

ネットワークアウェアな P2P型安否情報共有システムの提案

生出 拓馬^{1,a)} 武田 敦志^{2,b)} 高橋 晶子^{3,c)} 菅沼 拓夫^{4,1,d)}

受付日 2013年5月14日, 採録日 2013年10月9日

概要: 大規模災害時などにおける構成, 機能, 性能などが不安定なネットワーク環境上で, 可能な限り安定したネットワークサービスを提供するためには, 限られたネットワーク資源を節約するように各ノードが動作し, ノード自身やネットワークにかかる負荷を能動的に分散することが求められる. そこで我々は, ノード間の動的負荷分散を効率的に実現する P2P 型安否情報共有システムを提案する. 本システムでは, 我々がこれまで提案してきた構造化 P2P ネットワークである Waon (Well-distribution Algorithm for an Overlay Network) をネットワーク基盤技術として用いることで, ネットワークの再構築やネットワーク維持にかかるコストの増加をともなわずに動的負荷分散を実現する. 本論文では, P2P 型安否情報共有システムの設計・実装を通じて Waon の導入による効果を検証し, 大規模災害時におけるネットワークサービスへの Waon の適用可能性について議論する.

キーワード: P2P, オーバレイネットワーク, 安否情報共有システム, 災害時通信

Design of A Network-aware P2P Safety Confirmation System

TAKUMA OIDE^{1,a)} ATSUSHI TAKEDA^{2,b)} AKIKO TAKAHASHI^{3,c)} TAKUO SUGANUMA^{4,1,d)}

Received: May 14, 2013, Accepted: October 9, 2013

Abstract: There are many issues to be addressed when we provide network services which are unstable in their structure, functions and performance, during large-scale disasters. In order to provide stable network services under the unstable situation, computer nodes need to work autonomously to save network resources and to balance the loads of the nodes and networks. In this paper, we propose a P2P Safety Confirmation System based on our proposed structured P2P network called Well-distribution Algorithm for an Overlay Network (Waon). In this system, each node sufficiently achieves autonomous dynamic load balancing by using Waon framework, without incurring additional maintenance costs. We design and implement the system and evaluate the effectiveness of our proposal. Moreover we discuss about the feasibility of the proposed framework to build communication support systems in natural disaster.

Keywords: P2P, overlay network, safety confirmation system, communication support in natural disaster

¹ 東北大学大学院情報科学研究科
Graduate School of Information Sciences, Tohoku University,
Sendai, Miyagi 980-8579, Japan

² 東北学院大学
Tohoku Gakuin University, Sendai, Miyagi 981-3193, Japan

³ 仙台高等専門学校
Sendai National College of Technology, Sendai, Miyagi 989-3128, Japan

⁴ 東北大学サイバーサイエンスセンター
Cyberscience Center Tohoku University, Sendai, Miyagi 980-8578, Japan

a) oide@ci.isc.tohoku.ac.jp

b) takeda@cs.tohoku-gakuin.ac.jp

c) akiko@sendai-nct.ac.jp

d) suganuma@isc.tohoku.ac.jp

1. はじめに

東日本大震災では, 通信インフラの壊滅的な損壊と, 通信トラフィック集中回避のために実施された通信事業者の通信制限により, 被災者を含め, 利用者間での通信が著しく困難となった. そのような中で, Google Person Finderをはじめとした様々なクラウド型ネットワークサービスが提供され, 電子掲示板や twitter などの既存サービスでも

本研究の一部は, 第一著者が仙台高等専門学校に在学中に実施された.

積極的な災害関連情報の共有が行われた。しかしながら、インターネットへのアクセス自体が困難であった被災地では、これらのサービスの利用には制限があったことに加え、既存サービスを利用した情報共有では、大量の情報の中から被災者が自分の欲しい災害関連情報を効果的に得ることは困難であった [1]。

本論文では、災害時に特定の人の安否を確認する、安否情報共有システムに焦点を当てる。安否情報共有システムのアーキテクチャとしては、情報共有の効率化の観点から、クラウドのサーバ上に安否情報を配置してクライアントから参照するサーバ・クライアント型での構成が一般的である。しかしながら、被災地における情報利用に十分配慮しつつ、構成、機能、性能などが不安定なネットワーク環境上で、可能な限り安定した広範なサービスを提供する必要があることから、以下のような要件を満たすシステムとする必要がある。

- (P1) 被災を免れ、稼働可能なノードのみで安否情報の共有を実現する。
- (P2) 個人の安否情報に加え、避難者数や避難所の利用状況などの集計情報を容易に把握できる。
- (P3) 特定のノードに負荷を集中させず、被災状況とその変化に応じた動的な負荷分散を実現する。

これらの要件を満たすシステムとして、本論文では、P2P ネットワークに基づく、P2P 型安否情報共有システムを提案する。

具体的には、我々が構造化 P2P ネットワークの 1 つとして提案してきた、Well-distribution Algorithm for an Overlay Network (Waon) [2] を用いる。構造化 P2P ネットワークは、主に高い通信効率や検索性能が要求される際に用いられるが、動的負荷分散実現のためにネットワーク構造が静的・固定的に決定される場合が多いため、機能の多様性を実現することが困難であった。具体的には、構造化 P2P ネットワークにおいては、ノード ID やコンテンツ ID によらないハッシュ値などによるノード、コンテンツの配置を行うことで負荷分散を実現するため、本質的にノードの物理的な位置情報やコンテンツの意味情報などといった固有情報が失われる。したがって、それらの情報を利用することで実現されるアプリケーションレベルの様々な機能、たとえば地理範囲検索やコンテンツ範囲検索などの機能を提供することが困難であった。これに対し Waon では、ネットワーク構造に制限を加えることなく、ネットワーク環境の変動やノード間の性能差に応じて、ノードが自律的に動的負荷分散を行う。この手法では、ノードはネットワークを再構築せずに動的負荷分散を実行するため、ネットワークの再構築に必要なコストを削減する。さらに、ノード間の物理的な近傍性を考慮したネットワークを構築するため、実際の伝送距離が短縮でき、不安定なネットワーク環境上においてネットワーク資源を効果的に

利用可能となる。

以上から、本研究では、Waon をネットワーク基盤技術として用いることで、上記の要件を満たし、ネットワークの再構築やネットワーク維持にかかるコストの増加をとまわずに動的負荷分散を行う、ネットワークアウェアな P2P 型安否情報確認システムを実現する。本システムは、動作可能なノードのみでアプリケーションレベルでのオーバーレイネットワークを構成し、そのうえで安否情報を共有する。安否情報としては、避難者の氏名、年齢、避難場所などの情報を含み、利用者は複数条件により安否情報の検索ができる。各ノードが保持する安否情報は、ネットワーク環境やノード間の性能差に応じて動的に伝送され、自律的な負荷分散を実現する。

