

大規模かつ柔軟なセンサー共有を実現する 仮想統合センサーネットワーク

寺西裕一^{†1,†2} 中内清秀^{†1} 西永望^{†1}
石芳正^{†2} 川上朋也^{†2} 濱口雄人^{†2}
義久智樹^{†2} 西尾章治郎^{†2}

広域に分散して存在する多数のセンサーネットワークを、複数のアプリケーションがリソースの独立性を保った上で柔軟に共有可能とする仮想統合センサーネットワーク (VFSN: Virtual Federated Sensor Network) を提案する。本稿では、VFSN のコンセプト、および、ネットワーク仮想化技術とオーバーレイネットワーク技術に基づき VFSN の構成・再構成を行うプラットフォームシステムの設計を示す。また、設計に基づくシミュレーションによる大規模環境下での VFSN 構成・再構成の性能評価を示す。

Large-scale and Flexible Sensor Resource Sharing on Virtual Federated Sensor Network

YUUCHI TERANISHI,^{†2,†1} KIYOHIDE NAKAUCHI,^{†1}
NOZOMU NISHINAGA,^{†1} YOSHIMASA ISHI,^{†2}
TOMOYA KAWAKAMI,^{†2} YUTO HAMAGUCHI,^{†2}
TOMOKI YOSHIHISA^{†2} and SHOJIRO NISHIO^{†2}

We propose a new concept for sensor network sharing which we call 'VFSN: Virtual Federated Sensor Network' that enables multiple applications to share widely distributed sensor networks flexibly, preserving resource isolations. In this paper, the concept of VFSN and a VFSN platform system design to enable flexible structuring and re-structuring utilizing network virtualization and overlay network technologies. We also show an evaluation result by a simulation based on our design under a large-scale network environment.

1. はじめに

無線センサーの高機能化や低価格化に伴ない、至る所に配備されたセンサーやその観測値データを、ネットワークを介して共有・活用するアプリケーションが現実的になりつつある。実現されるアプリケーションは、気象状況や交通渋滞状況等に基づいて情報推薦や警告通知を行なうサービス、地球温暖化や大気汚染などを監視する環境サービス、商店街等でユーザーの行動を解析するマーケティングサービスなど、多岐にわたる。

各センサーは、センサーデバイス間が主に無線で接続されるセンサーネットワーク (SN) を構成し、シンクと呼ばれる特別なノードを経由して、ネットワーク接続される。ネットワーク接続されたセンサーの観測値データを共有する形態としては、観測値データを集中サーバに保存する形態がまず考えられる。この形態をとるシステムは数多く検討され^{1)~4)}、Pachube⁵⁾ 等、既に稼働しているサービスもある。しかし、この形態は、膨大な SN を収容する大規模環境でのスケラビリティの確保が難しい。このため、シンクをピアとした P2P アーキテクチャでオーバーレイネットワークを構成し、相互接続することで観測値データを共有する統合センサーネットワーク (FSN: Federated Sensor Network) の実現法が検討されてきている^{6)~9)}。FSN では、シンクにデータベースをもち、観測値を管理・共有する。また、シンクはセンサーへ直接アクセスするためのプロキシとしても動作する。例えば、カメラセンサーを遠隔から操作するといった動作も実現し得る。

FSN において、アプリケーション毎に必要な観測値データの集約・加工・変更といった処理は、FSN のノード上で分散処理することでデータ転送量を省くことができる。例えば、観測値の平均値を SN 毎に得る処理は、SN から全ての観測値データを集めてから計算するのではなく、SN 毎に平均値を計算してから集約するほうが効率が良い。こうした分散処理の実現には、FSN の各ノードにアプリケーション独自の機能を実現するプログラマビリティが要求される。X-Sensor⁸⁾ はこのようなプログラマビリティをモバイルエージェントにより実現している。

しかし、FSN は、基本的にシンクが動作する機器等の物理リソースを全アプリケーションが共用するため、ネットワーク上のデータ流通やデータ処理のためのリソースがアプリケーション毎に分離されない。よって、他のアプリケーションが膨大な量のデータ処理を要

†1 情報通信研究機構/National Institute of Information and Communications Technology

†2 大阪大学/Osaka University

求をした場合など、帯域や処理能力の圧迫等の影響を受けてしまう。

本研究の目的は、大規模環境においてFSNを複数のアプリケーションが共用する際、データ流通やデータ処理のためのリソースをアプリケーション毎に分離しつつ、独自のデータ流通・処理方法を柔軟に導入でき、かつ、ネットワーク構成やメンバとなるSNを自由に設定可能とするプラットフォームを実現することにある。

