

## P2P ネットワーク自動再構築による 連続ストリーミング配信法の提案

ピラヴォン ミナサイ<sup>†</sup> 敷田 幹文<sup>‡</sup>

<sup>†</sup> 北陸先端科学技術大学院大学情報科学研究科

<sup>‡</sup> 北陸先端科学技術大学院大学情報科学センター

<sup>† ‡</sup> 〒 923-1292 石川県能美市旭台 1-1

E-mail: <sup>† ‡</sup>{minaxay, shikida}@jaist.ac.jp

あらまし 本論文では、中央管理サーバが必要でない、Pure P2P ネットワーク技術を用いた連続ストリーミング配信法を提案する。P2P 配信では、すべてのクライアントは配信サーバに集中アクセスすることなく、クライアント間でストリーミングデータをやり取りすることができ、配信サーバの負荷やネットワーク帯域幅を減少させることができる。しかし、クライアント・サーバ型ストリーミング配信と異なり、P2P 型ストリーミング配信では、クライアント (Peer) の参加・切断は自由に行われるため、不安定かつ高遅延な配信ネットワークになる。そこで本稿では、配信 Peer が切断した時に注目し、特性の近い配信 Peer を 2 台ごとにクラスター化する。クラスター内の配信 Peer には主 Peer と副 Peer を定義し、互いに監視する。配信 Peer を二重化することにより、配信 Peer の主 Peer が切断しても、受信 Peer は同じストリーミングデータを持つ副 Peer に迅速に切り替えられる。

キーワード P2P ストリーミング配信, Peer の二重化, Peer のクラスター化

## Proposal of Live Streaming Distribution using Self-Restructuring P2P Network

Minaxay PHILAVONG<sup>†</sup> and Mikifumi SHIKIDA<sup>‡</sup>

<sup>†</sup> School of Information Science, Japan Advanced Institute of Science and Technology

<sup>‡</sup> Center for Information Science, Japan Advanced Institute of Science and Technology

<sup>† ‡</sup> 1-1 Asahidai, Nomi-shi, Ishikawa, 923-1292 Japan

E-mail: <sup>† ‡</sup>{minaxay, shikida}@jaist.ac.jp

**Abstract** This paper proposes the live media streaming system by using Pure P2P network technology. In P2P streaming network, all clients need not to access streaming server, and the client itself can distribute streaming data to other client, that cause the reduction of load and bandwidth for the server. However, it is different from Client-Server system, in P2P streaming system, client can join and leave randomly, and that makes streaming network unstable and delay. The proposed architecture is designed by grouping 2 sending Peers with same functions into a cluster. In the cluster, 2 Peers are defined as Main-Peer and Sub-Peer. When one of sending Peer (or Main-Peer) fail or leave the system, receiving Peer can shift to Sub-Peer, which contain the same streaming data rapidly, and maintain to receive the streaming data.

**Keywords** P2P Streaming system, Double Peers buffering, Peer Clustering

## 1 まえがき

現在、コンピュータのプロセッサの高速化、メモリの大容量化、ネットワークの広帯域化や常時接続インターネットの普及により、インターネット上で様々な種類のサービスが提供されるようになってきている。特に音楽や動画のようなマルチメディアのストリーミング配信の利用が広まっている。

しかし、ストリーミング配信は連続性と即時性を求められる。従来のストリーミング配信方式では、ユーザがインターネットを通じてメディア配信サーバにアクセスし、ストリーミングデータを受信して、到着したデータを順次再生する。いわゆるクライアント・サーバ型 (C/S 型) である。しかし、クライアント台数が増えると、配信サーバにアクセスが集中することから、高品質のストリーミング配信を維持するためには配信サーバの処理能力を上げなければならないし、ネットワーク帯域幅を増強する必要がある。そのため、ストリーミング配信用の設備投資コストが増大になる。

より安いコストかつ安定的なストリーミングを配信するため、Peer-to-Peer (P2P) 技術を用いたストリーミング配信が注目されている。P2P ネットワークによるストリーミング配信は、全てのクライアントが配信サーバにアクセスすることなく、クライアント間でストリーミングデータのやり取りができる。新たな通信形態である。しかし、P2P によるファイル共有と異なり、ストリーミングデータは連続性が求められる。P2P ストリーミング配信では、クライアント (Peer) は自由に配信ネットワーク構造に参加・切断が行われている。しかし、上位レベルの配信 Peer が配信システムから離脱する場合、受信 Peer はストリーミングデータを受信できなくなり、他の配信 Peer の再検索を行う。そのため、受信 Peer が遅延や不安定なストリーミング再生になる。