本論文では、Waon に基づく P2P 型安否情報確認システムの設計と実装、および初期評価実験について述べる。具体的には、P2P 型安否情報確認システムへの Waon の適用に関する詳細設計を行った。また、本システムの詳細設計に基づきシミュレーション実験のためのシステム実装を行った。さらに、シミュレーション実験を通して得られた結果から、他の構造化 P2P ネットワークアルゴリズムに比べ、動的負荷分散が効果的に行えることが検証でき、本システムにおける Waon の適用による効果を示した。

本論文の構成は以下のとおりである。2 章では、既存研究および Waon の概要について述べる。3 章では、Waon を適用した安否情報確認システムの設計について述べる。4 章では、実装したシミュレーションシステムによる実験結果から、安否情報確認システムにおける Waon の効果について述べる。5 章でまとめる。

2. 関連研究

2.1 安否情報共有システム

大規模災害時における運用を想定した情報共有システムの研究が進められている。文献 [3] では、スマートフォンなどの携帯端末間のすれ違い通信を用いることで災害情報を通信可能エリアまで効率的に伝送し、Google Person Finder や Twitter などの公衆サービスへ登録するシステムが提案されている。このシステムでは、医者やボランティアなどが自身の携帯端末を持って移動し災害情報を伝搬させることで、インターネットアクセスの断絶した地域の情報を収集する機能を実現している。

しかし、公衆サービスやインターネットを利用している限り、こうして収集された災害情報に対して、それらを真に必要な被災地の人々がアクセスすることは困難である。また、一般的な SNS では、目的の災害関連情報を的確に参照することが難しい点や、災害情報の中に安否情報が含まれる場合は個人のプライバシーを考慮する必要があることなど、課題が多い。

一方、被災地での通信インフラに関しては、機器の障害

により構成、機能、性能などが著しく変動し、利用者間のコミュニケーションに多大な影響を与えた。これに対し、スマートフォンや無線機器を用いたネットワーク構築技術 [4], [5], 劣悪なネットワーク環境下における通信技術 [6], DTN の対災害応用, メッシュ型無線ネットワーク構築基盤技術などの研究開発が進められている。これらの技術の進展により、地域のローカルネットワークが利用可能な状況となっていることを前提とした、サービス構成・提供技術の高度化が検討可能となっている。

以上から、本論文における安否情報共有システムの要件として以下の3点を定義する。

- (P1) 被災を免れ、稼働可能なノードのみで安否情報の共有を実現する：**インターネットが利用できない状況において、被災を免れた稼働可能な残存機器のみによって、地域のローカルネットワークを利用して災害対応用のオーバーレイネットワークを即時的に構築する。そのうえで必要最低限の災害関連情報のみを被災者間で共有する仕組みを提供する必要がある。
- (P2) 個人の安否情報に加え、避難者数や避難所の利用状況などの集計情報を容易に把握できる：**(P1)を充足することによりオーバーレイネットワーク上のノードに安否情報が分散配置され、それによる副作用として各種情報の集計が困難となることが予想される。したがって、安否情報が分散する場合でも、特定個人の安否情報の検索のほか、避難者総数、特定条件に適合する避難者の数、避難所の利用状況など、集計情報を容易に把握することが可能な仕組みを提供する必要がある。
- (P3) 特定のノードに負荷を集中させず、被災状況とその変化に応じた動的な負荷分散を実現する：**文献 [7], [8]では、あらかじめ構築したネットワーク上における災害情報共有システムが提案されている。これらのシステムでは、サーバの役割を担うネットワーク機器が情報を集約して配布することで利用者間での災害情報の共有を実現している。しかしながら、計算機・ネットワーク環境が急激に変化する大規模災害時には、電力供給の遮断などによるサーバの突然の停止、サーバアクセスのためのネットワークの切断や負荷集中などにより、サービス利用が不可能となる場合がある。これは、災害時に残存した機器を用いてサーバを臨時に構築する場合でも、被災地の不安定なインフラ状況から同様のことがいえる。したがって、特定ノードに負荷を集中させず、被災エリアの違いによる被害の偏りや被災後の時間的な経過による状況変化に柔軟に対応して、動的に負荷を分散させる仕組みを提供する必要がある。

2.2 P2P ネットワーク

サーバを必要とせずノード間で動的に情報を持ちあう

P2P ネットワークは、拡張性や耐故障性に優れ、大規模災害時のような不安定なネットワーク環境における情報共有環境の構築に適している。特に、DHT (Distributed Hash Table) は構造化 P2P において最も有名なアルゴリズムであり、これまでに Chord [9], CAN [10], Pastry [11], Tapestry [12] など、DHT に基づく様々な構造化 P2P が提案されている。DHT では高い拡張性とネットワーク上のコンテンツへの確実な到達可能性が保証されており、DHT に基づくネットワークを利用したシステムを構築することで 2.1 節で述べた要件 (P1) を充足することができる。しかし、DHT はそのネットワーク構造が静的・固定的に決定されるために、環境の変化に応じた動的な負荷分散や完全一致検索以外の検索機能の実現が困難であり、要件 (P2), (P3) の充足は困難である。

Mercury [13] は DHT を用いずにネットワークを構築することで動的な負荷分散を実現する構造化 P2P の 1 つである。Mercury では、負荷密度の低いノードを 1 度離脱させ、負荷密度の高いノードの隣に再加入させる leave-rejoin 法を用いることで、特定の端末へ負荷を集中させず、周囲の環境に応じた動的な負荷分散が可能である。また、DHT と比較してネットワーク構造への制限が少ないことから、コンテンツの属性ごとに柔軟な範囲検索が実現可能である。そのため、Mercury に基づくネットワークを利用したシステムを構築することで要件 (P1), (P2), (P3) のすべてを充足することができる。しかし、負荷分散を実行するたびにノードがネットワーク上で離脱と加入を繰り返すため、そのたびにネットワークの再構築が生じ、ネットワーク維持のための通信コストが増加するという問題がある。この問題は災害時において非常に限定されたネットワーク資源を多分に消費するだけでなく、ネットワーク分断のリスクを向上させる一因となりうる。

2.3 Waon

Waon は、我々がこれまで提案してきた構造化 P2P ネットワークである。Waon では、Chord と同様にリング状の ID 空間を用いる。Waon におけるすべてのノードとコンテンツはこの ID 空間上に配置され、その位置関係に基づいたリング状のオーバーレイネットワークを構築する。また、コンテンツのノードへの割当ても ID 空間上の位置関係に基づいて決定される。具体的には、それぞれのノードは、自身のノード ID から順方向に最近隣のノード ID までの ID 空間を自身の管理領域とし、その領域に含まれる ID を持つコンテンツを管理する。Waon も Mercury と同様に DHT と比較してネットワーク構造への制限が少なく、ノード ID の変更によってリングネットワーク上のノードの位置を自由に変更可能である。そのため、環境の変化に応じて自身の管理領域を増減させることで動的な負荷分散を実現する。また、コンテンツ ID の割当ても自由に行う