近年、リソース管理の独立とプログラマビリティを両立させるネットワーク仮想化技術の標準化や検討が進みつつある^{10),11)}。ネットワーク仮想化技術によって、上位サービスは仮想ネットワークへ新たな機能を組み込むことができ、かつ、他の仮想ネットワークに影響を与えることはない。これまでに、GENI¹²⁾、ORBIT¹³⁾等の大規模プロジェクトにおいて無線環境を含むネットワーク仮想化が検討されている他、AMPHIBIA¹⁴⁾のように異種の無線ネットワークを統合したネットワーク仮想化の検討も進められている。

FSNを仮想化する上では、FSNの機能レベルの仮想化、個別SNの物理リソースの管理や、発生する観測値データの処理まで含むプログラマビリティが要求される。しかし、こうした機能は、個々のデータに対するアクセス制御では実現できず、また、ネットワーク仮想化技術の範疇でもこれまで具体的な検討は行われていない。

本研究では、FSNをネットワーク仮想化技術との組み合わせにより仮想化し、SNのリソースを共有するアプリケーションを、品質を保った上で柔軟に実現可能とする**仮想統合センサーネットワーク (VFSN: Virtual Federated Sensor Network)**を提案する。VFSNによって、アプリケーション毎に独自の機能を持ち品質を保証できるFSNを実現でき、かつ、物理的リソースを共有した経済的なFSNの運用が可能となる。

本稿では、VFSNのコンセプト、および、ネットワーク仮想化技術に基づくVFSNプラットフォームの実現法を提案する。提案するVFSNプラットフォームによって、アプリケーション毎に動的に構成、再構成、削除される専用のFSNを構成できる。

2. VFSN: 仮想統合センサーネットワーク

2.1 仮想統合センサーネットワークの概要

仮想統合センサーネットワーク (VFSN: Virtual Federated Sensor Network) は、センサーネットワーク (SN) のリソースをサービス毎に相互共有し、統合センサーネットワーク (FSN) として扱うための仮想ネットワークである。VFSNは従来のネットワーク仮想化の特徴であるプログラマビリティとリソース分離の機能をFSN向けに拡張するものと位置付けられる。

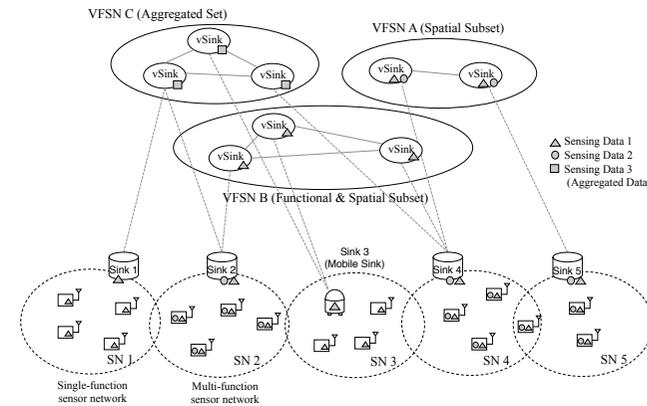


図1 VFSNの構成例

図1は、VFSNの構成例である。

VFSNでは、**仮想シンク (vSink)**が、各SNを仮想化する役割を担う。vSinkは、各VFSN専用で動作し、必要なセンサーや観測値データを扱う仮想的なシンクである。vSinkは複数のSNに対応する場合もある。VFSNはvSink間で構成される論理的なオーバーレイネットワークとなる。オーバーレイネットワークの構造は、VFSNによって異なる。

図1は、5つのSNが3つのVFSNを構成している様子を示している。SNとして、移動するロボット等が観測値データを収集する移動シンクによるSNも想定する(図におけるSN3)。

VFSNは、対象とするSNに含まれるセンサー、または、SNが発生する観測値データそれぞれの部分集合や加工情報を相互共有する。VFSNを用いるセンサー情報サービスは、センサーや観測値データへの検索処理やアクセスをプログラマビリティを持たせつつ行なえる。

VFSNにおいて対象とするセンサー、観測値データとしては、以下が挙げられる。

- 空間部分集合
地理的な部分領域のみの観測値データを共有したい場合など、全SNが対象とする観測領域全体のうち、一部分のみの観測値データや対応するセンサーを相互共有する。
- 時間部分集合
指定された一定の周期の観測値データのみを相互共有する。

