

モバイル端末による被災状況の分散集約に向けた 制御情報管理方式の検討

内山 彰^{1,a)} 樋口 雄大^{1,b)} 廣森 聡仁^{1,c)} 梅津 高朗^{2,d)} 山口 弘純^{1,e)} 安本 慶^{—3,f)}
東野 輝夫^{1,g)}

概要：効率の良い災害対応のためには、被災状況を迅速に把握することが重要である。現状、このような情報は実際に捜索隊や自治体職員が足を運んで確認しているが、捜索隊が現場に到着する前にオンラインで情報を集約することができれば、迅速な救助活動の実現につながる。そこで本稿では、被災状況の集約に対するユーザ要求（クエリ）を効率良く実現するための情報処理基盤の設計開発を行う。開発する基盤はDTN バンドル層とアプリケーション層間のミドルウェア層として位置づけられ、各ノードは資源情報ならびに災害情報を分散管理する。この下で、各ノードはどのように振る舞えば指定されたクエリを全体として実現できるかを検討し、そのアルゴリズムの一例を示している。

1. 序論

地震や災害の多発する日本では「防災・減災」に資する技術の開発は最も重要な研究テーマの一つであり、東日本大震災の甚大な被害を踏まえて平成 23 年 8 月に策定された第 4 期科学技術基本計画でも「安全・安心な都市基盤の構築」が最優先課題とされている。また、欧州委員会の FP8(Horizon 2020) 計画や米国 NSF 研究でも安全・安心な都市基盤の構築に関する研究の推進が謳われている。特に大災害時においては、道路や建築物の損壊、車両による道路の閉塞や人の分布などの情報を広域で迅速かつ円滑にセンシング・集約し、人や物資の輸送戦略や救命救急従事者の配置戦略、被災者への情報配信などを可能とすることが求められる。一方で、インターネット網のみならず、電話網すら寸断される状況においては、3G/4G や WiFi など従来のパラダイムに基づくデータ集約や固定カメラなどのインフラによる状況把握だけでは機能せず、災害時でも有

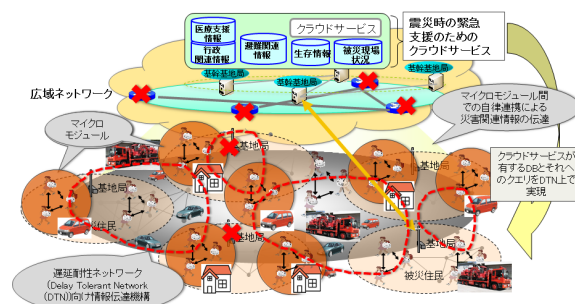


図 1 情報センシング集約機構

効性を発揮する新しいパラダイムとアーキテクチャに基づく情報センシング集約機構が必要となる。

例えば、災害時において、国連の国際搜索救助諮問グループ (INSARAG) のガイドライン [1] では、各搜索隊は担当地域の被災状況 (建物の被害や被災者数) を把握した後に、優先順位を付けて搜索活動に当たることが求められている。効率の良い災害対応のためには、被災状況を迅速に把握することが重要であり、例えば、各地の傷病者分布を調べることで救急隊の派遣計画を立て、災害医療の効率化を図ったり、道路状況を把握することで緊急車両の通行可能な経路を調べ、短時間で目的地に到着するといったことが考えられる。現状、これらの情報は実際に搜索隊や自治体職員が足を運んで被災状況を確認しているが、搜索隊が現場に到着する前にオンラインで情報を集約することができれば、迅速な救助活動の実現につながる。

スマートフォンやウェアラブルデバイスは、周辺の写真を送信することで帰宅難民の混乱状況などをレポートす

1 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Os-
aka University

2 滋賀大学経済学部
Faculty of Economics, Shiga University

3 奈良先端科学技術大学院大学
Nara Institute of Science and Technology

a) uchiyama@ist.osaka-u.ac.jp

b) t-higuti@ist.osaka-u.ac.jp

c) hiromori@ist.osaka-u.ac.jp

d) ta-umedu@biwako.shiga-u.ac.jp

e) h-yamagu@ist.osaka-u.ac.jp

f) yasumoto@is.naist.jp

g) higashino@ist.osaka-u.ac.jp

る機能を有し、車両は広範囲での道路閉塞状況の収集やバッテリー制約に縛られないデータ処理が可能である。また対象街区の公衆 WiFi 基地局や救助隊が臨時に敷設する無線基地局は、内蔵ストレージを活用することで周辺デバイスからのデータを集約・計算するマイクロクラウドの役割を果たすこともできる。このように、処理性能やストレージ容量、モビリティや通信範囲などが異種多様である環境で、「マイクロモジュール群」を知的かつ機能的に連携させるため、遅延耐性ネットワーク (Delay Tolerant Network(DTN)) 上で情報センシングデータベースならびにその分散処理機能を実現することが強く望まれる。

