

## 予防的流量制御による通信の実時間性の改善

兵頭 和樹<sup>†</sup> 小森 翔太<sup>†</sup>  
市村 哲<sup>†</sup> 田胡 和哉<sup>†</sup>

ロボットや自動車等の制御系では、実時間制御が多数のプロセッサを用いて行われており、リアルタイム性を持つ通信系が重要である。一方、コスト等の観点から広く一般に利用されているイーサネットを従来専用の通信媒体を利用していた分野へ適用したいという要求が高まっている。本研究では、多数のプロセッサが協調して実時間制御を行う分散実時間制御機構を対象に、スイッチ接続のイーサネットを利用して低遅延でゆらぎの小さい実時間通信と高スループットの非実時間通信の両立を目指す。本研究では、実時間タスクの周期性に着目し、実時間通信の実施を予測し事前に流量制御を行うことで通信の実時間性を改善する予防的流量制御を提案する。提案する方式の実現法を検討するとともに評価実験により有効性を確認したので報告する。

### A Real-time Communication with Preventive Flow-control

KAZUKI HYODOU,<sup>†</sup> SHOTA KOMORI,<sup>†</sup> TETSU ICHIMURA<sup>†</sup>  
and KAZUYA TAGO<sup>†</sup>

In this paper, we propose a new approach toward achieving low-latency real time communication and high throughput non real time communication over switched ethernet. Real time networks are important in distributed real time controlling systems for mechanical systems such as humanoid robots. Otherwise, it is demanding application, where ethernet replaces CAN for performance and cost reasons. However, because ethernet switches lack build-in policing features, nodes connected by switched ethernet need to be cooperative. We focus that the real time task for feedback controlling are executed periodically. We design and experiment preventive flow control that is limiting the flow of non real time communications in advance of real-time transfers.

#### 1. はじめに

ロボットや自動車等の制御系では、複雑な実時間制御が多数のプロセッサを用いて行われており、リアルタイム性を持つ通信系が重要である<sup>1),2)</sup>。さらに、このような複雑な分散処理系では、実時間通信以外の通信も行われることが考えられる。たとえば、実時間制御系の性能を段階的に改善するためには、学習系やシミュレーションと実際の実時間制御系を結合して運用する必要がある<sup>3)</sup>。そのため、測定結果や種々の制御表がプロセッサ間で交換されなければならない。このとき、システムのコストを考えると実時間性を要する通信とそうでない通信は、同一の通信経路を用いて行われることが望ましい。同様に、一般に広く利用されている汎用の通信媒体を利用できることが望ましい。通常のスイッチ接続によるイーサネットを用いて、

制御に用いられる実時間性を要する通信 (以下、実時間通信) とそうでない通信 (以下、非実時間通信) を同一の通信経路上で共存させるための方法を考えてみる。

一般に、機械系の制御では、周期的な観測とそれに基づく計算が行われている。したがって、制御系を分散化すれば、ネットワークを用いて周期的な実時間通信が行われることになる。分散実時間制御系を安定に動作させるためには、通信キュー長を制限し、このような性質を持つ通信を遅延やロスを起こすことなく伝送することが重要である。非実時間通信を混在させると、通信系の各部にあるパケットキューの長さが長くなり、場合によってはパケットロスや遅延を起こす可能性が生じる。この問題を、パケット優先度等の特別なネットワーク機能を利用せずに、汎用の通信媒体で解決することがここでの目標である。

ここでの実時間性の著しい特徴として周期性が挙げられる。実時間通信の実施にあわせて事前に流量を変更させることでパケットキューの長さを適切に管理する方法を考える。具体的には、実時間通信の周期から

<sup>†</sup> 東京工科大学コンピュータサイエンス学部  
Department of Computer Science, Tokyo University of  
Technology

実施される時刻を予測し、予告パケットを実時間通信と衝突する経路に非実時間通信を行っているノードに送出する。予告パケットにより、たとえば必要な時間だけ非実時間通信を停止することで、実時間通信への影響を排除しながら効率良く非実時間通信を行うことが可能となる。ここでは、予告パケットによる流量制御法について検討するとともに、方式全体の有効性について実システムに基づく評価を行った。評価実験では、同等の遅延特性を達成する場合、提案方式は静的な流量割り当てに比べ約 10 倍の非実時間通信の平均流量を利用可能であるなど、提案方式の有効性を確認した。

