

# 広帯域大規模分散環境における時間の共有-グループ同期-

山口 政博 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

## 概要

広帯域大規模分散環境上で、音声を使用した対人コミュニケーションを行なう場合、各ノード間で伝送遅延が大きく異なるため、全ノードが平等に発言を行なえないことがある。また各ノードで情報の伝達順序が異なり、対人コミュニケーションを行なう場合、問題点となる。本稿では、音声を使用した対人コミュニケーションをモデル化することで、これらの問題点を明確にする。また、これらの問題点を解決するために新たにグループ同期を定義し、グループ同期を実現するための方法を提案する。

## SyncWare on Global Distributed Broadband Environment -Group Synchronization-

Masahiro Yamaguchi, Koji Okamura and Keijiro Araki  
Graduate School of Information Science,  
Nara Institute of Science and Technology  
E-mail: {masahi-y, oka, araki}@is.aist-nara.ac.jp

## abstract

On global distributed environment, when we communicate using voice, we cannot only speak fairly, but also each node cannot receive packets in the same order as generated because transmission delay at each node is different each other. We clear these problems through human communication model using voice. In order to solve these problems, we define "Group Synchronization" and propose the algorithm to implement this synchronization.

## 1 はじめに

近年の計算機ネットワークの進歩により、遠隔地にいる人間同士が計算機ネットワークを利用して、対人コミュニケーションを行なうための様々なツール [1][2] が提案されている。これらのツールは、連続メディアを利用して対人コミュニケーションを行なう。そのため、連続メディア処理を行なう上で必要な同期機構について様々な研究がなされている [3]。

しかし、従来の連続メディアに対する研究では単一メディア内同期及びメディア間同期に焦点が置かれ、異なるノードから送信された複数の連続メディア間情報の順序まで考慮した同期については、研究されていない。

この同期について、ローカルなネットワーク上で対人コミュニケーションを行なう場合、ネットワークの遅延が無視できるほど小さかった。そのため、各ノードで情報の伝送される順序について考慮する必要がなく、複数ノードから送信された情報の順序を保証するための機構について、考慮する必要はなかった。しかし、広域分散環境上では、ネットワークの物理的な距離による伝送遅延 [4] が無視できないほど大きいため、各ノード間の遅延時間に大きな相違がある。そのため、複数ノードから送信された情報を受信する

と、ノード毎で順序が異なる場合が生じる。そこで本稿では、この矛盾を回避するための同期機構、グループ同期を定義し、実現するためのアルゴリズムについて提案する。

その際、グループ同期の必要性を考えるために、広帯域大規模分散環境上で行なう対人コミュニケーションをモデル化することにより、大規模分散環境上で対人コミュニケーションを行なった場合、生じる問題点を明らかにし、グループ同期の必要性について述べる。

## 2 広帯域大規模分散環境上でのコミュニケーション

近年のネットワーク技術の発展はめざましく、現在、上位層に対し品質保証 (QoS: Quality of Service) を行なうことができるネットワークがある。また、分散環境上で人間がコミュニケーションを行なう場合、アプリケーションが下位層に対し、必要な要求を行なうことでアプリケーションを使用することができる。しかし、アプリケーションからの要求をネットワーク部分が提供する QoS のみで満たすことは非常に困難である。

上記のことを、図 1 に示す。図 1 は、分散環境上で対人コ

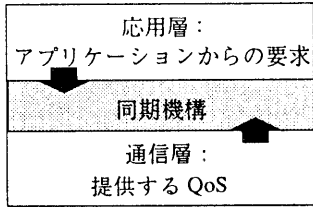


図 1: コミュニケーションにおける階層構造

コミュニケーションを行なう場合を階層構造で表したものである。最下層の通信層は使用するネットワークが提供する QoS(Quality of Service) をがし、最上層の応用層は人間がコミュニケーションを行なう上でのアプリケーションからの要求を表す。図 1 では、通信層が提供する QoS 上で応用層からの要求を満たすために、第 2 層において同期をとるための機構が必要となる。

そこでこの章では、なぜ同期機構を必要とするのか、を 3 章で述べるために、通信層が提供する QoS と応用層からの要求について明確にする。

## 2.1 応用層：アプリケーションの要求

ここでは、人間がコミュニケーションを行なう上で必要なアプリケーションの要求について述べる。

人間がコミュニケーションを行なうためには様々なメディアを使用する。これらのメディアは、状況に応じて単独で、または組み合わせて使用されるが、本稿では即時性を必要とする対人コミュニケーションを行なう場合に使用される、音声を利用した対人コミュニケーションに着目している。