一方で、被災時や緊急時には被災状況や環境が時々刻々と変化し、DTN で活用できる資源情報も流動的に変化するため、集約すべき災害情報に加えてそれらの資源情報も分散環境で管理・活用することで、効率的なデータ処理・転送・集約の方法を各ノードで自律分散的に決定できる可能性がある。例えば、災害情報の発生領域とのデータ転送遅延や成功率、ノード群の移動方向などから災害情報に対するアクセス遅延や成功率を予測し、ノード群のストレージ容量を考慮してデータ処理結果のランデブーポイントを決定することなども可能となる。一方で災害情報を必要とするユーザはこれらを意識せず、時空間および必要なデータのみを指定するだけでそれらの情報が取得できることが望ましい。

本稿ではこのようなユーザ要求 (クエリ) を効率良く実現するための情報処理基盤の設計開発を行う。開発する基盤は DTN バンドル層とアプリケーション層間のミドルウェア層として位置づけられ、各ノードは資源情報ならびに災害情報を分散管理する。このもとで、各ノードはどのように振る舞えば指定されたクエリを全体として実現できるかを検討し、そのアルゴリズムの一例を示している。これまで提案されてきた DTN における情報集約とは異なり、本基盤では DTN 上で、分散した情報を一元的な仮想データベースサービスとしてユーザに提供するための基盤技術の実現を目指している。データのみならず、DTN 上の計算資源の分散管理と更新、およびそれらのデータや資源を活用した高度なデータ問い合わせ処理を実現することを目指している点で従来方法とは異なるアプローチをとっている。具体的な例題を用いてその可用性を示す。

2. 関連研究

2.1 災害時通信網

災害時にも高い可用性を維持する高信頼な通信インフラの実現に向けて、これまでに様々な取り組みが行われている。NerveNet [3] では、各地に設置された基地局間でメッシュネットワークを構築する。このとき、基地局間の経路を複数用意しておくことで、災害により一部の経路が利用できなくなった場合でも、通信を継続できる仕組みを実現

している。Never Die Network (NDN) [4] は、動作環境の悪化に対する堅牢性の向上を目指したネットワークである。NDN では、衛星通信や WiFi, WiMAX 等の複数の通信手段を組み合わせ、ネットワークの状態に応じて自律的・適応的に制御方法を変化させる。劣悪な環境においては、通信品質を落として、ネットワークの接続性の維持を優先する。これらの取り組みは災害時の情報通信にとって必要不可欠であり、災害が起きたときに備えてできる限りの対策を講じておくべきである。我々の提案する情報処理基盤は、このような災害時通信網と連携することで、情報集約の効率をより高めることができる。例えば、災害時通信網が利用可能な地域では、基地局を代表ノードとして制御情報を集中管理することで、ユーザ要求に応じて必要最小限のセンサ群から被災状況を収集することも可能である。

2.2 In-Network Data Aggregation

無線センサネットワークの分野では、センサノードの消費電力の軽減を目的として、データをネットワーク内で集約しながら収集する In-Network Data Aggregation の研究が盛んに行われている [5]。In-Network Data Aggregation は、ノード間で木構造のトポロジを構成する structured な方式と、特定のトポロジを前提としない unstructured な方式に大別される。structured な方式の代表例としては、TAG [6] や PEGASIS [7] 等がある。これらの方式では、シンクノードを根とする木構造を構成し、ノード間の親子関係に従って葉から根に向かってデータ収集と aggregation を繰り返す。これらの方式は、少ないメッセージ数で効率良くデータ集約を行える利点がある一方、あるリンクにおいて通信が失敗したり、ノードが故障したりした場合に、そこから下の部分木に属する全ノードの情報が失われてしまうため、障害への堅牢性が課題となる。

一方、unstructured な方式の代表例としては、Synopsis Diffusion [8], Adaptive Scalable Counting (ASC) [9] などが挙げられる。これらの方式では、親子関係を定めず、隣接ノード間でデータ交換と集約を繰り返すことにより、最終的に複数の経路で集約されたデータがシンクに届けられるため、structured な方式とは対照的に高い堅牢性が実現できる。一方で、同一ノードが観測したデータが複数の経路でノード間を伝搬する過程で、同じデータを複数回 aggregation の結果に含めてしまう二重カウントの問題が生じる。そこで、duplicate-insensitive aggregation と呼ばれる工夫により、二重カウントの問題を回避しつつ、確率的にデータ集約結果の正しさを保証する仕組みが提案されている。

