

PDE-II におけるメディア間同期機構の実現に対する考察

1H-6

稲垣 英太郎 岡村 耕二 荒木 啓二郎  
 奈良先端科学技術大学院大学 情報科学研究科

1. はじめに

マルチメディアを扱うアプリケーションは、同期した一群の連続メディアを処理できる必要がある。UNIXに代表される従来のOSではマルチメディア処理における連続メディア間の同期機構（以後単に同期機構）を提供していないため、現状ではプログラマが同期という時間的な制約を意識して開発を行なっている。しかしプログラマの負担を軽減するためにも、システム側で同期機構を提供することが望ましい。

本稿では、我々が現在開発を進めている分散並列OS PDE-II(Parallel Distributed Environment II)<sup>[2]</sup>上 に実装される同期機構について述べる。

2. 同期の品質とその保証

テレビ電話で円滑なコミュニケーションを図る場合、音声より明瞭に再生するために、音声の優先順位を動画像より高くする必要がある。このため、計算機上でマルチメディアを利用するには連続メディア間の優先順位が必要になる。本章では連続メディア間の同期と連続メディア間の優先順位は不可分であるとし、以下に両者を統合した同期モデルを提案する。

2.1 データ処理モデルと同期の品質

マイクやビデオに代表される表示メディア<sup>[2]</sup>やアプリケーションなど、マルチメディアのデータを処理する実体をエンティティと総称する<sup>[2][4]</sup>。動画像や音声など周期的な連続メディアを計算機上で利用するためには、その連続性を損なわないよう、エンティティが連続メディアのデータ（以後単にデータ）を周期的に処理する必要がある。図1に二つのデータ処理の進行を表すデータ処理モデルを示す<sup>[1]</sup>。

図1の縦軸が表す論理時間とは、一周期当たりのデータ処理が終ると一周期時間だけ進む時間である。横軸は実時間の流れを表す。図1中の正方形のブロックは、一周期時間当たりに行なう連続メディアデータの処理を表し、ブロックの一辺は周期時間の長さを表す。ブロックの左端はエンティティがデータ処理を開始した時刻、即ち同期機構がエンティティにデータを供給した時刻を示す。ブロックの右端はデータ処理のデッドラインである。

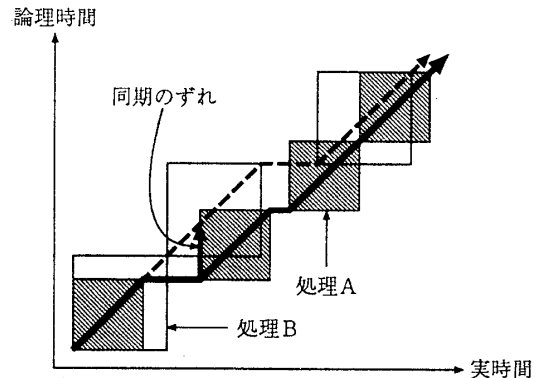


図1: 同期の品質

論理時間を表す縦軸の値は離散的な値をとるが、同期のずれを算出する場合、縦軸の値を連続的な値として捉える必要がある。そのためデータ処理の進行は、図1のブロックに沿って右斜め上に伸びた太線の様に進むと仮定する。

ここで、網がかかった小さなブロックを処理A、大きな白いブロックを処理Bとする。処理Aと処理Bが同期している状態とは、それぞれのデータ処理の進行が一致している時である。しかし、分散環境下では通信の過程で不定期な遅延が発生するため、完全に同期をとることは難しく、実際には図1の様に同期のずれを伴って処理が進行する。我々はその同期のずれの許容範囲を「同期の品質」と定義した。

図1では処理Bは処理Aに同期しているが、この時「処理Bは処理Aに従う」と言い、この場合における同期の品質を「処理Aに対する処理Bの同期の品質」と呼ぶ。

2.2 データスキップと遅延挿入

図2に同期のずれの修正方法<sup>[1]</sup>を示す。ここでは網がけの濃い小さなブロックを処理A、白い大きなブロックを処理Bとし、処理Bは処理Aに従うとする。またここでの同期の品質は、「処理Aに対する処理Bの同期の品質」である。

処理Bの進行が処理Aより進み過ぎて同期の品質を超過した場合（図2のB-3参照）、同期機構は処理Aが再開されるまで処理Bへのデータ供給を停止する。このように、処理Bの論理時間の進行をデータ供給が再開されるまで停止して同期のずれを修正する処理を

“Media Synchronization Mechanism for PDE-II”  
 E.INAGAKI, K.OKAMURA and K.KARAKI  
 Graduate School of Information Science, Nara Institute  
 of Science and Technology