## 2. 背 景

### 2.1 分散実時間制御機構

はじめに、ここで対象とするシステムとして分散実時間制御機構について述べる。

分散実時間制御機構とは、複数の計算機を通信回線を用いて結合した処理系を用い、全体としてロボットや自動車等の複雑な機械システムの実時間制御を行うための機構である。単一の計算機を用いた実時間制御に比べ、全体として大きな処理能力が得られ、機械の複雑さにあわせて制御系も柔軟に再構成できる。また、機械装置と計算機を組み合わせた機械モジュールを実現することで、コストの軽減につながるなどの大きな利点を持っている。

一方、全体姿勢の制御や学習による最適化処理など、複数のモジュールの状況を総合して判断を下す必要があり、制御系全体ではきわめて複雑な分散協調処理が実時間で行われることになる。これを実現するために実時間性を持つ通信系が重要である。

### 2.2 分散実時間制御機構のための通信

前節で述べた分散実時間制御機構に要求される通信として以下のものが挙げられる。

- (1) フィードバック制御に要求される通信
- (2) 例外事象発生のお知らせに要求される通信
- (3) 学習系の実現に要求される通信

サーボ系の制御では、周期的に制御プログラムを実行し、センサ等から得られる実際の制御量や測定値を基に微調整を行うフィードバック制御が行われる。対象とするシステムは、センサやアクチュエータの個々に制御装置を付加したモジュールによって構成されることを想定しており、システム全体が協調して動作するためには、フィードバック制御に必要なデータをネットワークを介して各モジュール間で共有する必要がある。実際に、制御理論の分野において、フィードバッ

ク制御内にネットワークを含むさいの制御の安定性に関する研究が行われている<sup>4),5)</sup>。(1)は、このフィードバック制御に必要なデータを、他のモジュールやセンサから周期的に得るための通信である。これらのデータは周期的に与えられ更新され続けるので、フィードバック制御に致命的な問題を生じない限りはある一定の割合でデータが消失しても良い<sup>5)</sup>。しかし、制御への影響を考えるとデータ消失を可能な限りおさえ、制御周期を守れるよう低遅延で通信が完了される必要がある。

(2)は、接触、転倒や妨害の発生などの突発事象や機械の故障を知らせるための通信である。これらのデータはサイズは最小であるが、確実かつ可能な限り速く届けられる緊急パケットとして定義される必要がある。

また、実時間制御系の性能を段階的に改善するためには、学習系やシミュレーションと実際の実時間制御系を結合して運用することが必要である。このため、測定結果や種々の制御表をプロセッサ間で交換するための通信が必要である。これらのデータは実時間性を必要とすることは稀であるが、比較的データサイズが大きく、要求される平均流量も高い。また、制御効率改善の観点から、実時間通信を妨げない範囲で、可能な限り早くデータが届けられることが要求される。

分散実時間制御機構においては、これらが個別に達成されるのではなく、複数のカテゴリの通信を統括的に管理し、全体として可能な限り要求事項が満たされるようにすることが重要である。

### 2.3 イーサネットと実時間通信

制御機器を相互接続する通信媒体として専用の規格が用いられてきたが、インターネット利用者の拡大によるイーサネット機器の普及を背景に、イーサネットを、従来、専用規格が利用されていた分野へも適用したいという要求が高まっている。機器の価格や入手性、豊富なソフトウェア資源が利用可能であり、イーサネットを利用することはコスト等の観点から非常に利点大きい。

実時間処理系にイーサネットを利用することが困難であった理由の一つは、その通信制御に CSMA/CD (Carrier Sense Multiple Access with Collision Detection) 方式を採用していることにある。この方式では、伝送経路内で衝突(コリジョン)が発生すると、乱数から算出する時間を待機してから再送処理を行う。このため通信の遅延時間を保証することができない。これに対して、Bello<sup>6)</sup>らは、バス共有型のイーサネット上での衝突緩和による実時間性の向上を達成している。

一方、イーサネットでは LAN を構築するさいにスイッチを利用することが一般的になっている。スイッチを利用することでコリジョンの発生は緩和できるが、スイッチの負荷が大きくなるとスイッチ内のバッファのキュー長が長くなり遅延の増大やバッファオーバーフローによるパケットロスが起こりえる。このため、非実時間通信を混在させる環境では、そのままでは通信の実時間性を保証することは困難である。

