

## 予約アクセスによる時間同期処理の試み

坂下善彦 太田賢\* 水野忠則\*

三菱電機(株) 情報システム研究所  
静岡大学工学部

伝送ネットワークの発展に伴い、そこに結合されたコンピュータの間で処理される情報は、極めて高速に行われる。しかし、ネットワークを介すが故に絶対的な通信時間が存在する。ここでは、例としてマルチメディア情報を対象にし、その時間的変化関係を保証することを目的にしている。利用するネットワーク状況を把握して、且つ、処理の対象とする資源の数とその特質を考慮し、予め処理を起動する時間を考慮する予約アクセスの方法を検討した。通信トラフィックにおける従属接続型の待ち合わせの考え方を基にして、トラフィック密度と伝送効率に注目した。コンピュータ内部でのバッファリングや処理の分配の要素については、今後の検討対象課題とし、プロトコルレベルでの仕様検証を基本対象とし、サービスの振る舞いの動作を観測した。

A study of an estimation for considering execution time  
related to the network environments

Yoshihiko SAKASHITA, Ken OHTA\*, Tadanori MIZUNO\*

Information Systems Laboratory  
Mitsubishi Electric Corporation  
5-1-1, Ofuna, Kamakura, Kanagawa 247, Japan  
\* Faculty of Engineering, Shizuoka University  
3-5-1, Johoku, Hamamatsu, Shizuoka 432, Japan

In the distributed communication systems, the communication time might be finite, even though, a communication speed shall be very fast. On processing multimedia information, the time synchronization factors are very important. This report introduces the estimation time for accessing the distributed resources regarding the status of network environments. Based on the principle of "theory of cascaded queuing processing", we have tried to estimate the traffic intensity and the access performance.

## 1 はじめに

コンピュータシステムの形態は集中型から分散型へと移行している。この現象は、ハードウェアシステム構成の視点から見た特徴を表しているが、ソフトウェアシステムの視点からは、異なる視点から見た特徴がある。クライアント・サーバという形態は、そのサービスあるいは実現されるプログラムの構造から観た場合に、供給と消費、あるいはサービスと依頼の関係が存在する側面を表している。

分散処理では、ネットワークに結合されたコンピューティング資源や情報資源を物理的な距離を越えて、互いに操作・利用して、最終目的であるサービスをユーザに提供することが、目的とする所である。ネットワークを介して結合しているコンピュータの間で、実行主体であるプログラムが互いに関連しながら動作し、それぞれのプログラムが自律した機能を備えて互いに協調して動作するコンピューティング環境を目指す [1][2]。

そこでは、予め動作の順序が決められた状況にプログラムが置かれるのではなく、そのネットワークコンピューティング環境に存在する各プログラムが置かれている状況から、サービスに対して期待される目的に向かって、成すべきアプローチをサブゴールとして自らが作り、それを実行することによって、全体として最終の目的を達成する、ネットワークとコンピューティングによる処理機構を目指す。

本報告では上記の枠組の中で、マルチメディアを扱う場合を対象として、時間関係を含む構造関係を如何にして保存しながら処理を実施できるか [3]、という点に焦点を当て、そこに発生し得る様々な要件を挙げ、それらに対処するための対策案を示し考察した内容を報告する。

## 2 スレッドの分配機構

分散オペレーティングの研究において、例えば大西等の DM-1 [4] では、複数のノードで並列に実行されるプログラムをノード間通信を意識せずに記述できる。そこでは、ネットワークを介したスレッド間通信の頻度が大きくなり、このために、ネットワークの通信量を適当な量に保つためにスレッド間の通信量を考慮する。例えば、Mach [5] では、資源の割り当ての単位 (Task) とプロセッサ割り当ての単位 (Thread) を、分割する事によって、オーバーヘッドの小さい処理の実行を狙っている。

