P2P ネットワークにおける木構造に基づく複製更新伝播法

渡辺俊貴[†] 原 隆浩[†] 木戸裕樹[†]
中通 実[†] 西尾章治郎[†]

近年の計算機の高性能化やネットワークインフラの発達により，Peer-to-Peer（P2P）ネットワークへの関心が高まっている．P2P ネットワークでは，検索効率やデータの可用性向上，負荷分散のためにデータを複製し，ネットワーク上の複数のピアに配置することが有効である．しかし，データに更新が発生した場合，複製を持つピアに対して，更新情報を通知する必要がある．本論文では，複製を持つピアからなる n 分木の論理ネットワークを形成し，その木に沿って更新を伝播することにより，負荷分散と遅延減少を両立した複製更新伝播法を提案する．提案方式では，更新伝播木中の各ピアが，自身の先祖と子の情報を保持することにより，ピアの退出時などに木の再構成を行うことができる．

An Update Propagation Method for Replicas Using a Tree Structure in Peer-to-Peer Networks

Toshiki Watanabe,[†] Takahiro Hara,[†] Yuki Kido,[†]
Minoru Nakadori[†] and Syojiro Nishio[†]

Due to recent advances in computers and network technologies, there has been increasing interest in peet-to-peer networks. In peer-to-peer networks, data items are usually replicated to improve searching efficiency, data availability, and load distribution. However, when a data item is updated, the update information should be immediately notified to all peers holding the replicas. In this paper, we propose a strategy that creates an n-ary tree whose root is the owner of the original data and the other nodes are peers holding its replicas, and propagates the update information according to this tree for achieving not only load balancing but also delay reduction. In this strategy, peers participating the tree maintain the information on their ancestors and children nodes, and reconstruct the tree using the information when some peers in the tree fail.

1. はじめに

近年の計算機の高性能化やネットワークインフラの発達により，Peer-to-Peer（P2P）ネットワークへの関心が高まっている[2),3),6),8)~10)]．Web サービスなどの既存のネットワークサービスで最も主流となっているクライアント・サーバモデルでは，単一のサーバがすべてのサービスを提供するため，クライアント数の増加にともない，サーバの処理負荷が増大してしまう．また，サーバに障害が発生した場合，サービス全体が停止してしまう．一方，P2P ネットワークでは，クライアント，サーバのような明確な役割分担はなく，互いにサービスを提供しあうことにより，単一障害点がなくなり，高いネットワーク耐性やスケーラビリティを実現できる．

P2P ネットワークの中でも，インデックス情報を管理するサーバが存在せず，完全な分散環境であるピュア P2P ネットワークの構成（トポロジ）は，構造（structured）型と非構造（unstructured）型に分類される．構造型トポロジを用いた P2P システムは，DHT（Distributed Hash Table）などを用いて，ネットワークのトポロジやデータを検索するために必要なキー（データとデータを提供するピアの識別子の組）の配置を厳密に決定するシステムである[8),9)]．一方，非構造型トポロジを用いた P2P システムでは，DHT のように，明確な方針に従って決定された検索トポロジが存在しないため，データの検索には，フラッディングやその派生型などの無作為な検索方法が用いられるが，トポロジに制約がなく，柔軟な検索ができ，任意の論理ネットワークを構成できる[5)]．

P2P ネットワークでは，検索効率やデータの可用性向上，負荷分散のために，データを複製し，ネットワー

† 大阪大学大学院情報科学研究科マルチメディア工学専攻
Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University

ク上の複数のピアに配置することが有効であると考えられている．一方，リアルタイムの天気予報やニュース速報，分散 Web コンテンツなどの分散ファイルシステムサービスでは，複数のピアで共有しているデータに更新が発生する環境が想定される．データに更新が発生した場合，更新データの通知が遅れると，更新前の古い複製にアクセスする可能性が高くなる．そのため，古い複製へのアクセスが許されないアプリケーションでは，複製を所持するすべてのピアに対して，更新データを即座に通知する必要がある．

本論文では，各ピアが非構造型 P2P ネットワークにおいて，他のピアが所持するオリジナルデータおよび複製にアクセスする環境を想定する．各ピアは，自身のデータ記憶領域に他のピアが所持しているデータの複製を作成する．ピアがデータを要求する場合，ネットワーク内に検索クエリをフラッディングすることにより，データを所持するピアが存在するかを問い合わせる．検索において，要求データを所持するピアを複数発見した場合，それらのうち論理ネットワーク上のホップ数が最も小さいピアにアクセスするものとする．データへのアクセスが行われると，既存の複製配置方式によって，複製が作成され，そのデータに関する更新伝播用論理ネットワークへの参加が行われる．また，ピアのデータ記憶領域には限りがあるため，新しくデータの複製を作成する際に，他の複製を削除しなければならない場合がある．複製を削除するときには，削除したデータに関する更新伝播用論理ネットワークからの脱退が行われる．

複製を持つピアに更新データを通知するための単純な方法として，オリジナルデータを持つピア（以下では，オリジナルノードと表記する）が，そのデータの複製を持つすべてのピアのネットワーク上での識別子（IP アドレスなど）を管理しておき，更新が発生するたびに複製を持つすべてのピアに更新データを直接伝播する方法（以下では，放射伝播法と表記する）が考えられる．この方法を用いた場合，複製を持つすべてのピアに短時間で更新データを通知することができるが，更新伝播時の負荷がオリジナルノードに集中する．さらに，複製を持つピア数の増加にともない負荷が線形的に増加するため，大規模なネットワークには適していない．

この問題を解決する別の更新伝播法として，複製を持つ各ピアが，同じ複製を持つ他の 1 つのピアへ，直線的に更新を伝播する方法（以下では，直線伝播法と表記する）が考えられる．この方法により，更新伝播時における各ピアの負荷を分散することができるが，複製を持つピアの増加に従って，更新を伝播する経路は線形的に長くなる．したがって，オリジナルノードから複製を持つすべてのピアへ更新の伝播が完了するまでに，大きな遅延が生じてしまう．

そこで，本論文では，オリジナルノードを根とし，複製を持つピアを内部節点とする n 分木の論理ネットワーク（更新伝播木）を形成し，この更新伝播木に沿って更新データを伝播する複製更新伝播法を提案する．この手法により，更新伝播時の負荷分散と遅延減少を両立させることができる．さらに，複製の作成・削除時や，ネットワーク障害などによるピアの退出が発生した場合にも，更新伝播木は自律分散的に再構成される．