これらの In-Network Data Aggregation は提案する情報処理基盤で被災状況や資源情報を管理する際、スケーラビリティを確保するために必要な仕組みであり、本研究では状況に合わせた既存手法の利用を想定している。

2.3 DTN ルーティング

DTN におけるルーティングプロトコルについても数多くの既存研究が存在する [10]. 既存のプロトコルは Deterministic な方式と Stochastic な方式の 2 通りに大別することができる. Deterministic な方式 [11] は, 事前に全ノードの動きが分かっている理想的な環境を前提とし, 将来のノード同士の遭遇時刻に基づき, time-evolving graph における最短経路問題として最適なノード間経路を決定する. しかし, ノードモビリティの正確な予測は実際には困難であるため, 主として性能比較の対象として用いられる.

このため, 多くの DTN ルーティングプロトコルでは Stochastic なアプローチが採用されている. Stochastic なアプローチでは, 遭遇したノードに対してデータ転送を行うかどうかを, 位置情報や遭遇履歴等に基づき確率的に決定することで, ネットワークトポロジに関する知識が無い DTN 上において, 宛先へのデータ配送を実現する. 文献 [12] では, 宛先ノードの位置情報が既知であると仮定し, 遭遇したノードの移動方向と自身の移動方向をもとに, 遭遇ノードにデータを転送するかどうかの決定を行っている. また, 文献 [13] のようにモビリティの制御が可能なノードを仮定し, 効率の良いデータ配送経路を決定する方法も考案されている.

本基盤では, これらのルーティングプロトコルを適切に組み合わせることにより, DTN を介して, ノードの資源情報や被災状況のセンシング結果を効率的に集約・管理する仕組みを実現する.

2.4 時空間データベース

提案する情報処理基盤は, 時間や空間と紐付いたデータを持つ多数のモバイル端末群が DTN を介して疎に連結された, 分散型の時空間データベースとみなすことができる. データベース分野では, 時々刻々と変化する人やモノの位置情報を格納した時空間データベース上で, クエリを効率的に処理するための仕組みが活発に研究されている. SINA [14] や SEA-CNN [15] では, 類似する複数の時空間クエリをグルーピングする共有実行と, 過去のクエリの実行結果を逐次的に更新することでクエリの再実行を回避するインクリメンタル実行とを組み合わせることで, 高いスケーラビリティを実現している. こうしたアイデアの一部は, 我々の情報処理基盤上でのクエリ実行にも応用することができると考えられるが, 通信網が寸断された災害現場においては, 従来の時空間データベースのように, データベース内に格納されたすべてのオブジェクトの位置情報が集中管理されている環境を前提とすることは現実的でない. 提案する基盤では, DTN を介してモバイル端末のリソースやデータの空間的分布を最小限のオーバーヘッドで分散管理するとともに, クエリの配送・実行手順をネットワーク内で自律的に最適化する機構を実現することにより, クエ

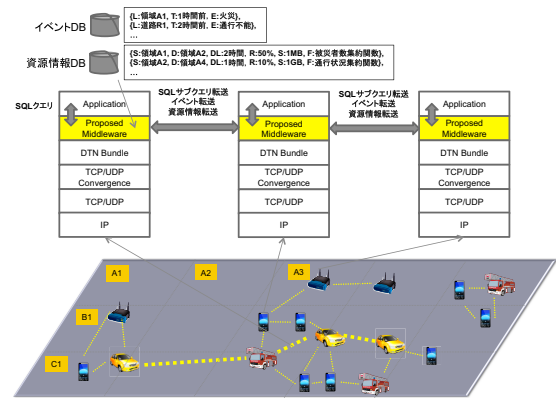


図 2 提案する情報処理基盤の概要

リ実行結果の品質や遅延時間を確率的に保証することを目指している.

3. 提案する情報処理基盤

3.1 アーキテクチャ

本基盤は, 人々が有するスマートフォン, 緊急車両や一般車両の車載器, 移動基地局, 街路の高機能カメラなどの「ノード」と, それらの間の遭遇時の狭域ピアツーピア通信, もしくは広域網を介した通信を想定する. ただし広域網は被災のために多くの地域で到達不要であるような状況を想定する. それらのノードは一定量のストレージと計算機能を有するものとし, 街路のカメラ等を除き, 大部分は移動ノードであるとする. 各ノードはセンサーやカメラ等を有し, それらからの画像や音, テキストといった入力データを, 時空間メタデータを与え, 後述する「イベント」として格納することができる. メタデータはデータの時空間に関する情報を付与することができる. また自身の計算機能やストレージ残量, 予想残存バッテリー量などを「資源情報」として格納しておく.