### 3. 予防的流量制御

本章では、前述のイーサネット上で実時間通信を実現するさいの問題点を考慮し、分散実時間制御機構のための通信をスイッチ接続のイーサネットを用いて実現するための一方式について検討する。

#### 3.1 基本方針

2.2 節で述べた、実時間分散制御機構のための通信として、本研究では低遅延で遅延のゆらぎが小さい実時間通信と高スループットの非実時間通信の両立を目標とする。

2.3 節で述べたとおり、スイッチ接続のイーサネット上で実時間通信を妨げる要因は、スイッチを含む通信経路にある各バッファのキュー長が長くなり遅延の増大やパケットロスが起こることによる。一般に、遅延とスループットはトレードオフの関係にある。スループットを高めると通信経路上のキューにパケットが溜る可能性が増え、遅延の増加につながる。逆に、スループットを抑えればキューにパケットが溜る可能性は減り、キューでの待ち時間が減ることで各パケットの遅延が小さくなる。

非実時間通信の総平均流量を大幅に制限することである程度キューの長さを管理することは可能であるが、瞬間的に通信が集中するとキューが溢れる可能性がある。また、この方法ではスループットとの両立はできない。低遅延の実時間通信と高スループットの非実時間通信を両立するためには、総平均流量を大幅に制限することなくキュー長を適切に管理する方式が必要である。

一方、2.1 節で述べたように、ここでの実時間通信の著しい特徴として周期性が挙げられる。本研究では、この実時間通信の周期性を利用し、実時間通信の実施にあわせて事前に非実時間通信の流量を制御することでキュー長を適切に管理する方式を提案する。

通常時は、同一経路の流量がネットワーク性能を超えないよう流量制限を行うことでキュー長が大きく成長するの防ぎ、周期的な実時間通信の実施にあわせて非実時間通信の流量を大幅に制限する。たとえば、実

時間通信の実施直前に非実時間通信を一定期間停止すれば、各パケットキューに溜っていたパケットが処理され、即時に実時間通信のパケットを処理することが可能となる。

これまでに述べた、実時間通信の実施前にキュー長が適切になるようネットワーク上の通信流量を制御する方式を予防的流量制御と呼ぶことにする。以下、予防的流量制御の詳細について述べる。

#### 3.2 流量制御方式

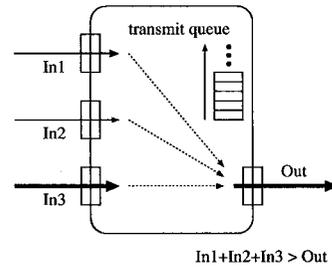


図 1 キュー長の成長

Fig. 1 Growing queue length.

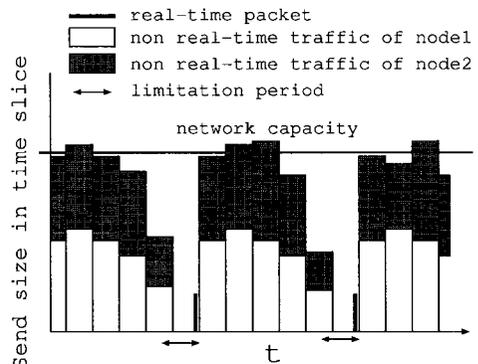


図 2 流量制御の例

Fig. 2 Example of flow-control.

本節では、予防的流量制御における各ノードへの流量割当および各ノード上での流量制御の方式について述べる。

まず、流量割当について考える。スイッチにおいてキュー長が成長するのは、同一ポートへ出力されるパケットが複数のポートからネットワーク性能を超えて入力された時である(図 1)。スイッチでキュー長が大きく成長するのを防ぐためには、同一ポートへ出力する各ポートに入力される流量を  $r_i$  とし、ネットワークの処理性能を  $C$  とすると、 $\sum_{i=1}^N r_i < C$  でなけれ

ばならない。これは、同一経路の総流量がネットワーク性能を超えないように各ノードの送信量を制御する必要があることを意味する。しかし、これだけでは流量の時間的なゆらぎにより微視的にみた場合にネットワーク性能を超える流量が発生する可能性があり、完全にキュー長の成長を防ぐことはできない。