音声を使用した対人コミュニケーションは、会議・講演会・雑談など様々な形態がある。しかし、音声による対人コミュニケーションにはそれぞれ共通する要素があり、その要素を抽出することにより、音声をを用いた対人コミュニケーションの要求を共通化することができる。そこで、アプリケーションの要求を考えるために、音声による対人コミュニケーションのモデル化を行なう。モデル化により、第 2 層の同期機構を考える上で必要となる、音声による対人コミュニケーションが行なう要求を明確にする。

また本稿では、音声による対人コミュニケーションについてのみモデル化を行なっているが、他のメディアを利用した対人コミュニケーションのモデル化については、音声メディアと他のメディア間で、メディア間同期 [3] をとる方法により、音声メディアのモデル化のみで済ませることができる。

### 2.1.1 モデル化

人間が音声によるコミュニケーションを行なう場合、全参加者が平等な立場で行なわれることが望ましい。そこで、

音声による対人コミュニケーションをモデル化するために、以下のことを前提とする。

- 全参加者は、同一発言内容聞く
- 全参加者には、発言権が平等に与えられる。

上記の前提は最低限、音声による対人コミュニケーションを行なうのに必要となる。この前提がない場合、例えば一方が発言を行なったにも関わらず、他方では別の発言を聞いている状態において、コミュニケーションは成立しない。また、発言内容を聞いたにも関わらず、全く発言できない参加者がいたのでは、意志の疎通をはかることができず、やはりコミュニケーションは成立しない。そのため、上記のような前提を必要とする。

今回のモデル化では、対人コミュニケーションの形態の中で主だった項目を抽出し、そのパラメータを変更することで様々な対人コミュニケーションを表す。以下はその項目である。

1. 参加者の構成
  - (a) 参加者数
  - (b) 参加者の把握
  - (c) 参加者の指定
  - (d) 参加(退出)できる時間

### 2. 話し手

### 3. 聞き手

以下、順に説明していく。

#### 1. 参加者の構成.

参加者の構成は以下の項目に細分化され、パラメータを変更することにより様々な形態に対応することができる。

- (a) 参加者数
 

参加者数がどのように変化するかを表す。

  - 固定. 参加者数は固定されていて、その人数でのみ行なわれる
  - 流動. 参加者数は特に決められていない
- (b) 参加者の把握
 

誰が参加しているのか把握できるかどうかを表す。

  - 可能. 全参加者メンバがわかる
  - 不可能. 誰が参加しているのかわからない
- (c) 参加者の指定
 

特定の参加者が参加しなければならないのか、を表す。

  - 指定. 参加者メンバが決められている
  - 不定. 誰が参加してもよい

表 1: モデル化の例

形態	参加者				話し手	聞き手
	数	把握	指定	時間		
1 対 1	固定	可能	指定	指定	複数	同期、非同期
会議 (議長あり)	固定, 流動	可能	指定	指定	1 人	同期
会議 (議長なし)	固定, 流動	可能	指定	指定	複数	同期
雑談	固定, 流動	可能, 不可能	不定	不定	複数	非同期
講演会	固定, 流動	不可能	不定	指定, 不定	1 人	同期
研究会	固定, 流動	不可能	不定	指定, 不定	1 人	同期

(d) 参加 (退出) できる時間

参加者が時間によって変化するかどうかを表す。

- 指定. 途中参加できない
- 不定. 途中参加 (退出) できる

2. 話し手

話し手が同時に何人まで発言できるか, 言い替えれば聞き手が何人まで同時に話し手の発言を聞くことができるか, を表す。

- 1 人. 話すことができるのは同時に 1 人まで。または, 聞き手は同時に 話し手 1 人の発言のみ聞くことができる。
- 複数. 話すことができるのは同時に 複数人。または, 聞き手は同時に複数の話し手の発言を聞くことができる。