本基盤の概要を図 2 に示す. 本アーキテクチャは DTN のバンドル層とアプリケーション層間のミドルウェアとして位置付け, 以降 Smart Query Processing 層 (SmartQP 層) とよぶ. 各ノードのアプリケーション (以降ユーザとよぶ) に対しては, 対象領域のどの「区画」にどの程度の「資源」が利用可能か, の情報が提供されるが, それらも DTN 環境で集約されることから, 空間的に不完全でかつ時間的に劣化した情報であり, かつノード間の資源情報の一貫性は一般的には保証されない. SmartQP 層からユーザに提供されるアプリケーションインタフェースを介し, ユーザは SQL に基づき記述された時空間クエリとそれに対する最適化基準の要求 (低遅延優先, データ網羅性優先など), ならびにクエリの結果を伝達すべき対象領域, の 3 つを指定することで, SmartQP 層ではそのクエリに合致したイベントを集約して対象領域に伝達する機能を提供する. このためには, 各ノードは自身のユーザのみならず,

表 1 イベント情報 DB の例

(a) イベント情報 DB

ID	Location	Time	Type	Value1	Value2	Image	Video
000001	Road R3	[Oct.10 10:00, 15:00]	HazardByCrowd	Severe	-	-	-
000002	Region A1	[Oct.10 15:00, 16:00]	BuildingFire	Danger	-	-	-
000003	Region A1∪A2	[Oct.10 14:00, 15:00]	FoodSupplied	100	-	-	-
000004	BusStop S1	Oct.10 16:00	Queue	Long	100	-	-
000005	TaxiStop S2	Oct.10 10:00	-	-	-	000005.jpg	-
000006	TaxiStop S2	Oct.10 11:00	-	-	-	000006.jpg	-

(b) (a) のイベント情報 DB からのクエリによるイベント情報生成例

ID	Location	Time	Type	Value1	Value2	Image	Video
000007	TaxiStop S2	Oct.10 10:00	Queue	Short	40	-	-
000008	TaxiStop S2	Oct.10 11:00	Queue	Long	120	-	-
000009	TaxiStop S2	[Oct.10 10:00, 11:00]	Queue	Growing	+80	-	-

表 2 資源情報 DB の例

ID	Location	Time	Type	Value1	Value2	Value3
000001	Region A1	[Oct.10 12:00, 15:00]	ACCESSIBILITY	A2	92%	1h
000002	Region A1	[Oct.10 10:00, 14:00]	ACCESSIBILITY	A2	40%	2h
000003	Region A1	[Oct.10 12:00, 13:00]	ACCESSIBILITY	A1	80%	1h
000004	Region A1	Oct.10 14:00	STORAGE	1GB	-	-
000005	Region A2	Oct.10 14:00	PROCESSING	CountHuman(jpg)	-	-
000006	Region A2	Oct.10 14:00	PROCESSING	CountHuman(mp4)	-	-
000007	Region A2	Oct.10 15:00	PROCESSING	DetectQueue(e1,e2)	-	-

直接通信可能なノード (DTN により将来遭遇するノード, あるいは固定的に隣接するノード, あるいは広域網を介して通信可能なノードなど) 間で協調しながら SQL クエリやイベント情報ならびに資源情報を交換しながら効率よく処理を実行する必要がある。以下ではそれらのノードをまとめて隣接ノードとよぶ。与えられたクエリに対し, それを各ノードがどのように解釈し, どの隣接ノードに対してどのようなクエリを再発行することで, 最初に投入されたクエリを DTN 環境で効率よく実現できるか, が課題であり, そのために SmartQP 層はどのような機能を有し, それをどのようなしたアーキテクチャを実現すべきかを検討することが本稿の目的である。

SmartQP 層全体が (分散して) 有するイベント情報 DB の例を表 1(a) に示す。イベントとはある時空間領域における特定の事象を表すエンタリであり, 各行が 1 つのイベントに対応し, 列はタイプ列に指定された属性ごとに意味解釈が与えられる (簡単のため, 各ノードはタイプごとの意味解釈を知っていると仮定する)。例えば, 000001 は道路 R3 において 10 月 10 日の 10 時から 15 時の間に避難者等の通行者による通行障害が発生したことを表すイベント HazardByCrowd が発生し, その程度は Severe であったことを表す。また, 000004 は, バス停 S1 において 10 月 10 日の 16 時において Queue が観測され, その長さは Long, 人数は 100 人であったことを示している。000005 は撮影場所および時刻情報が付与された写真であることを表す。