以下では，2 章で関連研究について述べ，3 章で木構造に基づく複製更新伝播法について説明する．4 章で，障害耐性の向上を目的とした更新伝播木の再構成法について説明する．5 章で提案手法の性能評価を行い，最後に 6 章で本論文のまとめと今後の課題について述べる．なお，本研究の成果の一部は，文献 7) において公表している．

2. 関 連 研 究

P2P ネットワークサービスに関する研究分野では，データの複製配置に関する研究がさかんに行われているが，複数のピアで共有されるデータに更新が発生する環境を考慮した研究は，あまり行われていない．しかし実環境では，データに更新が発生する場合が想定され，更新データの通知が遅れると，更新前の古いデータにアクセスしてしまう可能性が高くなる．そのため，複製を持つすべてのピアに対して，即座に更新データを通知する機構が必要となる．以下では，従来研究における複製更新伝播法について説明する．

2.1 確率的な更新伝播法

Datta らは，P2P ネットワーク上の複製を持つピアに対して，確率に基づいて更新データを伝播させる方法を提案している[4]．データを更新したピアは，同じデータの複製を持つピアのうちのいくつかを隣接ピアとし，それらのピアに対してある確率（更新伝播率）で更新データを伝播する．一度更新を伝播したピアは，部分リストと呼ばれるリストに加えられ，更新データと一緒に隣接ピアに通知される．更新データを受け取ったピアは，リストに存在しない隣接ピアに対して，更新伝播率に基づいて更新データを伝播する．この更新伝播率の値は，更新発生元ピアからの論理ネットワーク上のホップ数が大きくなるにつれて小さくなる．つまり，更新データが多くのピアに伝わるにつれて，隣

接ピアに更新データを送る確率が低くなる．しかしこの方法では，すべてのピアに更新が伝播される保証がない．さらに，1 つのピアに対して複数の経路を通って更新が伝播されることがある．

2.2 チェイン構造を用いた更新伝播法

Wang らは，複製を持つピアを一直線のチェイン上に配置する論理ネットワークを形成することにより，更新伝播時のオーバヘッドを抑えつつ，ネットワーク耐性の向上を実現する方法を提案している[10]．この方法では，各ピアは，左右 m 個ずつのピア情報（IP アドレスなど）を自身の調査ノードとして保持する．チェイン上のピアで更新が発生した場合，更新発生元ピアは，左右 m 個の調査ノードに更新データを送信する．それぞれの方向の m 個の調査ノードのうち，最も遠くにあるオンライン上のノードを更新伝播責任ノードとし，更新データが送られてきた方向とは反対方向の調査ノードに対して，同様に更新データを伝播し，次の更新伝播責任ノードを決定する．以上の操作を，左右両方向のピアに対して繰り返すことにより，オンライン上のすべてのピアに更新データを伝えることができる．しかしこの方法では，複製を持つピアを一直線上に並べるため，ピア数が多くなると，すべてのピアに更新データを伝播するまでの遅延時間が長くなってしまう．また，更新データを多くのピアに送信するピアとまったく送信しないピアが現れ，更新伝播時の負荷の偏りが大きくなってしまう．なお，文献 10) では，任意のピアが更新を発生させる環境を想定しているのに対し，本論文で提案する手法では，オリジナルノードのみが更新を発生させる環境を想定している．

2.3 複製所持ピアのリストを利用した更新伝播法

文献 11) では，非構造型 P2P ネットワークにおける問合せの効率化やトラヒックの減少を目的として，効率的なデータ配置方法について議論しており，その中でデータの更新伝播の方法についても述べている．この手法では，オリジナルノードは，そのデータの複製を持つすべてのピアからなるリストを管理する．データに更新が発生した場合，オリジナルノードはそのリストを分割し，分割数と同数のピアをリストから選択して，更新データおよび分割したリストを送信する．更新データおよびリストを受け取ったピアは，そのリスト内のピアへ，更新データとさらに分割したピアのリストを送信する．この操作を繰り返すことにより，更新データを伝播させる．

この手法において，分割数を一定とした場合，本論文の提案手法と類似した更新伝播の動作となる．しかし，この手法では，オリジナルノードは，複製を持つすべてのピアの情報を管理しなければならない．したがって，複製の作成や削除が頻繁に起こるような環境では，リスト管理のための負荷が大きくなってしまう．また，各ピアは，他ピアのネットワークへの参加状態をつねに把握しているわけではないので，更新伝播時にピアの存在を確認するために多少の遅延が発生してしまう．

一方，本論文で提案している更新伝播法では，更新伝播用に木構造型の論理ネットワークを形成し，その木に沿って更新データを伝播する．したがって，オリジナルノードを含め各ピアは，木構造維持に必要となる，同じ複製を持つ少数のピアの情報のみを管理しているだけで，すべてのピアに確実に更新データを伝播することができる．また，ピアの参加や脱退に応じてつねに木構造を修復するため，更新伝播時にピアの参加状態を確認する必要がなく，そのための遅延が発生しない．

3. 木構造に基づく複製更新伝播法

本章では，木構造に基づいて複製の更新を伝播する方式を提案する．提案方式では，検索に用いるネットワークとは別に，オリジナルノードを根とし，複製を持つピアを内部節点とした n 分木の論理ネットワーク（更新伝播木）を形成し，この更新伝播木に沿って更新データを伝播する．これにより，更新伝播時の負荷分散と遅延減少を両立させる．この方式では，更新伝播木に参加する各ピアが **図 1** のように，更新伝播木における親の方向に対して k 個分の先祖ノード，および，子ノードの情報（IP アドレスなど）を管理しておき，更新伝播木上での参加位置の決定，および複製削除時などの更新伝播木の再構成を自律分散的に行う．

このように，各ピアが自身の k 個上位の先祖ノードの情報を所持することにより，ネットワーク障害な

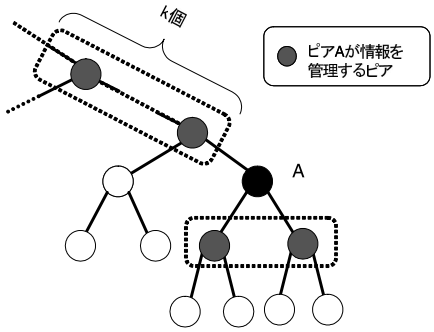

図 1 更新伝播木上のピアの管理情報
Fig. 1 Information managed by a peer in an update propagation tree.