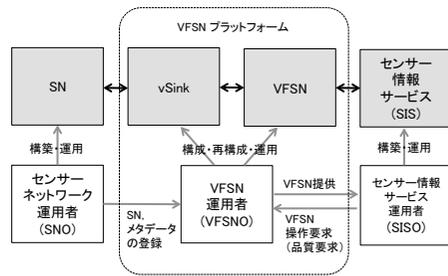


図 2 VFSN のサービスモデル

- 機能部分集合

雨量センサー、風力センサー、温度センサー、湿度センサーなど複数の種類の機能のうち、指定されたある一つの種類の機能の観測値データやセンサーを相互共有する。

- 属性部分集合

属性がある条件を満たすセンサーやその観測値データを相互共有する。例えば、同じ組織名が所有するセンサーのみ共有する場合などが相当する。

- 加工情報集合

平均値などの集約情報、時間的、空間的に補間された情報、指定された特徴値や異常値などを共有する。異機種センサー間の単位の違いを統一化する処理も含む。

図 1 では、SN1, SN3 は、センサーデータ 1 のみを扱う単機能 SN、他は、センサーデータ 1、センサーデータ 2 を扱う複数機能 SN である。VFSN A は、SN4, SN5 の空間部分集合を、VFSN B は、SN2, SN3, SN4 のセンサーデータ 1、すなわち、空間部分集合の機能部分集合を、VFSN C は、SN5 以外の SN の加工情報集合を、それぞれ共有している。VFSN C において、SN1 と SN2 は一つの vSink に収容されている。

2.2 サービスモデル

図 2 は、VFSN の構築・管理・利用を行なう上でのサービスモデルを示している。VFSN には以下の役割が存在する。

- センサーネットワーク運用者 (SNO: Sensor Network Operator)
- 仮想統合センサーネットワーク運用者 (VFSNO: Virtual Federated Sensor Network Operator)
- センサー情報サービス運用者 (SISO: Sensor Information Service Operator)

VFSN 機能は、VFSN プラットフォームによって実現・提供される。VFSN を利用するセンサー情報サービス (SIS) から見ると、VFSN プラットフォームから、要求に応じた独自の FSN が提供されるように見える。1 つの VFSN が、1 つの SIS または SISO に割り当てられる。SISO は、ネットワーク仮想化技術が提供する独立したネットワークリソースのもと、必要な機能を持つ VFSN を利用することができる。

VFSN プラットフォームは、要求に応じた vSink の生成や vSink から成る VFSN の構成を行なう。各 SN は、vSink から要求に応じて観測値データやセンサーへのアクセスを提供する。vSink によって VFSN が構成されると、各 SIS は、VFSN を通じてセンサーおよび観測値データへアクセスすることができる。

2.3 VFSN の操作と再構成

VFSN の構成、すなわち、どのような vSink を生成し、どのようなオーバーレイネットワーク構造とするかは、SISO からの操作指示に従って動的に決まる。一方、VFSN に加わる vSink のメンバー管理は、SISO からの操作指示に加え、SNO からの操作指示、すなわち、SN による自律的な構成も可能とする必要がある。これは、SN リソースの他への提供や利用、および、その中止を SNO 同士の合意に基づいて決める相互利用の運用形態に対応するためである。

また、SISO からの操作指示として、既に存在する複数の VFSN を接合 (分離) する構成も可能とする。これにより、例えば、災害時に、一時的に組織 A と組織 B の SN を結合し、観測領域を広げる、基本的機能を持つ VFSN を組み合わせてカスタムメイドなセンサーデータ流通サービスを構成可能とする、といった要求に対応できる。

VFSN を接合する際、接合後の VFSN のみ動作させ、元の VFSN は動作させない構成法と、接合後の VFSN と元の VFSN を別の VFSN として継続動作させる構成法が考えられる。後者の構成法では、VFSN 間での流通制限が必要となる場合がある。例えば、プライベート VFSN とパブリック VFSN を接合するとき、パブリック VFSN からプライベート VFSN への転送は許可するが、プライベート VFSN からパブリック VFSN への転送は許可しない、といった制限が考えられる。

以上の要求へ対応できるように、VFSN の構成に関する操作として、以下を扱う。

- VFSN の生成 (create)
- VFSN の削除 (destroy)
- VFSN への vSink の追加 (add)
- VFSN からの vSink の削除 (delete)

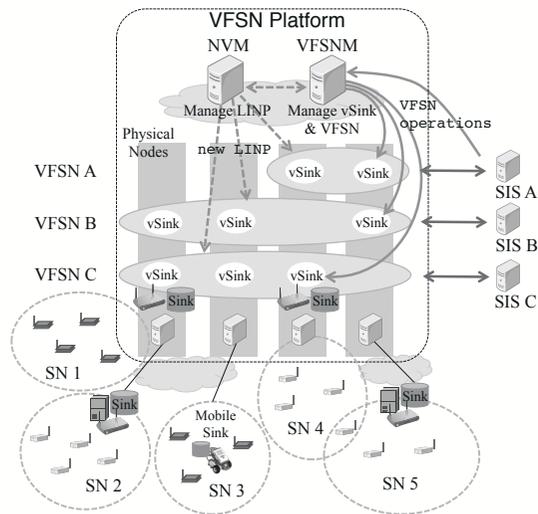


図3 VFSN プラットフォームのシステム構成例

- vSink の VFSN への参加 (join)
- vSink の VFSN からの離脱 (leave)
- VFSN 間の接合 (unite)
- 接合された VFSN 間の流通設定 (unite-set)
- 接合された VFSN の分離 (separate)