この背景から、安定な P2P のストリーミング配信の研究が注目されている。先行研究 [7] では、中央管理サーバを用いてデータブロックの配信管理や、ネットワーク構成の管理による P2P ストリーミング配信 (Hybrid P2P) が研究されている。Peer の参加・離脱は管理サーバを経由で行われ、組織内でのストリーミング配信である。

そこで、本稿では Pure P2P による P2P ストリーミング配信を提案する。データブロックの配信とネットワーク構成の管理は Peer 自律で行われる。

また、送信 Peer をグループ化し、Peer の 2 台ごとにクラスターを作成する。クラスター内は特性の近い 2 台の Peer から構成される。送信 Peer は主 Peer と副 Peer に二重化される。送信 Peer が分断された際には、受信 Peer がクラスター内にある、前の主 Peer から同じストリーミングデータファイルを持つ副 Peer へと自動的に切り替えることにより、P2P ネットワークでの連続ストリーミング配信を実現することを目的とする。

本稿では、2 章で従来方式のストリーミング配信方式について述べ、その問題点を説明する。3 章で提案方式の詳細なアルゴリズム及び動作等について記述し、4 章では提案方式について議論する。

## 2 従来方式

従来のストリーミング配信である Unicast 配信と Multicast 配信及び既存の P2P ストリーミング配信の研究について、特徴と問題点を述べる。

### 2.1 Unicast-based と Multicast-based

Unicast 配信の代表である Content Distribution Network (CDN) はクライアントの近いところに Cache サーバを置き、ユーザは少ないネットワークホップを経由し、ストリーミングデータを受信することができ、ネットワーク帯域幅の問題や配信遅延の問題を回避できるが、設備の投資コストや管理が問題になる。例えば、Akamai Technologies 社のオンデマンド分散型コンピューティングプラットフォームは、65 カ国、1,100 ネットワークにある 14,000 台以上のサーバから構成されている [1]。

一方、Multicast 配信である IP Multicast 技術では、ネットワーク層 (ルータ) でパケットを複製し、特定の宛先のクライアントにパケットを送信することにより、バックボーンネットワークの帯域幅を有効に利用できるが、IP Multicast 配信をルータがサポートする必要があり、実装や管理上の問題がある [2]。

### 2.2 P2P によるストリーミング配信

IP Multicast をネットワーク層で実装することが困難であることから、アプリケーション層 (End-Host) マルチキャスト (Application-Layer Multicast) に関する研究が行われている [3, 4, 5, 6]。ALM 技術を用いた P2P ストリーミング配信に関する研究は、先行研究 [7] では Hybrid P2P によるストリーミ

ング配信であり、スケジュール管理サーバやネットワーク構成管理サーバが必要となり、Peerの参加/切断は管理サーバを経由して、ネットワーク構造が構成される。また、ネットワーク規模は組織内のサブネットワークに制限されている。

現在、P2P技術を用いた、インターネット放送用ソフトウェアであるPeerCast[8]では、Join/Leaveアルゴリズム(図1)でシステムが構成される。Peerが参加したい場合は、Joinアルゴリズムにより、参加Peerはソースノードに接続要求を出し、接続する。しかし、ソースノードが接続不可能な時には、配信Peerは自分の子Peerを選択し、親Peer候補として、参加したいPeerに子Peerの情報を返す。参加Peerは届いた子Peerの情報に基づいて、子Peerに接続要求させる。参加Peerは接続可能になるまでこのJoinアルゴリズムを繰り返す。また、Peerが切断する時には、Leaveアルゴリズムを用いて、離脱配信Peerは自分の受信Peer(子Peer)に、自分の親Peerの情報を渡し切断する。しかし、ネットワーク障害などにより、子Peerが新親Peerの情報を受信できない場合は、子Peerは再Joinすることができなくなり、ソースノードに再接続要求しなければならないため、Joinアルゴリズムをソースノードから再び行う。そのため、再検索の時間がかかり、連続ストリーミング受信できなくなる[9]。