3. 聞き手

聞き手は, 他人の発言を聞きながら話すことができるか否かを表す。

- 非同期. 聞き手は, 他人の発言を聞きながら話すことができる。
- 同期. 聞き手は他人の発言を聞きながら話すことはできない。

2.1.2 モデル化の例

2.1.1 節の項目内のパラメータを変化させることにより, 様々な形態の音声による対人コミュニケーションを表すことができる。

表 1は, 音声による対人コミュニケーションの様々な形態をモデル化した例である。例えば, 議長のいる会議では参加者人数が固定で, 参加者の把握ができ, 参加者指定, 話し手は 1 度に 1 人しか話すことができず, 聞き手は同期的に発言を行なう, といったタイプであることがわかる。

2.2 通信層: ネットワークが提供する QoS

ネットワークが提供する QoS には様々なパラメータが考えられる。ここでは後に述べる第 2 層の同期機構からの要求を満たすことができるような QoS パラメータを挙げる。

1. 送受信するデータに対して, スループット  $B$  の保証がなされている。この値は,

$$B[\text{bit/sec}]$$

で表され, ネットワークの帯域が  $B$  で指定した値で確保される。

2. 遅延時間を保証することができる。この時間は実遅延時間 ( $real\_delay\_time$ ) と, 遅延時間の揺らぎ ( $jitter$ ) により,

$$real\_delay\_time \pm jitter[\text{msec}]$$

で表される時間である。

3. ネットワークの信頼性が保証されている。実際の値は, パケット消失率 ( $packet\_loss\_rate$ ) により,

$$packet\_loss\_rate = 10^{-n}[\%]$$

で表される。ここで,  $n$  は正の整数とする。

通信層は, 以上のような QoS を提供し, 本稿ではこの要求が満たされるものとする。また, 大規模分散環境上では, 常に伝送遅延 [4] がある。この伝送遅延は, ネットワーク距離が長くなるほど大きくなり, 第 2 層を考える上で必ず考慮する必要がある。

3 グループ同期機構

2章で通信層が提供する QoS と, 応用層が行なう要求について, 明らかになった。3章では, 第 2 層で何も制御しない場合に生じる問題点を挙げ, 2章で明らかにした, 通信層が提供する QoS と, 応用層が行なう要求を用いることで, 新たにグループ同期を定義する。

### 3.1 問題点

今、応用層の要求を例えば、参加者数固定、参加者の把握ができ、参加者指定、参加時間指定、話し手は1人で、聞き手は同期的に会話を行なうタイプの対人コミュニケーションとしよう。また、通信層は2.2節のQoSを提供するネットワークであるものと仮定する。

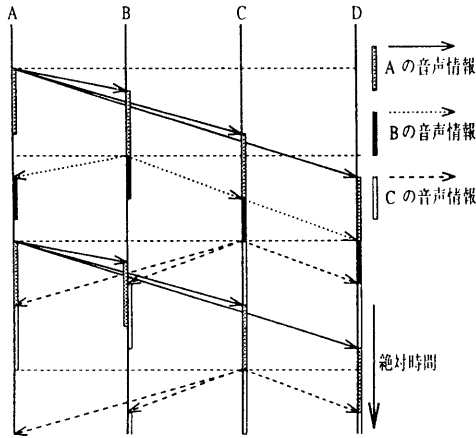


図 2: 伝送遅延の相違による問題

以上のとき、応用層の要求内の、話し手1人、聞き手は同期的に会話を行なう場合の問題点について考える。

遅延の異なるネットワークを通して、各ノード間でコミュニケーションを行なうとき、第2層で何も制御を行なわなければ、平等さに欠ける場合がある。例えば、A,B,C,Dの4者で遅延のあるネットワークを使用して、対人コミュニケーションを行なう場合を考える。図2は、この様子を表した図である。縦軸は絶対時間、各参加者からの矢印は始端が発言を始めた時間、終端が各参加者に発言が到着した時間、矩形は各参加者の発言を示す。

この例では、第2層で何も制御を行っていない。そのため、他の参加者の音声にDに絶えず到着し、Dはいくら待っても発言することができない。この場合、2.1.1節で示した、音声を用いた対人コミュニケーションの前提、「全参加者には、発言権が平等に与えられる」ことに矛盾する。したがって第2層は、ネットワーク距離が遠い参加者に対しても、「平等に発言」できるための制御が必要である。

また、各ノード間で遅延時間の異なるネットワークで会話を行なったとき、会話に矛盾が生じる場合がある。例えば、A,B,Cの3者で遅延時間のあるネットワークを使用して対人コミュニケーションを行なったとする。図3は、この様子を表した図である。