上記のうち、join、leave は、SN の自律的動作による構成に対応する操作であり、SNO が指示する。その他は、基本的に SISO からの指示に従う操作である。unite 操作では、元の VFSN の継続・非継続利用の設定をあわせて指示する。unite-set は、元の VFSN の動作を継続する unite 操作がなされた VFSN において、片方向・双方向いずれの流通を許可するかの許可設定を指示する。separate は、unite 操作がなされた VFSN を元に戻す。

これらの操作によって、メンバとなる vSink を、トップダウンで、または、自律的に変更可能な、柔軟な VFSN の構成・再構成が可能となる。

3. VFSN プラットフォームの設計

本章では、VFSN 機能を提供するプラットフォームのネットワーク仮想化基盤上の実現

法を示す。ネットワーク仮想化基盤として、PlanetLab¹⁵⁾ や CoreLab¹⁶⁾ に代表されるスライスネットワーク仮想化方式を想定する。

以下では、仮想ネットワークにおける、論理的に隔離されたネットワークの単位を LINP(Logically Isolated Network Partition) と呼ぶ¹⁰⁾。

3.1 VFSN プラットフォームのシステム構成

図3は、VFSN プラットフォームのシステム構成例を示している。図における SN および VFSN は、図1と同様の構成をとっている。VFSN プラットフォームの機能要素としては、ネットワーク仮想化基盤上の LINP を管理するネットワーク仮想化マネージャ NVM(Network Virtualization Manager)、VFSN を管理する仮想統合センサーネットワークマネージャ VFSNM(Virtual Federated Sensor Network Manager)、および、物理ノードが存在する。NVM は、PanetLab¹⁵⁾ における PLC (PlanetLab Central) に相当する。VFSNM および NVM は、VFSNO により管理・運用される。また、VFSN プラットフォームを構成する物理ノードが各 SN を収容し、LINP 上に VFSN を構成する。

VFSN プラットフォームにおいては、各 VFSN に含まれる SN が持つ機能や位置等の属性情報はメタデータとして集中管理する。各 SN のシンクは、対象とする観測値データの種類や担当する領域等、VFSN の構成において必要となる属性情報をメタデータとして VFSNM へ登録する。

3.2 vSink の実現

図4は VFSN 物理ノードおよび vSink の構成例を示している。VFSN プラットフォーム上の各物理ノードは、ネットワーク仮想化基盤として複数の SN を接続可能とするため、仮想マシン毎に無線インタフェースを具備可能な構成¹⁴⁾をとる。

物理ノード上には、各 SN が属する VFSN の LINP に対応する仮想ノード (vNode) が存在する。vSink は各 vNode 上に動作する。図4は、2つの SN を収容し、2つの vSink、2つの VFSN が動作している状態を示している。

各 SN を VFSN プラットフォームに接続する際の運用形態は、2種類ある。一つは、シンクを VFSN プラットフォーム上の物理ノード上に動作させ、VFSNO の管理下に置く形態である。図4においては、SN1 はこの形態で動作している。もう一つは、シンクを VFSN プラットフォーム上の物理リソースを用いずに動作させ SNO の管理下に置く形態である。図4においては、SN2 はこの形態で動作している。

前者の形態では、各 SN は観測値データの保存を VFSN プラットフォームに委託する形となる。vSink とシンクが同一の物理ノード上で動作するため、データ通信のオーバーヘッド

