

## 対戦ゲームにおけるメンバ間公平性保証方式の性能評価

石川 貴士\* 石原 進\*† 井手口 哲夫\*† 水野 忠則‡

筆者らは、比較的即応性が要求されるネットワーク対戦ゲームやネットワークオークションにおける端末間の遅延差により起こるユーザ間の不公平さを解決するメンバ間公平性保証方式 (*ICEGEM*: Impartial Communication Environment for GamE Members) を提案している。本方式では、反応時間に基づく公平性保証、動的なサーバ・クライアントでの制限時間、制限時間の短縮により端末間での時刻の同期、端末間同期などの複雑で困難な制御を必要としないで、完全なユーザ内の公平性を実現する。本稿では、本方式を用い、ネットワーク対戦型早押しキーボードタイピングゲームのプロトタイプを実装し、遅延差が生じる端末間で実験を行った。その結果として、反応時間に基づく公平性保証では、端末間の遅延差に関係なく、公平にアプリケーションが利用できることが確認できた。

### Implementation and Evaluation of *ICEGEM* Prototype System, with Guaranteed Fairness of Users' Response

Takashi Ishikawa\*, Susumu Ishihara\*†, Tetsuo Ideguchi\*† and Tadanori Mizuno†

In this paper we report about the implementation and the evaluation of the prototype system of *ICEGEM* (Impartial Communication Environment for GamE Members), which can control the order by users' response time. In real-time network applications, such as network games, network auction, etc., the users often have event inversion on heterogeneous network environment which have low-speed networks and high-speed networks. Because of this, the fairness between users can not be guaranteed. We have proposed *ICEGEM* which guarantees the fairness between users' response time. In this paper we show an implementation of the prototype system of network key-typing game on *ICEGEM*, and evaluated whether users are able to use application fairly bear little relation to the difference of communication latencies.

#### 1 はじめに

ギガビットイーサネットなどの高速ネットワークが普及しつつある。一方で、モバイル端末を持ち歩き、出先から携帯電話・PHSの比較的低速な無線回線を利用したモバイルコンピューティングが一般化してきている。また、高速ネットワークの普及により、今後ネットワークを介した即応性が要求されるアプリケーションの利用がさらに広まることが予想される。KDDとカプコンでは、70[msec]以内で届く高速性の対戦型ゲーム用ネットワークサービス「マッチングサービス」[1]の提供開始を2000年3月から始めている。また、PlayStation2ではCATVと接続した高速なネットワークゲーム環境が計画されている。このような高速ネットワーク

と低速ネットワークの混在する環境においては、即応性の要求されるネットワークアプリケーション、例えばネットワーク対戦ゲーム、オークション、電子会議システムなどを利用する際に、端末間の遅延差によりイベントパケットの順序がサーバ側で逆転する現象が起きる。それによりユーザの反応時間に対する不公平さ図1を生じたり、また因果関係が保たれなくなることがある。この結果、ネットワーク対戦ゲームで、回線のスピードが遅いため敵の動きを確認できなかったり、自分の動きが遅くなったりにしてゲームに負けてしまうなどの問題が生じる。

筆者らは、上記のような不公平性の解消のための機構として、対戦ゲームにおけるメンバ間公平性保証方式 (*ICEGEM*: Impartial Communication Environment for GamE Members) [2][3]を提案している。本方式は、反応時間に基づく公平性保証、動的なサーバ・クライアントでの制限時間、制限時間の短縮によりメンバ間で公平なサービスを受けることができる。

本稿では、本方式の詳細とプロトタイプによる評価について述べる。以下、まず端末間の遅延差による不公平性について述べ、関連研究とその問題を上

\*静岡大学大学院理工学研究科  
Graduate School of Science and Engineering,  
Shizuoka University  
†愛知県立大学情報科学部  
Faculty of Information Science and Technology,  
Aichi Prefectural University  
‡静岡大学情報学部  
Faculty of Information, Shizuoka University

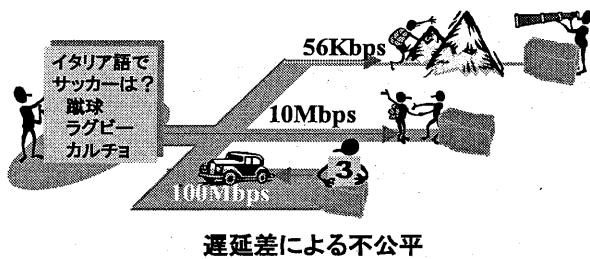


図 1 不公平性

げ、本システムの反応時間に基づく公平性保証、動的なサーバ・クライアントでの制限時間、制限時間の短縮の詳細について述べる。次に、反応時間に基づく公平性保証を用いた、ネットワーク対戦型早押しキーボードタイピングゲームのプロトタイプについて述べ、遅延差が生じる端末間での評価実験について報告する。