ことが可能であり、同一避難所の安否情報をまとめてリングネットワーク上に配置することで避難所の利用状況の集計などへ応用が可能である。

Waon の特長は以下のとおりである。

- (C1) 拡張性に優れた構造化 P2P である：Waon におけるノードやコンテンツの検索にかかる処理数は $O(\log N)$ であり、これは Chord # [14] で用いられているノード間のホップ数に基づくルーティングテーブルを用いることによって実現する。
 - (C2) ネットワークの再構築などにもなう通信コストの増加を防ぐ動的負荷分散を実現する：動的負荷分散ではノードにかかる負荷に応じて ID 空間上のノード位置を変更し、管理コンテンツ数と通信コストの分散を実現する。
 - (C3) 範囲検索などの柔軟な検索機能を実現する：コンテンツの ID 空間上の位置に制限は設けないため、コンテンツ名の順序関係を保存した状態でネットワーク上に配置することができる。そのため、コンテンツの範囲検索だけでなく、複数条件による検索ができる。
 - (C4) 物理ネットワークのトラフィックを減少させることができる：Waon ではノードの ID 空間上の位置に制限は設けないため、ノードの物理的な近傍性を考慮したネットワークを構築できる。また、そのネットワークは動的負荷分散の実行で再構築されることはない。
- これらの特長のうち、特長 (C1) によって要件 (P1) を、(C3) によって (P2) を、(C2) によって (P3) を、それぞれ充足する。加えて、(C2) と (C4) によって、Mercury の問題であった通信コストの増加を防ぎ、不安定なネットワーク環境下において限られた資源の有効な利用が期待でき、(P3) をより効果的に解決することが可能である。

3. P2P 型安否情報共有システムの設計

3.1 概要

P2P 型安否情報共有システムは大規模災害時の利用を想定した情報共有システムである。1 章で述べたシステムに対する要件を満たすために、ネットワーク基盤技術として Waon を用いる。本システムの主な利用者は、被災地における地方自治体を想定し、自治体間の通信基盤はあらかじめ利用可能であることを前提とする。すなわち、災害発生後にはクラウド上などのサーバを利用せず、自治体間で、既知の残存機器により、アプリケーションレベルでのオーバーレイネットワークを構成する。こうすることで、信頼性の高い安否情報のみを、被災者のプライバシーを考慮したうえで共有する。

以降の節では、P2P 型安否情報共有システムへ Waon を適用する際の、詳細設計について述べる。

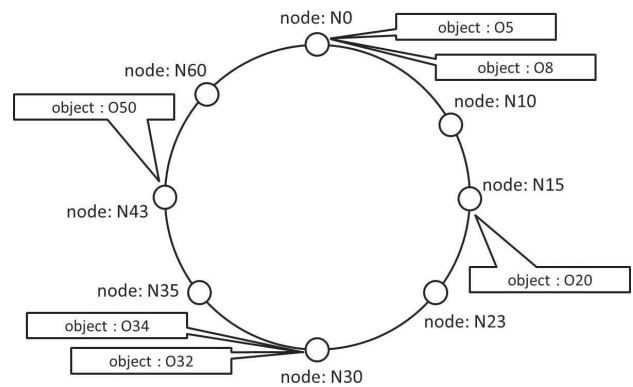


図 1 ネットワーク構造

Fig. 1 Network architecture.

3.2 ネットワーク構造

Waon では、被害を免れた機器のみで図 1 に示すリング状のオーバーレイネットワークを構築する。P2P ネットワークの特徴を生かし、突然のネットワーク断絶や、バッテリー電源の枯渇によってネットワークを構築している機器がネットワーク上から離脱した場合も、残りの機器によって欠落した機器の機能を維持する。また、Waon ではリング状のネットワークを構築するためのルーティングテーブルと、効率的な通信を実現するためのルーティングテーブルの 2 種類の情報のみを用いて様々な機能を実現する。関連研究における他の構造化 P2P の多くは、ネットワーク環境の変化やノードの性能差に応じて動的負荷分散を実行する際、これらのルーティングテーブルの再更新や、通信で用いるルーティングテーブル以外の情報の管理を行う必要がある。しかし Waon では、動的負荷分散のためにルーティングテーブル以外の情報の交換、ルーティングテーブルの更新を必要とせず、必要最低限の通信のみでネットワークサービスの機能を実現する。

Waon におけるノードは以下の要素によって構成される。

```

node           := < id, successor, predecessor,
                    route, object >
node.id        := INTEGER
node.successor := {node0, node1, ..., noder-1}
node.predecessor := {node0, node1, ..., noder-1}
node.route     := {node0, node1, ..., nodelog N-1}
node.object    := {object0, object1, ...}
    
```

ここで、 r はリングネットワークを構築するルーティングテーブルの経路長、 N はネットワーク全体のノード数である。通常、 r は 2 以上 $\log N$ 以下の値をとる。

オーバーレイネットワーク上での各ノードの位置は、それに割り当てられる一意の id ($0 \leq id < 2^M$) によって決まり、負荷分散の実行時には動的に変更ができる。各ノードは、自身の *successor* ノードとして、自身より順方向側に位置する r 台のノードとのリンクを、また、自身の *predecessor* ノードとして、自身より逆方向側に位置する

r 台のノードとのリンクを、それぞれ保持する。これらのリンクはリングネットワークを維持するために用いられる近距離用リンクであり、他のノードのネットワークへの参加や離脱を検出するために定期的に更新される。また、近距離用リンクとは別に、各ノードは *route* として、ノード間のホップ数に基づいたルーティングテーブルを構築する。このルーティングテーブルは、自身から順方向側に 2^n ($n = 0, 1, 2, \dots, \log N - 1$) だけ離れたノードとのリンクを保持する。このルーティングテーブルは $O(\log N)$ の通信効率を実現する遠距離用リンクであり、近距離用リンクと同様に定期的に更新される。*object* はノードが管理しているコンテンツである。

一例として、図 1 における $N30$ の要素を示す。なお、 $r = 2$ とする。

```

N30.id           := 30
N30.successor    := {N35, N43}
N30.predecessor  := {N23, N15}
N30.route        := {N35, N43, N0}
N30.object       := {O32, O34}

```

3.3 ノードの動作

3.3.1 ノードの参加

ノードがネットワークに参加する際は、あらかじめネットワークに参加しているノード（紹介ノード）と通信が行える必要がある。そのため、本システムではあらかじめシステムを利用する地方自治体どうしのノードの IP アドレスは既知であるものとする。ノードの参加時には紹介ノードと近隣ノード情報を交換し、割り当てられたノード ID に従って、自身の近隣ノード情報が得られるまで問合せを再帰的に実行して近距離用リンクの更新を行う。この期間中はリングネットワーク上に正しく配置されておらず自身の管理領域が決定していないため、参加ノードは他ノードから送られてきた安否情報に関するリクエストをすべて破棄する。近隣ノード情報が得られた参加ノードは、その時点から他ノードからの安否情報に関するリクエストに応じるものとし、順次自身の遠距離用リンクの更新を行う。なお、新たに登録されたノードはネットワークに新たに参加したノードだけではなく、分断されていたネットワークが統合されて復帰したノードであることも考えられる。この場合、3.5 節で述べる動的負荷分散にともなうノード ID の変更によって、統合時にノードのリングネットワーク上での並び順が不適切であることが考えられる。しかし、本システムではネットワークの再構築を極力行わないという観点から、それらのノードに対して ID を割り当て直すことはせず、正規の手段によって参加したノードに対してのみ ID の割当てを行うものとする。