「遅延挿入」と定義する。

一方図2のB-4の様に、処理Bの進行が処理Aより遅れ過ぎて同期の品質を超過した場合、同期機構はB-4を捨ててB-5をエンティティに供給する。このように、処理Bの論理時間の進行を一時的に加速することで同期のずれを修正する処理を「データスキップ」と定義する。

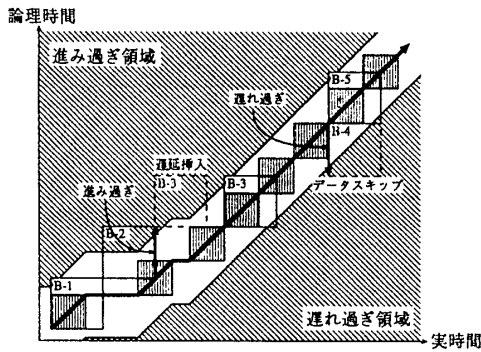


図2: データスキップと遅延挿入

図2に示すように各処理の従属関係は連続メディア間の優先順位を反映しており、優先順位の低い処理はより優先順位の高い処理に従う。

### 3. 同期機構

同期機構は“QOSに基づいたマルチメディア処理モデル”[2][5]におけるQOSポイントである。図3にPDE-IIの同期機構を示す。

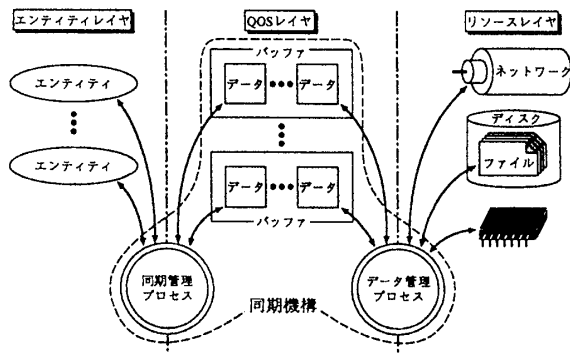


図3: 同期機構

エンティティレイヤは、表示メディアやアプリケーションをエンティティとして実現し、その処理を行なうレイヤである。リソースレイヤはネットワークやメモリ、ファイルなどシステムの有限なリソースを扱う。QOSレイヤは、エンティティレイヤとリソースレイヤの間に立って同期を調停するためのレイヤで、そこに

同期をとる連続メディアの数と同数のバッファが用意される。

データ管理プロセスはリソースからデータを受受あるいは読み出し、QOSレイヤにあるバッファにそのデータを書き込む。同期の管理は、同期管理プロセスがエンティティへのデータ供給を管理することで行われる。同期管理プロセスは各データ処理の同期のずれと同期の品質を監視し、ある連続メディアの処理が遅れ過ぎて同期の品質を超過したらデータスキップを、進み過ぎて同期の品質を超過したら遅延挿入を適用して、複数の連続メディア間の同期を図る。

### 4. 実装に関する考察

同期管理プロセスはその処理に時間的な制約を持つ。そのため同期管理プロセスを最終的にPDE-II上に実装する場合には、ある一定周期時間毎に起動することが保証された周期スレッド<sup>[3]</sup>を用いる。一方データ管理プロセスは、リソースからデータを受けとりQOSレイヤのバッファにそのデータを渡す役割を担うが、同期管理プロセスのように時間的な制約を持たないため特に実装上の制約はない。QOSレイヤのバッファは同期管理プロセスとデータ管理プロセスの両者がアクセスするため、その実装には共有メモリ空間を用いる。

### 5. おわりに

現在、本稿で述べた同期機構の有効性を確認するために、UNIX上に同期機構のプロトタイプを実装中である。この実装が終了した後、プロトタイプを用いて様々な同期の実験を行う予定である。

### 参考文献

- [1] Anderson, D.P. and Homsy, G.: “A Continuous Media I/O Server and Its Synchronization Mechanism”, IEEE, Computer, Oct. 1991.
- [2] 岡村, 吉川, 稲垣, 荒木: “PDE-IIの概要～QOSに基づいたマルチメディア処理モデル～”, 情報処理学会, 第48回全国大会, 1994.
- [3] 吉川, 岡村, 荒木: “PDE-IIにおける実時間同期のための周期スレッドの提案”, 情報処理学会, 第48回全国大会, 1994.
- [4] 田中, 岡村, 荒木: “PDE-IIにおけるマルチメディアアプリケーションの実装について”, 情報処理学会, 第48回全国大会, 1994.
- [5] 岡村, 吉川, 稲垣, 荒木: “QOS指定可能なマルチメディアモデルの提案”, 情報処理学会, マルチメディア通信と分散処理ワークショップ, 平成5年11月.