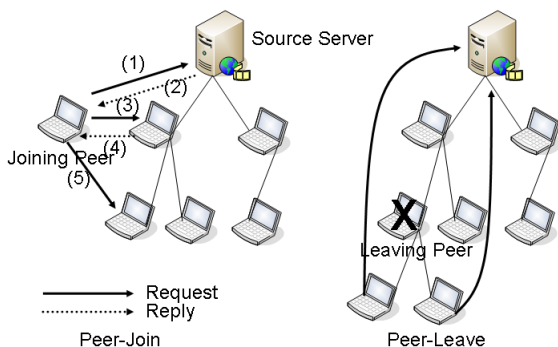


図1: PeerCastのJoin-Leaveアルゴリズム

### 3 二重化P2Pによるストリーミング配信

本稿では、中央管理サーバを必要としないPure P2Pによる連続ストリーミング配信手法を提案する。以下に手順とアルゴリズムを述べる。

本提案方式は、3つのアルゴリズムから構成される。まず、“参加アルゴリズム”で、参加したいPeer

からP2P配信ネットワークに参加し、検索技術により、隣接のストリーミングデータを持つ配信Peerに接続要求を出す。次に、“グループ化アルゴリズム”により、参加Peerと配信Peerがコネクションが成立する場合、参加Peerは配信Peerの元に管理され、配信Peer(ParentPeer)は参加Peer(新ChildPeer)と今あるChildPeerとグループ化して、クラスターを作成する。最後のアルゴリズムは、“切断アルゴリズム”で、配信Peerはストリーミングネットワークから離脱する場合、受信Peerは連続ストリーミングの受信を維持する。

#### 3.1 Peerの参加アルゴリズム

参加したいPeer(NewPeer)は(図3)、P2P検索技術を用いて、隣接している接続可能配信Peerの検索を行い、ストリーミングデータを持つ配信Peer(Sending Peer)に接続要求を送信し、候補ParentPeerとする。接続する前に候補ParentPeerは自分のParentPeerを確認し(図4)、上位レベル階層のクラスター内にあるPeerが接続可能かを確認する(図3, 行4-7)。それによって、ツリーのバランス(全体ツリーではない)を考慮し、ネットワークによる配信遅延を減らすことで、受信Peerはストリーミング配信サーバからの配信リレー時間をより小さくできる(図2)。上位レベル階層のクラスター内にあるPeerが接続不可能である場合は、候補ParentPeer(MainPeer)及び同クラスター内のSubPeerの接続可能かを確認する。全てのMainPeerとSubPeerが接続不可能の場合、その候補ParentPeerのChildPeerの一つを選択し、そのChildPeerの情報を、参加したいPeerに渡し、接続要求をさせる。以降からはChildPeerのクラスター内にあるPeerだけに接続可能かどうかを確認し、接続可能まで繰り返す。

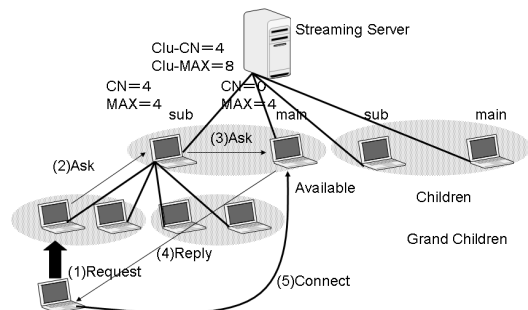


図2: Peerの参加アルゴリズム

```

/* Receiving Peer */
01. Add(streaming network)
02. Search(neighbor Peer)
03. Request(JOIN)→ neighbor Peer
04. for (t=0; t<TIMEOUT; t++)
05. if Receive(JOIN_ACK)
06. do Wait(GOTO)
07. if Receive(GOTO=Peer)
08. Connect(NewPeer, Peer) break
09. else Receive(JOIN_FAIL)
10. else Receive(JOIN_FAIL)
11. if Receive(JOIN_FAIL)
12. do Search(neighbor Peer)