図3では、Aが絶対時間で先に発言を始めている。その後、BはAの発言を受け取った後に発言を開始したが、C

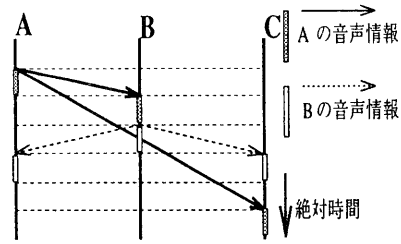


図 3: 伝送遅延の相違による問題その2

では、Bの発言が先に到着してしまい、そのままCで再生を行なうと、Bの後にAが再生されてしまう。

この場合、Aの発言に関連してBが発言を行っていたなら、会話の前後が入れ替わってしまうことになり、Cは発言に矛盾を感じるであろう。

上記の例は、2.1.1節で示した、音声を用いた対人コミュニケーションに対する前提、「全参加者は、同一発言内容を聞く」ことに矛盾する。したがって、このような発言順に対して矛盾が生じさせないための制御を行なう必要がある。

上記の2つの例は、各ノード間でネットワークの伝送遅延時間が大きく異なるために生じるものである。そのため、第2層の同期機構は各ノード間での伝送遅延時間の違いを考慮する必要がある。

### 3.2 グループ同期

大規模分散環境上では、各ノード間の伝送遅延の相違によって、2.1.1節で示した前提に矛盾が生じることを3.1節で述べた。そこで、これらの問題点を解決するために必要な第2層の同期機構、グループ同期を定義する。

グループ同期は、

1. 各ノードでの情報送信・受信が平等に行なわれる
2. 1.の事項を保証するために、各ノードで受信する情報の順序は同一ものであり、また情報は絶対時間で送信された順序で再生を行なう

と定義する。

上記の定義に基づいたグループ同期を実現することで、2.1.1節で示した前提を満たしつつ、大規模分散環境上で音声を使用した対人コミュニケーションを行なうことが可能となる。

## 4 アルゴリズム

3.1節で述べた音声による対人コミュニケーションのタイプをもとに、グループ同期を実現するためのアルゴリズムを提案する。以下のアルゴリズムは、2.2節で述べたQoSを提供するネットワークを使用する。便宜上、スループットB

は送受信されるデータに対し、十分満たすことができる帯域が確保されるものとし、遅延時間内の  $jitter = 0$ 、パケット消失率内の  $n$  は十分大きいものとする。また、第3層でのモデルのタイプは3.1節で指定したものを使用する。

ここで、3章で示した、第2層の同期機構、グループ同期が、3.1節で示したモデルのタイプに対する役割について、再度以下に示す。

- ネットワーク距離が遠い・近いに関わらず、参加者に対して平等に発言できる場を与える。
- 各ノードが発言の再生を行なう場合、発言の順序の前後が入れ替わることがあってはならない。

また本アルゴリズムでは次のことを仮定する。

- 各ノードは対人コミュニケーションを開始時間を基準として、同一の時間軸（絶対時間）を持つ
- 参加者はあらかじめわかっている

本アルゴリズムは、同一時間軸上で誰も発言していないときに、最初に発言した者に対して優先的に発言権を与える。また、発言権を得た者以外が発言を行なっても受信ノードで破棄する。また再生される情報は、各参加者で適当な遅延時間を付加することで、全参加者で同一時間に再生される。上記のように制御することで、グループ同期を実現する。

以下にこのアルゴリズムを詳しく説明する。このアルゴリズムは、2つの部分からなり、最初の部分は前処理によって、各ノード間の中で最大遅延時間を探し、全ノードで最大遅延時間を同一値にするための前処理である。この前処理は、次の部分で絶対時間で同一時間に同じ発言を再生するために必要である。次の部分は、実際に参加者がコミュニケーションを行なう部分（セッション）内での処理を表す。

#### 4.1 前処理

この処理は、各ノード間の中で最も大きい伝送遅延時間を最大伝送遅延時間とし、その値を各ノードで同一値にするための処理である。最大遅延時間は、4.2節で全ノード同一時間で発言を再生するために必要とする。