同様に, 資源情報 DB の例を表 2 に示す。資源情報 DB の各行はある領域に関する「資源」の情報である。資源とはその領域に存在するノード群の特性を表し, (i) 領域への到達可能性 (ACCESSIBILITY), (ii) データ容量 (STORAGE) ならびに (iii) データ処理能力 (PROCESSING), に大別される。(i) の到達可能性とはその領域に対し, ある領域からどの程度の信頼性 (到達率) および遅延時間でデータ送信可能かを示し, 例えば 000001 では, 10 月 10 日の 12 時から 15 時の間に, 領域 A1 に対し, 領域 A2 から到達率 92%, 遅延時間 1 時間で到達した実績があることを示している。(ii) のデータ容量は領域全体で保持可能なデータ容量を示し, (iii) のデータ処理能力はイベント情報 DB あるいは資源情報 DB の特定のカラムデータを引数として新しいイベント情報および資源情報を生成するデータ操作言語として設計される。例えば表 2 の資源情報 000005 は, 領域 A2 において, 操作 CountHuman(jpg) を実行できるノードが 14:00 に存在したことを示し, それを用いて表 1(a) のイベント 000005 および 000006 の jpg ファイルをそれぞれ処理した場合に登録されるイベント 000007 および 000008 を表 1(b) に示す。また, 資源情報 000007 は領域 A2 において操作 QueueDetection を実行できるノードが 15:00 に存在したことを示し, 表 1(b) のイベント 000009 はそれを用いて同表のイベント 000007 および 000008 から生成したものである。これらのようなデータ操作は DB のスキーマとともに予め定義されているものとする。

各ノードの SmartQP 層は、あるクエリを受けた場合に、自身が有するイベント DB と資源情報 DB をもとに、そのクエリをどのように分割してどの隣接ノードにサブクエリとして投入するかの「アルゴリズム」を有する。理想的には、クエリを処理するノードが、(1) 他ノードが保有するイベント情報 DB、ならびに (2) それらのノードまでの DTN を介した予想到達可能性や予想遅延時間からなる資源情報、を一元的かつリアルタイムに把握可能であれば、それらの完全な情報を用いて、各ノードは受け取ったクエリに対し、どのノードにどのようなサブクエリを発行すれば与えられたクエリが（なるべく短い遅延で、かつなるべく確実に）実現可能かをある一定の方針のもとで判断できる。これに対し実際に得られる情報は自身と遭遇ノードのみからであり、ノード遭遇時のデータ交換容量ならびにストレージ容量にも一定の制限があることから、遭遇ノードと資源情報 DB を交換したり、資源情報取得のための単純な Epidemic Routing ベースのクエリを発行したり、利用可能であれば平常時の交通統計情報や道路地図等から推定したノード遭遇の期待時間等を用いて資源情報 DB を補完しながら、クエリの分散計算に必要な情報を集約する。

3.2 ユーザクエリの定義

クエリは SQL 文で記述する。通常の SQL と同様、レコード検索 SELECT や論理演算子 AND, OR, 条件指定 WHERE, テーブル指定 FROM などが利用できるが、ユーザはイベント情報や資源情報の存在を意識することなく記述させるため、テーブルを指定する演算子 FROM の引数として、時空間（領域および時間範囲）を指定する点が通常と異なる。また、条件判定を行う関数や平均、総和などを計算する集計関数も利用できる。なお、ユーザはイベント情報 DB のスキーマは把握しているものとする。

以下の例は領域 A1 から火災が発生しているイベントを取得するクエリである。

火災発生領域の検索

```
SELECT *
FROM A1
WHERE Type = "Fire"
```

また、以下の SQL クエリは領域 A2 内の被災者数を取得する。

被災者数の検索

```
SELECT Value1
FROM A2
WHERE Type = "Human" OR "Queue" OR "Crowd"
```