```

図 3: 受信 Peer 側の参加アルゴリズム

```

/* Sending Peer */
01. if Receive(JOIN)←NewPeer
02. Reply(JOIN_ACK)
03. for (t=0; t<TIMEOUT; t++)
04. Ask(ParentPeer)
05. if ParentPeer_Connection<MAX
06. do Reply(GOTO=ParentPeer) break
07. ParentPeer_Connection++
08. else Ask(MainPeer)
09. if MainPeer_Connection<MAX
10. do Reply(GOTO=MainPeer) break
11. MainPeer_Connection++
12. else Ask(SubPeer)
13. if SubPeer_Connection<MAX
14. do Reply(GOTO=SubPeer) break
15. SubPeer_Connection++
16. else Replay(GOTO=ChildPeer) break

```

図 4: 配信 Peer 側の参加アルゴリズム

### 3.2 Peer のグループ化と二重化アルゴリズム

図 5 に示すように、まず、参加したい Peer は P2P ストリーミング配信ネットワークに参加し、ストリーミング配信 Peer に接続要求を出し、要求を受け取った配信 Peer は候補の配信 Peer になる。ストリーミング配信 Peer(及びサーバ) は接続制限台数 (MAX) を設定する。配信 Peer を ParentPeer と呼び、受信クライアントを ChildPeer と呼ぶ。前述の“参加アルゴリズム”により、要求 Peer は配信 Peer に

コネクションの成立を判定する。配信 Peer に接続されている ChildPeer は、“サーバからの位置・距離 (Prefix IP Address), RTT(Round Trip Time)”と“提供帯域幅 (Shared BandWidth)”と“接続時刻・順番 (Connection Order)”からなる 3 つのパラメータ (図 6) を考慮し、特性の近い ChildPeer を 2 台ごとにグループ化し、クラスターにする。クラスター内の 2 台の ChildPeer は互いに監視し、ParentPeer は ChildPeer のクラスターを管理する。クラスターにある 2 台の Peer は参加する Peer の配信 Peer になり、受信 Peer (Receiving Peer) にとって、直接の配信 Peer は MainPeer と呼び、MainPeer と同クラスター内にある Peer は SubPeer と呼ぶ。これにより、受信 Peer は二重化の配信 Peer を持つ。

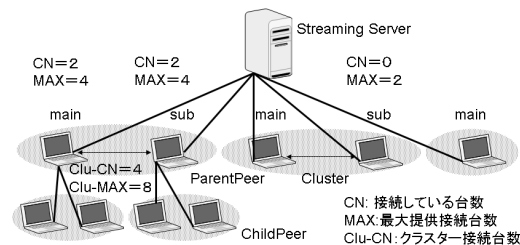


図 5: Peer のグループ化、二重化アルゴリズム

```

/* Grouping */
01. if Connect(NewPeer, Peer)
02. Peer→Measure(RTT(NewPeer, Peer))
03. Peer→Get(PrefixIPAddress(NewPeer))
04. Peer→Get(SharedBandWidth(NewPeer))
05. if PrefixIPAddress(NewPeer)=
06. PrefixIPAddress(ChildPeer)
07. if SharedBandWidth(NewPeer)=
08. SharedBandWidth(ChildPeer)
09. do Group(NewPeer, ChildPeer) break
10. if RTT(NewPeer)=RTT(ChildPeer)
11. if SharedBandWidth(NewPeer)=
12. SharedBandWidth(ChildPeer)
13. do Group(NewPeer, ChildPeer) break
14. else Group(by connection order)

```

図 6: 受信 Peer のグループ化アルゴリズム

### 3.3 Peer の切断アルゴリズム

Peer は ChildPeer を持っていない時、Peer が切断すると、ParentPeer の ConnectionNumber(現在接続している台数) を 1 減少し、クラスターの総合 Con-

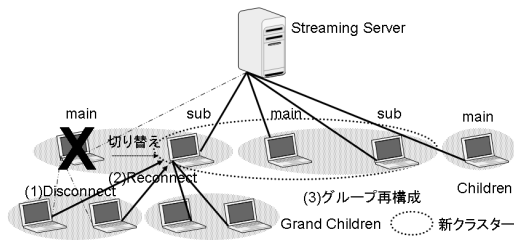


図 7: Peer の切断アルゴリズム

/\* Leaving \*/

1. Cluster(MainPeer, SubPeer)
2. Connect(ChildPeer, MainPeer)
3. if Disconnect(MainPeer)
4. Connect(ChildPeer, SubPeer)
5. Group(SubPeer)