どによっていくつかのピアが退出してしまった場合にも，各ピアが所持するピア情報のみを用いて，局所的に更新伝播木を修復することが可能となる．ただし，k の値を大きくすることにより，ピアの参加や脱退によってピアの位置が変化したときに，より多くのピアに新しい位置情報を伝えなければならず，木構造維持に必要なメッセージ数が多くなることが考えられる．

以下では，新規ピアの更新伝播木への参加方法，および複製の削除による更新伝播木からの脱退の方法を説明する．

3.1 新規ピアの参加

新たに複製を作成したピアは，そのデータの更新伝播木に参加する必要がある．ここで，新たに更新伝播木に参加するピアを新規参加ピア，新規参加ピアの参加位置の決定を行うピアを責任ノードと呼ぶ．最初は，クエリに応答したピアが責任ノードとなる．更新伝播木への参加手順は以下のとおりである．

(1) 責任ノードは，自分自身が更新伝播木の根（オリジナルノード）以外であれば，更新伝播木上の自身から k 個上位の先祖ノード（先祖ノードが k 個未満の場合は根）にあたるピアに子の数 x を問い合わせる．このとき，$x < n$ の場合は，新規ピアをそのピアの子とし，そのピアは自身の子に関する情報として，新規ピアを追加する．さらに，新規ピアはそのピアから $k-1$ 個分（先祖ノードの数が $k-1$ 個未満の場合は根まで）の先祖ノードの情報を受け取り，そのピアと $k-1$ 個の先祖ノードの情報を，自身の先祖ノードの情報として記録する．$x = n$ の場合は，同様の処理を $k-i$（$i = 1, 2, \cdots, k-1$）個上位の先祖ノードに対して，新規ピアの参加位置が決まるまで繰り返し（i を 1 つずつ増やしながら）行う．これらの処理により，責任ノードが新規ピアの参加位置を決定できなかった場合，および，自身が更新伝播木の根のとき，手順 (2) へ進む．

(2) 責任ノードは，自身の子の数 y を確認し，$y < n$ のときは，新規ピアを自身の子とし，$k-1$ 個分（先祖ノードの数が $k-1$ 個未満の場合は根まで）の先祖ノードの情報を新規ピアに送る．また，責任ノードは，自身の子に関する情報に，新規ピアを追加し，手順を終了する．$y = n$ のとき，手順 (3) に進む．

(3) 責任ノードは，自身の子の中からランダムに 1 つの子を選択し，その子を新たに新規ピアの責任ノードとする．新たに責任ノードとなったピアは，(2) 以降の手順に従って，新規ピアの更新伝播木における参加位置を決定する．

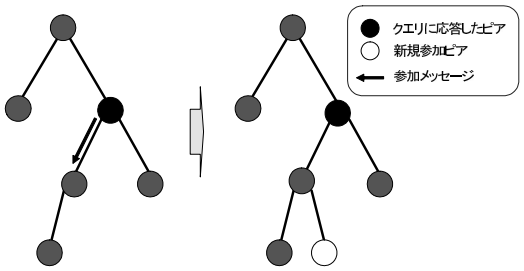

図 2 ピアの参加例
Fig. 2 An example of peer's participation.

$n = 2$ とし，2 分木を作成する場合の参加手順の動作例を，**図 2** に示す．図 2 では，最初の責任ノードを含む先祖ノードの子の数が n の場合を表しており，新規ピアは最初の責任ノードの子の子として，更新伝播木に参加している．

3.2 複製の削除によるピアの脱退

一般的に，ピアのデータ記憶領域には限界があり，新たにデータの複製を作成する際に，他のデータの複製を削除しなければならない場合がある．複製を削除すると同時に，そのピアはそのデータの更新伝播木からも脱退することになるが，その際，木が分断されてしまい，脱退するピアの子孫ノードは，以後の更新情報を受け取ることができない．そのため，分断された木を修復する必要がある．ここで，更新伝播木から脱退するピアを脱退希望ピアと呼ぶ．更新伝播木の修復の手順は以下のとおりである．

(1) 脱退希望ピアが葉ノードの場合，脱退希望ピアは親ノードに木から脱退することを伝える．その後，親ノードは脱退希望ピアの情報を，自身の持つ子ノードに関する情報から削除する．脱退希望ピアは，親ノードを含む k 個分の先祖ノードの情報を削除し，手順を終了する．

(2) 脱退希望ピアが子を持つ場合，脱退希望ピアは自身の子の中から 1 つをランダムに選択し，脱退メッセージを伝える．メッセージを受け取ったピアは，自身が葉ノードでない限り，同様の手順で子を選択しメッセージを伝播させる．脱退メッセージが葉ノードに達したら，その葉ノードの位置を脱退希望ピアの位置と入れ替える．葉ノードの親ノードは，その葉ノードの情報を，自身の持つ子ノードに関する情報から削除し，手順 (3) へ進む．

(3) 入れ替えられたピアは，新たな位置における親と，深さ k 分の子孫ノードもしくは葉ノード

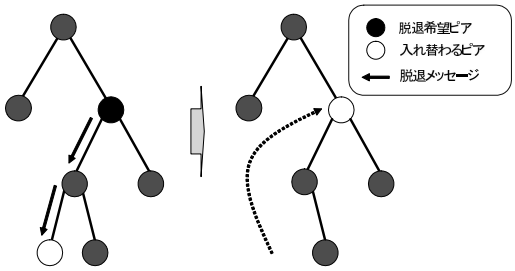

図 3 ピアの脱退例
Fig. 3 An example of peer's exit.

にまで，ピアが入れ替わったことを通知する．通知を受け取った親と子孫ノードは，自身の持つ子ノードおよび先祖ノードに関する情報において，脱退希望ピアを，入れ替えられたピアの情報に変更する．その後，脱退希望ピアは，先祖ノードや子ノードに関する情報をすべて削除し，手順を終了する．

$n = 2$ の場合において，脱退ピアが内部節点であったときの，脱退手順の動作例を**図 3** に示す．

4. 障害発生時における更新伝播木の再構成

3.2 節で説明した脱退手順では，ネットワーク障害や機器の故障などによる，周辺ピアへの通知のない退出（以下では，これを “不当な退出” と呼ぶ）を考慮していない．一方，実環境における P2P ネットワークでは，ピアの不当な退出が発生する場合も多い．ピアがネットワークから不当に退出した場合，更新伝播木の修復を行うことができず，更新データをすべてのピアに伝播することができなくなる可能性がある．更新伝播木が分断した場合に，検索用の論理ネットワークを用いてクエリをフラッディングさせ，データ（複製）を持つピアを発見し，更新伝播木に再参加する方法も考えられるが，大きなトラヒックが発生してしまう．そこで提案手法では，管理している k 個分の先祖ノードの情報を用いて，木の分断が発生したときに，分断した箇所の周辺のピアのみで木を再構成する．そのため，再構成に要するトラヒックを抑制できる．提案手法では，最大 $k-1$ 個までの連続した先祖ノードの不当な退出まで修復が可能となる．