## 2 端末間の遅延差による不公平性

### 2.1 逆転現象

高速なネットワークと低速なネットワークの混在する環境において、即応性の要求されるネットワークアプリケーションを利用する際に、端末間の遅延差によりイベントパケットの順序がサーバ側で逆転する現象が起きる。この逆転現象の例を図 2 に示す。S はサーバ、 $C_1$ ,  $C_2$  はクライアント、 $R_1$ ,  $R_2$  はそれぞれのクライアントがパケットを送信した時刻、 $B_1$ ,  $B_2$  はサーバがそれぞれのクライアントからのパケットを受信した時刻である。低速なネットワークを使用している  $C_1$  は、高速なネットワークを使用している  $C_2$  よりも早い時刻にパケットを送信したにもかかわらず、サーバ側では  $S-C_1$ ,  $S-C_2$  間の遅延差のために到着順序が逆転していることが分かる。このように、クライアント側でのイベントパケット送信時刻とサーバ側でのイベントパケットの到着時刻が逆転してしまうことにより、対戦ゲームなどにおいてクライアント間の不公平が生じる。

### 2.2 関連研究

このような逆転現象に対して因果関係の保存と、ユーザ反応時間との公平性を保証するための手法として [4][5][6][7][8] の研究がある。これらの研究は、サーバ側で一定時間、クライアントからのイベントパケットを待ち合わせた後、順序の制御を行っている。この一定時間の待ち合わせと、順序の制御を繰り返すことによって端末間の遅延差による不公平性を解消している。ここで、サーバ側での一定の制限時間を設けている理由は、応答のないクライアントをいつまでも待たせては、即応性が損なわれるためである。そのため、たとえ制限時間を超えてからパ

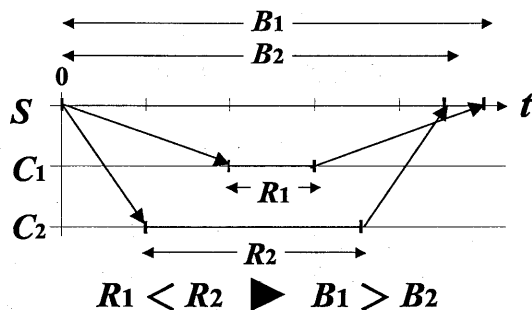


図 2 逆転現象

ケットが届いたとしてもサーバ側で破棄する手法が用いられている。

また、ネットワーク通信ゲームに特化したソフトウェア開発企業である株式会社ドワンゴは、高速通信ゲームシステム「DWANGO」[9]を商品化している。このシステムは、アクションゲームのようにデータ伝送に同時性を要求するジャンルのゲームに適した高速通信ゲームシステムで、毎秒 60 フレーム同期通信を可能にする。現在、Dreamcast 用ネットワークゲームのサーバシステムとして採用されている。

### 2.3 問題点

[4][6][7] の研究では、NTP (Network Time Protocol) により全サーバ・クライアント間で時刻の同期をはかり、全クライアントの遅延時間を測定し、クライアントでの出力時刻を同期させる手法がとられている。出力時刻を同期させる理由は、特定の端末が早く情報を得ることによるクライアント間の不公平さをなくすなどの理由がある。また、全サーバ・クライアント間で時刻の同期を行い、それぞれのクライアントとサーバ間のメッセージ伝達遅延に基づいて、すべてのクライアント間での表示時刻を合わせることによって、クライアントのイベントパケット送信時刻から順序の制御を行っている。しかしながら、遅延時間は常に変動しているので完全にクライアント側で到着時刻をすべてのクライアントで合わせることは難しい。

また、実験環境が比較的高速なネットワークであること、クライアント数が少数であることから、低速・高速のネットワークが混在したインターネット上での利用を考えると課題が残る。また、制限時間までに全クライアントからの反応パケットがサーバに到着しても制限時間まで待ち合わせてから順序の制御を行うため、サーバ側での無駄な待ち時間が発生する。そこで、筆者らは関連研究とは別のアプローチとして、全サーバ・クライアントでの時刻を同期させることを必要としない、反応時間に基づく公平性保証を提案している [2][3]。本方式は、即応性を向上させ、クライアント側でも制限時間を設けることにより、無駄なトラフィックを軽減する。次節では、本方式の詳細について述べる。