図 8: Leave Algorithm

nectionNumber を 1 減少する。Peer が ChildPeer を持っているとき、Peer が切断する時に、Peer のクラスター内にある SubPeer に ChildPeer を切り替えて、接続する。その時、クラスター内では ParentPeer が一つの Peer となる。SubPeer は自分の ChildPeer と前の MainPeer の ChildPeer の担当になることにより、SubPeer の負荷が増大し、そのため、SubPeer は同レベルのクラスターに参加させる。クラスターに参加するのは、参加する予定のクラスター内の Peer の BufferData (一時的に保存しているストリーミングデータブロック) が SubPeer の BufferData に近いものであると定義する。新しく構成したクラスターは 2~3 台の Peer の構成になる。新クラスター内では、ChildPeer のクラスターの担当役割を再構成し、再グループ化する (図 7)。

クラスター内に ChildPeer を持っていない SubPeer が離脱する場合でも、ChildPeer を持つ MainPeer がグループ化アルゴリズムを用いて、他のクラスターに参加する。よって、受信 Peer は二重化の送信 Peer を維持することができる。

### 3.4 システムの概略

提案方式に基づくシステムの概要 (図 9) は、P2P の検索機能、Peer のグループ化機能、Peer の管理機能からの 3 つ構成される。

- P2P 検索機能: P2P ストリーミング配信ネットワークで、参加 Peer から隣接の配信 Peer を検索する機能

- Peer のグループ化機能: 送信 Peer は自分の特性の近い 2 台ごとの子 Peer をグループ化する機能
- Peer の管理機能: MainPeer, SubPeer, ChildPeer はお互いに監視する機能

この 3 つの機能により、物理路ネットワークから論理ツリー型のネットワーク構造を構築し、最適化されたマルチキャストルーティング (データフローの最適化) を行い、ストリーミングデータの送受信し、Peer の参加/切断によるネットワーク構造を自動再構築する。

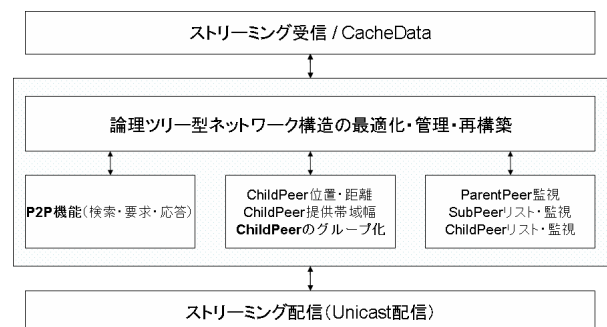


図 9: システムの概略図

## 4 考察

本章は提案方式と従来方式を比較し、提案方式について議論を行う。

### 4.1 Peer の参加

PeerCast では、参加 Peer はまずソースノードに接続要求を出し、接続可能になるまで、ソースノードから子 Peer へ、トップダウン型で検索する。そのため、参加 Peer がストリーミングデータを受信できるまで、検索時間がかかる。

一方、本提案方式では、参加 Peer はソースノードから接続要求せず、P2P の検索機能を用いて、隣接 Peer に接続要求するため、ソースノードからの検索時間を短縮することが可能である。また、配信 Peer は隣接 Peer であることにより、Peer 間におけるネットワークの遅延が減少される。

### 4.2 Peer ネットワークの構成・管理

先行研究 [7] では、中央管理サーバによる配信ネットワークを構成・管理する。組織内でのストリーミングを想定し、Peer の参加・離脱は中央管理サーバを経由して迅速に行う。しかし、大規模なネット

ワークでの P2P ストリーミング配信する場合, 中央管理サーバによる全体のネットワークにある Peer を管理することは困難である.

本提案方式では, 中央管理サーバを必要とせず, 物理 P2P ネットワーク構造を論理ツリー構造にし, ネットワーク構造を効率的に管理する. 根である配信サーバから, ParentPeer は ChildPeer を管理することで, ネットワーク全体の管理コスト・制御メッセージを減少することができ, 大規模なストリーミング配信システムを自律的に分散管理することが可能である. MainPeer, SubPeer と ChildPeer は互いに監視することにより, ネットワーク障害によって一時的に MainPeer から ChildPeer に通信できない場合でも, ChildPeer は SubPeer を経由して MainPeer の状態を確認することができる.

