

## 並列計算機によるリアルタイム音響情報処理

4P-9

### — 音楽音響信号に対するビートトラッキングシステム —

後藤 真孝 村岡 洋一

早稲田大学 理工学部

#### 1. はじめに

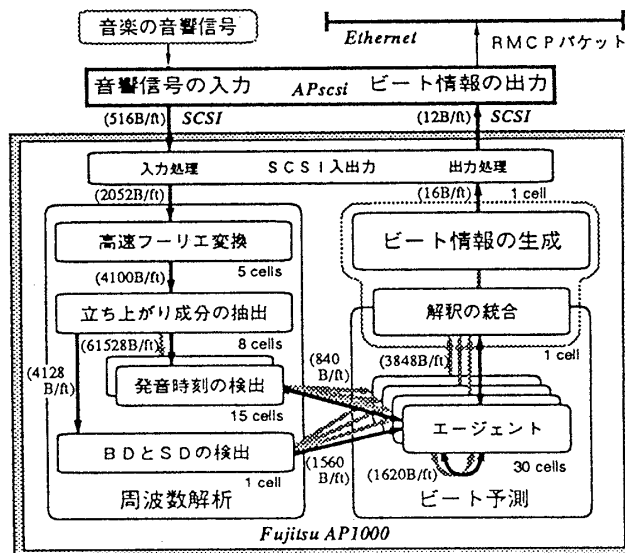
本稿では、並列計算機の音楽情報処理への応用例として、音楽音響信号に対してリアルタイムにビートを認識するシステム(BTS)の、並列計算機への実装について述べる。BTSは、音楽聴取過程の計算機による実現を目指す試みの1つである。このような音響信号からの聴覚知覚処理を実現するには、単なる信号処理だけでなく音楽的知識などを用いた高次の判断が必要である。これらの処理は計算量が多く、並列度も高いため、人間のようにリアルタイムに知覚するには並列処理するのがよい。

BTSの実装は、異なる並列処理方式を組み合わせることを行う。処理の質が異なる複数の計算をリアルタイムに実行するために、計算機全体として大局的に見るとパイプライン状にデータを処理する。そしてパイプラインの各ステージにおいて、さらにパイプライン処理やデータパラレル処理、分散協調処理をおこなう。各プロセッサが質的に異なる処理をおこなうことにより、BTSを構成する各処理モジュールをプロセッサへダイレクトにマッピングすることができる。さらに、並列計算機を構成するプロセッサ数を増やすことにより、リアルタイム性を損なうことなく処理を質的に拡張し易い。

#### 2. BTSのAP1000への実装

BTSは、市販のCompact Discなどから得た複数の楽器音を含む音響信号に対し、人間が音楽に合わせて手拍子を打つように、曲のビート(4分音符)の位置を認識する。その際、ビートの存在する時刻を認識するだけでなく、そのビートが強拍か弱拍かも判断する。認識した結果はビート情報として、入力した曲に合わせてリアルタイムにネットワーク上へ送信される。BTSは、ロック・ポップスの主にドラムスがビートを刻む曲を対象とする。入力曲は4/4拍子とし、テンポは70~180 M.M.(4分音符/分)の間で、曲中を通じてほぼ一定であるものとする。

以上を実現するBTSの処理手順の概要を述べる。まず、音響信号の入力においてA/D変換された音響信号に対して、周波数解析をおこない、各周波数帯域ごとの発音時刻と、バスドラム(BD)とスネアドラム(SD)の発音時刻を検出する。次に、ビート予測の複数のエージェントが、過去に得られた発音時刻から様々なビートの解釈を出力する。各エージェントはビートの間隔を求め、次のビート時刻の予測とビートタイプの判断をおこなった後に、解釈の確信度を自己評価する。そして、最も確信度の高い解釈の結果に基づいて、ビート情報の生成をおこなう。最後にビート情報の出力が、ネットワークを通じて他のアプリケーションプログラムへとビート情報を送信する。具体的な処理は、文献[1]に詳しい。