3.3 資源情報の定義

資源情報は領域内のノードが有する資源の特性（到達可能性、容量、処理能力）に関する情報である。すなわち、領域が保持する資源とは、その内部に存在するノード群によって提供可能な資源である。ただし、領域 p の資源情報は、各ノードが保持する資源の総和ではなく、 p 内での遅延およびノード間の到達率を考慮した資源量を表す。ノード間は DTN により接続されるため、資源情報を生成する際には領域内あるいは領域間のノード間通信遅延や到達率を把握する必要がある。DTN におけるエンドノード間通信は、ノードモビリティによるネットワークトポロジの変化に依存するため、時間経過（遅延の増加）とともに到達率は向上する。本稿では、以下のようにノード i から見たノード j の資源量 $r(i, j, d)$ を定義する。

$$r(i, j, d) = P(i, j, d) \times r_j$$

$P(i, j, d)$ は i から j への遅延 d 以下での到達率であり、 r_j はノード j の資源量を表す。理想的には各ノード i から見た時の他ノードの資源量をノード毎に保持することが望ましいが、ノードの移動に追従して情報を更新することは非常に困難であるため、本稿では領域単位でノード群をグループ化し、領域 p 内の遅延 d 以下での平均到達率 $\bar{P}(p, d)$ と p 内に存在するノード群の総資源量との積を、領域 p に到達できた場合に遅延 d 以下で利用可能な資源量 r_p と定義する。

$$r_p = \bar{P}(p, d) \times \sum_{i \in p} r_i$$

このもとで、領域 p を仮想的な 1 ノードと見なし、以下のようにノード i から見た領域 p の資源量 $r(i, p, d)$ を定義する。

$$r(i, p, d) = P(i, p, d) \times r_p \quad (1)$$

ノード間の到達率と同様、 $P(i, p, d)$ はノード i から領域 p への遅延 d 以下での到達率を表す。DTN におけるノード間遅延解析に関する研究がこれまでにいくつか提案されている [16], [17]。これらの研究では、ノード組が遭遇するまでの時間に関する解析が行われており、この時間はエンド間通信に他ノードを介さない場合のみを仮定したときの遅延と等しい。本研究では、これらの研究に基づき各領域内での通信遅延に対する到達率の分布（到達率分布）が与えられるものとし、領域ごとの資源量の計算に利用する。

3.4 資源情報の更新

資源情報は特定の領域に関連付けられるため、その資源を有するノードの移動、およびその資源情報を有するノードの移動に合わせた更新を行う必要がある。前者で

は、ノードの移動予測を行い、領域指定をノードの移動に応じて広範囲とすることで、情報の劣化を表現する。例えば時刻 t において領域 R の資源情報があれば、時刻 $t + \Delta t$ においては、 R から離脱するノードの予想移動量に応じて R を拡大することで対処できる。ただし、 R の隣接領域から R への流入もあるため、それらの領域情報を重畳することで新しい資源情報を一定時間毎に生成することで対処する。後者では自身から他領域への到達可能性情報を有していれば、自身の移動量を加味し、新しい到達率と遅延に更新することで対処する。また、ノード間遭遇時に資源情報を交換することでより広範囲の資源情報を管理できるが、すべての情報を交換することは現実的でないため、例えば BlockTree [18] のように、領域をセル分割し、同一セルに属するノード群の資源情報のみを優先的に伝搬させる方法などを採用し、限られた通信容量やデータ容量での情報共有を実現する。

4. クエリ分散処理アルゴリズム

クエリが与えられると、各ノード i はイベント情報 DB $E(i)$ を検索し、該当するエントリが見つかった場合、クエリの結果としてそのまま指定された宛先へ転送する。そうでない場合、自身が保持する資源情報 DB $R(i)$ に従って、できる限りクエリを実行可能な最小単位に分割し、サブクエリを発行する。クエリの単位は空間、オペレーションに対して定義される。

対象領域はメッシュ状に分割され、それぞれのセルがクエリの空間的な最小単位となる。空間的なクエリの分割アルゴリズムは以下の擬似コードで表される。

空間的クエリ分割アルゴリズム

```
foreach(entry in R(i))
  if (Q.Region includes entry.Region){
    IssueRegionQuery(entry.Region);
    Q.Region -= entry.Region;
  }
```

クエリ Q の FROM 句で指定されている領域 $Q.Region$ に対し、 $R(i)$ のエントリにより小さい領域 $entry.Region$ が含まれていれば、現在のクエリ Q を $entry.Region$ に対して発行し、 $entry.Region$ は元のクエリの対象から除外する。残ったクエリは geographic routing により、さらに別のノードに転送することで、いずれ最小単位に分割される。

それぞれのオペレーションには別途最小の実行単位とその実行順序が DAG として定義される。空間的クエリ分割と同様、オペレーションについても $R(i)$ に従ってできる限りこれらの最小実行単位に分割を行うが、オペレーションの依存関係により、他のオペレーションの実行結果を待機する場合も起こる。このような条件は、サブクエリの SQL の WHERE 句で指定される。オペレーション分割アルゴ