### 3 ICEGEM の提案

#### 3.1 概要

関連研究で想定している環境は、全クライアント・サーバが比較的高速なネットワークに接続されており、通信が LAN などの狭い範囲で行われていたこと、クライアント数が少ないものであった。しかしながら、これは非常に限られた環境であり、対戦ゲームなどは LAN に限らず、インターネット、無線ネットワーク、モデム接続など、遅延差の大きい環境で利用されることが予想される。そこで、比較的即応性の要求されるネットワークゲームを対象としたメンバ間公平性保証方式 (ICEGEM) を提案する。想定環境は、以下の通りである。

アプリケーション：ゲーム、オークション

ネットワーク：LAN、インターネット

通信速度：ギガビット～32Kbps

クライアント数：0～20

TCP：接続、離脱処理命令用

UDP：ゲーム処理命令用

接続・離脱といったパケットロスが許されない命令は TCP により、ゲーム内での順序の制御を行うパケットロスが許される命令は UDP により転送する。UDP 上で転送するメッセージでは、即応性を重視しつつ、ユーザ間の公平性を保証するものとする。ICEGEM の主な 4 つの特長を以下に示す [2]。

#### 反応時間に基づく公平性保証

クライアント側の画面に表示されてからユーザが次のイベント動作を行うまでの反応時間により公平性保証する

#### クライアント側の制限時間

クライアント側でユーザの反応時間に制限時間を設け、サーバ側にパケットを送信しないことでトラヒックの軽減、無駄な負荷を下げる

#### サーバ側の動的な制限時間

サーバ側での制限時間を全クライアントの遅延時間の変動から動的に変更する

#### サーバ側の制限時間の短縮

制限時間に関係なく、順序の制御が行える状態になれば、その時点で制限時間を短縮する

### 3.2 詳細

#### 3.2.1 記号の説明

以下の説明では、サーバがイベントをクライアントにイベントパケットを送信し、クライアントがそ

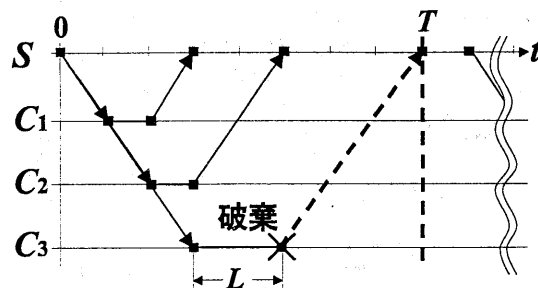


図 3 クライアント側の制限時間

れに対する反応を返す 1 サイクルを 1 ターンと呼ぶことにする。また、ICEGEM で定義する記号を以下に示す。

$S$  : サーバ

$C_i$  : クライアント ( $i = 1, \dots, n$ )

$T_j$  :  $j$  ターン目の  $S$  側の制限時間 ( $j = 1, \dots, \infty$ )

$L_j$  :  $j$  ターン目の  $C$  側の制限時間

$R_{i,j}$  : ユーザの反応時間

$B_{i,j}$  : 返答パケットの受信時間

$D_{i,j}$  : 反応時間を含めない往復遅延時間

$P_{i,j}$  : 反応時間を含めない往復遅延時間

#### 3.2.2 反応時間に基づく公平性保証

1 つ目の特長として反応時間に基づく公平性保証について説明する。まず、サーバが全クライアントにイベントパケットの送信を行う。各クライアントはパケットを受信し、反応命令と反応時間  $R_{i,j}$  の情報を付加し、サーバに送信する。サーバは各クライアントからの  $T_j$  までに受信したパケットを、クライアント側の反応時間  $R_{i,j}$  に基づいて順序の制御を行う。こうすることにより、全クライアント・サーバ間で時刻の同期をとる必要はなく、またすべてのクライアントで同時に表示させなくても公平性の保証ができる [3]。