B/ft = bytes / frame-time 1 ft = 11.61 msec 1 B/ft = 86.13 bytes/sec

図1: AP1000への実装

これらの計算量が多い処理をリアルタイムにおこなうために、富士通の分散メモリ型並列計算機AP1000上にBTSを実装した(図1)。実装したAP1000では、64台のセルと呼ばれる要素プロセッサが、2次元トラス状に接続されている。BTSではこれらのセルを8つのグループに分け、各グループに対して異なる処理モジュールを割り当てる。各処理モジュールのセルの数を、図1の各長方形の右下に示す。

各処理モジュール間でデータを流すことで、計算機全体としてはパイプライン状に処理する。矢印はグループ間の大局的なデータの流れを表しており、主に入力から出力へとU字型にデータが流れる。実際の通信量を、矢印の横に示す。

以下では、各処理モジュールの実装について述べる。

##### 2.1 音響信号の入力

音響信号の入力は、AP1000のセルの1つとSCSI接続されたワークステーション(Sun SPARC Station 10, 以下APscsiと呼ぶ)によりおこなう。A/D変換やビート情報のネットワーク上への出力は、すべてAPscsi上でおこなう。

APscsiは音響信号を22.05kHz, 16bitでA/D変換し、256点ごとにブロック化してAP1000へ転送する。この1ブロックに相当する時間を1フレーム時間(11.61msec)と呼び、フレーム時間(ft)をAP1000内のすべての処理の時間単位とする。

##### 2.2 周波数解析

周波数解析の処理モジュールはそれぞれ計算の質が異なるため、別々の並列処理方式を用いる。

1. 高速フーリエ変換(FFT) (パイプライン処理, 5セル)  
A/D変換されたデータに対して、観測区間(4ft)を時間軸方向に1ftずつずらしながらFFTをおこない、周波数スペクトルを得る。各セルは逐次のFFTをおこなうが、それぞ

れが1ftずつ時間的にずれてパイプライン状に処理する。1回のFFTの計算はおよそ4.2ftかかるため、5つのセルがFFTをおこなうことで、毎ftごとに結果を得ることができる。この結果は、次の8つのセルに対してマルチキャストされる。

2. 立ち上がり成分の抽出 (データパラレル処理, 8セル)  
FFTの結果から周波数の立ち上がり成分を抽出する。この処理は、8つのセルによるデータパラレル処理で計算する。全周波数帯域を8つの帯域に等分し、各セルが自分の担当する帯域の立ち上がり成分を抽出する。この結果を、発音時刻の検出をおこなう15個のセルと、BDとSDの検出をおこなう1個のセルに対してそれぞれ送信する。
3. 発音時刻の検出 (コントロールパラレル処理, 15セル)  
立ち上がり成分の抽出結果から、15個の発音時刻の検出器が、それぞれ異なる条件(パラメータ)で発音時刻を求める。各検出器に1つずつセルを割り当て、並列に発音時刻を検出する。各検出器は、次節で述べるエージェントの組に対応しており、検出結果を対応する組へ送信する。検出器のパラメータの値は、対応する組のエージェントによって動的に調整される。具体的には、ピーク時刻を検出する際の感度や、検出する周波数帯域がパラメータとして変更される。

BDとSDの検出は、立ち上がり成分の抽出結果に対して、BDとSDの各特徴周波数を自動的に獲得しておこなう。これは1つのセルでおこない、その結果をビート予測の30個のエージェントに対して送信する。

### 2.3 ビート予測 (分散協調処理, 30セル)

周波数解析の結果を30個のエージェントが並列に解釈し、次のビート時刻をそれぞれが予測する。この解釈の結果は、解釈の統合処理へと集められ、最も確信度の高い解釈が選択される。これにより、あるエージェントの解釈がはずれても、他のエージェントが正しく解釈している限り、ビートを見失わずにトラッキングできる。

各エージェントはそれぞれ1つのセルに実装され、異なる戦略によりビートを解釈する。エージェントは戦略を変更する3つのパラメータを持ち、他のエージェントと通信することにより、お互いに異なるパラメータの値を設定する。こうして入力を各エージェントが異なる視点から調べることで、様々なビートの解釈の結果を得る。

これらのエージェントは、2つずつの組に分けられる。同じ組の2つのエージェントは協調し合いながら、両者が同じビートの間隔で、お互いにビートの間隔の1/2ずれた時刻を予測する。これにより、一方が4分音符の裏拍を予測していても他方が正しい表拍を予測できる。