1. 各ノード（参加者）は、通信層から提供される各ノード間の遅延時間（delay-time）の中から、最大の値を最大遅延時間として記憶する。その最大遅延時間を付加したパケットを全ノードに送信する。
2. 1.を受信した各ノードは、自身の持っている最大遅延時間とパケット内の最大遅延時間を比較して、大きい方を自身の最大遅延時間として記憶する。

以上のような、操作を行なうことで、全参加者はネットワーク中の最大遅延時間を獲得することができる。

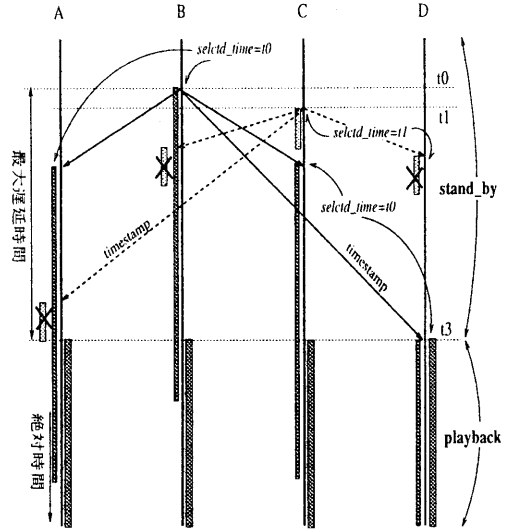


図4: セッション

#### 4.2 セッション

この処理は、実際に音声を使用した対人コミュニケーションを行なうための処理である。

各ノードは、次に再生すべきデータが送信され始めた時間を格納するための *selctd.time* を持つ。

この処理では2つの状態、発言可能状態、*stand\_by* と、発言再生状態 *playback* である。それぞれの状態は、さらに発言の送信側、受信側に分けられる。

##### stand\_by

この状態から、セッションは開始する。どの参加者も発言していない状態では、*selctd.time* の値はクリアされている。

##### 送信側

発言したいノード（参加者）は、自身が発言を開始した時間と *selctd.time* とを比較し、自身の発言の方が先、もしくは *selctd.time* に何も入っていないならば、その発言開始時間を *selctd.time* に入力する。その後、発言データパケットに発言開始時間を *timestamp* として付加し、参加者に対して送信する。もし、*selctd.time* の方が先の時間であった場合、すでに自分より先に発言したノードがいるとわかるので、パケット送信を行なわない。

##### 受信側

パケットを受け取ったノードは、パケット内の *timestamp* と自身の *selctd.time* を比較し、

*timestamp*の方が遅い時間であれば、そのバケットを破棄する。*selctd.time*の方が遅い時間であれば、そのバケット内の発言の方がより早い時間に開始されたとみなす。そこで、*selctd.time*をバケット内の*timestamp*の時間に書き換え、バケットの発言情報を記憶する。その後、*selctd.time*から前処理で求めた最大遅延時間だけ待って、**playback**状態になる。

#### playback

実際に発言情報を再生する部分である。

#### 送信側

発言が再生される。発言中のノードで自身の発言が再生されたなら、発言権を得たことになり、そのまま発言を続けることができる。その後、発言が終了すると、最大遅延時間だけ遅れて自身の発言が再生終了する。その後、*selctd.time*を消去して、**playback**は終了し、**stand\_by**状態に戻る。また、他ノードの発言が再生された場合、発言を行なうことはできず、受信側の制御に移行する。

#### 受信側

再生されている発言ノードのバケットのみ受信し、他ノードからのバケットは全て破棄する。再生終了後、*selctd.time*を消去して、**playback**は終了し、**stand\_by**状態に戻る。

以上の処理により、全参加者の発言の機会は平等なものとなり、また会話の順序に矛盾を生じることがない。すなわち、グループ同期を行なうことができる。

図4は、セッションの流れを図で表したものである。各ノードの直線は、絶対時間を表し、矩形は、各参加者の発言で、直線の左側は各ノードでの発言の受信、右側は、発言の再生を表す。

図4を使用して、本アルゴリズムについて説明する。例えば、**stand\_by**状態のとき、他の参加者が発言を行っていない時刻 $t_0$ で、Bが発言を試みたとする。Bは*selctd.time*の値が何も入っていないことから、時刻 $t_0$ 以前に誰も発言を行っていないことを知り、発言開始時間 $t_0$ を*timestamp*に入れ、各ノードに送信し、自身の*selctd.time* =  $t_0$ とする。