1つのノードの中でのタスクとN個のスレッドと

の間の応答時間は、実行時間を  $T_e$  とすると、 $n \times T_e$  である。他方、1つのノードにあるタスクとN個のノードにあるスレッドとの間での応答時間は、通信時間を  $T_c$  とすると、 $T_e + T_c$  となる。ここで、通信時間  $\Delta T_c$  と、処理時間  $\Delta T_e$  の比が、

$$\Delta T_c / \Delta T_e > n - 1$$

となった場合に、N個のノードにまたがるN個のスレッドの分配を制御する。筆者等は、スレッドの分配を制御する代わりに、次の2つの方法を検討する。

- $n =$  固定の場合に、 $T_c$  をいかにして小さくするか、
- スレッドの制御を行う手段として、通信状況をアプリケーションが観察して、要求するサービスの質を考慮しながら、許容できる範囲の質が確保できるようにスレッドを分配制御する。

本報告では、前者の場合について、マルチメディアを例にして周りの環境の様子を観察して自身の振る舞いを調整する試みを行なう。

処理の群としての Task が要求されて Queue に入り、それが処理される Thread にそれぞれが図1のように割り付けられる。この割り付けがネットワークを介して行なわれる際に、通信時間が発生する。

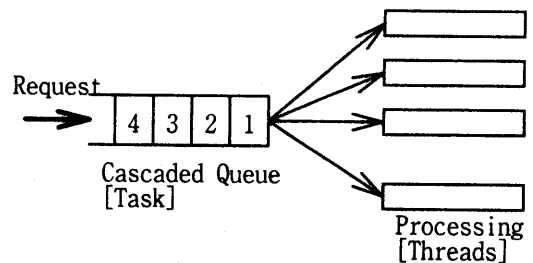


図1: Queueにある要求を Thread にアサインする

## 3 検討のモデル

マルチメディアを構成する要素がネットワーク上に存在し、あるクライアントがこれらのマルチメディア要素を構造的あるいは時間的連結の関係を保存する狙いをもって、ネットワークを介してアクセスして獲得して目的通りに処理を実行する場合を想定する。

### 3.1 マルチメディアの要素

意味的な構造関係が時間同期の要素と強く関係して形成されている。そのために必要とされる要素を、文献 [3] に示した。その実際を観察するために、分散システムにおける通信サービスとプロトコルの仕様記述の手法を用いてその動作をシミュレーションにより観察した [6]。そこでは、マルチメディアの時間変化要素とその記述を汎用的に行う事を検討し、プロトコル設計用言語 T-PROMERA (Protocol Meta Language) により、適用している。

### 3.2 クライアント側からの要求

基本的には、通信トラフィックのモデルにおける従属接続型の待ち合わせ理論のモデルに従って、要求が処理系に渡され、そこでの処理結果が戻ってきて、要求が満たされる。一般には、M/M/S システムのモデルで論じられる [7]。

そこでの、処理バッファが実際には有限 (N) となるために、要求の発生元 (I) に対してフィードバックがかかり図 2 のようにブロッキング (Pb) される。

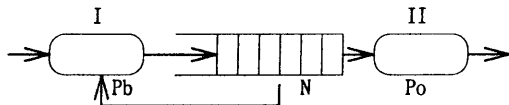


図 2: フィードバックによるブロッキング

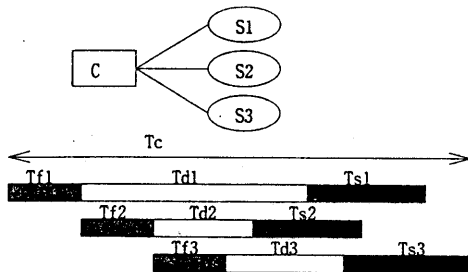


図 3: 予約の時刻と構成要素の獲得時刻との関係

### 3.3 サーバ側からの応答

要求を受けた複数のサーバから要求を発生したクライアントに向けて、一斉にあるいはシリアルに応

答を図 3 のように返す。それぞれの対象情報 (Si) に対して、直列に起動を掛けられ (Tfi)、それぞれのネットワーク状況に依存した時間 (Tdi) が経過した後で、それぞれの処理時間 (Tsi) を費やして、獲得される。この時に、ネットワーク上は種々の競合が発生して、必ずしも期待された構造関係を保存して応答がなされる保証はなくなる。