提案手法では，ピアの不当な退出に適応するため，更新伝播木上の各ピアが親ノードに対して，ネットワーク上に存在しているかを確認するためのメッセージ（確認情報）を定期的に送信する．これにより，親ノードの不当な退出を検出できる．ここで，親ノードの不当な退出を検出したピアを修復責任ノードと呼ぶ．木の再構成の手順は，以下のとおりである．

(1) 更新伝播木を構成する各ピアは，親ノードに対して定期的に確認情報を送り，親の生存を確かめる．各ピアは，確認情報に対する親からの応答が得られなかった場合，さらに 1 つ上位の親に確認情報を送る．この操作を，生存するピアが見つかるまで繰り返すことにより，連続する先祖の脱退数 z，および，脱退したピアの位置（脱退認識箇所）を調べる．ここで，初めて生存が確認できた先祖ノードを修復管理ノードとする．

(2) 修復管理ノードから，脱退認識箇所がすでに他のピアによって修復されていることが修復責任ノードに伝えられた場合，修復責任ノードは自身の z 個上位の先祖ノードの情報を修復済みのピア情報に変更する．その後，未修復ノード数 z の値を 1 減らし，その修復済みのピアを新たな修復管理ノードとする．脱退認識箇所がすでに他のピアによって修復されているという情報が修復責任ノードに伝えられる限りこの操作を繰り返し，$z = 0$ となった場合，手順 (5) へ進む．脱退認識箇所が未修復であれば，手順 (3) へと進む．

(3) 修復責任ノードが葉ノードであれば，修復責任ノードは自身が修復管理ノードの子として参加し，修復管理ノードから $k-1$ 個分（先祖ノードの数が $k-1$ 個未満の場合は根まで）の先祖ノードの情報を受け取る．また，通知を受けた修復管理ノードは，自身の子に関する情報を，入れ替えられたピア（修復責任ノード）の情報に書き換え，手順を終了する．

(4) 修復責任ノードが葉ノードでない場合，修復責任ノードは自身の子の中から 1 つをランダムに選択し，先祖ノードの脱退メッセージを伝播させる．脱退メッセージが葉ノードに達したら，その葉ノードを修復管理ノードの子とし，その葉ノードは修復管理ノードから $k-1$ 個分（先祖ノードの数が $k-1$ 個未満の場合は根まで）の先祖ノードの情報を受け取る．また通知を受けた修復管理ノードは，自身の子に関する情報を，(元) 葉ノードの情報に書き換える．(元) 葉ノードの親であったノードは，その (元) 葉ノードを自身の子に関する情報から削除する．その後，未修復ノード数 z の値を 1 減らし，(元) 葉ノードを新たな修復管理ノードとする．z が 0 でない限り，手順 (2)〜(4) の操作を繰り返し，$z = 0$ となれば，手順 (5) へ進む．

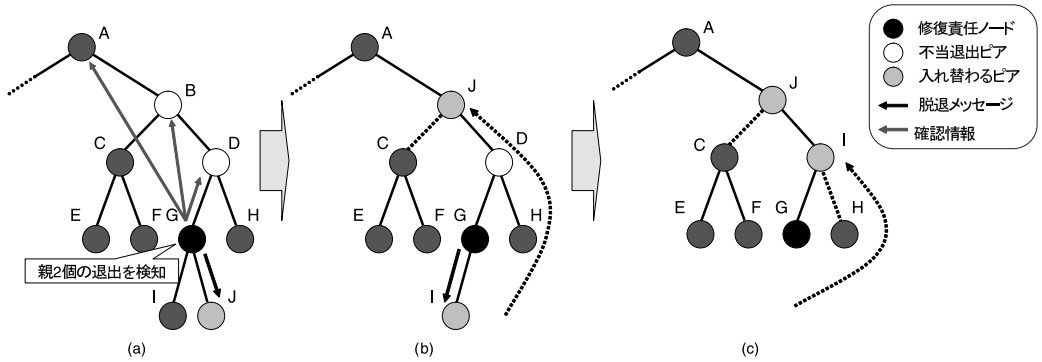

図 4 不当な退出発生時の木の修復法
Fig. 4 An example of repairing an update propagation tree when a failure occurs.

(5) 修復責任ノードは，自身が所持している先祖ノードの情報，および自身の情報を，深さ z 先の子孫もしくは葉ノードにまで伝播させる．その情報を受け取ったピアは，自身の先祖ノードの情報を書き換え，手順を終了する．

この一連の操作により，不当退出を検出したピア，およびそのピアの子孫ノードを木に再参加させることができる．不当退出を検出したすべてのピアがこの操作を完了させることにより，木の修復が完了する．

$n = 2$（2 分木）の場合における，ピアの不当な退出発生時の更新伝播木の修復例を，**図 4** を用いて示す．図 4 (a) のように，ピア B，D が不当に退出した場合，ピア C，G，H が親ノードに対して確認情報を送信することにより，退出を検出することができる．ここでは，ピア G が最初に退出を検出したものとする．ピア G がピア D に対して送る確認情報への応答が得られなかった場合，さらに 1 つ上位の親であるピア B に確認情報を送る．ピア B からも応答が得られない場合，次にピア A に確認情報を送る．ここでピア A からの応答が得られると，脱退している先祖ノードの数が 2（$z = 2$）であると分かる．ピア G は葉ノードではないので，自身の子の中からランダムに選択したピア J に脱退メッセージを送る．ピア J は葉ノードであるため，ピア B とピア D のうち，最も根に近いピア B の位置とピア J の位置を入れ替え，z の値を 1 減らす（図 4 (b)）．ここで，$z = 1$（> 0）なので修復を続け，ピア G はピア I に脱退メッセージを送る．ピア I は葉ノードなので，ピア D の位置とピア I の位置を入れ替え，z の値を 1 減らす（図 4 (c)）．$z = 0$ となり，ピア G は，更新された先祖ノードの情報を伝えるべき子ノードを持たないので，ピア G が担当する修復を完了する．

この時点では，ピア C およびピア H は，更新伝播木が修復されたという情報を得ていない．その後，ピア H が親ノードの退出を検出した場合，ピア D，B，A へと確認情報を送信し，ピア A を修復管理ノードとする．ここでピア H は，ピア A から，すでにピア B の位置がピア J によって修復されていることが伝えられる．情報を受け取ったピア H は，自身が管理するピア B の情報をピア J の情報に書き換え，z の値を 1 減らし，新たにピア J を修復管理ノードとする．次にピア H は，ピア J から，ピア D の位置がすでにピア I によって修復されていることを伝えられる．ピア H は，自身が管理するピア D の情報をピア I の情報に書き換え，z の値を 1 減らす．ここで，$z = 0$ となり，ピア H は，更新された先祖ノードの情報を伝えるべき子ノードを持たないので，ピア H が担当する修復を完了する．ピア C も同様の手順を終えた時点で木の修復が完了する．