本研究では、同一経路の総流量が概ねネットワーク性能を超えないように各ノードに流量割当を行うとともに、各ノードでは単位時間当たりの送信量を制限することで流量制御を行う。各ノードの単位時間当たりの送信量は、実際の利用状況や、実時間通信の実施にあわせて動的に変更する(図2)。実時間通信の直前の送信量を大幅に制限することで、キューに溜っているパケットが処理され、実時間パケットのキューでの待ち時間を大きく減らすことができる。また、実時間通信の実施までに余裕のあるタイムスロットでは、ネットワーク性能を超える流量が発生しても許容でき、平均して高いスループットを得ることが可能である。

### 3.3 予告パケット

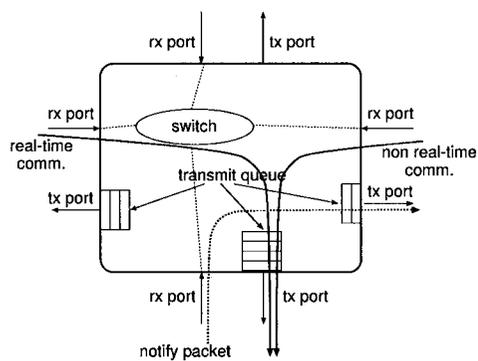


図3 予告パケット送信の例  
Fig. 3 Example of sending notify packet.

前節で、各ノードの単位時間当たりの送信量が周期的な実時間通信の実施にあわせて制限されることを述べた。この実時間通信の実施を各ノードに通知するために、事前に予告パケットを送信する。ここでは、この予告パケットについて考える。

実時間通信の実施にあわせて流量制限される通信は、その実時間通信に影響する経路のみの通信を制限すれば良い。具体的には、実時間パケットを送出するノードからの非実時間パケットの送信、実時間パケットを受信するノードへの非実時間パケットの送信である。

実時間パケットを送信するノードは、自分で実施を予測可能であるので、実時間パケットを受信するノード

に対して非実時間パケットを送信しているノードへのみ予告パケットを送れば良いことになる。

予告パケットの通知は、実時間パケットを受信するノードから、そのノードへ非実時間パケットを送信するノードへのみ行う(図3)。

これらにより、実時間通信の実施予告による流量制限の影響を最小限に抑えられるとともに、実時間通信を阻害することなく効率の良い非実時間通信を行うことが可能となる。

### 3.4 緊急パケット

例外発生のお知らせに利用される緊急通信は、その実施を予測することが困難であり、これまでに述べてきた実時間通信の周期性を利用した方式を適用することはできない。

しかし、適切な流量制御を行うことでスイッチのキュー長を管理することが可能であり、通信の遅延を予測することができる。本システムでは、周期的な実時間通信の有無に関わらず、3.2節で述べた通常時の流量制御を行う。また、緊急パケットはブロードキャストされる特殊な予告パケットとし、これを連続して送信することで、ネットワークは定常状態となり緊急パケットは安全に伝達される。また、緊急パケットの送信はACKを受け取ることで停止する。これにより、緊急パケットの到着が保証される。

## 4. 設 計

本章では、3章で述べた予防的流量制御を行う機構の設計について述べる。

### 4.1 インタフェース

予防的流量制御は、下位層での事前の流量制御により低遅延でゆらぎの小さい実時間通信と高スループットの非実時間通信を両立する方式である。したがって、アプリケーションは従来のプロトコルおよびインタフェースを利用できる。ここでは、UDP/IPとソケットインタフェースを利用することを考える。

予防的流量制御では、非実時間通信の単位時間当たりの送信量を制御するとともに実時間通信の即時送信が行われなければならない。そこで実時間通信と非実時間通信を区別するために新たにSOCK\_RT\_DGRAMというソケットタイプを追加する。これは、即時送信されるデータグラム型のソケットを意味する。

これにより、従来のソケットを利用して作成されたプログラムを少ない変更で実時間通信を行うことができる。

### 4.2 周期の検出

実時間通信の周期の検出は受信側で行う。実時間通

信の受信を行うアプリケーションは、ソケットタイプに SOCK\_RT\_DGRAM を指定し、あるポートをバインドして受信待ちをする。SOCK\_RT\_DGRAM でバインドされたポートでは、パケットの到着時刻を記録し、到着間隔を計測することで周期を検出する。