#### 3.2.3 クライアント側の制限時間

無駄なパケットの送信により、サーバへの負荷を軽減するために、サーバでの制限時間に加えて、クライアント側でも制限時間を設け、この制限時間を超えたパケットはサーバに送信することなくその場で破棄をする (図 3)。サーバ側の制限時間  $T_j$  は、最大往復遅延時間とクライアント側の制限時間  $L_j$  の和である。このため、 $L_j$  を超えたパケットをサーバに送信してもサーバの制限時間を超える可能性が高いと考えられるからである。ここで、クライアント  $C_1$  のように高速なネットワークで接続されているユーザはクライアント側の制限時間  $L_j$  を超えたパケットを破棄せず、サーバに送信してもサーバ側の制限時間  $T_j$  内に到着可能であると予想でき

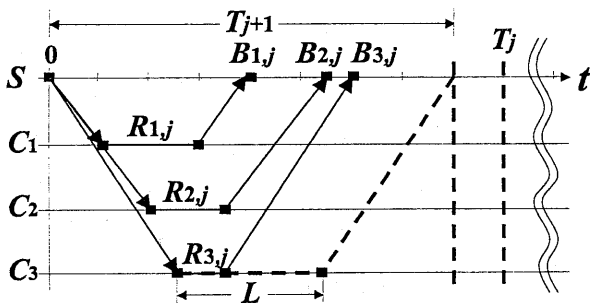


図4 制限時間の変更

るが、公平性を保証するため、平等に packets を破棄することにする。これにより、無駄なトラフィックの軽減をし、またサーバへの無駄な負荷を軽減する。

1 ターン内に packets が到着しない場合として、クライアント側の制限時間による packets の破棄 (A) と途中の通信路での packets ロス (B)、サーバ側の制限時間による packets の破棄 (C) という計 3 つケースが考えられる。プロトタイプシステムのようなゲームなど、packet 1 つの重要性より即応性が重要となる場合には、(A)、(B) のケースを区別する必要がないこともある。しかしながら、オークションのようなアプリケーションでは、packet ロスがアプリケーションのサービスを損なう恐れがある。本稿では、(A)、(B) を区別しないケースを想定する。また (C) のケースでは、受信された packet はアプリケーションのサービスに対しては影響を与えないが、次のターン以降のクライアントの反応時間の予測には利用できる。

### 3.2.4 サーバ側の動的な制限時間

まず最初にサーバに対してクライアントが接続を行い、全クライアントが接続した段階で各クライアントの初期往復遅延時間  $D_{i,0}$  の設定を行うため、サーバは全クライアントに対して packets を UDP により送信する。それに対し全クライアントは  $R_{i,0} = \text{const}$  でサーバに反応を返す。サーバは受信した packet から各クライアントの初期往復遅延時間  $D_{i,0}$  の設定を行う。反応時間を含めない初期往復遅延時間は、 $D_{i,0} = B_{i,0} - \text{const}$  となる。

1 ターン目のサーバ側の制限時間の初期値  $T_1$  は、予測最大往復遅延時間とクライアント側の制限時間  $L_1$  によって決定する。1 ターン目の予測往復遅延時間を  $P_1$  とし、全クライアントでの最大の往復遅延時間を  $\max(D_{i,1})$  とすると  $T_1$  は、

$$P_1 = \max(D_{i,0})$$

$$T_1 = P_1 + L_0$$

となる。なお、予測往復遅延時間  $P_j$  については後で詳しく説明する。UDP による packet の欠落を考慮し、初期設定においてのみ往復遅延時間は全クライアントの情報が得られるまで繰り返し行う。

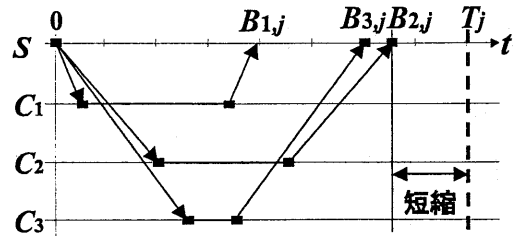


図5 全packet到着時の短縮

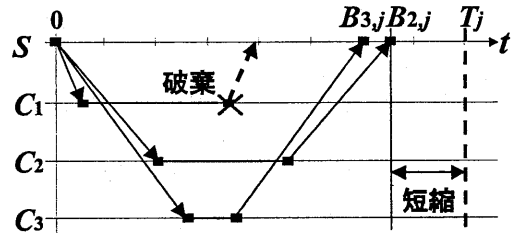


図6 全packet未到着時の短縮

遅延時間の変動に伴い、1 ターンごとにサーバ側の制限時間の変更を行う。j ターン目のクライアントの往復遅延時間は受信した全クライアントからの packet により、 $D_{i,j} = B_{i,j} - R_{i,j}$  となる。(j+1) ターン目のサーバ側の制限時間  $T_{j+1}$  は、(j+1) ターン目の予測往復遅延時間  $P_{j+1}$  とクライアント側の制限時間  $L_{j+1}$  の和により、