1 つの修復責任ノードが修復操作を行っている間に，他のピアからの修復要求が送られてきた場合，修復中であることを伝え，後から送られてきた修復要求を待機させておく．その後，先の修復が完了次第，待機中の修復要求に応じる．また，脱退メッセージを葉ノードへと伝播させている途中で，自身は葉ノードではないにもかかわらず，子がすべて不当に退出してしまっているために，脱退メッセージを葉ノードまで伝播させることができない場合がある．その場合，脱退メッセージの伝播および更新伝播木の修復を中断し，自身の子孫ノードの修復を待ち，修復が完了した後で，再び脱退メッセージの伝播を再開させる．

5. 性能評価

本章では，提案手法の性能評価のために行ったシミュレーション実験の結果を示す．評価では，非構造 P2P ネットワークでのデータ共有を想定した．

5.1 シミュレーション環境

シミュレーション実験における想定環境について説明する．P2P ネットワークに参加するピアの数を 1,000 から 5,000 までの間で変化させ，それらがべき法則（Power-Law Random Graph: PLRG）[1] に従うネットワークを構成するものとした．ここで，i 番目のピアの隣接ピアの数を d_i とし，d_i を以下の式で与えた．

$$d_i = \lfloor 70 \cdot i^{-0.4} \rfloor \quad (1)$$

このように，一部のピアにリンクが集中する環境を実現した．データの種類を 100 とし，全ピアのうち，ピア番号が 1 から 100 までのピアがそれぞれ，データ番号 1 から 100 のデータのオリジナルを持つものとした．各ピアはそれぞれ，1 タイムスロットごとに 0.1 の確率であるデータを要求する．データ要求の発行は Zipf 分布に従うものとし，データ番号が小さいデータほど要求が頻繁に発生するものとした．具体的には，j 番目のデータの要求確率を q_j とし，以下の式で与えた．

$$q_j = \frac{j^{-\alpha}}{\sum_{k=1}^{100} k^{-\alpha}} \quad (2)$$

ここで，α はデータの要求頻度の差を決定するパラメータであり，Zipf 係数と呼ばれる．シミュレーションでは，Zipf 係数 α は 0.5 とした．クエリの伝播にはフラッディングを用い，TTL は十分に大きな値とし，検索が必ず成功する環境とした．各ピアは，クエリに応答したピアのうち，検索ネットワーク上のホップ数が最も近いピアに対してアクセスするものとした．複製の配置方式には，パス複製法を用いた．パス複製法では，クエリを発行したピアからクエリに応答したピアまでの経路上にあるすべてのピアに複製を配置する．更新伝播木は 2 分木（$n = 2$）とした．

各データのサイズはすべて等しく，複製を保有可能な数はすべてのピアで 10 とした．各ピアが複製を作成する際にデータ記憶領域に空きがない場合は，所持していた複製の中で最も古い複製を削除し，新たな複製を作成するものとした．また，オリジナルデータは削除しないものとした．

以上のような環境において，10,000 タイムスロットのシミュレーション実験を行った．以下では，データ番号 100 のデータに注目して，このデータの更新伝播木に関する評価結果を述べる．

5.2 他の複製更新伝播法との比較

まず，不当な退出が発生しない環境を想定し，直線伝播法，放射伝播法，文献 10) のチェイン構造（$m = 3$），および提案手法（$k = 1, 4$）について，更新伝播時の遅延および平均負荷の比較を行う．

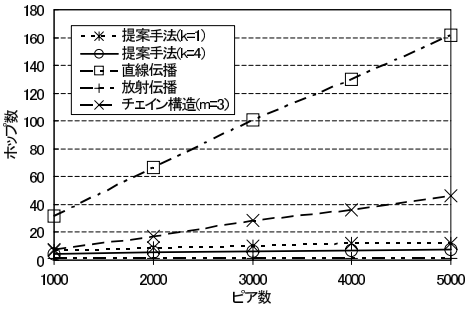

図 5 平均遅延
Fig. 5 Average delay.

5.2.1 ピア数に対する遅延の変化

オリジナルノードから各ピアへ更新が伝播される際に要した論理ホップ数の平均（平均遅延）を**図 5** に示す．放射伝播法を用いた場合，オリジナルノードが，複製を持つすべてのピアに更新を伝播するため，平均遅延はつねに 1 となっている．直線伝播法を用いた場合，1 つのピアにしか更新を伝播しないため，ピア数の増加に従って平均遅延も線形的に増加している．チェイン構造を用いた更新伝播法の場合，ほとんどのピアが m 個のピアに更新を伝播させるため，直線伝播法と比較すると平均遅延は小さく抑えられる．しかし，ピア数の増加に従って，平均遅延が線形的に増加してしまう．一方，提案手法では，平均遅延が対数オーダに抑えられている．また，$k = 1$ よりも，$k = 4$ の場合の方が平均遅延が小さくなっている．これは，新規ピアが更新伝播木に参加する場合，k 個上位の先祖ノードの子の数が n 未満のノードを発見するため，k の値が大きくなるほど，新規ピアが木の上部に参加でき，その結果，木の偏りが小さくなり，より高さの低い完全 n 分木に近くなるためであると考えられる．このことを検証するため，同様の環境において，提案手法における k の値を変化させた場合の遅延への影響を調べた．$n = 2$ とし，k を変化させた場合の平均遅延，最大遅延をそれぞれ**図 6**，**図 7** に示す．これらの図では，完全 2 分木における平均遅延および最大遅延についても，検証のために示している．図 6 より，k の値が大きくなるにつれて，平均遅延が小さくなり，根から各節点までの平均ホップ数が小さくなることが分かる．また図 7 より，k の値が大きくなるにつれて，最大遅延が小さくなり，根からの最大ホップ数，つまり木の高さが低くなることが分かる．以上の結果より，k の値が大きくなるほど木の偏りが小さくなり，完全 n 分木に近くなることが確認できる．

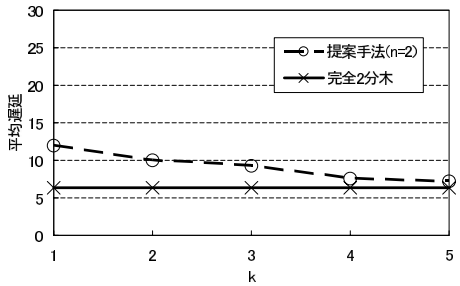

図 6 k と平均遅延（提案手法）
Fig. 6 k and average delay (proposed strategy).