3.3.2 リングネットワークの維持

ネットワークに参加するノードは、定期的に自身の近距

離用リンクを更新し、他ノードの参加や離脱の検出を行う。具体的には、自身の近隣ノードと近距離用リンクの情報を交換し、その中に新たなノード情報が含まれていた場合はネットワークに参加したノードとして経路表に追加する。また、応答のない近隣ノードを検出した場合はネットワークから離脱したノードとして経路表から削除する。この動作は、ノード単位で参加や離脱が行われた際はもちろん、ネットワークレベルで分断や統合が行われた際にも同様に実行される。

3.3.3 ノードの離脱

ノードがネットワーク上から離脱する際は、それまでに管理していたすべての安否情報を自身の直近の *predecessor* にすべて譲渡した後に離脱を行う。不安定なネットワーク環境によって突然ノードが離脱した場合は *predecessor* に安否情報を譲渡することなく離脱してしまうが、その際他ノードが 3.4.2 項で述べる安否情報の復旧を実行するため安否情報が失われることはない。

3.4 コンテンツの流通

3.4.1 ID 生成

ノードや安否情報のリングネットワーク上での位置は、それらに付与される ID によって決定する。Waon では、1 度構築されたネットワークはノードの参加や離脱がない限り再構築はされない。そのため、本システムにおけるノードの順序関係においても、頻繁に通信を行う必要のある近隣ノードとは物理的なネットワーク上でも近接していることが望ましい。近年、物理的な近傍性を考慮したネットワークの構築手法として、GPS を用いる手法やランダムマーク法 [15] を用いる手法などが提案されている。本システムでもそれらの手法を導入することで物理的な近傍性を考慮したリングネットワークの構築を行う。これにより、通信の物理的な通信路を短縮することができ、大規模災害時における限られたネットワーク資源を有効に使うことができる。また、安否情報の ID は、それらに含まれる避難者の氏名などの情報をもとに生成する。このとき、ハッシュ関数を用いずに、もとの情報を含んだ状態で ID を生成することで、ネットワーク上での安否情報の並び順が保存され、3.4.4 項で述べる安否情報の範囲検索が可能となる。

3.4.2 安否情報の追加

安否情報の構造を以下のように定義した。

```

object           := < id, name, age, place, detail >
object.id        := INTEGER
object.name      := STRING
object.age       := INTEGER
object.place     := STRING
object.detail    := STRING

```

ここで、*name*, *age*, *detail* はそれぞれ避難者の氏名、年

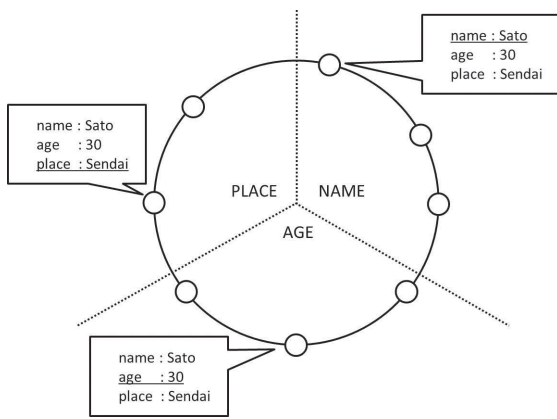


図 2 複製の生成

Fig. 2 Information replication.

年齢、詳細情報であり、*place*には避難所の情報を格納する。安否情報のIDは*name*, *age*, *place*をもとに複数生成され、それらは安否情報の複製を配置する際に用いられる。

ネットワーク上への安否情報の追加登録は、避難者の氏名、年齢、避難場所、詳細情報を指定することで行う。このとき、あらかじめ設定してある登録者情報が自動的に付加される。追加された安否情報は、与えられたIDに基づいて決定された1台のノードによって管理されるが、不安定なネットワーク環境下ではそのノードが突然ネットワークから遮断されてしまうことを考慮する必要がある。そのため、安否情報の追加の際には自動でその安否情報の複製を生成し、ネットワーク上の異なる場所へ配置する。具体的には、図2に示すように、リング状のID空間を3つに分割し、それぞれのID空間上で、避難者の氏名、年齢、避難場所のそれぞれの情報をもとに生成したIDを割り当てて追加を行う。こうすることで、ノードの突然の離脱によって安否情報が失われてしまう可能性は低くなる。また、本システムのノードIDの割当てには物理的な近傍性を考慮することができる。そのため、同一の情報をリングネットワーク上の離れた位置に分散配置させることによって、ネットワーク環境が不安定な被災地から追加された安否情報を、ネットワーク環境が比較的安定している被災地外のノードへ素早く拡散させる効果が期待できる。なお、ID空間の分割数を3としたのは、3.4.4項で述べる安否情報の範囲検索を、避難者の氏名、年齢、避難場所のそれぞれの属性ごとに実行できるようにするためである。それぞれのID空間では、各属性について前後関係が保存されて安否情報が配置される。そのため、目的の属性のID空間に向けて範囲検索を実行することで、属性ごとの範囲検索が可能となる。

また、IDの異なる3つの同一安否情報は相互にIDを生成することが可能なため、ノードの突然の離脱によって損失した安否情報は、その複製を保持するノードによって復旧が可能である。ただし、損失した安否情報の検出は困難

であるため、各ノードは定期的に自身の保持するすべての安否情報に対して複製の生成を行い、それにより重複した安否情報はネットワーク上から削除される。なお、同一の安否情報を持つ3台のノードがいっせいにネットワークから離脱した際にはその安否情報が失われてしまうことになるが、本システムでは少なくとも1台のノードは地理的に離れた場所のノードであり被害を免れることが期待できるため、情報損失の確率は低いと考えられる。

3.4.3 安否情報の削除

登録された安否情報の削除は、登録時に付加された登録者情報を入力することで行う。本システムでは1度離脱したノードが保持していた安否情報が再加入時に再びネットワーク上に拡散されることや、複製機能の実行などによって1度登録された安否情報が定期的に拡散されることから、安否情報を削除機能の実行によって完全にネットワーク上から消滅させることは困難である。これは、本システムが大規模災害時における安否情報の拡散を想定した際の、迅速な安否情報の共有を目的としているためである。ただし、東日本大震災の際に実際に用いられた公衆サービススペースの支援システムの多くと異なり、本システムでは、当事者のノードのみで構成されるプライベートなネットワーク上でのみ安否情報を扱う。そのため、安否情報が不特定多数の人々によって必要以上に長期間閲覧されることはなく、一定のレベルでのプライバシーの保護を実現する。