$$P_{j+1} = \alpha P_j + (1 - \alpha) \{ \max(D_{i,1}) \}$$

$$T_{j+1} = \beta P_{j+1} + L_{j+1}$$

と求められる。予測往復遅延時間は、前ターンの予測往復遅延時間と現ターンの最大の往復遅延時間の和とする。また、 $\alpha (0 \leq \alpha \leq 1)$  は前ターンの予測遅延時間に与える重みづけで、小さいほど変動を大きくし、 $\beta$  は遅延変動を吸収するマージンである。 $\alpha$  の値の調整で、ネットワークの状況に伴い常に変化する遅延時間に対応させ、次のターンでのサーバ側の制限時間  $T_{j+1}$  を動的に変更する (図 4)。なお、現段階では  $T_j$  を超えて到着した packet は、 $T_{j+1}$  を決定する際に  $\max(D_{i,j})$  の計算に用いていないが、継続的にサーバ側の制限時間を超える packet が存在する場合には、これらの packet の情報の利用が必要であろう。

### 3.2.5 サーバ側の制限時間の短縮

図 5 のように、サーバ側の制限時間  $T_j$  までに全クライアントからの返答 packet を受信した時点で順序の制御を開始できるので、その時点から 1 ターンとし、制限時間まで時間を短縮する。図 6 に示すように、サーバは必ずしも制限時間  $T_j$  までに全クライアントからの返答 packet を受信できるわけではない。これは、クライアント側の制限時間によって、packet が送信されずに破棄されていたり、ネットワークの途中で packet が欠落している場合があるからである。そこで本方式では、クライアン

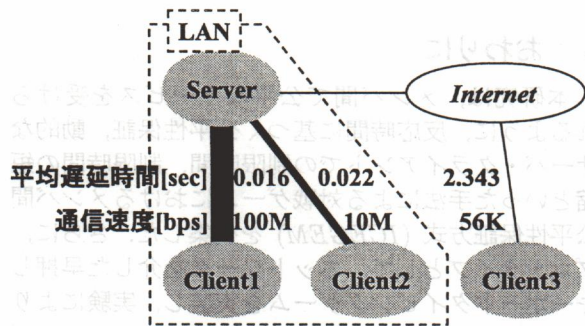


図 7 実験環境

トからの応答を得るたび、前ターンの最大の往復遅延時間を持つクライアントの往復遅延時間にクライアント側の制限時間を加算した時間まで待ち合わせる。一般に最大の往復遅延時間が、

$$\max_i (D_{i,j-1}) - L_j < D_{k,j-1}$$

を満たすクライアント  $k$  が存在し、そのうちサーバがまだ返答パケットを受信していないクライアントの集合を  $Y$  とすると、

$$T_j = \beta \max_{k \in Y} (D_{k,j-1}) + L_j$$

まで待ち合わせるか、もしくは、クライアント  $k \in Y$  からのパケットをすべて受信した時点で、サーバは順序の制御を開始する。この短縮により、サーバでの無駄な時間を省略することができ、即応性を向上させることができる。

## 4 プロトタイプの実装と評価

### 4.1 プロトタイプシステム

提案方式による公平性保証を評価するため、プロトタイプシステムとしてネットワークを介した早押しキーボードタイピングソフトを実装した [10]。現在実装されている機能としては、反応時間に基づく公平性保証とクライアント側の制限時間である。このシステムは、3人それぞれのプレイヤーが高速ネットワーク、中速ネットワーク、低速ネットワークからサーバに接続し、キーボードの早押しゲームを行うものである。サーバからタイピングするキー（アルファベット）がクライアントに同時に送信され、クライアントではキーを受信し、表示されたアルファベットを打ち込む。クライアントの画面にアルファベットが表示された時刻から、正しいキーが押されるまでの時間を反応時間とし、クライアントはサーバ側にその情報を送る。サーバは3台のクライアントから送られてきた反応時間により、順位付けを行う。以上のゲーム上の命令パケットをUDPにより伝送する。一方、遅延時間の初期設定やサーバで集計された順位のフィードバック、その他接続確認などの命令パケットをTCPにより伝送する。