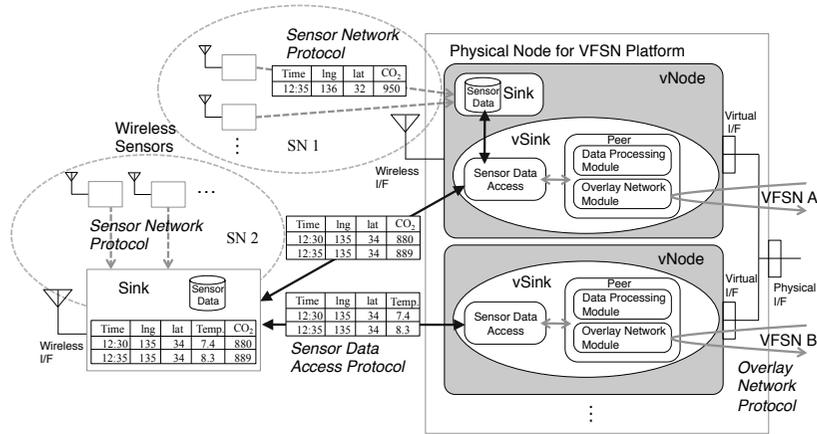


図 4 物理ノードおよび vSink の構成

を削減できる利点がある。後者の形態では、各 SN が VFSN プラットフォームとの間のアクセスネットワークを確保することになる。vSink と物理シンクが別の物理ノードで動作するため、データ転送などのオーバーヘッドが生じるが、SN 管理の独立性は維持できる。この際の際のデータ転送プロトコルとしては、IEEE 1888¹⁷⁾ 等の能動的、受動的なセンサデータ授受が可能なプロトコルを用いることを想定する。また、機能部分集合となる VFSN では、物理シンクから一部のデータを取り出せる必要がある。図 4 では、SN2 は温度と CO₂ 濃度を観測する機能を持つ SN であり、VFSN A 向けには、CO₂ 濃度を、VFSN B 向けには、温度を提供している。

vSink は、SN へのデータアクセス部、および、オーバーレイネットワークモジュール、データ処理モジュールを持つピア部から成る。ピア部は、PIAX¹⁸⁾ のようにオーバーレイネットワーク機能やデータ処理機能をプラグインできるソフトウェアにより実現する。

3.3 VFSN 操作の実現

create / destroy / add / delete 操作

SISO が VFSN を利用するには、まず、VFSNM に対し、VFSN の生成 (create) 要求を出す。VFSN の構成要求としては、対象とする vSink 集合、および、vSink 集合間で構成するオーバーレイネットワークの種類が指定される。

VFSNM は、VFSN の構成要求とメタデータに基づき、いずれの SN を対象とするかを

決定したのち、NVM を経由して、対象 SN に対応する各物理ノード上に vNode を生成、すなわち、LINP を生成する。次に、vNode 上に VFSN の構成要求に対応する vSink のプラグインソフトウェアを送り込んで動作させ、vSink 間にオーバーレイネットワークを構成させる。

不要となった VFSN は、VFSNM への destroy 要求により削除する。destroy は LINP を削除しリソースを開放することで実現される。

既に存在する VFSN に対してあとから SN を追加 (add)、削除 (delete) する操作も、同様に VFSNM を経由して行なう。SN の add 要求では、SN に対応する vNode を生成後に vSink を生成し、オーバーレイネットワークに参加させる。SN の delete 要求では、対応する vSink をオーバーレイネットワークから離脱させたのち、vNode を削除する。

join / leave 操作

SN が自律的に VFSN へ参加、離脱する形態では、まず、各 VFSN に SN に対応する vNode が割り当てられていることが前提となる。また、VFSN へ参加するには、SNO 間の合意により、対象とする VFSN に既に参加している SN に対応する vSink の情報を得ている必要がある。

join 操作では、他の vSink または、VFSNM から vSink のプラグインソフトウェアを入手した上で、SN に対応する vSink を vNode 上に生成し、既知の vSink を起点に、または、VFSNM を経由してオーバーレイネットワークに参加する。また、必要に応じて、VFSNM にメタデータを登録する。

SNO が SN のリソースを離脱させる leave 操作では、対応する vSink をオーバーレイネットワークから離脱させる。

unite / unite-set / separate 操作

unite 操作は、結合の対象となる VFSN それぞれに add され、VFSN 間のメッセージ転送を受け持つ IX ノード (Inter-overlay eXchange Node) を生成させることで実現することを基本とする。また、結合の対象となった VFSN に対応する LINP は、結合後もそのまま維持する。

IX ノードは、VFSNO の管理下に置き、VFSNM からの指示により、VFSN プラットフォーム上に IX ノードを生成する (図 5)。また、IX ノードには、接合対象となる VFSN に関するメタデータを、VFSNM から送信し管理させる。

unite の組み合わせごとに IX ノードは 1 つ以上動作させる必要がある。IX ノードは、VFSN プラットフォーム上のどの物理ノードで動作しても構わないが、いずれで動作すると