3.4.4 安否情報の検索

本システムでは、安否情報の範囲検索と、複数条件による検索を実現する。本システムにおける安否情報のIDは3.4.1項で述べた方法に基づいて生成されるため、ネットワーク上での安否情報の並び順が保存される。ここでの並び順とは、割り当てられたIDの順に安否情報を並べた際の前後関係のことであり、本システムでは類似する属性値を持つ安否情報どうしが隣り合って配置されることになる。そのため、*route*を用いて範囲検索の始点となるノードを検索し、*successor*を用いて範囲検索の終点となるノードまでをたどることで、範囲内に含まれるすべての安否情報を取得する。また、複数条件による検索では、1つの条件に対して範囲検索を実行し、得られた結果から残りの条件にも一致する安否情報のみを抽出することで実現する。これらの検索機能は、コンテンツIDの生成にハッシュ関数を用いる従来の構造化P2Pでは実現が困難である。WaonにおいてはコンテンツIDの割当てを自由に行うことができるため、範囲検索などの柔軟な機能を実現することが可能である。

図3にコンテンツの検索機能の擬似コードを示す。なお、ノード間の距離をそれぞれのIDを用いて以下に定義し、擬似コード中に含まれる関数は以下の動作を行う。

```

1: function SEARCHOBJECTS(n, fromId, toId)
2:   result = NULL;
3:   target = n.routing(fromId);
4:   repeat
5:     result = result + target.getObjects(fromId, toId);
6:     target = target.successor[0];
7:   until distance(toId, fromId) ≥ distance(target.Id, fromId)
8:   return result;
9: end function

```

図 3 コンテンツ検索の擬似コード

Fig. 3 Pseudo-code for searching objects on a node n .

$$distance(id_1, id_2) = \begin{cases} id_1 - id_2 & (id_1 > id_2) \\ id_1 - id_2 + 2^M & (id_1 \leq id_2) \end{cases}$$

$n.routing(id)$ ノード n の保持する $route$ に基づいて問合せを繰り返し、 id を管理領域とするノードを検索する。このノードの検索にもなうホップ数の増加は $O(\log N)$ となる (N はネットワーク全体のノード数)。

$n.getObjects(id_1, id_2)$ ノード n の管理する安否情報から、 id_1 から id_2 内の id を持つすべての安否情報を返す。

範囲検索の実現によって、本システムでは安否情報の効率的な検索だけではなく、避難所における避難者の受け入れ状況を知ることができる。また、管理している安否情報や検索して得られた安否情報から自動的に避難所の情報を取得してリスト化する。このリストから避難所を選択して範囲検索を実行することで、その避難所に避難している被災者の情報を一括して検索できる。この検索結果を参照することで、その避難所における避難者の人数や年齢分布が把握でき、避難者の受け入れ先の検討や支援物資の分配の際に役立てることが期待できる。

3.5 動的負荷分散

本システムでは、Wacon の動的負荷分散手法の導入により、特定のノードに負荷が集中することのない安定したネットワークを自動的に構築する。本システムにおける負荷値の定義は、管理している安否情報の数とする。その理由は、すべての安否情報はその利用法の性質上、ネットワーク上から等しく参照されると見なせること、および、安否情報 1 つあたりの容量は小さく個体差はないと判断できるためである。そのため、動的負荷分散が実行されネットワークが安定した状態では、すべてのノードがネットワーク全体の安否情報を等しく分配していると見なすことができ、ネットワークを構成しているノード数や、ネットワーク上に登録されている安否情報の総数の推定を行うことができる。

本システムにおける動的負荷分散の手順を以下に示す。

(1) それぞれのノードは、他のノードに対して、そのノード周辺にどれだけの負荷がかかっているかの問合せを

```

1: function UPDATEROUTE(n)
2:   node = n.successor[0];
3:   load = n.getLoad(node);
4:   distance = 0;
5:   for i=0 until distance < distance(node.id, n.id) do
6:     n.route[i] = node;
7:     distance = distance(node.id, n.id);
8:     node = n.route[i].route[i];
9:     load = load + n.route[i].getLoad(node);
10:    i = i + 1;
11:   end for
12:   averageLoad = load / 2^n.route.getLength();
13: end function

```

図 4 経路表更新の擬似コード

Fig. 4 Pseudo-code for updating the route of a node n .

```

1: function UPDATELOCATION(n, averageLoad)
2:   overLoad = n.getLoad(n.successor[0]) - averageLoad;
3:   if overLoad > 0.0 then
4:     num = n.shortenManagementArea(overLoad);
5:     for i=0 to num-1 do
6:       n.predecessor[0].addObject(n.object[i]);
7:       n.removeObject(n.object[i]);
8:     end for
9:   end if
10: end function

```

図 5 ノード位置変更の擬似コード

Fig. 5 Pseudo-code for updating the location of a node n .

行う。この問合せは、図 4 に示すように自身の経路表を更新する際に同時に行う。この更新は、ネットワーク上のノードが生存しているかを確認するために必ず必要となる更新であり、負荷分散の実行にあたって通信コストが増加することはない。

- (2) 問合せによって得られた負荷情報からネットワーク全体の平均負荷値を推測し、自身の負荷値のほうが上回れば、自身を過負荷ノードと見なす。
- (3) 過負荷ノードは図 5 に示すように自身の管理領域を減らすように自身の ID を変更してネットワーク上の位置を変更する。
- (4) 過負荷ノードはノード位置の変更によって管理する必要のなくなった安否情報を近隣ノードに対して譲渡する。

このうち、 $n.getLoad(node)$ 関数では、ノード n からノード $node$ の直前までのすべてのノードが管理している安否情報数の合計を返す。すなわち、 $n.getLoad(n.successor[0])$ では n の管理している安否情報数を返し、 $n.route[2].getLoad(n.route[3])$ では $n.route[2]$ から $successor$ 方向に 4 台分の安否情報数の和を返す。このとき、 $n.route[2]$ のノードは自身の経路表を更新する際に $n.route[3]$ のノード (すなわち $n.route[2].route[2])$ まで

のノードが管理している安否情報数の和をあらかじめ計算して保持しているため、安否情報数の和を計算するために他ノードへ問い合わせる必要はない。擬似コード中に含まれるその他の関数に関しては以下の動作を行う。

n.route.getLength() ノード *n* の保持する *route* の経路長を返す。すなわち、ネットワークが安定状態の場合は $\log N$ を返す (N はネットワーク全体のノード数)。

n.shortenManagementArea(overLoad) ノード *n* の管理する安否情報について、*overLoad* を超えない最大の安否情報が *n* の管理対象外となるよう *id* を変更する。その後、管理対象外となった安否情報数を返す。

n.addObject(object) ノード *n* の管理する安否情報として *object* を追加する。

n.removeObject(object) ノード *n* の管理する安否情報から *object* を削除する。

なお、図 4 における経路表更新時に応答のないノードを発見した場合は、そのノードよりも遠い位置のノードに関する情報を *route* から削除する。これは *route* がノード間の相対的な位置関係に基づいた経路表であるためであり、これにより一時的に遠距離のノードへのリンクは失われるが、リングネットワークが分断されない限りノードや安否情報への到達可能性は保証される。また、図 5 における安否情報の授受時に応答のないノードを発見した場合は、ただちにコンテンツの授受を中止する。このとき、ノード *n* は *predecessor[0]* への *add* リクエストに対する返信を参照し、正しく *add* が完了したことを確認したうえで *remove* を実行するため、ノードの突然の離脱による直接的な安否情報の損失以外にノードが安否情報を誤って破棄することはない。

