

## 1. 序論

計算機の処理においては常に高速な処理が求められている。計算機の高速化の手法として、複数のプロセッサによる並列処理がある。しかし複数のプロセッサを持つ並列計算機は非常に高価であるため、容易に利用できない。このためネットワークを用いて仮想的に並列計算機を構成することにより、誰でも安価に並列処理が可能となり、スーパーコンピュータに匹敵する性能を得ることが期待されている。

本研究では、MPI(Message Passing Interface)と呼ばれる並列処理の規格を用いている MPICH というソフトウェアを使用し、並列計算の有用性を実験的に評価する。MPICH は無料で提供されており、並列プログラミングでは主流となっている。本研究では複数の wav を mp3 に並列圧縮し、どれくらい時間が短縮できたかを検証する。

## 2. 研究内容

### 2.1 実験内容

本研究ではサイズ約 36MB の wav ファイルを mp3 に変換し、ファイル数および使用するホスト数を変えながらその変換時間を計測する。表 1 に使用した計算機の詳細を示す。

表 1: 使用したホスト

使用ホスト				
ホスト名	Kimura	Watanabe	Ishi-5	Magisian
OS	Windows XP Pro	Windows XP Pro	Windows XP Pro	Windows 2000
CPU	Pentium 1000Mhz	Pentium 1000Mhz	Pentium4 1.9GHz	Pentium4 1.9GHz
RAM	512M	512M	640M	670M
CPU 数	1	1	1	1

### 2.2 並列エンコーダ

エンコーダは主に、lame を用いたダイナミックライブラリ gogo.dll を使用した。wav ファイルは予め、ホストの MPICH が動作する共通のディレクトリにそれぞれ同じファイル名を "C:\¥USERNAME¥mpi" に "audio\_X.wav" という名で保存しておく ( $1 \leq X$ )。

wav のエンコードの振り分けは、MPICH 特有のプロセス毎に自動的に振り分けられるランクを用いて wav ファイルのエンコードを自動的に振り分ける。

## 3. 結果・考察

表 2 に並列エンコードした処理時間の結果を示す。それぞれの処理時間は 5 回実行させて平均を取った時間である。複数の wav を mp3 にエンコードする時間が示すとおり、wav ファイルとホストの数が比例した時間になって、ホストの数と処理時間は反比例した時間になっており、期待した処理時間が得られた。

表 2: wav → mp3 の変換時間 (秒)

ホストの数	wav ファイルの数			
	1	2	4	8
1	5.936	11.594	22.151	49.867
2	5.934	6.165	12.147	25.223
3	5.713	5.733	11.896	18.396
4	5.648	5.751	6.620	12.687

## 4. 結論

複数の wav ファイルを mp3 にエンコードする時は並列処理を使用することが大変有効的であることがわかった。

しかし今回使用した並列計算機はあまり変わらない性能の計算機ばかりだったので、極端に低スペックと高スペックの計算機で並列処理した場合の wav の並列処理の振り分けを考慮したアルゴリズムであれば、更に高速な実行処理にできたかもしれない。

## 参考文献

- 1) P. パチェコ 著, 秋葉博 訳: MPI 並列プログラミング, 培風館 (2001)
- 2) W. グロップ, E. ラスク, T. タークル 著, 畑崎隆雄 訳: 実践 MPI-2, ピアソン・エデュケーション (2002)