## 4 要求割当方式による予測

この応答の際のネットワークの利用環境として、要求割当方式 (Demand Assignment Technique) の手法によりネットワークの様子を観る。即ち、個々のサーバへの要求とその応答が、ネットワーク伝送路を全く無駄なく利用して送受を実施できれば、個々のマルチメディア要素をそれぞれのサーバから順番に直列に受け取ることが出来、期待する構造関係を崩すことなく受け取る事が出来る。

しかし、一般には、受け取るクライアント側のバッファ処理とか、バケットの長さの制限から、クライアントと特定のサーバの間でのやり取りで、ネットワーク上での無駄が発生してしまう。この間隙を縫って、クライアントは他のサーバとの間で、送受を実行できる機会が発生する。

データバケットの衝突を避けるために、各ノードがデータバケット伝送用のタイムスロットを事前に予約する予約プロトコル方式が知られている [8][9]。

### 4.1 予約回線の設定

予約回線の設定には、スロット式アロハ・プロトコルによるものと TDMA によるものがある。前者は、予約回線の接続時に衝突を生ずるが、予約バケットの接続時間がデータバケットに比べて極く短いので、予約回線の容量は小さくなる。後者は、予約回線接続時の衝突はないが、ノード数に比例した予約回線の容量を必要とする。

### 4.2 データ回線の設定

データ回線の設定れも、集中制御方式と分散制御方式がある。前者は、集中制御装置が空スロットの状態を管理していて要求に対して、空スロットの位置を教えるものである。後者は、予約回線でブロードキャストされる予約申告バケットを全ノードが監視し、その情報から空スロットの位置を把握する。

## 5 時間予約によるアクセス

筆者等は、マルチメディア処理を時間制約を備えた並行プロセスと観て、各メディアを再生する場合の時間関係の要求の保存を、形式記述技法を用いてシミュレーションによりその振る舞いを観測した。そこで検討した、LOTOS-T あるいは T-PROMELA では、時間指定を記述できるように拡張されている [6]。

### 5.1 予約時間の概念

ヒューリスティックにネットワークの状況を試行バケットを送り出してそれが戻ってくる時間を観測して、実際に送出するバケットの時間位置を制御する方式が試みられている [8][9]。これは、種々のコンピュータが混在する環境において、ネットワーク上の各コンピュータで刻々と変わっていく負荷状況とジョブの特性に影響を受ける環境を、比較的成本が小さく且つ自分が配送したジョブの特性のみを知ればよいことを考慮して、ビジーバックによりネットワーク環境の情報を得る方法である。これにより、過去に配送したジョブの内、未だ結果の戻ってこないものの予想処理時間を予測することができる。と期待される。

マルチメディア情報をその構造関係要素を含めて表現する一手段として、図 4 に示すように木構造によって表すことができる。

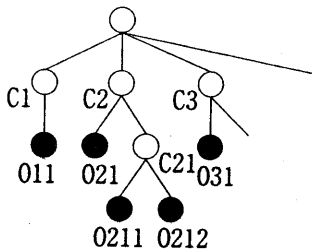


図 4: 木構造によるマルチメディア情報の構造

そこには、互いの構成要素の構造的そして時間的關係が保存されることが要求される。この木構造ではビジュアルにはこれらの時間關係を表すことは難しく、各ノードに関連する付帯情報などで、プバティあるいはプロファイル情報として付加される。これらの構成要素が、この時間關係情報を参照してネットワークを介して起動されて、サーバから送信されて、再現される。

再現 (表示・再生) させたい特定の時刻を目標に、

個別に構成要素を取り出す起動をネットワークの状況を予測して実行する。

### 5.2 予約時間の定義

唯一つの要素を対象にした場合、大きく次の時間要素が存在する。

- 起動要求がネットワークを介してサーバに届くまでの時間
- サーバ内部で処理が実行されるまでの時間
- 処理された情報がネットワークを介して要求元に届くまでの時間