図 6 に動的負荷分散の概要を示す。図 6 より、動的負荷分散が行われてもノードの並び順は変わらず、それによ

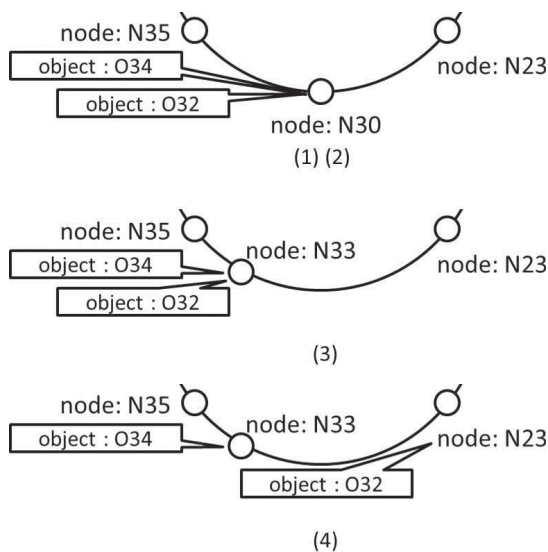


図 6 動的負荷分散の概要
Fig. 6 Overview of load balancing.

りそれぞれのノードの経路表を更新する必要がないことが分かる。そのため、本システムではネットワークの再構築を必要とせず負荷分散を実現し、大規模災害時における限られた資源の効率的な利用を実現する。また、本システムではネットワーク上の一部のエリアにノードが集中して配置され、安否情報の複製をネットワーク上に拡散しても 1 台のノードによって管理される場合があり、複製機能の効果が低くなることが考えられる。しかし、動的負荷分散の実行によって等しい数の安否情報を管理するようにノードの位置が変更されるため、複製は異なるノードによって管理されることになる。そのため、動的負荷分散の実行によって安定したネットワークを維持するだけでなく、システムの信頼性を向上させる。

4. 評価実験

4.1 シミュレーションシステムの実装

本システムの有効性を確認するために、Waon をネットワーク基盤技術として P2P 型安否確認システムのシミュレーションシステムの実装を行った。実装にはオーバーレイネットワーク構築ツールキットである Overlay Weaver[16] を用いたが、Overlay Weaver は主に DHT に基づく構造化 P2P ネットワークの構築に用いられるため、Waon や Mercury の実装のために適宜フレームワークの拡張を行った。

本実装では、3 章で述べた設計のうち、3.4.1 項に記載の「ノードの近傍性を考慮した ID 生成」については未実装となっている。

ノードや安否情報に与える ID 長は 40 バイト ($0 \leq id < 2^{320}$)、近距離用リンクの経路長 $r = 2$ とした。また、ノードに与える初期 ID は、ノードの物理的な近傍性を考慮した ID 生成の代替として IP アドレスから生成したハッシュ値に基づいて決定した。そのため、本実装では、ネットワークの物理的なトラフィックの軽減と、安否情報の複製を被災地外のノードへ転送する機能は未実装である。

4.2 安否情報の範囲検索に関する実験

本システムの範囲検索が正しく機能するかを確認するために、8 台のノードで構成されるネットワーク上に表 1 に示す安否情報を登録し、実験を行った。図 7 に安否情報配置後における、あるノードの管理している安否情報のリストを示す。本実験で共有する安否情報は 16 名分であるが、複製を含めて実際には延べ 48 名分の安否情報がアップロードされている。Waon の動的負荷分散の実行によりそれらは 8 台のノードによって平等に割り当てられ、図 7 に示すように 1 台あたりの管理安否情報数は 6 名分になる。なお、本論文での実験環境は想定する規模と比較してかなり小さいものとなっているが、個々の安否情報は非常に小さいデータ量であるため、ネットワーク規模が機能に与える影響は少ないと判断した。

表 1 実験で用いた安否情報
Table 1 Sample safety information.

Name	Age	Place	Detail
oide	20	sendai	SampleSafetyInformation
horikawa	20	yamadera	SampleSafetyInformation
horikawa	50	yamadera	SampleSafetyInformation
tanaka	55	shiogama	SampleSafetyInformation
sato	15	rifu	SampleSafetyInformation
tanaka	20	rifu	SampleSafetyInformation
takahashi	30	sendai	SampleSafetyInformation
saito	25	tomiya	SampleSafetyInformation
takahashi	20	tomiya	SampleSafetyInformation
saito	25	rifu	SampleSafetyInformation
takada	20	rifu	SampleSafetyInformation
takahashi	25	sendai	SampleSafetyInformation
sato	60	tomiya	SampleSafetyInformation
takahashi	40	yamadera	SampleSafetyInformation
sato	50	sendai	SampleSafetyInformation
takahashi	40	sendai	SampleSafetyInformation

LOCAL DATA				
NAME	AGE	PLACE	DETAIL	
takahashi	30	sendai	SampleSafetyInfor...	▲
takahashi	40	sendai	SampleSafetyInfor...	
takahashi	25	sendai	SampleSafetyInfor...	
sato	50	sendai	SampleSafetyInfor...	
horikawa	50	yamadera	SampleSafetyInfor...	
takahashi	40	yamadera	SampleSafetyInfor...	

図 7 管理コンテンツ
Fig. 7 Local objects.

SEARCH RESULT				
NAME	AGE	PLACE	DETAIL	
oide	20	sendai	SampleSafetyInfor...	▲
takahashi	30	sendai	SampleSafetyInfor...	
takahashi	40	sendai	SampleSafetyInfor...	
takahashi	25	sendai	SampleSafetyInfor...	
sato	50	sendai	SampleSafetyInfor...	

(a) result ("search * * sendai")

SEARCH RESULT				
NAME	AGE	PLACE	DETAIL	
oide	20	sendai	SampleSafetyInfor...	▲
takahashi	25	sendai	SampleSafetyInfor...	

(b) result ("search * 2* sendai")

図 8 検索結果
Fig. 8 Search results.