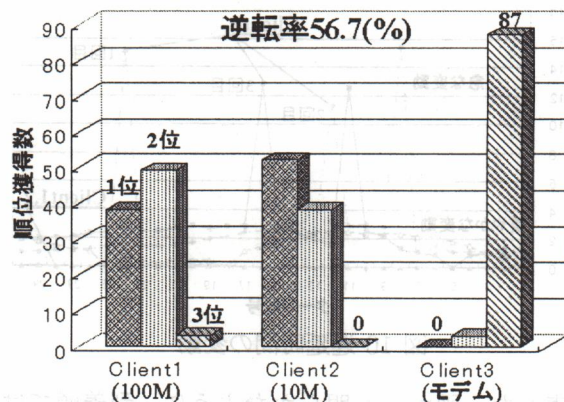


図 8 到着順

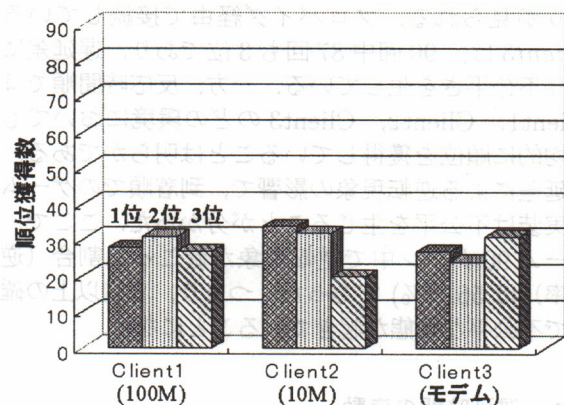


図 9 反応時間順

### 4.2 実験

図 7 に実験環境を示す。高速なネットワークとして、サーバと同じ LAN にある 100Mbps で接続されたマシンを、中速なネットワークとして、サーバと同じ LAN にある 10Mbps で接続されたマシンを、低速なネットワークとして、プロバイダ経由のモデム (56kbps) により接続されたマシンを用意した。また、それぞれの平均往復遅延時間は、0.016, 0.022, 2.343[sec] で、送信メッセージのデータ部のサイズは 1Byte である。実験では、3人のプレイヤーが環境 Client1, Client2, Client3 をそれぞれ交代して、1 ゲーム 30 回、3 ゲーム合計 90 回のキータイピングを行った。環境を交代させた理由は、個人的なキータイピングの速さの違いが実験に影響しないようにするためである。この環境下において、パケットの到着順と反応時間順の比較、遅延時間の変動、クライアント側の制限時間の効果について評価した。

### 4.3 到着順と反応時間順

反応時間による順序の制御によって公平性が保証されているかを調べるため、到着順と反応時間順を比較する。図 8, 図 9 は、それぞれのクライアントが 3 回のゲームにおいて、サーバでの到着順と反応時間順による順位獲得数の違いを表したものである。横軸にクライアントを、縦軸にそのクライアントが 3 回のゲームで獲得した順位の合計数

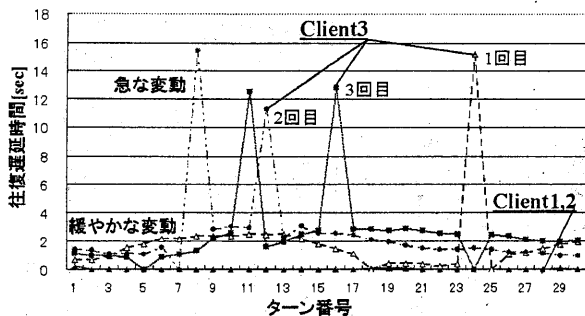


図 10 遅延時間の変動

を表した。図からも明らかなように、到着順ではネットワークの環境の影響で、順位獲得数に大きな偏りが見られる。プロバイダ経由で接続している Client3 は、90 回中 87 回も 3 位であり、遅延差により不公平さを生じている。一方、反応時間順では Client1, Client2, Client3 のどの環境においても平均的に順位を獲得していることは明らかである。遅延差による逆転現象の影響で、到着順でのゲームの実装は不公平を生じることが分かった。ここで、ゲーム 90 ターン中で逆転現象が起こった割合（逆転率）は 56.7(%) であった。つまり、5 割以上の確率で不公平な状態が起きていることになる。

