

# 立体音響処理の ARM プロセッサ上での実時間実装

武智 翔吾<sup>†</sup> 辻野 孝輔<sup>†</sup> 尾上 孝雄<sup>‡</sup> 越智 裕之<sup>†</sup> 中村 行宏<sup>††</sup>

<sup>†</sup>京都大学 情報学研究科 <sup>‡</sup>大阪大学 情報科学研究科

<sup>††</sup>立命館大学 総合理工学研究機構

## 1 はじめに

携帯情報機器の急速な性能向上により、近年では、音声・映像コンテンツを場所を問わず容易に閲覧することができる。携帯機器上での音声再生の臨場感を向上する手法として、音声に三次元的な音源の方向感を与える立体音響処理が注目されている。携帯機器上で利用可能なヘッドフォンもしくはステレオスピーカによる再生で上記の臨場感を得る手法としては、頭部伝達関数を用いた処理が一般的に用いられる。

頭部伝達関数を用いた立体音響処理は一人での聴取に適する。携帯機器上では必然的に一人での聴取を行うことが支配的なため、携帯機器への搭載は立体音響処理の有望な応用の一つである。本稿では、携帯機器向けを中心に広く普及した組み込みプロセッサである ARM プロセッサ上での立体音響処理の実時間実装について報告する。携帯機器上で立体音響処理を実時間で行うことにより、聴取者の操作に応じた処理が可能となり、より多彩な応用が可能となる。

## 2 立体音響処理

人間が音像の位置を三次元的に知覚することを音像定位と呼ぶ。音像定位の要因としては、両耳に到達した信号間の遅延差や音圧差、頭部や耳介などでの回折によって生ずる音波の周波数特性の変化、部屋の壁や床などでの反射や回折などが存在する[1]。これらの音像定位の手がかりを含む、音源から人間の外耳道までの音声の伝達関数を頭部伝達関数と呼ぶ。頭部伝達関数は音源の方向と距離に依存する。頭部伝達関数の特性を音源の位置情報を含まない通常の音声に重畳することにより、仮想的な音像定位を伴うステレオ音声を合成することが可能である。

頭部伝達関数は複雑な周波数特性を示すため、頭部伝達関数を単一の FIR ないし IIR フィルタで精度良く再現するには高い次数のデジタルフィルタを用いて近似する必要がある。そのため、携帯機器上での立体音響処理においては、フィルタ演算に必要な演算負荷とメモリ量が大きいことが実装上の課題となる。これに対し、演算に必要な演算負荷とメモリ量を削減する手法が小林ら[2]によって提案されている。この手法では、図 1 のブロック図に示すように、入力音声を周波数分割し、異なった構成のフィルタで各サブバンド信号を処理する。本稿ではこの手法を基本に立体音響処理の実装を行う。

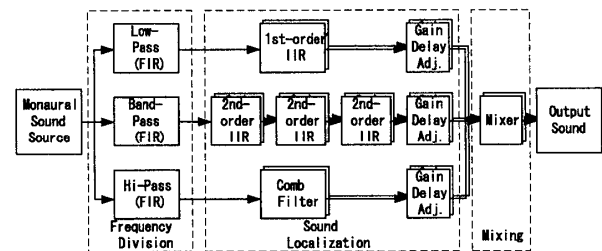


図 1: 周波数分割に基づく立体音響処理

## 3 処理システムの構築

立体音響処理の実時間処理の検証を目的とし、図 2 に示す構成のシステムを構築した。このシステムでは、ARM9 コアを搭載した TI OMAP5912 プロセッサの評価用ボードである OSK5912 を利用し、外部サーバからファイルシステムを NFS でマウントするとともに、外部ソフトウェアで立体音響処理の制御を行う。以降の評価は、この検証用システム上で行う。