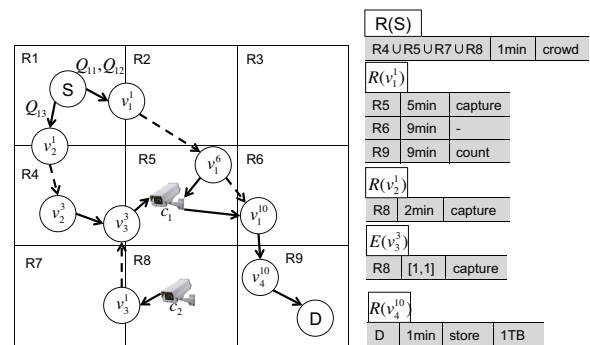


図 3 クエリ処理の例

リズムは以下の擬似コードで表される。

オペレーション分割アルゴリズム

```
foreach(entry in R(i))
  if (Q.operation == entry.operation){
    IssueOperationQuery(entry.operation);
    Q.operation -= entry.operation;
  }
```

5. クエリの分散処理例

図 3 にクエリ処理の例を示す。図中の v_i^t はノード v_i の時刻 t における位置を表す。ノード S が領域 $R5 \cup R8$ の時刻 1 から 5 における密度をノード D に知らせたい場合、以下のようにクエリ Q_1 を記述する。

群衆人数取得クエリ記述例

```
SELECT count DESTINED D
FROM EventTable(R5 U R8,5,10)
WHERE Type = image_crowd
```

FROM 句で検索対象とする仮想的な時空間データベースを指定し、Type = image_crowd と指定することで群衆人数を表すエントリを参照する。さらに、SELECT 句に count オペレーションを指定することで、Type で指定した群衆画像に対して人数をカウントする。count オペレーションは群衆画像に対して画像処理を適用することで実行される。このように各オペレーションに必要な処理は事前に定義されているものとする。DESTINED 句でノードの ID または位置を指定することで、クエリ結果の受け取り先を指定できる。

ノード S の中間層では、このクエリに対してまず自身の event table $E(S)$ を参照し、対応する時空間データベースのエントリを保持しているか調べる。図 3 の例では対応するエントリが存在しないため、隣接ノード v_1, v_2 の event table $E(v_1), E(v_2)$ を参照する。どちらのテーブルにも対応するエントリが無い場合、自身および隣接ノードの resource table $R(S), R(v_1), R(v_2)$ を参照し、効率の良

いくエリ処理方法を決定する。これらの resource table より、 v_1, v_2 からそれぞれ $R5, R8$ まで到達可能なことが分かる。また、 $R(v_1)$ から $R9$ で count オペレーションが可能なのも分かる。したがって、ノード S は Q_1 をサブクエリ Q_{11}, Q_{12}, Q_{13} に分割し、 Q_{11}, Q_{12} を v_1 へ、 Q_{13} を v_2 へ発行する。 Q_{11}, Q_{13} はそれぞれ領域 $R5, R8$ に対するクエリであり、 Q_{12} は領域 $R9$ において count オペレーションを実行するクエリである。これらのサブクエリは以下のよう

サブクエリ Q_{11}

```
SELECT capture DESTINED R9
FROM EventTable(R5,1,5)
WHERE Type = crowd
```

サブクエリ Q_{12}

```
SELECT count DESTINED D
FROM EventTable(R9,5,10)
WHERE Type = query
AND (Reference = Q11 OR Reference = Q13)
```

サブクエリ Q_{13}

```
SELECT capture DESTINED R9
FROM EventTable(R8,1,5)
WHERE Type = crowd
```

これらのサブクエリを受け取った v_1, v_2 は DTN バンドル層に従って、 $R5, R8$ の時空間データベースにアクセスする。時刻 1 における領域 $R8$ の capture オペレーションの実行結果は時刻 3 において v_2 が遭遇したノード v_3 が event table $E(v_3)$ に保持している。このため、サブクエリ Q_{13} については $R8$ に転送されず、 $E(v_3)$ に格納されているクエリの結果が $R9$ へ転送される。同様に、サブクエリ Q_{11} は DTN バンドル層に従って転送され、指定された時空間データベースにアクセスし、その結果を宛先に指定された $R9$ に転送する。サブクエリ Q_{12} を受信した $R9$ 内に存在する count オペレーションを実行可能なノード v_4 は、Reference に指定されている Q_{11} または Q_{13} の結果が届いた場合、count オペレーションを実行し、結果をノード D へ転送する。