図 8 (a) に、避難場所が "sendai" の安否情報の範囲検索の結果、図 8 (b) はその条件に 20 代の避難者という条件を加えた複数条件による検索結果をそれぞれ示す。図 8 (a) より、条件に合うすべての安否情報をネットワーク上から検索できた。また、図 8 (b) より、図 8 (a) の検索結果からさらに 20 代の安否情報を抽出して表示できた。

4.3 動的負荷分散に関する実験

動的な負荷分散を実現する Waon と Mercury それぞれのアルゴリズムについて、その特性を評価するための評価実験を行った。比較対象のアルゴリズムとして Mercury を選んだ理由は、Random Sampling によってランダムに選択し

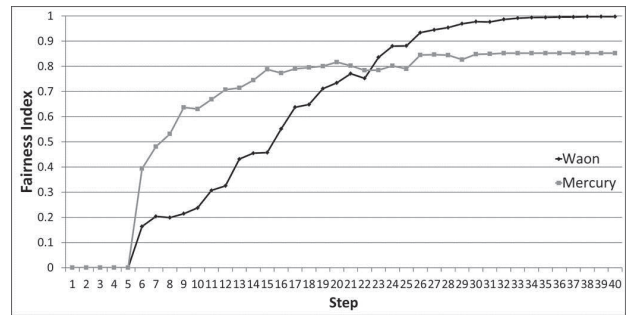


図 9 Fairness Index の推移
Fig. 9 Sequential network Fairness Index.

たノードから負荷値を収集し、今回の提案における Waon と同様にネットワーク全体から平均負荷値を推定する手法を採用しているためである。文献 [2] においては、Waon の負荷分散アルゴリズムの比較対象として Chord+load balancing 方式を採用しているが、文献 [2] の提案アルゴリズムでは Waon は自身の周辺のノード (predecessor) から負荷状況の情報を収集し、ネットワークの局所部分から平均負荷値を推定する手法を採用しており、比較対象としては不適切との判断から、Mercury との比較検討を行っている。

Mercury に与えるパラメータは文献に則り $k = 4, k_1 = 4, k_2 = 4, d = 3, \alpha = \sqrt{2}$ とした。ただし、両アルゴリズムともに安否情報の複製は作成しないものとし、Mercury における hub の数は 1 とした。

本実験では 16 台のノードを起動してネットワークを生成し、動的負荷分散の評価には Fairness Index (FI) [17] の値を用いた。Fairness Index はオブジェクト間の公平性を評価する $1/N$ から 1.0 までの指標であり (N は全体のノード数)、以下の式によって算出される。

$$FI(x_1, x_2, \dots, x_N) = \frac{(\sum x_i)^2}{N \sum x_i^2}$$

また、動的負荷分散の実行 1 回を 1 step とし、ノードは以下の動作をすることとした。なお、本実験における安否情報は偏った文字列の安否情報が登録される状況を考慮し、ランダムに生成した偏った文字列から 3.4.1 項で述べた手法に従って ID を生成した。

steps 1-5 ネットワークが安定するまで待機する。

step 6 ID を偏らせて生成した安否情報を 200 個配置する。

steps 7-40 ネットワークが安定するまで待機する。

図 9 に各アルゴリズムの動的負荷分散の評価値を、図 10 にノード間で通信されたメッセージ量を示す。結果より、Waon の動的負荷分散は収束までに時間を要するが、ネットワークの再構築が不要であるためにノード間で交わされるメッセージ量がほとんど増加しないことを確認した。一方、Mercury の動的負荷分散は Waon よりも早く収束するが、ネットワークの再構築を繰り返すためにノード間で交

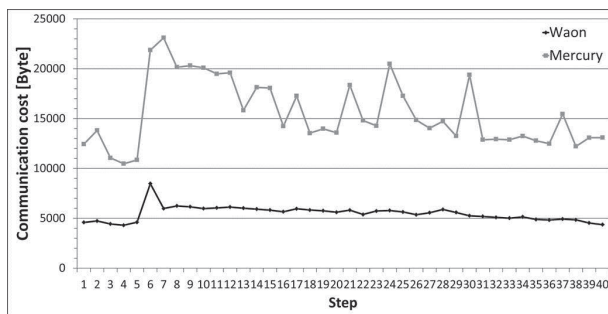


図 10 メッセージ量の推移
Fig. 10 Sequential communication cost.

わされるメッセージ量が増加することを確認した。

4.4 考察

4.2 節の実験結果から、3.3 節で述べたノードの機能が適切に動作して P2P ネットワークを構築し、3.4.2 項で述べた安否情報の追加および複製が行われ、ネットワーク上のノード間で適切に安否情報が分散して配置されたことを確認した。また、3.4.4 項で述べた安否情報の検索が実現され、目的の安否情報を的確に検索できたことを確認した。これにより要件 (P1) と (P2) が充足され、ファイル共有システムで必要不可欠である範囲検索などの柔軟な検索機能を実現し、実用に向けた安否情報共有システムとしての機能を実装できた。

また、4.3 節の実験結果から、3.5 節で述べた通信コストの増加を防ぐ動的負荷分散が実行され、周囲の環境に応じて動的に負荷分散を実行したことを確認した。これにより不安定なネットワーク環境下において少ない通信量で機能を実現するシステムを実装できたといえる。また、Mercury との比較によって、Waon がよりネットワーク資源が不足する大規模災害時における利用に適していることを確認した。以上から、(P3) が充足されたといえる。

なお、Waon の値の収束が遅い理由として、Mercury における負荷分散との方向性の違いが考えられる。Mercury における負荷分散アルゴリズムは、ネットワーク全体の平均負荷値を推定した後に、最も負荷がかかっていないと推測されるノードが一時的にネットワーク上から離脱し、最も負荷がかかっていると推測されるノード付近に再加入することで負荷の分散を実現する。そのため、Mercury ではより高速に収束状態に向かう一方で、ネットワークの再構築が頻繁に行われる。一方、Waon における負荷分散アルゴリズムでは、自身の周辺のノードすべてが過負荷であることが分かっていたとしても、自身の predecessor に対してのみ負荷分散を実行する。そのため、収束状態に至るまでにより多くの時間がかかるが、その間ノードはネットワークの再構築が不要となる。今回想定している環境ではノード間での通信は最小限に抑えることが求められるため、Waon による負荷分散の方がより適している。しかし、

収束の速度が遅いという問題のほかにも、収束の速度がコンテンツの偏り具合に影響されるという問題が既知の課題としてあげられている。そのため、パラメータの最適化やアルゴリズムの効率化によって、収束速度を Mercury などの既存研究と遜色のない速度にすることが今後の課題としてあげられる。

4.5 本システムの適用可能性

本システムの考え方は、安否情報共有システムに限らず、災害時の様々な情報共有システムに適用可能であると考えられる。以下に本方式の、災害時情報共有システム一般における利点について考察する。