## 4 立体音響処理の最適化実装

ARM プロセッサ上での立体音響処理の効率的な実現のために、固定小数点演算を用いたフィルタの最適化実装を行った。

Realtime implementation of 3-D sound processing on ARM processor

<sup>†</sup> Graduate School of Informatics, Kyoto University

<sup>‡</sup> Graduate School of Information and Science, Osaka University

<sup>††</sup> Ritsumeikan University

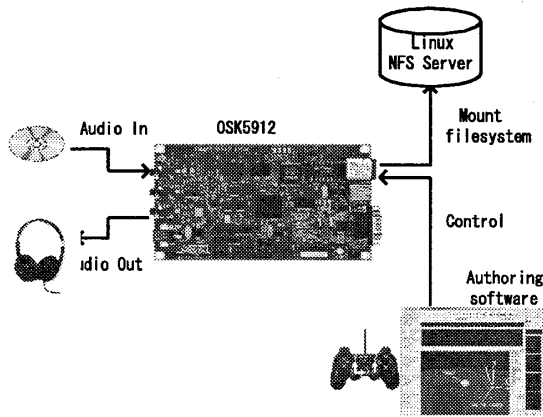


図 2: システムの構成

#### 4.1 固定小数点化

図 1 のフィルタ構成を対象とした立体音響処理フィルタの自動設計手法が辻野ら[3]により提案されている。設計結果は浮動小数点表現で得られるため、結果を固定小数点化する必要がある。固定小数点演算のオーバーフローを防止し、同時に演算精度を確保するため、 $L_2$  ノルムを用いたスケールリングを行った。オーバーフローは加算を行った場合に発生する。フィルタへの入力からフィルタ内の加算点までの伝達関数  $F(e^{j\omega t})$  を考え、どの加算点についても  $F(e^{j\omega t})$  の  $L_2$  ノルム

$$\|F(e^{j\omega t})\|_2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |F(e^{j\omega t})|^2 d\omega$$

が 1 を越えないようにフィルタのゲインを調整し、固定小数点表現のフィルタ係数を得た。

#### 4.2 コード最適化

フィルタ演算の固定小数点コードを C 言語で記述し、以下の最適化手法をコードに対して適用して演算を効率化した。

- フィルタ演算におけるループを展開し、分岐命令を削減して動作を高速化する (loop unrolling)
- 除算命令を逆数の乗算に置換する
- 実行順序の制御によりロード演算を削減する

最適化手法の適用前後におけるプロファイル結果を表 1 に示す。また、プロファイル条件を表 2 に示す。手動でのコード最適化により、演算サイクル数を 49.6%削減することができた。検証に用いた OMAP5912 プロセッサに搭載された ARM9 コアの動作周波数は 192MHz であり、最適化されたコードの実時間処理が可能である。システム上で立体音響処理を実行し、実時間処理が実現できていることを確認した。

表 1: プロファイル結果

	最適化前 (MCycles/sec)	最適化後 (MCycles/sec)
周波数分割	89.5	34.6
フィルタ処理	11.4	9.6
混合	11.8	9.4
合計	119.1	60.0

表 2: プロファイル時の条件

コンパイラ	gcc 3.3.1
最適化オプション	-O3
処理音のサンプリング周波数	44.1kHz
処理音の量子化ビット	16bit
フィルタ係数の量子化ビット	16bit

## 5 結論

本稿では、ARM プロセッサを用いた立体音響処理の固定小数点実装について報告した。周波数分割に基づく立体音響定位手法[2]を用い、ARM プロセッサ上へ立体音響定位処理をリアルタイム実装することに成功した。C 言語レベルでのコード最適化により、サンプリング周波数 44.1kHz における実時間処理に必要な CPU サイクル数は 60.0 MCycles となった。最適化を行う前と比較して、実行サイクル数を約 50% 削減することに成功した。

## 参考文献

- [1] Jens Blauert, "Spatial Hearing: The Psychoacoustics of Human Sound Localization", MIT Press, 1996.
- [2] Wataru Kobayashi, Noriaki Sakamoto, Takao Onoye, and Isao Shirakawa, "3D acoustic image localization algorithm by embedded DSP", IEICE Trans. Fundamentals, E84-A, No.6, 2001.
- [3] Kosuke Tsujino, Wataru Kobayashi, Takao Onoye, and Yukihiro Nakamura, "Automated design of digital filters for 3-D sound localization in embedded applications", Proc. IEEE International conference of Audio, Speech, and Signal Processing, V349-352, 2006.

## 謝辞

本研究は株式会社アーニス・サウンド・テクノロジーの協力を得て行われた。小林様をはじめとする同社の方に深く感謝する。