#### 4.4 遅延時間の変動

図 10 は、3 回のゲームでの Client3 の遅延時間の変動を表したものである。この図からも、モデムおよびインターネット経由で接続されている Client3 の遅延時間は、急に大きく変動することが分かる。

#### 4.5 クライアント側の制限時間の効果

次にクライアント側の制限時間で破棄したパケットが、サーバ側に送信されているとするとどれだけの割合でサーバ側の制限時間を超えるかについて調べた。 $L_j = 0.800[\text{sec}]$ ,  $\alpha = 0$ ,  $\beta = 1$  としたときにクライアント側で制限時間を超えるものは、90 回中 17 回存在した。その中で、サーバ側に送信した場合もサーバ側での制限時間により破棄されるものは、13 回であった。よって、76.5(%) となり、高い確率でサーバ側でも破棄されることが分かった。しかしながら、サーバ側の制限時間は、クライアント側の制限時間と往復遅延時間によって計算されたものであるため、破棄率は 100(%) により近い値であることが望まれる。この場合は遅延時間が短くなり、次のターンの制限時間内にサーバが受信できた回数が多かったことになる。逆に  $\beta$  の値を大きくすれば、サーバ側で破棄される確率が減ることになる。今回は最も単純なケースについて評価を行ったが、今後、これらのパラメータの適切値に関して検討する必要がある。

## 5 おわりに

本研究は、メンバ間で公平なサービスを受けられるように、反応時間に基づく公平性保証、動的なサーバ・クライアントでの制限時間、制限時間の短縮といった手法による対戦ゲームにおけるメンバ間公平性保証方式 (ICEGEM) を提案した。さらに、プロトタイプとして、ネットワークを介した早押しキーボードタイピングゲームを実装し、実験により反応時間に基づく公平性保証、クライアント側の制限時間について評価を行った。その結果、反応時間に基づく公平性保証を行ったものは、ゲームの上での順位を公平に獲得することができた。

今後の課題として、適用されるアプリケーションをゲームとその他のオークションのようなものに分類し、それぞれについてのシステムの設計の検討が必要である。また、オークションなどのように反応時間が非常に重要なパラメータとなるアプリケーションにおいてはクライアント側での反応時間の偽証行為について対策が必要である。このほか、ユーザの反応時間の予測方式について適切なパラメータの検討のほか、サーバ側の動的な制限時間、制限時間の短縮といった手法の実装を行い、公平性、即応性について評価を行う予定である。

## 参考文献

- [1] マッチングサービスの提供開始について：  
<http://www.kdd.co.jp/press00/00-002r.html>
- [2] 石川貴士, 石原進, 井手口哲夫, 水野忠則: リアルタイム性の強いネットワークアプリケーションの公平性を考慮した通信方式の提案, 1999 年電気関係学会東海支部連合大会, 578, pp. 289, (1999.9).
- [3] 石川貴士, 石原進, 井手口哲夫, 水野忠則: リアルタイム性の強いネットワークアプリケーションの公平性を保証した通信方式の提案, 第 59 回情処全大, 3V-4, pp. 551-552, (1999.9).
- [4] 徳永博之, 関野公彦, 久保田創一, 佐藤栄: リアルタイムグループウェアにおけるイベント順序制御の一考察, 情処学 DPS 研報, 82-30, pp. 171-176, (1997.4).
- [5] 伊関宏心, 富永英義: マルチメディアシステムにおけるイベント順序制御手法の提案, 電子情報通信学会総合大会, B-7-174, pp. 295, (1998).
- [6] 桑子純一, 瀬崎薫: 分散環境におけるメディア同期, 信学技報, SSE98-100, IN98-81, pp. 67-72, (1998.9).
- [7] 橘芳郎, 石橋豊, 田坂修二:  $\Delta$  因果公平性保証を用いたメディア同期方式の性能評価実験, 信学技報, SSE98-203, IN98-175, pp. 25-30, (1999.3).
- [8] Y. Ishibashi and S. Tasaka: A group synchronization mechanism for live media in multicast communications, in *Conf. Rec. IEEE GLOBECOM'97*, pp. 746-752, (1997.11).
- [9] DWANGO: <http://www.dwango.co.jp/>
- [10] 石川貴士, 石原進, 井手口哲夫, 水野忠則: 対戦ゲームにおけるメンバ間公平性保証方式: 実装と評価, 第 6 回モバイルマルチメディア通信ワークショップ, MoMuC-J6, pp. 1-8, (2000.3).