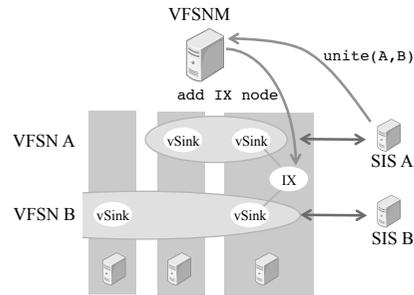


図5 unite 操作

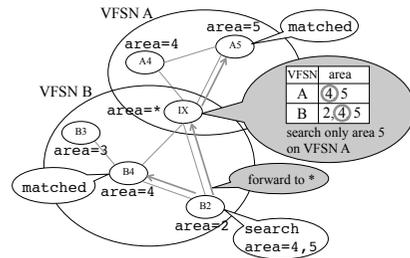


図6 IX ノードの実現

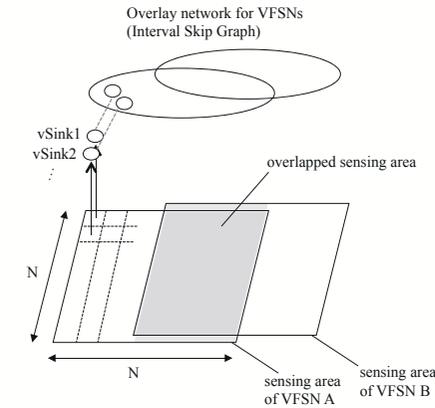


図7 アプリケーションの構成

効率的となるかは、流通するデータ量やオーバーレイネットワークの構造に依存する。unite の対象となる VFSN 間のメッセージの流通は、すべて IX ノードを経由する。従って、IX ノードによって、VFSN 間のメッセージの流通を制御できる。unite-set 操作は、メッセージの流通において、どの VFSN から、どの VFSN へのメッセージの転送を許可するかを、IX ノードに設定する。また、separate 操作では、全ての関連する IX ノードを delete する。

3.4 IX ノードの実現

VFSN のオーバーレイネットワークは、unite された別 VFSN へ転送する必要があるメッセージを IX ノードへ送信し、転送させる機能を持つ必要がある。

汎用的な実現法として、IX ノードが全てのメッセージを受信できる動作をするよう、オーバーレイネットワークを拡張する方法が考えられる。IX ノードが全てのメッセージを受信できるようにする方法として、オーバーレイネットワークに、全てのノード検索条件にマッチする Wildcard キーを導入する方法がある。

Wildcard キーは、オーバーレイネットワークによって実現方法は異なる。たとえば、Skip Graph¹⁹⁾では、キーの比較を行なう比較演算において、Wildcard キーを特別扱いし、キーを指定したノード検索を実行する際に、メッセージ転送先の一つとして加えるという方法で実現できる。このとき、IX ノードは、別の VFSN へメッセージを転送する際、受信元の VFSN で既に要求にマッチしたセンサーや観測値データの要求に回答していることが、メタデータを用いて判断できる場合、転送しなくて良い。例えば、図6においては、VFSN B において既にクエリに回答した area 4 については、VFSN A で回答する必要がないため、VFSN A 向けには、area 5 のみのクエリが転送されている。

4. 大規模アプリケーションによるシミュレーション評価

前章で示した設計に基づく VFSN プラットフォームにおいて、大規模アプリケーションとなる VFSN を動作させるシミュレーション評価を行なった。

4.1 アプリケーションの構成

アプリケーションとして、TSAR⁹⁾と同様の構成を取る VFSN をシミュレートした。TSAR は、Interval Skip Graph と呼ばれる構造化オーバーレイネットワークを構成し、一次元の範囲属性を持つセンサーへのアクセスを、ノード数 n に対しメッセージ数 $O(\log n)$ で実行可能とするシステムである。本シミュレーションでは、TSAR において、指定されたエリア内に設置された SN から観測値データを収集可能とするため、各 SN の担当する領域を、空間充填曲線 Z-ordering で表現し、エリアを指定した検索を行なえるアプリケーションを構成した。また、IX ノードに対応できるように、Wildcard キー拡張を施している。

図7は、本シミュレーションの動作環境を示している。図のとおり、一部の共通の観測領域をもつ2つの VFSN が存在し、それぞれが観測領域を縦横 N 個で格子状に分担する SN を収容する環境(すなわち、 $N = 10$ のとき、100 の SN が参加している)を用いている。

この構成において、 $N = 0 \sim 100$ に変化させた場合のシミュレーションを行なった。最大で各 VFSN について $N=100$ すなわち、 $100 \times 100 \times 2 = 20,000$ の SN が含まれる構成となる。