6. 結論

本稿では、DTN における効率の良い災害情報集約を実現するための情報処理基盤の設計開発を行った。開発する基盤は DTN バンドル層とアプリケーション層間のミドルウェア層として設計され、各ノードは資源情報ならびに災害情報を分散管理する。分散管理した情報を一元的な仮想データベースとしてユーザに提供するための基本設計を検討するとともに、そのアルゴリズムの一例を示した。今後

は様々なクエリを実現可能なアルゴリズムの詳細設計を行うと共に、災害時のみならず、通常時でも適用可能な設計の検討を進めたいと考えている。

謝辞 本研究は JSPS 科研費 26220001 の助成を受けたものです。

参考文献

- [1] United Nations: *International Search and Rescue Advisory Group Guidelines and Methodology* (2012).
- [2] 松本直人: 事例に学ぶ東日本大震災における情報発信, 情報処理学会論文誌, Vol. 54, No. 3, pp. 1021–1027 (2013).
- [3] Inoue, M., Ohnishi, M., Peng, C., Li, R. and Owada, Y.: NerveNet: A Regional Platform Network for Context-Aware Services with Sensors and Actuators, *IEICE Transactions on Communications*, Vol. E94-B, No. 3, pp. 618–629 (2011).
- [4] Uchida, N., Takahata, K., Shibata, Y. and Shiratori, N.: A Large Scale Robust Disaster Information System Based on Never Die Network, *Proceedings of IEEE 26th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 89–96 (2012).
- [5] Fasolo, E., Rossi, M., Widmer, J. and Zorzi, M.: In-network aggregation techniques for wireless sensor networks: a survey, *IEEE Wireless Communications*, Vol. 14, No. 2, pp. 70–87 (2007).
- [6] Madden, S., Franklin, M. J., Hellerstein, J. M. and Hong, W.: TAG: A Tiny AGgregation Service for Ad-hoc Sensor Networks, *SIGOPS Oper. Syst. Rev.*, Vol. 36, pp. 131–146 (2002).
- [7] Lindsey, S., Raghavendra, C. and Sivalingam, K.: Data gathering algorithms in sensor networks using energy metrics, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 9, pp. 924–935 (2002).
- [8] Nath, S., Gibbons, P. B., Seshan, S. and Anderson, Z. R.: Synopsis Diffusion for Robust Aggregation in Sensor Networks, *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pp. 250–262 (2004).
- [9] Fan, Y.-C. and Chen, A.: Energy Efficient Schemes for Accuracy-Guaranteed Sensor Data Aggregation Using Scalable Counting, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 24, No. 8, pp. 1463–1477 (2012).
- [10] Zhang, Z.: Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges, *IEEE Communications Surveys Tutorials*, Vol. 8, No. 1, pp. 24–37 (2006).
- [11] Ferreira, A.: Building a reference combinatorial model for MANETs, *IEEE Network*, Vol. 18, No. 5, pp. 24–29 (2004).
- [12] Tasiopoulos, A., Tsiaras, C. and Tzoumpis, S.: On the cost/delay tradeoff of wireless delay tolerant geographic routing, *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–9 (2012).
- [13] Chen, Z. et al.: Ad Hoc Relay Wireless Networks over Moving Vehicles on Highways, *Journal of Parallel Distributed Computing*, Vol. 63, pp. 75–86 (2003).
- [14] Mokbel, M. F., Xiong, X. and Aref, W. G.: SINA: Scalable Incremental Processing of Continuous Queries in Spatio-temporal Databases, *Proc. SIGMOD*, pp. 623–634 (2004).

- [15] Xiong, X., Mokbel, M. F. and Aref, W. G.: SEA-CNN: Scalable Processing of Continuous K-Nearest Neighbor Queries in Spatio-temporal Databases, *Proc. ICDE*, pp. 643–654 (2005).
- [16] Picu, A., Spyropoulos, T. and Hossmann, T.: An analysis of the information spreading delay in heterogeneous mobility DTNs, *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–10 (2012).
- [17] Karagiannis, T., Le Boudec, J.-Y. and Vojnovic, M.: Power Law and Exponential Decay of Intercontact Times between Mobile Devices, *IEEE Transactions on Mobile Computing*, Vol. 9, No. 10, pp. 1377–1390 (2010).
- [18] Stigl, D., Gross, C., Nobach, L., Steinmetz, R. and Hausheer, D.: BlockTree: Location-aware decentralized monitoring in mobile ad hoc networks, *Proceedings of IEEE 38th Conference on Local Computer Networks (LCN)*, pp. 373–381 (2013).