Bの発言は伝送遅延を伴って、各参加者に到着する。このとき、Bの発言がCに到着する以前、時刻 $t_1$ にCが発言を試みたとする。Cは、自身の*selctd.time*の値に何も入っていないことから、自身よりも先にBが発言を行なったことを知らない。そのためCは、*timestamp* =  $t_1$ として、各ノードに送信し、自身の*selctd.time* =  $t_1$ とする。

Cの発言を受信した各ノードは、最大遅延時間だけ待ち、その待ち時間内に到着したBのデータバケット内*timestamp* =  $t_0$ からCはBよりも後で発言を行なったことがわかり、Cの発言は破棄される。その後、Bの発言は発言開始時刻から最大遅延時間だけ待ち、時刻 $t_3$ のときに**playback**状態になり、絶対時間軸上で同一時間に再生される。再生後、*selctd.time*の値は消去され、**stand\_by**状態に戻って再び発言を行なうことができる。

## 5 関連研究

グループ同期を実現するためには、本稿で述べたアルゴリズムの他に、放送型通信アルゴリズム [7] を利用する方法、分散同期アルゴリズム [8] を利用して各ノードが発言権を得る方法などが考えられる。

しかし、[7]の方法では、各ノードからの情報受信の度に、受信確認を行う必要がある。この方法では、音声を利用した対人コミュニケーションのようなリアルタイム的な要素をもつ場合に対しては、通信のオーバーヘッドが大きくなり、会話の持つリアルタイム性を損なってしまふ、という問題点がある。それに対し、本稿で提案したアルゴリズムでは、最初に前処理を行なうことで、通信に対するオーバーヘッドを軽減している。

また、[8]では、発言に対する順序制御が完全に行なうことができないため、先に発言をした参加者より、後から発言した参加者が優先されてしまふ、といった問題点がある。それに対し、本アルゴリズムでは一定の時間だけ音声の再生を遅らせることによって、この問題点を解決している。

## 6 おわりに

本稿では、人間が音声を使用してコミュニケーションを行なう場合の、モデル化を行ない、その1モデルをもとにグループ同期について定義した。また、そのグループ同期を実現するための方法について提案した。今後は、このアルゴリズムを実装し評価を行なう予定である。さらに、今回は1モデルについてのみの問題点を考慮したが、今後は他のモデルについても検討を行なう予定である。

## 参考文献

- [1] T. Turletti, "The INRIA Videoconferencing System (IVS)", *ConneXions - The Interoperability Report*, Vol. 8, No. 10, pp. 20-24, Oct. 1994.
- [2] H. Schulzrinne, "Guide to NeVoT 3.28", GMD Fokus, Berlin, June, 1995, available at [gaia.cs.umass.edu/pub/hgschulz/nevot/nevot.ps.gz](http://gaia.cs.umass.edu/pub/hgschulz/nevot/nevot.ps.gz)
- [3] 岡村, 稲垣, 田中, 松尾, 吉川, 荒木, "マルチメディア基盤システム PDE-II におけるメディア間同期", 日本ソフトウェア学会第11回大会論文集, pp.201-204, 1994.
- [4] 天海, 村上, 釘本, 岡, 伊藤, 後藤, 伊藤, "長距離超高速インターネットの特性解析", マルチメディア通信と分散処理ワークショップ, pp.149-156, Oct. 1994.
- [5] C. Topolcic, "Experimental Internet Stream Protocol, Version 2 (ST-II)", RFC1190, Oct. 1990.
- [6] L.Zhang, S.Deering, D.Estrin, S.Shenker, D.Zappala, "RSVP: A New Resource Reservation Protocol", *IEEE Network*, Vol.7, No.5, Sep. 1993.
- [7] 滝沢, 中村, "放送型通信アルゴリズム", 情報処理学会誌, Vol.34 No.11 pp.1341-1348, Nov. 1993.
- [8] T.Johnson, "A Performance Comparison of Fast Distributed Synchronization Algorithms", Technical report, U.Florida Dept. of CIS, 1994, available at [ftp.cis.ufl.edu/cis/tech-reports/tr94/tr94-032.ps.Z](http://ftp.cis.ufl.edu/cis/tech-reports/tr94/tr94-032.ps.Z)