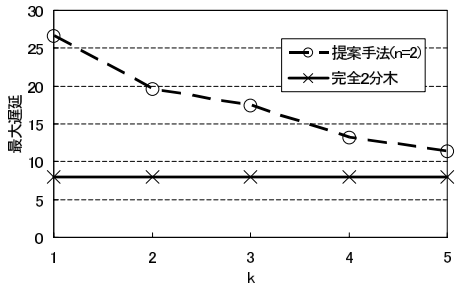

図 7 k と最大遅延（提案手法）
Fig. 7 k and max delay (proposed strategy).

5.2.2 ピア数に対する負荷の変化

各ピアが次に更新を伝播したピア数の平均値（平均負荷）を**図 8** に示す．ここで，放射伝播法では，オリジナルノードが，複製を持つすべてのピアに更新を伝播するため，ピア数の増加に比例して平均負荷が増加し，他の更新伝播法と比べて大幅に性能が悪くなる．そのため，図 8 では放射伝播法による結果は省略している．直線伝播法を用いた場合，各ピアは複製を持つ 1 つのピアに更新を伝播するため，平均負荷はつねに 1 となっている．チェイン構造を用いた更新伝播法の場合，オリジナルノードのみ左右両方向（最大 6 個）のピアに更新を伝播するが，それ以外のピアは，チェイン上の 1 方向のピアにのみ更新を伝播させればよく，最大でも 3 である．そのため，ピア数が増加しても，平均負荷は 3 程度に抑えられる．提案手法では，更新を子にのみ伝播するため，1 つのピアが更新を伝播させるべきピア数は最大でも n（$= 2$）である．そのため，ピア数が増加しても，平均負荷は n（$= 2$）以下になる．$k = 1$ と $k = 4$ ではほとんど差が現れず，k の値による平均負荷の差は小さいといえる．

以上の実験結果から，直線伝播法では平均遅延が，放射伝播法では平均負荷がきわめて大きくなってしまう．チェイン構造を用いた更新伝播法では，これら 2 つの手法と比較して負荷分散と遅延減少が実現できているが，ピア数の増加に従って，平均遅延が線形的に

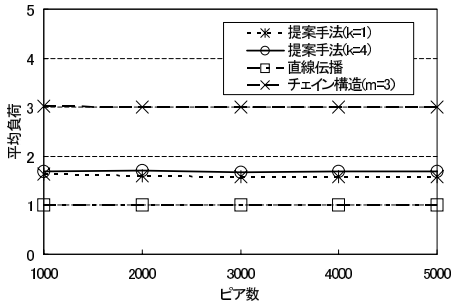

図 8 平均負荷
Fig. 8 Average load.

大きくなってしまう．m の値を大きくすることにより，平均遅延の増加量を減らすことができるが，平均負荷が高まるため，負荷分散と遅延減少の両立は難しい．一方，提案手法では，平均負荷を n 以下に保ちながら平均遅延を対数オーダに抑えることができ，負荷分散と遅延減少を両立させることができている．また，k の値を大きくすることにより更新伝播木を完全 n 分木に近づけることができ，平均遅延を小さく抑えられる．

5.3 不当退出発生時の木構造維持コストの変化

次に，不当な退出が発生する環境において，提案手法における k の値を変化させた場合の，木構造維持に必要なメッセージ数を評価した．

ネットワーク上のピアは，一定の確率で不当に退出するものとした．更新伝播木に参加する各ピアは，確認情報を 1 タイムスロットごとに送信し，不当な退出を検出した場合は，即座に木の修復を行うものとした．不当な退出によって木の分断が修復できない場合，検索ネットワークにクエリをフラッディングし，そのデータのオリジナルまたは複製を持つピアを発見するものとした．このとき，発見したピアに対して参加要求を送り，分断した部分木を更新伝播木へ復帰させる．

提案手法では，k 個以上の連続する先祖ノードの不当な退出が発生した場合にフラッディング操作が必要であり，多くのメッセージが発行される．それに対し，$k - 1$ 個までの連続する先祖ノードの不当な退出であれば，管理しているノード情報を用いて，少ないメッセージ数で木の修復を行うことができる．そのため，k の値を大きくすることで，木に再参加させるために必要なメッセージ数を減少させることができる．一方，k の値が大きくなるにつれて，管理するノード情報が増加するため，複製削除によるピアの脱退などによって更新伝播木の構成が変化したときに，各ピアの情報を更新するために必要なメッセージ数が増加する．

そこで，総ピア数を 5,000 とし，不当退出発生確率

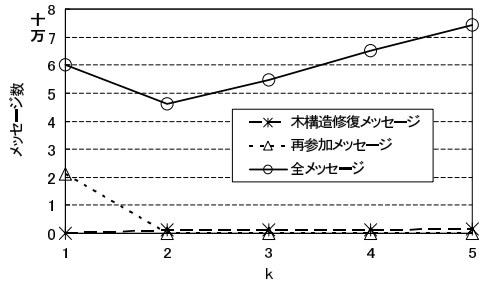

図 9 k とメッセージ数（不当退出確率 0.001）
Fig. 9 k and number of messages (failure rate: 0.001).

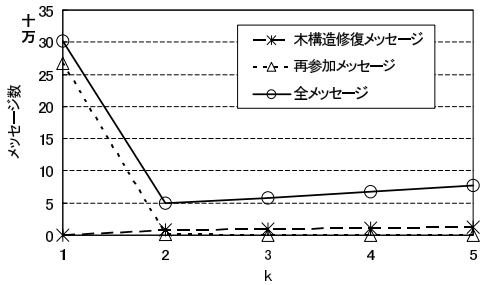

図 10 k とメッセージ数（不当退出確率 0.01）
Fig. 10 k and number of messages (failure rate: 0.01).

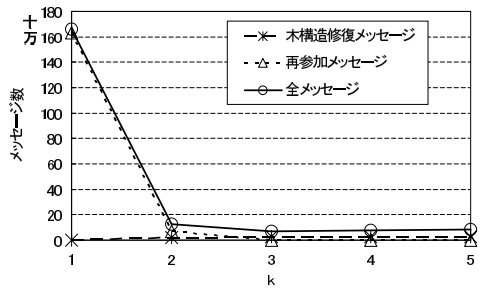

図 11 k とメッセージ数（不当退出確率 0.1）
Fig. 11 k and number of messages (failure rate: 0.1).