このように受信側で実時間通信の周期を検出することで、UDP/IP プロトコルに変更を加えることなく実現できる。

#### 4.3 仮想ネットワークを利用した流量制御

非実時間通信を行うノード間で TCP/IP の仮想ネットワークを利用し、仮想パスを構築することを考える。仮想パスは、初めて非実時間通信が行われるさいにノード間で接続を確立し、以降の同一ノード間の全ての非実時間通信はこの仮想パスを通して行われる。非実時間通信を行うソケットが全て閉じられると仮想パスも閉じる。

非実時間通信は、単一の TCP/IP リンクを用いて IP パケットをカプセル化することで、総平均流量の制御を行う。これにより、流量管理は仮想パス毎に仮想ネットワークを実現する TCP/IP の Window 管理機能を用いて行えば良く、流量制御を単純化できる。また、予告パケットは実時間通信を受信するノードから、仮想パスが張られている相手ノードにのみ送信すれば良い。

### 5. 実装

本章では、提案する方式の有効性を確認するために行った実装について述べる。

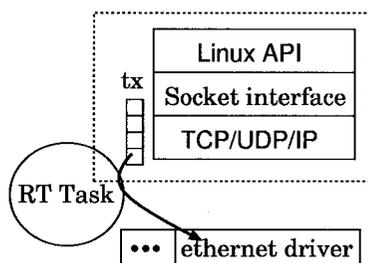


図 4 流量制御機構の実装図  
Fig. 4 Implementation of flow-control system.

今回、実時間通信の事前予告による流量制限が有効に働くことを確認するために、各ノードでの流量制御機構と予告パケットの処理部分を実現した。

非実時間通信の流量制御は、RTLinux を用いて、周期的にスケジューリングされる実時間タスクによって実現した。この実時間タスクは 1 回スケジューリング

される毎に、予め指定されたサイズ分のパケットを上位層の送信キューから取り出しイーサネットドライバに渡す(図 4)。予告パケットを受け取ると、次のタイムスロットの送信量をゼロとする。これにより非実時間通信の流量制御を行う。今回、実時間タスクの周期は 1 ミリ秒とした。

また、予告パケットは実時間パケットを送信するノードからブロードキャストによって送信される。

### 6. 評価実験

#### 6.1 実験方法

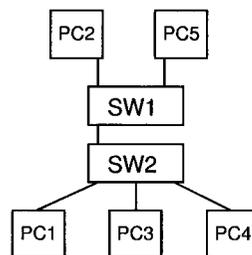


図 5 実験装置図  
Fig. 5 Experimental system.

表 1 実験装置の構成

Table 1 Construction of Experimental System.

各 PC の構成	
CPU	Intel Pentium-M 1.5GHz
NIC	Intel EtherExpress 100
OS	RTLinux 3.2pre with kernel-2.4.20
スイッチ	
SW1	BUFFALO LSM10/100-8W/H
SW2	BUFFALO LSW-TX-5EBC

実験は図 5 に示す環境で行った。実験装置の構成は表 1 に示すとおりである。

周期的に実施される実時間通信として PC1 から PC2 へ、RTLinux を用いて 10 ミリ秒毎に UDP パケットを送信している。また、非実時間通信として PC3, PC4 から PC5 へ通常の TCP 通信を行った。

受信ノードでの影響を排除し、スイッチでの実時間通信への影響を見るために、2つのスイッチをカスケード接続し、PC1 から PC2 への実時間通信と、PC3, PC4 から PC5 への非実時間通信がスイッチでのみ競合するようにしている。

PC3, PC4 から PC5 への非実時間通信は、一定サイズのデータのバースト転送と休止状態を繰り返す。

## 6.2 解 析

表 2 RTT の測定結果 (制御無し)

Table 2 Experimental Result on RTT (without control).

非実時間通信	最大	平均	標準偏差
無	132.1	122.4	0.93
有	10565	6341	4642

単位：マイクロ秒

まず、スイッチの性能を解析するために、流量制御がない場合の実時間通信のラウンドトリップタイム (RTT) を測定した。測定は、非実時間通信が無い場合と有る場合とで行った。非実時間通信は、PC3, PC4 からそれぞれ 60KB, 45KB のデータのバースト転送を一定間隔で繰り返した。このときの平均流量はそれぞれ、48Mbps, 36Mbps である。