- (1) 災害時の情報共有システムは、被災を免れた機器によって継続的にサービスを提供すべきである。Waon は単一障害点のない構造化 P2P であり、それぞれのノードの持つルーティングテーブルは定期的に更新される。Waon では不安定なネットワーク環境におけるノードの突然の参加や離脱を検出し次第ネットワークを修正するため、サービスの提供が途切れることはない。この利点はサーバの稼働が前提となっている他の情報共有システムと比較して有利である。
- (2) システムは限られたネットワーク資源を効率的に使用する必要がある。Waon では動的負荷分散による通信コスト増加が生じず、ノードの物理的な近傍性を考慮したネットワークを構築することで物理的なトラフィックを軽減する。これにより帯域が十分確保できないネットワークへの過負荷を低減させることが可能である。
- (3) システムは被災地から登録された災害関連情報を能動的に被災地外のノードに転送する必要がある。Waon ではノード ID やコンテンツ ID を自由に決定できるため、物理的な近傍性を考慮したネットワークを構成することで、比較的安定な地域を特定したり、できるだけ自身から離れた地域にあるノードに複製を配置するような ID の割当てが容易に行える。これにより情報の欠落を防ぐと同時に、被災地外への情報伝搬効果的に実現できる。
- (4) システムは柔軟な災害関連情報の検索機能を提供する必要がある。Waon ではコンテンツの ID を自由に決定できるため、情報の持つ属性ごとに範囲機能を実行することができる。また、情報のリスト化など、特定情報の検索以外の利用も容易に実現可能である。

5. おわりに

本論文では、大規模災害時におけるネットワークアウェアな P2P 型安否情報共有システムの提案を行った。本システムでは、我々が提案してきた構造化 P2P ネットワークである Waon をネットワーク基盤技術として利用し、通

信コストの増加を最小限に抑える動的負荷分散と、柔軟な安否情報の検索機能を同時に実現した。また、実験結果より、本システムの効果を検証した。

今後の課題として、未実装機能であるノードの物理的な近傍性を考慮したネットワーク構築機能を実現したのちに、ネットワーク環境が不安定な状況下での本システムの動作実験を行う。具体的には、ノードの参加や離脱を考慮するネットワーク環境において、リングネットワークを適切に維持し、複製や範囲検索が正しく動作することを検証する。また、負荷値の定義をノードの性能やネットワーク環境を考慮するように拡張することで、より厳密な意味での負荷の分散を実現する。これにより、特に被害の大きい地域のノードは情報の登録と閲覧のみを行い、比較的被害の少ない地域のノードが安否情報の管理を引き受けるなど、ノード間でネットワーク上における自律的な役割分担を可能とするなどの、本方式の高度化を進める予定である。

謝辞 本研究の一部は、公益財団法人中島記念国際交流財団の助成を受けて実施したものである。

参考文献

[1] IPA 独立行政法人情報処理推進機構：プレス発表 IPA「災害に対応する IT システム検討プロジェクトチーム」の活動結果について報告, IPA (オンライン), 入手先 (<https://www.ipa.go.jp/about/press/20130128.2.html>) (参照 2013-08-01).

[2] Takeda, A., Oide, T. and Takahashi, A.: Simple Dynamic Load Balancing Mechanism for Structured P2P Network and its Evaluation, *International Journal of Grid and Utility Computing*, Vol.3, No.2, pp.126-135 (2012).

[3] 小山 由, 水本旭洋, 今津真也, 安本慶一: 大規模災害時の安否確認システムと広域無線網利用可能エリアへの DTN に基づいたメッセージ中継法, 技術報告 29, 奈良先端科学技術大学院 (2012).

[4] 佐藤剛至, 柴田義孝: 災害情報システムのための動的ネットワーク再構成手法に基づいたコグニティブ無線の研究, 技術報告 40, 岩手県立大学 (2011).

[5] 大瀧 龍, 重安哲也, 浦上美佐子, 松野浩嗣: 自律的無線ネットワークを用いた被災情報提供システム—被災地域の地形を考慮した無線ノード置局アルゴリズムの提案, 情報処理学会論文誌, Vol.52, No.1, pp.308-318 (2011).

[6] Fall, K.: A Delay-tolerant Network Architecture for Challenged Internets, *Proc. 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '03*, pp.27-34, ACM (2003).

[7] 蛭田瑞生, 鶴岡行雄, 多田好克: 災害情報共有システムの提案, 技術報告 2, 電気通信大学 (2012).

[8] Sasaki, Y. and Shibata, Y.: A Disaster Information Sharing Method by the Mobile Servers in Challenged Networks, *2012 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pp.1048-1053 (2012).

[9] Stoica, I., Morris, R., Liben-Nowell, D., Karger, D., Kaashoek, M., Dabek, F. and Balakrishnan, H.: Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications, *IEEE/ACM Trans. Networking*, Vol.11, No.1, pp.17-32 (2003).

[10] Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S.: A Scalable Content-addressable Network, *Proc. 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '01*, pp.161-172, ACM (2001).

[11] Rowstron, A. and Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-scale Peer-to-peer Systems, *Proc. IFIP/ACM International Conference on Distributed Systems Platforms* (2001).

[12] Zhao, B.Y., Huang, L., Stribling, J., Rhea, S.C., Joseph, A.D. and Kubiatowicz, J.D.: Tapestry: A Resilient Global-Scale Overlay for Service Deployment, *IEEE Journal on Selected Areas in Communications* (2004).

[13] Bharambe, A.R., Agrawal, M. and Seshan, S.: Mercury: Supporting Scalable Multi-Attribute Range Queries, *Proc. ACM SIGCOMM* (2004).

[14] Schutt, T., Schintke, F. and Reinefeld, A.: Range Queries on Structured Overlay Networks, *Computer Communications* (2008).

[15] Sylvia, R., Mark, H., Richard, K. and Scott, S.: Topologically-aware Overlay Construction and Server Selection, *21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, Vol.3, pp.1190-1199 (2002).

[16] Kazuyuki, S., Yoshio, T. and Satoshi, S.: Overlay Weaver: An Overlay Construction Toolkit, *Computer Communications* (2008).

[17] Jain, R., Chiu, D. and Hawe, W.: A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems, Technical Report TR-301, DEC Research (1984).



生田 拓馬 (学生会員)

2013年仙台高等専門学校専攻科修了。現在、東北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、エージェント指向コンピューティングに関する研究に従事。



武田 敦志 (正会員)

2005年東北大学大学院情報科学研究科博士後期課程修了。同年東北文化学園大学情報工学科助手。2010年東北大学院大学情報科学科講師。2011年同大学准教授。オーバーレイネットワーク、ネットワークセキュリティ、マルチメディア通信等の研究開発に従事。FIT2007論文賞等受賞。博士(情報科学)。



高橋 晶子 (正会員)

2007年東北大学大学院情報科学研究科博士後期課程修了。2006年(独)日本学術振興会特別研究員(DC2), 2007年同(PD)。2009年仙台高等専門学校助教。2013年同准教授。知識処理, エージェント指向コンピューティング, P2P ネットワーク等の研究開発に従事。博士(情報科学)。



菅沼 拓夫 (正会員)

1997年千葉工業大学大学院博士後期課程修了。同年東北大学電気通信研究所助手。同大学助教授, 准教授を経て, 2010年から東北大学サイバーサイエンスセンター教授。ネットワーク管理技術, ネットワークミドルウェア, エージェント指向コンピューティング, 共生コンピューティング等の研究開発に従事。博士(工学)。