を 0.001，0.01，0.1 としたときの，メッセージ数が最小となる k の値を検証した．実験結果を，それぞれ**図 9**，**図 10**，**図 11** に示す．図中の木構造修復メッセージとは，ピアの不当な退出が発生した際に，各ピアが管理する情報を用いて局所的に更新伝播木を修復するために必要なメッセージ数，再参加メッセージとは，各ピアが管理するピアの情報だけでは更新伝播木の修復が不可能な場合に，フラッディングによって更新伝播木に再参加するために必要なメッセージ数を表す．全メッセージは，木構造修復メッセージと再参加メッセージに加え，新規ピア参加時や，複製削除などの正当な手続きをふんだピアの退出時にかかるメッセージ数を含めた，木構造維持に必要な全メッセージ数を表す．

すべての不当退出確率において，木構造修復メッセージは，k の値が増加するにつれて多くなる．これは，修復処理によって，ピアの更新伝播木上の位置が変更された場合に，より多くのピアの先祖ノードの情報を更新する必要があるためである．一方，再参加メッセージは，$k = 1$ の場合には，不当な退出による木の分断が修復されないため，多くのピアがフラッディングによる更新伝播木への再参加の操作を行わなければならず，メッセージ数がきわめて多くなっている．$k = 2$ 以上であれば，木を修復できる可能性が高くなり，フラッディング操作を行う回数が少なくなるため，メッセージ数も少なく抑えられる．その中でも，ピアの不当な退出の発生確率が低い環境では木の分断が発生しにくく，$k = 2$ でも高確率で木の修復を行うことができる．逆に，不当な退出の発生確率が 0.1 のように高い環境では，k の値を大きくする必要があり，図 11 では，$k = 3$ のときにメッセージ数が最小となっている．

全メッセージに関しては，図 9 のように不当退出発生確率が低い環境では，k の値を大きくした場合，局所的な木の修復によるメッセージ数の減少量よりも，先祖ノード情報の更新によるメッセージ数の増加量が大きくなる．したがって，k が 4 以上であれば，$k = 1$ のときよりも多くのメッセージ交換が必要になってしまう．逆に，図 10，11 のように不当退出発生確率が高い環境では，木の分断が発生しやすいため，k の値を大きくとることが有効である．図 9，10 では $k = 2$ が，図 11 では $k = 3$ のときにメッセージ数が最小となっている．全体としては，ピアの不当な退出を即座に検出できる環境においては，$k = 2$ であれば十分であるといえる．

5.4 n の影響

n を変化させた場合の，提案手法の平均遅延と最大遅延，平均負荷，木構造維持に必要な全メッセージ数に対する影響を調べた．総ピア数を 5,000 とし，不当退出発生確率を 0.01 に統一した場合の結果をそれぞれ**図 12**，**図 13**，**図 14**，**図 15** に示す．

図 12 および図 13 より，すべての n の値に対して，k の値が増加するにつれて平均遅延，最大遅延ともに小さくなるという同様の特徴を確認できる．また，n の値が大きい場合ほど，遅延が小さく抑えられている．これは，n の値が大きいほど各ピアが多くの子ノードを持つことができるため，木の高さを低く抑えられるからである．

一方，n の値を大きくすることにより，各ピアが一度に更新を伝播するピア数が増加する問題がある．

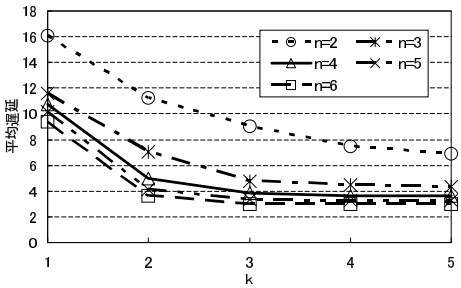

図 12 n と平均遅延
Fig. 12 n and average delay.

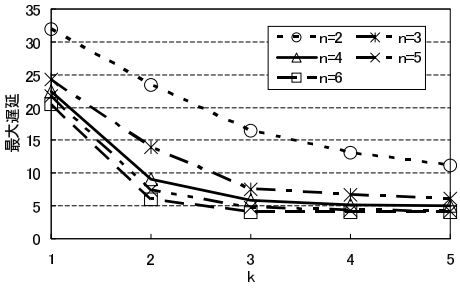

図 13 n と最大遅延
Fig. 13 n and max delay.

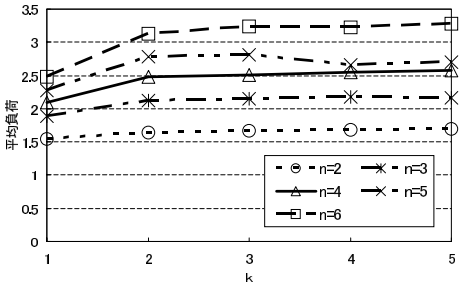

図 14 n と平均負荷
Fig. 14 n and average load.

図 14 より，n の値が増加するにつれて，更新伝播時の平均負荷が増加することが確かめられる．

図 15 の結果から，すべての n の値において，$k = 1$ のときにメッセージ数が最も多く，$k = 2$ 以上の場合はメッセージ数を少なく抑えられるという，図 10 と同様の特性を示すことが確かめられる．また，n の値が大きいほど，木構造維持に必要なメッセージ数が少なくなっている．これは，n の値が大きくなるほど，木の高さが低くなることに起因する．提案手法では，ピアの参加や脱退などによりピアの位置情報が更新された場合，新しく情報を変更されたピアから k 個（葉までの下位ノード数が k 個未満の場合は葉まで）の下位ノードにまで，新たな位置情報を伝える必要がある．このとき，木の高さが低い場合は，あるピアに対して k 個下位までのノードが存在しない可能性が高いため，伝播するメッセージ数が少なくなる．このような理由から，n の値が大きく，木の高さが低いほど，伝播するメッセージ数が減少する．

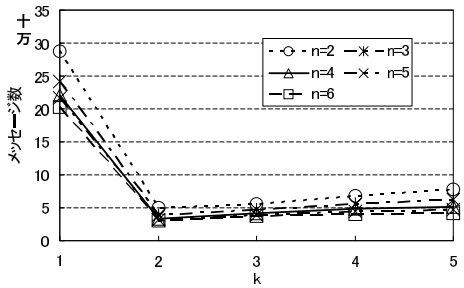

図 15 n とメッセージ数
Fig. 15 n and number of messages.

