(k)})}{\Delta C_i(\mathbf{P(k)})} = \frac{\lambda\left(\mathbf{P(k)} - \delta_{\mathbf{P_i}}\right) - \lambda\left(\mathbf{P(k)}\right)}{C\left(\mathbf{P(k)} - \delta_{\mathbf{P_i}}\right) - C\left(\mathbf{P(k)}\right)} \quad (6)$$

La direction l conduisant à la valeur de κ_l la plus élevée est conservée et la variable associée modifiée de la manière suivante $\mathbf{P(l+1)} = \mathbf{P(l)} - \delta_{\mathbf{P}}$

4 Résultats sur un récepteur W-CDMA

Notre approche a été testée sur un récepteur WCDMA composé des blocs suivants : *ADC*, *FIR*, *de-spreader*, *decoder*, *channel estimation* et *un-scrambler.* La courbe présentée à la figure 3 montre l'évolution du coût associé à chaque bloc au cours des différentes itérations. Le coût d'implantation correspond à la consommation d'énergie. Le coût global est une fonction croissante à chaque itération. Dans l'algorithme W-CDMA, l'ADC, le filtre FIR et le bloc *de-spreader* fonctionnent à une fréquence plus élevée que les blocs d'estimation de canal et de décodage. Il est naturel que ce filtre FIR consomme le maximum d'énergie et donc possède le coût maximal.

Comme présenté à la figure 3, pour cette application, 12 itérations sont nécessaires pour atteindre la convergence. Les courbes de Pareto, caractérisant le coût du sous-système en fonction de la contrainte de précision, sont utilisées pour déterminer la direction de déplacement d'après l'équation 6. Dans cette expérimentation, pour atteindre la contrainte de performance, le niveau maximal de bruit est de $-60\ dB$. Le temps d'exécution sur Matlab de l'algorithme d'optimisation est de quelques minutes sur un PC. L'application WCDMA complète peut être optimisée en utilisant l'algorithme classique *min*$+1$ bit en considérant toutes les opérations de l'application et ainsi aucun de niveau de hiérarchie. Les résultats obtenus montrent que le nombre d'évaluation des performances requis par l'algorithme *min+1 bit* est d'un ordre de grandeur ($10\times$) plus élevé par rapport à notre proposition d'approche hiérarchique basée sur l'algorithme *Max* $-\delta_P\ dB$.

5 Conclusion

Une approche hiérarchique basée sur le principe de diviser pour régner pour l'optimisation de la largeur des données a été présentée. Une approche mixte combinant la simulation et une technique analytique est utilisée pour déterminer les performances du système. Le modèle de source de bruit unique pour caractériser le comportement en précision finie d'un sous-système est proposé. Notre approche d'optimisation peut être appliquée de façon récursive à chaque niveau de la hiérarchie tant que la technique classique d'optimisation de la largeur des opérations ne peut pas être utilisée. Les résultats d'expérimentation montrent le gain en termes de temps pour optimiser un système complet.

Références

[1] S. Kim, K. Kum, and S. Wonyong, "Fixed-Point Optimization Utility for C and C++ Based Digital Signal Processing Programs," *IEEE Transactions on Circuits and Systems II*, vol. 45, no. 11, pp. 1455–1464, November 1998.

[2] M. Coors, H. Keding, O. Luthje, and H. Meyr, "Fast Bit-True Simulation," in *IEEE/ACM Design Automation Conference 2001 (DAC 01)*, Las Vegas, US, June 2001, pp. 708 – 713.

[3] F. Berens and N. Naser, *Algorithm to System-on-Chip Design Flow that Leverages System Studio and SystemC 2.0.1*, Synopsys Inc., March 2004.

[4] Mentor Graphics, *Algorithmic C Data Types*, version 1.3 ed., Mentor Graphics, March 2008.

[5] B. Widrow, "Statistical Analysis of Amplitude Quantized Sampled-Data Systems," *Trans. AIEE, Part. II :Applications and Industry*, vol. 79, pp. 555–568, 1960.

[6] R. Rocher, D. Menard, P. Scalart, and O. Sentieys, "Analytical accuracy evaluation of Fixed-Point Systems," in *12th European Signal Processing Conference (EUSIPCO 2007)*, Poznan, Poland, September 2007.

[7] D. Menard, R. Serizel, R. Rocher, and O. Sentieys, "Accuracy Constraint Determination in Fixed-Point System Design," *EURASIP Journal on Embedded Systems*, vol. 2008, p. 12, 2008.

[8] M.-A. Cantin, Y. Savaria, and P. Lavoie, "A comparison of automatic word length optimization procedures," *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, vol. 2, pp. II–612–II–615 vol.2, 2002.