本提案方式では, ParentPeer はストリーミングデータを複製し, ChildPeer まで送信するが, ParentPeer の処理能力が低い場合や提供帯域幅が少ないなど, ChildPeer のストリーミング受信に影響がある. そのため, ネットワーク構造は完全木構造だけではなく, ChildPeer は上位レベルの多 ParentPeer からストリーミングデータを受信可能にし, メッシュ型木構造 [10] によるストリーミングの遅延が短い P2P ネットワークについては今後の課題にする.

### 4.3 Peer の離脱

PeerCast では, ParentPeer が切断した時, ネットワーク障害などで ChildPeer が前 ParentPeer から新 ParentPeer をもらえない場合, ChildPeer はソースノードから, 接続可能かつ同データファイルを持つ Peer を再検索することになる. そのため検索の時間が増えると, ストリーミング配信の連続性を維持することが困難になる. 本研究の提案方式では, 参加する Peer を 2 台ごとにグループ化し, 互いに監視することにより, ChildPeer は自分の配信 ParentPeer を MainPeer とし, 同クラスター内にある ParentPeer を SubPeer とする. MainPeer が離脱・切断されると, ChildPeer は同じデータファイルを持つ SubPeer に迅速に切り替えることにより, 連続ストリーミング配信が可能になる. 主 Peer がいなくなった副 Peer は他のクラスターに参加し, 送信 Peer の二重化を維持することができる. その結果, テレビやラジオ放送のような長時間・連続ストリーミング配信に適用可能であると期待できる.

## 5 むすび

本研究では, Pure P2P 型によるストリーミング配信法を提案した. 物理 P2P ネットワーク構成を論理ツリー構造にし, Peer がネットワーク構成を自律で管理する. 参加する Peer を 2 台ごとにグループ化し, 送信 Peer を二重化することにより, ParentPeer(MainPeer) が切断した時, ChildPeer は ParentPeer の同クラスター内の SubPeer に即時に切り替えることができ, ストリーミングの連続性を保ちながら, 大規模な長時間連続ストリーミング配信に適用できることを示した.

今後の課題として, 提案方式のアルゴリズムに基づき, プロトタイプの実装を実装し, シミュレーションする. これにより, システムを評価し, 本研究の提案方式の有効性を示す.

## 参考文献

- [1] Akamai Technologies, Inc. <http://www.akamai.com/>
- [2] C. Diot, B. Levine, J. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the IP multicast service and architecture", IEEE Network, Vol.14, No. 1, pp. 78-80, January 2000.
- [3] Y. H. Chu, S. G. Rao, S. Seshan, and H. Zhang, "A Case for End System Multicast", ACM SIGMETRICS 2000, pp. 1-12, ACM Press, June 2000.
- [4] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel, "ALMI: An application level multicast infrastructure", USITS 2001, pp. 49-60, March 2001.
- [5] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast", ACM SIGCOMM 2002, pp. 205-217, ACM Press, August 2002.
- [6] 三村 和, 中内 清秀, 森川 博之, 青山 友紀, "RelayCast: ピアツーピア型ストリーム配信のためのミドルウェア", 電子情報通信学会技術研究報告, IN2002-43, Vol. 102, No. 214, pp. 7-12, July 2002.
- [7] 末次 信介, 若宮 直紀, 村田 正幸, 小西 弘一, 谷口 邦弘, "物理網構成を考慮したハイブリッド型 P2P 動画ストリーミング配信機構の提案と評価", 電子情報通信学会技術研究報告, CQ2004-57, Vol. 104, No. 187, pp. 47-52, July 2004.
- [8] PeerCast. <http://www.peercast.org/>
- [9] 國近 洋平, 甲藤 二郎, 大久保 榮, "予備親探索機能を有したアプリケーションレベルマルチキャスト", 電子情報通信学会 情報ネットワーク研究会, IN2003-210, Vol. 103, No. 691, pp. 7-12, March 2004.
- [10] S. W. Tan, A. G. Waters, and J. Crawford "MeshTree: Reliable Low Delay Degree-bounded Multicast Overlays", ICPADS 2005, pp. 565-569, July 2005.