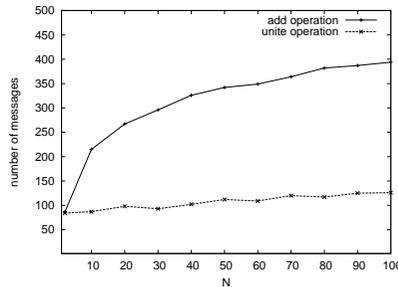


図 8 add, unite のコスト

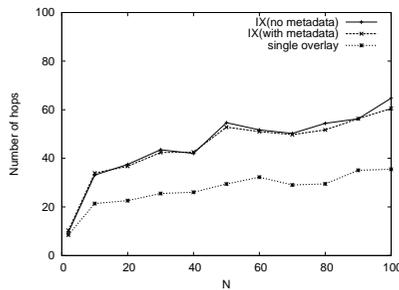


図 9 ノード検索遅延

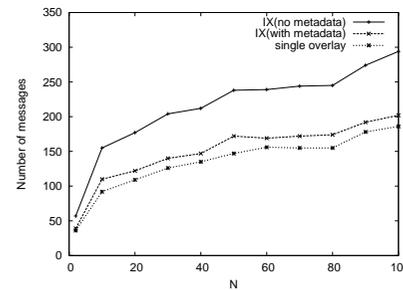


図 10 ノード検索コスト

4.2 シミュレーション評価

図 8 は、add および unite のコスト (ノード間転送メッセージ数) を示している。‘add operation’ は、オーバーレイネットワークへ 1 つ vSink を加える際にかかったコストの平均を示している。‘unite operation’ は、VFSN A と B が既に構成されている状態において、unite 操作をした場合のコストを示している。

add 操作は、Interval Skip Graph へ vSink の担当領域に相当する Z-ordering で表現されるキーを追加する操作となっている。1 つの vSink が複数のキーを保持する場合があるが、ノード数の増加に対するコストは、Interval Skip Graph の特徴により、対数関数的な増加に抑えられる。unite 操作では、IX ノードのために追加された Wildcard キーの追加が各 VFSN に対して行なわれる。この操作は、2 つの Skip Graph 構造に対して、それぞれキーを 1 つずつ追加する操作で実現できるため、コストは小さく、かつ対数関数的な増

加に抑えられている。

図 9, 10 は、ランダムな位置、大きさ (最大値は SN 4 個の担当領域) のエリアを指定した検索を 1000 回、ランダムな vSink から発行した場合に、目的とする vSink に到達するまでに要した最大遅延 (ホップ数) および、コストの平均値を示している。‘IX (no metadata)’ は、VFSN A と VFSN B を unite により結合した状態で、IX がメタデータを持たず、全ての検索要求を他方へ転送する場合を示している。‘IX (with metadata)’ は、各 VFSN の観測領域に関するメタデータを持ち、重複する領域へは検索要求を転送しない場合である。また、‘single overlay’ は、VFSN A と VFSN B の観測領域をカバーするノードのみでオーバーレイネットワークを構成する場合の最大ホップ数を示している。‘single overlay’ が、同ノード数での理想値となる。

最大ホップ数は、メタデータあり、なしにかかわらず、IX 経由で 2 つの VFSN を横断して目的ノードへ到達した場合のホップ数となるため、ほぼ同等となる。一方、コストについては、メタデータを用いる場合は抑制されており、この環境では理想値と遜色ないコストとなっている。また、いずれについても、一つの構造化オーバーレイネットワークを構成する場合と同様に、ノード数の増加に対するホップ数、コストが unite 操作後の VFSN においても対数関数的な増加に留められることが確認できる。

5. 関連研究

複数のセンサーネットワークを P2P アーキテクチャで統合する FSN に関する研究は、FSN のための構造化オーバーレイネットワークの構築方法⁹⁾ や、センサーネットワーク間の異種性の解消に関する検討⁶⁾⁻⁸⁾ に注力されてきた。TSAR⁹⁾ は、Skip Graph¹⁹⁾ を拡張して、センサーネットワーク上の観測値や属性に基づく分散管理を行なう。X-Sensor⁸⁾ は、FSN において、FSN 上の観測値データの処理やクエリをモバイルエージェントを用いて行なっている。GSN^{6),7)} は、処理結果を用いて新たに定義したストリームを仮想センサーとして定義できる。また、異種のセンサーネットワークのメタデータを生成する方法²⁰⁾ の検討もなされている。しかし、上記はいずれも、センサーネットワーク間を接続する際、物理リソースはすべて共有され、本稿で述べた VFSN のようなリソース分離の概念を検討していない。

また、センサーネットワークの物理リソースを仮想化する試みとして、VSN²¹⁾ がある。これは、ひとつの無線センサーネットワークの部分集合のみを用いればよい複数のアプリケーションにおいて、センサーの物理リソースを共有する仕組みである。VSN は、SN 単体の仮想化方法であり、SN 間の接続は考慮していない。VFSN は、VSN を FSN に拡大

するコンセプトと位置付けられる。

6. おわりに

本稿では、膨大なセンサーネットワークのリソースを共有するアプリケーションを、品質を保った上で柔軟に実現可能とする仮想統合センサーネットワーク (VFSN) のコンセプトを新たに提案した。また、VFSN の実現法として、ネットワーク仮想化技術を用いて、仮想化されたセンサーネットワークのシンク (vSink) 間でオーバーレイネットワークを動的に構成・再構成可能とするプラットフォームの設計を示した。

今後の課題として、本稿で示した VFSN 実現方式に関して、仮想シンクやメタデータの記述方式、IX ノードによるオーバーレイネットワーク接合に関して、IX ノードの最適な配置や、複数の IX ノードを用いて負荷分散を実現する方式などの検討があげられる。