### 2.4 ビート情報の生成・出力

最も確信度の高い解釈に基づいてビート情報を生成する。このビート情報は、SCSIを介してAPscsiへ転送される。APscsiはこれをRMCP(Remote Music Control Protocol)パケット[2]として、音楽に合わせてEthernet上へ送信する。これにより、音楽に同期したCGの生成や手拍子の音の出力など、ネットワーク上の様々な機器でビート情報を活用できる。

## 3. 時間管理のための同期機構

APscsiがA/D変換する際に、一定個数(256点)のデータが揃うまでブロックすることを利用して時間管理する。AP1000

例は、APscsiから転送されてきたデータによるデータ駆動方式で動作し、1ブロックを受信する度に同期をとる(11.61msec間隔)。この同期は、プロセス間で同期をとるためにおこなうのではなく、APscsiからデータを受信する時刻に、各セルの処理のタイミングを合わせるためにおこなう。

BTSの並列化方式では、あるグループが同期をとっているときに、別のグループが処理を続ける必要がある。AP1000のS-Netによるバリア同期は、現在のOSではすべてのセルで明示的に同期をとる必要があるため利用できない。そこで、ステータスハードウェアを利用して時間確認可能な同期機構を実現する。S-Netは、バリア同期をおこなう同期ハードウェア以外にステータスハードウェアを備えている。これは、全セルからの8ビットのステータスの論理和をハードウェアで求め、その結果を各セルが参照できる機構である。BTSでは、8ビットのステータスのうち上位3ビットを通常のステータス用(0~7)、下位5ビットを同期用(0~31)として割り当てる。

同期は以下のおこなう。SCSI入出力をおこなうセルがマスターとなり、APscsiからデータを受信する度に同期用ステータスをインクリメントする。他のすべてのセルはスレーブとなり、同期用ステータスの変化を検出して同期をおこなう。こうして同期をとる必要のあるセルだけが変化を監視して同期し、必要のないセルはそのまま処理を続けることができる。例えばFFTをおこなうセルは、同期用ステータスを5ftおきに飛ばして確認する。また、同期用ステータスの値が予測していた値よりも進んでいれば、自分の処理が遅れていることがわかる。しかし、現在の実装では、AP1000の処理能力はBTSの計算負荷よりも高く、正常動作しているときに処理が遅れることはない。そのため、処理が通常より大幅に遅れた場合には、異常が起きたと判断してシステムは停止する。

現在、AP1000全体のパイプラインをデータが通過する時間は11ftである。つまり、ある時点の出力結果には、最短で11ft(127.71msec)前の入力までが反映される。この遅延時間は、入力を直接加工して音響信号を出力する処理(リバンプなどのエフェクト処理)では問題になるが、BTSのように過去のビートを認識して、常に予測結果(次のビート時刻)を出力する場合には問題にならない。例えば、最も短い180M.M.の4分音符の長さでも333.33msecあるため、予測時に前回のビート時刻周辺の情報は十分利用できる。

## 4. おわりに

本稿では、並列計算機の音楽情報処理への応用例として、BTSの並列計算機AP1000への実装について述べた。BTSでは、複数の並列処理方式を組み合わせることで、質の異なる様々な処理をリアルタイムに実行することができる。BDが1,3拍目、SDが2,4拍目で多く鳴るロック・ポップスを入力して実験した結果、30曲中27曲に対して正しいビートをリアルタイムにトラッキングできた。今後は、各エージェントの予測アルゴリズムの改良により対応曲を増やすとともに、エージェント間の協調・統合の方式も改良していく予定である。

### 謝辞

AP1000の実行環境を提供して頂いた富士通研究所 並列処理研究センターに感謝いたします。

### 参考文献

- [1] 後藤 真孝, 村岡 洋一: 音楽音響信号に対するビートトラッキングシステム, 情処研報, Vol.94, No.71, 音楽情報科学 94-MUS-7-8 (1994).
- [2] 後藤 真孝, 橋本 裕司: MIDI制御のための分散協調システム — 遠隔地間の合奏を目指して —, 情処研報, Vol.93, No.109, 音楽情報科学 93-MUS-4-1 (1993).