ここでは、要求元あるいはサーバ内部での処理時間については検討の対象とはせず、固定とする。前述のビジーバックによりネットワーク状況の情報を獲得した後に、所要時間を予測し、その際に図 5 に示すように、起動に要する時間 ( $T_f$ ) と予測した時間 ( $T_d$ ) の分 ( $T_{es}$ ) だけ事前の時刻に起動をかけて、上の処理を実行する。

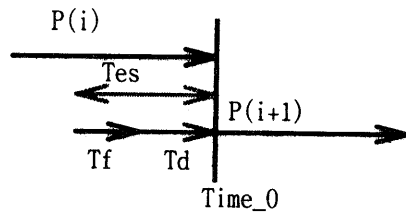


図 5: 予約時間と構成要素の獲得時刻との関係

LOTOS-T,

-a{t};B

時間  $t$  を絶対値から、 $t$  を変数とする予測時間関数として扱う。

即ち、時間要素は次のように定義する。

$t \rightarrow F(t)$

即ち、 $F(t)$  は、上述のネットワーク状況情報から得られた時間情報となる。

このような場合の振る舞いのシミュレーションは、文献 [6] において示されている。しかし、これらの構成要素は互いに種々の構造關係にあり、この關係を直接的に制御しながら実施する場合は、お互いの構造關係を考慮した時間情報を定めなければならない。例えば、並列の關係にある場合は、同時に複数

の構成要素がネットワーク上に同時に存在する事になり、互いに干渉して全送出時間が増大する。この状況を、次章において検討する。

## 6 検討と考察

パケット放送通信の手法における、パケットの衝突を回避するため、呼の設定時にデータ転送用のタイムスロットの予約を行う要求割り当て方式: Demand assignment techniques に準える。

### 6.1 伝送効率

図3にも示したように、複数のサーバから特定の要求元に向けて集中的に情報が送出されることになる。当然ながら、ネットワーク上では衝突が起こり再送パケットが多く存在することになる。割当方式には、代表的な Contention protocol 方式と Reservation protocol があり、前者では ALOHA protocol があり、後者では スロット式アロハプロトコルあるいは TDMA による予約回線の設定と、集中制御あるいは分散制御方式によるデータ回線設定がある。ここでは、固定割り付け方式による場合の伝送効率を観る。

ノードの数を  $V$  とし、1 フレームの間隔をデータスロットの数に換算して  $M$  とすると、予約のスロット数では  $M \times V$  となる。サーバ数を  $N$  とすると、予約スロット数は  $N$  となる。予約回線容量の全回線容量との比  $\gamma$  は、

$$\gamma = N / (M \times V)$$

であるから、伝送効率は、

$$1 - \gamma = 1 - N / (M \times V)$$

となる。  $V=5$  の場合の様子を図6に示す [10]。

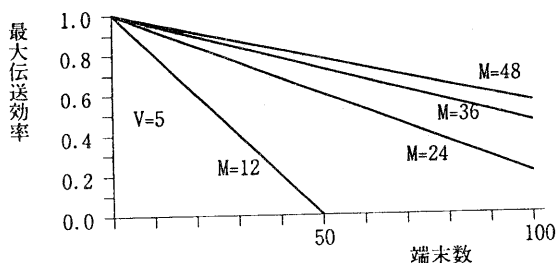


図6: 予約プロトコルの伝送効率との関係

### 6.2 予約処理との関係

一般に、予約プロトコルは、データ回線のタイムスロットを全部無駄なく利用できるために、予約回線分の損失はあるが、コンテンション・プロトコルに比べて伝送効率は向上する。その反面、予約処理のための伝送遅延時間が増大する。即ち、送出されるデータの量と、並行して同時に発生する送出処理の数に依存する。ここで、対象としているサーバからのデータの獲得では、同時に処理を要求されるサーバの数は数台と少ない。従って、データ・スロット換算の1フレームの間隔を大きくするほど効率は良くなる。