表 2 に示すように、非実時間通信が行われていない場合の実時間通信の RTT は、平均 122 マイクロ秒でありそのばらつきも非常に小さい。一方、同時に非実時間通信が行われているさいは、平均で 6341 マイクロ秒と“無”の場合の約 50 倍となっている。また、最大では 10 ミリ秒を超えており RTT のばらつきも非常に大きくなっている。このときの平均流量の合計は 84Mbps であり、ネットワークは飽和していないが、バースト転送が集中して行われるためスイッチのキューにパケットが溜り、実時間通信の遅延が大きく増加している。

ここでは RTT を測定しているが、非実時間通信は片方向のみで行われており、測定した RTT の増加分はほぼ片道通信遅延の増加と考えることができる。

予防的流量制御を導入することで、負荷の大きい非実時間通信が行われている場合でも低遅延でばらつきの小さい実時間通信が可能であることを示す。

### 6.3 実時間通信の遅延のばらつき

表 3 RTT の測定結果 (制御有り)

Table 3 Experimental Result on RTT (with control).

制御	最大	平均	標準偏差
制御無し	10565	6341	4642
静的流量割当	551.6	352.9	105.9
予防的流量制御	184.1	123.8	4.1

単位：マイクロ秒

流量制御を導入したさいの RTT の測定結果を表 3 に示す。比較として流量制御が無い場合、および、予告パケットを送信をせずに静的流量割当を行って流量制御を導入した場合も同時に示している。非実時間通信の流量割当は、PC3, PC4 の平均流量がそれぞ

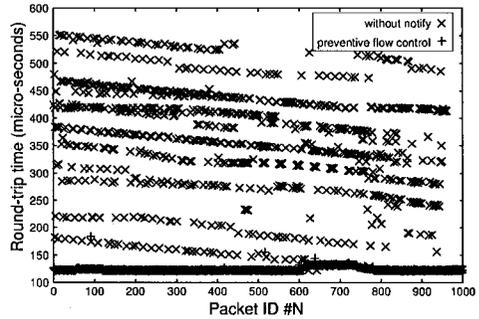


図 6 RTT のばらつき  
Fig. 6 Dispersion of RTT.

れ 48Mbps, 36Mbps で合計が 84Mbps となるように行った。これは、実験環境での最高スループットであった 94Mbps の 0.9 倍にあたる。予防的流量制御では、流量制御の単位時間毎の送信量がネットワーク性能を大きく超えることはなく、今回の実験では実時間通信の周期 10 ミリ秒毎に 1 ミリ秒の間通信が停止するためこの値となる。予防的流量制御を導入することで、長期的な平均流量は低下するが、短期的にはネットワーク性能の限界に近いスループットを達成できる。

静的に流量を割り当てて流量制御を行った場合でも、制御が無い場合に比べ大きく遅延が改善していることがわかる。また、予防的流量制御を導入することでさらに遅延が改善している。図 6 に示すように、特に遅延のゆらぎの改善が大きい。予防的流量制御を導入したさいは、ほぼ 120~130 マイクロ秒の遅延に納まっていることがわかる。これは、静的流量割当では実時間パケットの送信が非実時間パケットの送信と衝突する可能性があるのに対して、予防的流量制御ではそれぞれの送信が衝突することが無いためである。

同様の実験において静的流量割当で、予防的流量制御とほぼ同等の遅延特性を得るためには総平均流量を 10Mbps 以下にまでする必要があった。同等の遅延特性を実現するにあたって、予防的流量制御が非常に高いスループットを達成していることがわかる。

なお、この実験において実時間通信のパケットロスを確認されていない。

## 7. 関連研究

スイッチ接続のイーサネット上での実時間通信実現に関する研究として、EtheReal<sup>7)</sup>、Tenet<sup>8)</sup> や Loeser<sup>9)</sup> の研究がある。EtheReal は、実時間通信と非実時間通信が共存するためのスイッチそのものを構成する試みであり、既存のスイッチを利用する本研究とは

目的が異なる。また、Tenet は下位から上位まですべての層が実時間性を保証するプロトコルになっており、アプリケーションが既存の UDP/IP を利用して実時間通信を行う本研究とは異なっている。

Loeser らは、Traffic Shaping という単位時間毎のバースト送信量を厳密に制限する方式でスイッチ接続のイーサネット上でハードリアルタイム性を保証できることを解析的に示している。この方式では、バースト送信量を管理する単位時間を 1 ミリ秒としたとき、非実時間通信に 92Mbps の流量が割り当てられた状態で、1380 マイクロ秒の遅延を保証できるとしている。しかし、その流量の割当は静的であり、実時間通信の実施前にキュー長を管理するために動的に流量制御を行う本研究は Loeser らの試みの発展と位置付けることが可能である。また、流量管理の単位時間と非実時間通信の流量が同じであれば、本研究で提案した方式の方がより低遅延での実時間通信が可能である。