現在、我々は本稿の設計に基づき、VFSN プラットフォームの実機プロトタイプの実装を進めている。今後、上記であげた課題を検討しつつ、実用的な仮想統合センサーネットワークの実現を目指す。

謝辞 本研究に関し様々な観点より議論頂いた大阪大学村田正幸教授、(独) 情報通信研究機構ネットワークシステム総合研究室のメンバ諸氏に深く感謝する。

参考文献

- 1) A. Kansal, S. Nath, J. Liu, and F. Zhao, "SenseWeb: An Infrastructure for Shared Sensing," *IEEE Multimedia*, Vol. 14, pp. 8–13, 2007.
- 2) C.L. Fok, G.C. Roman, and C. Lu, "Towards A Flexible Global Sensing Infrastructure," *In ACM SIGBED Review, Special issue on the workshop on wireless sensor network architecture 2007*, Vol. 4, pp. 1–6, 2007.
- 3) K. Chang, N. Yau, M. Hansen, D. Estrin, "A Centralized Repository to Slog Sensor Network Data," *In Proc. of Intl. Conf. on Distributed computing in Sensor Systems*, 2006.
- 4) M. Balazinska et al. "Data management in the world-wide sensor web," *IEEE Pervasive Computing*, 2007.
- 5) U. Hague, "pachube," available from (<http://www.pachube.com/>).
- 6) K. Aberer, M. Hauswirth, and A. Salehi, "Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks," *In Proc. of the Intl. Conf. on Mobile Data Management (MDM 2007)*, pp. 198–205, 2007.
- 7) K. Aberer, M. Hauswirth, and A. Salehi, "Global Sensor Networks," *IEEE Communications Magazine, special issue on Advances in Service Platform Technologies*

for Next Generation Mobile Systems, 2006.

- 8) A. Kanzaki, T. Hara, Y. Ishi, T. Yoshihisa, Y. Teranishi, and S. Shimojo, "X-Sensor: Wireless Sensor Network Testbed Integrating Multiple Networks," *Wireless Sensor Network Technologies for the Information Explosion Era Studies in Computational Intelligence*, Vol. 278, pp. 249–271, 2010.
- 9) P. Desnoyers, D. Ganesan, and P. Shenoy, "TSAR: A Two Tier Sensor Storage Architecture Using Interval Skip Graphs", *In Proc. of the 3rd Intl. Conf. on Embedded Networked Sensor Systems*, pp. 39–50, 2005.
- 10) "Framework of Network Virtualization for Future Networks," Recommendation ITU-T Y.3011, 2011.
- 11) A. Nakao, "Network Virtualization as Foundation for Enabling New Network Architectures and Applications," *IEICE Trans. Communications*, Vol. E93-B, No. 3, pp. 454–457, 2010.
- 12) "GENI," available from (<http://www.geni.net/>).
- 13) D. Raychaudhuri et al., "Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols," *In Proc. of Wireless Communications and Networking Conference*, Vol. 3, pp. 1664–1669, 2005.
- 14) K. Nakauchi, K. Ishizu, H. Murakami, A. Nakao, and H. Harada, "AMPHIBIA: A Cognitive Virtualization Platform for End-to-End Slicing," *In Proc. of the IEEE Intl. Conf. on Communications (ICC)*, pp. 1–5, 2011.
- 15) L. Peterson, T. Anderson, D. Culler, and T. Roscoe, "A Blueprint for Introducing Disruptive Technology into the Internet," *ACM SIGCOMM Comput. Commun. Rev.*, Vol. 33, Issue 1, pp. 59–64, 2003.
- 16) A. Nakao, R. Ozaki, and Y. Nishida, "CoreLab: An Emerging Network Testbed Employing Hosted Virtual Machine Monitor," *In Proc. of the 2008 ACM CoNEXT Conference*, pp. 73:1–73:6, 2008.
- 17) "Ubiquitous Green Community Control Network Protocol," 1888-2011, IEEE Standard, 2011.
- 18) "PIAX," available from (<http://www.piax.org>)
- 19) J. Aspnes, and G. Shah, "Skip graphs," *ACM Trans. Algorithms*, Vol. 3, 4, Article 37, 2007.
- 20) H. Jeung et al. "Effective Metadata Management in Federated Sensor Networks," *In Proc. of IEEE Intl. Conf. on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2010)*, pp. 107–114, 2010.
- 21) A. P. Jayasumana, Q. Han, T. H. Illangasekare, "Virtual Sensor Networks - A Resource Efficient Approach for Concurrent Applications," *In Proc. of Fourth Intl. Conf. on Information Technology (ITNG '07)*, pp. 111–115, 2007.