以上の結果より，n の値を大きくすることにより，更新伝播時にかかる遅延が小さくなるだけでなく，木構造維持に必要なメッセージ数も小さく抑えられる．しかし，その一方で，更新伝播時の平均負荷が大きくなる．したがって，各ピアのデータの送信速度や処理能力，データサイズなどのシステム環境を考慮して，適切な n の値を設定する必要がある．

6. おわりに

本論文では，P2P モデルを用いたデータ共有サービスにおいて，ピアが共有するデータ（複製）に更新が発生する環境を想定し，木構造に基づく複製更新伝播法を提案した．提案手法では，各ピアが n 分木の論理ネットワーク（更新伝播木）を形成し，この木に沿って更新を伝播させることにより，更新伝播時の遅延減少と負荷分散を実現する．また，各ピアが更新伝播木上の k 個の先祖ノードの情報を管理することにより，ネットワーク障害などによって，ピアが正当な手続きをふまずに P2P ネットワークから退出した場合でも，自律分散的に更新伝播木を修復する．

提案手法の性能を評価するため，シミュレーション実験により，他の更新伝播法との比較を行った．その結果から提案方式は，ピア数の増加に対して平均負荷を定数オーダに抑えると同時に，平均遅延を対数オーダに抑えることを確認した．また，k の値を大きくすることにより，より平均遅延が抑えられることを確認した．さらに，提案手法において，k の値を変化させることによるメッセージ数の変化を調べた．その結果，木構造維持に必要なメッセージ数を最小にする k の値を検証し，ピアが P2P ネットワークから退出する

確率が高くなると，最適な k の値が増加することを確認した．最後に，n の値の変化による影響を調べた．その結果，n の値を大きくすることにより，更新伝播時の遅延や木構造維持に必要なメッセージ数が小さく抑えられるが，逆に更新伝播時の平均負荷が高くなることを確認した．

本論文では，頻繁に親ノードの生存を確認することで，即座にピアの退出を検出し，更新伝播木を修復する環境を想定していた．しかし，アプリケーションによっては，更新発生間隔が大きいデータもあるため，つねに木の修復を行う必要性が低い場合も考えられる．そこで今後は，データの更新頻度を考慮し，適切な間隔で木の修復を行う方法を検討する予定である．

謝辞 本研究の一部は，文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」，科学技術振興調整費「先進融合領域イノベーション創出拠点の形成：ゆらぎプロジェクト」，および科学研究費萌芽研究（17650029），特定領域研究（18049050）の研究助成によるものである．ここに記して謝意を表す．

参 考 文 献

1) Admic, L.A., Lukose, R.M., Puniyani, A.R. and Huberman, B.A.: Search in power-law networks, *Physical Review E.*, Vol.64, No.4, 046135 (2001).
2) Balakrishnan, H., Kaashoek, M.F., Karger, D., Morris, R. and Stoica, I.: Looking up data in P2P systems, *Comm. ACM*, Vol.46, No.2, pp.43–48 (2003).
3) Cohen, E. and Shenker, S.: Replication strategies in unstructured peer-to-peer networks, *Proc. SIGCOMM'02*, pp.177–190 (2002).
4) Datta, A., Hauswirth, M. and Aberer, K.: Updates in highly unreliable, replicated peer-to-peer systems, *Proc. ICDCS'03*, pp.76–85 (2003).
5) Gnutella. URL: http://www.gnutella.com
6) Lv, Q., Cao, P., Cohen, E., Li, K. and Shenker, S.: Search and replication in unstructured peer-to-peer networks, *Proc. ICS'02*, pp.84–95 (2002).
7) 中通 実，内田 渉，原 隆浩，前田和彦，西尾章治郎：Peer-to-Peer ネットワークにおける木構造を用いた複製更新の伝搬について，電子情報通信学会データ工学ワークショップ（DEWS 2004）論文集 (2004).
8) Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S.: A scalable content-addressable network, *Proc. SIGCOMM'01*, pp.161–171 (2001).
9) Stoica, I., Morris, R., Karger, D., Kaashoek, M.F. and Balakrishnan, H.: Chord: A scalable peet-to-peer lookup service for internet applications, *Proc. SIGCOMM'01*, pp.149–160 (2001).
10) Wang, Z., Das, S.K., Kumar, M. and Shen, H.: Update propagation through replica chain in decentralized and unstructured P2P systems, *Proc. P2P'04*, pp.64–71 (2004).
11) 山田太造，相原健郎，高須淳宏，安達 淳：Peer-to-Peer システム上での効率的なデータ配置による問合せ処理とロードバランスへの寄与，情報処理学会論文誌：データベース，Vol.45, No.SIG5(TOD22), pp.93–101 (2004).

(平成 18 年 5 月 2 日受付)
(平成 18 年 11 月 2 日採録)

渡辺 俊貴

2006 年大阪大学工学部電子情報エネルギー工学科卒業．現在，同大学大学院情報科学研究科博士前期課程在学中．P2P ネットワークにおける複製管理に興味を持つ．日本データベース学会の学生会員．

原 隆浩（正会員）

1995 年大阪大学工学部情報システム工学科卒業．1997 年同大学大学院工学研究科博士前期課程修了．同年同大学院工学研究科博士後期課程中退後，同大学院工学研究科情報システム工学専攻助手，2002 年同大学院情報科学研究科マルチメディア工学専攻助手，2004 年より同大学院情報科学研究科マルチメディア工学専攻助教授となり，現在に至る．工学博士．1996 年本学会山下記念研究賞受賞．2000 年電気通信普及財団テレコムシステム技術賞受賞．データベースシステム，分散処理に興味を持つ．IEEE，ACM，電子情報通信学会，日本データベース学会の各会員．

木戸　裕樹

2004 年大阪大学工学部電子情報エネルギー工学科卒業．2006 年同大学大学院情報科学研究科博士前期課程修了．現在，株式会社コーエー所属．

中通　　実

2004 年大阪大学工学部電子情報エネルギー工学科卒業．2006 年同大学大学院情報科学研究科博士前期課程修了．現在，トヨタ自動車株式会社所属．

西尾章治郎（フェロー）

1975 年京都大学工学部数理工学科卒業．1980 年同大学大学院工学研究科博士後期課程修了．工学博士．京都大学工学部助手，大阪大学基礎工学部および情報処理教育センター助教授，大阪大学大学院工学研究科情報システム工学専攻教授を経て，2002 年より大阪大学大学院情報科学研究科マルチメディア工学専攻教授となり，現在に至る．2000 年より大阪大学サイバーメディアセンター長，その後 2003 年より大阪大学大学院情報科学研究科長を併任．この間，カナダ・ウォータールー大学，ビクトリア大学客員．データベース，マルチメディアシステムの研究に従事．現在，Data & Knowledge Engineering 等の論文誌編集委員．本会理事を歴任．電子情報通信学会フェローを含め，ACM，IEEE 等 8 学会の会員．