### 6.3 予約アクセス

ここで対象としている予約アクセスを、効果的にするためには、図3に示したように、同時に要求する構成要素の数に依存して、最終的にデータが届くまでの時間を見積もり、その分だけ必要とする時刻より早く、起動してデータの獲得を開始しなければならない。即ち、先の予約時間  $F(t)$  は、上述のようなネットワーク状況を考慮した情報と、処理される対象情報の特質による情報、の両方の影響を受ける事を配慮しなければならない。この詳細については、プロトコル使用に基づくシミュレーションなどを実施して定量化したい。

しかし、定性的には、マルチメディア・データの特質から、そのデータ量は極めて大きいので、この図に示した条件である  $V=5$  は、適当ではない。マルチメディア情報を扱うことを考慮すると、情報量は多いが同時に起動されるサーバの数は少ないことを前提にすると、 $V$  の値は大きく  $N$  の値は小さい条件で評価する必要がある。

伝送効率を見て、サーバから送出されるデータが平均的にどの程度の遅延で到達するかを見込む為には、コンピュータ内部でのバッファリングや処理機構の要素も含めて、より正確な処理あるいは遅延モデルを構築して検討する必要がある。

## 7 おわりに

ネットワーク環境も技術の進歩が早く、現在は ATM が注目を集めており、将来への期待も大きく様々な研究開発が進んでいる。コミュニケーション速度が飛躍的に向上しても、ネットワーク上を行き交うデータの速度には、物理的制約や交信手順のための制約で、遅延は有限であり零にはならない。遅延の量が

特定のサービスを観た場合に、許容できる範囲であるか否か、あるいは許容できずとも、他の手段で補完出来るか否か、などの要素が存在し得る。ここでは、遅延時間の絶対値はともかく存在し得る遅延時間を如何にして、最小にするかという視点から、その一つのヒューリスティックではあるが、予約の概念を検討した。

ネットワークを利用する場合、そこで行われる手続き（各種のプロトコル）は当然ながら汎用的に構築されており、伝送される速度が技術の発展とともに速くなるが、絶対値はともかく有限である。このような環境で、扱う情報の種類が多岐にわたる、操作をする際に、その特徴に従った処理が必要になる。

今後は、自律型の協調分散システムを実現していく上で、アプリケーションの間での協調のための情報交換に向けて検討を進めていく予定である。

## 参考文献

- [1] 小塚宏、佐藤文明、宮崎一哉、福岡久雄；分散協調環境の提案、情報処理学会、マルチメディア通信と分散処理 59-11、 pp.81-88(1993,1)
- [2] Fumiaki SATO, Hiroshi KOZUKA, Kazuya MIYAZAKI, Hisao FUKUOKA; Noah: An Environment for Distributed Autonomous Systems., ISADS'95,(1995,4)
- [3] 坂下善彦；マルチメディア処理の要素、情報処理学会、マルチメディア通信と分散処理、 pp.183-190(1994,6)
- [4] 大西祐史、岡部寿男、大久保英嗣、津田孝夫；分散オペレーティングシステム DM-1 におけるスレッド分配機構、情報処理学会第 48 回全国大会,4-51,2H-8,(1994)
- [5] 西尾信彦； Real-Time Mach 3.0 上の連続メディア処理のための協調サーバ群の設計と実装、情報処理学会、システムソフトウェアとオペレーティング・システム、 63-8,(1994,3)
- [6] 大田剛、渡辺尚、水野忠則；時間変化要素を含む分散型通信システムの記述法、情報処理学会、マルチメディア通信と分散処理、 pp.x-y(1994,x)
- [7] 野口正一、木村英俊、大庭弘太郎；岩波講座 情報科学-5、情報ネットワークの理論、岩波書店,1982.
- [8] 染葉佳代子、渡辺尚、大田剛、水野忠則；分散型ディスパッチャを用いたジョブ配送法について、情報処理学会第 48 回全国大会 7D-3,pp.1-317 - 1-318(1994,x)
- [9] 渡辺尚、大田剛、水野忠則；双方向ビジーバックを用いた動的負荷分散方式について、電子情報通信学会春季全国大会 (1994)
- [10] Lam,S.;Satellite packet communication - multiple access protocol and performance. IEEE Trans.Commun. Vol.COM-27(1979)