## 8. おわりに

本論文では、分散実時間制御機構を対象として予防的流量制御による通信の実時間性の改善について提案した。ロボットや自動車等の、多数のプロセッサを用いて実時間制御を行う制御系では実時間性を持つ通信が重要である。また、このような分散処理系では実時間通信以外の通信も行われる。システムのコスト等を考慮するとこれらの通信は同一の通信経路を用いて行われることが好ましい。同様に、一般に利用されている汎用の通信媒体を利用できることが望ましい。

本研究では、分散実時間制御機構の通信媒体として一般に広く利用されているスイッチ接続のイーサネットを利用し、実時間通信と非実時間通信を共存させる方法について述べた。提案した方式では、実時間タスクの持つ周期性に着目し、実時間通信の実施にあわせて非実時間通信の流量を制限することで、通信経路にあるパケットキューの長さを調節することで通信遅延のばらつきを抑える。

実際の実装では、単位時間毎の送信量を抑制することで各ノードの流量を制御するとともに、実時間通信の実施前に予告パケットを送信する。予告パケットを受け取ると次のタイムスロットの送信量はゼロとした。また、予告パケットの送信は実施される実時間通信の経路に影響するものだけでなく、実時間通信への影響を排除し効率良く非実時間通信を行うことができる。

流量制御の単位時間を 1 ミリ秒に設定して行った評価実験では、流量制御が無い場合に比べ大幅に遅延が改善することを確認した。また、動的に流量制御を行

うことで、静的な流量割当に比べてもさらに遅延が改善することも確認した。このとき、同等の遅延特性を静的流量割当で達成するためには、総平均流量を提案方式の約 10 分の 1 にする必要あり、提案方式が非常に高スループット下で低遅延の実時間通信を実現することを確認した。これらにより、提案した実時間通信の実施にあわせて事前に流量制御を行う方式が有効であることを示した。

現在、ヒューマノイドロボットを準備しており、本研究で提案した方式の適用を進めている。今後、実際のロボットの制御系において評価を行う予定である。

## 参考文献

- 1) Kaiser, J. and Mock, M.: Implementing the Real-Time Publisher/Subscriber Model on the Control Area Network (CAN), *2nd Int'l Symposium on ObjectOriented Distributed Real-Time Computing Systems* (1999).
- 2) Mock, M. and Nett, E.: Real-Time Communication in Autonomous Robot Systems, *Proc. the 4th International Symposium on Autonomous Decentralized Systems (ISADS 99)*, pp.34-41 (1999).
- 3) 兵頭 和樹, 田胡 和哉, 伊藤 雅仁, 井上 亮文, 宇田 隆哉, 市村 哲, 星 徹, 松下 温: ロボット用オペレーティングシステム NOAH の構想, コンピュータシステム・シンポジウム論文集, pp. 113-120 (2004).
- 4) Seiler, P. and Sengupta, R.: An  $H_\infty$  Approach to Networked Control, *IEEE Transactions on Automatic Control*, Vol. 50, No. 3, pp.356-364 (2005).
- 5) Ishii, H.:  $H^\infty$  control with limited communication and message losses, 第 34 回 制御理論シンポジウム (2005).
- 6) Bello, L. L., Kaczynski, G. A. and Mirabella, O.: Improving the Real-Time Behavior of Ethernet Network Using Traffic Smoothing, *IEEE Transactions on Industrial Informatics*, Vol. 1, Issue 3, pp.151-161 (2005).
- 7) Varadarajan, S. and Chiueh, T.: EtheReal: A Host-Transparent Real-Time Fast Ethernet Switch, *Proc. International Conference on Network Protocols (ICNP)*, October (1998).
- 8) Banerjee, A., Ferrari, D., Mah, B.A., Moran, M., Verma, D.C. and Zhang, H.: The Tenet Real-Time Protocol Suite: Design, Implementation and Experience, *IEEE/ACM TON*, Vol.4, No.1, pp.1-10 (1996).
- 9) Loeser, J. and Haertig, H.: Low-latency Hard Real-Time Communication over Switched Ethernet, *Proc. 16th Euromicro Conference on*

