耐災害NDNベース分散型アプリケーションの情報伝達性能に対するユーザ移動モデルの影響評価

小河原 健生[1] 川原 圭博[1] 浅見 徹[1]

概要：災害時においては，安否確認や被災状況など周囲状況の的確な把握が被害拡大を防ぐ上で重要となる．しかしながら，そのような情報の伝達を担う携帯電話網などの通信インフラは大規模な災害時には機能しないことが多い．その原因の一つは，現在の情報インフラは，サービスを提供および管理するサーバが利用できなくなると，基地局といった末端設備が健在でも機能しなくなる，センター集中型の構造をとっていることである．本研究では，災害時に断線などによりサーバ類が利用できなくなった場合にも，基地局など残存した設備を利用して局所的なネットワークを構築し，サービスを継続させることを目指す．そのために，Named Data Networking(NDN) により自立分散型ネットワークを構築し，Hierarchical ID-Based Encryption(HIDE) による自律認証を行うことで，災害時にも通常時同様に利用できるサービスおよびアプリケーションを構成した．本稿では，ユーザの行動モデルがアプリケーションの情報伝達性能へ与える影響についてシミュレーションにより調査した．

User Movement Effect on Dissemination Performance of a Disaster Tolerant NDN-based Distributed Application

Takeo Ogawara[1] Yoshihiro Kawahara[1] Tohru Asami[1]

1. はじめに

大規模な災害は，社会に深刻なダメージを与える．2011年3月11日に発生した東日本大震災においてもその被害は甚大であり，人々に大きな影響を与えた．携帯電話網といった通信インフラについてもその被害は大きく，災害直後にはサービスを利用できない地域が多く見られた．

そのような災害時には，被災状況や家族および知人の安否確認など，的確な周囲状況の把握が被害の拡大を防ぐ上でも重要となる．しかしながら，前述のとおり，情報の収集や共有を担う携帯電話網といった情報インフラは災害時ほとんど機能しない．その原因の一つとして，携帯電話網のような情報通信インフラがコアネットワーク内のサーバ群による中央集中型の管理構成をとっていることがあげられる．サービスの認証，名前解決，アプリケーション等を提供するサーバが，どれかひとつでも利用不可能になると，ユーザはサービスを利用することができない．そのため，サーバや基地局自体が損壊した時サービスを提供できなくなることはもちろんであるが，設備自体に被災がない場合にも，基地局とサーバを結ぶ伝送路が被災することにより断線することによってもサービスは停止してしまう，実際，東日本大震災において，停電を除けば基地局が停波した原因の約6割はサービスを提供しているサーバと基地局をつなぐ経路が断たれたことによるものであった．この意味では，電話網もインターネットも同様であり，現行のシステムでは災害時にもサービスを提供することは困難である．災害時において携帯電話の利用に対する需要は高かったが，上述の原因により基地局自体は利用可能であった場合にもユーザはサービスを利用できない状況であった [2]．

本研究では，災害時に，インフラが中央集中管理を行なっているサーバ群から切断された状況下においても，末端設備を利用して局所的なネットワークを構築することを目指す．グローバルな接続ができなかったとしても，局所的な情報共有ができることは，大いに役立つはずである．たと

[1] 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology, The University of Tokyo

えば，避難所から地元の警察署，病院，消防署などへ連絡し救援を求めることができる．避難所間で情報共有を行うことで，離れ離れになってしまった家族や友人間で安否や周囲の被災状況の確認を行うこともできる．たとえ局所的なネットワークであっても，被災者の救援や被害拡大の防止に多いに役立つことが期待される．本研究では特に，そのような情報共有を行うアプリケーションについて考える．

これを実現するためには，各通信ノードが物理層からアプリケーションまでを統合して，自律的に動作し，物理的なダメージによらずサービスを提供できるアーキテクチャが重要になる．そこで，本研究では，Named Data Networking(NDN) を用いて自律分散型のネットワークを構成し，暗号化方式に Hierarchical ID-Based Encryption(HIDE) を用いてユーザによる自律的な認証を行う [3], [4].

2. 関連研究

NDN は，データの ID(Name) によって通信を開始するネットワークアーキテクチャである．Name はデータに対し割り当てられるもので，データを保管しているノードの位置に依らず不変である．これは，NDN では，データにアクセスするためにノードを識別する必要がないことを意味する．現在のインターネットアクセスでは，URL からドメイン名を取得し，Domain Name Server を利用し，Dynamic Host Configuration Protocol Server などによって割り当てられた IP アドレスを取得してデータにアクセスを試みる．そのため，Domain Name Server や Dynamic Host Configuration Protocol Server などによってネットワーク内のノードが管理されている必要がある．P2P ネットワークについても同様に，各ノードに固有の識別子を与え，データの ID からデータの位置を解決するサーバを利用することにより通信を行う．NDN ではそのようなネットワークの管理や名前解決を行うサーバが必要なく，災害時のサーバへのアクセスが期待できない状況で利用するのに適している．

NDN におけるデータへのアクセスは以下の様な流れで行われる．データを要求するノードは，Name をその中に含む Interest Packet を送信する．各ノードはそれぞれが保持している Forwarding Information Base を参照して，Name からパケットのフォワーディング先を判断し，データを持つノードまで Interest Packet をフォワーディングする．この時，Interest Packet をフォワーディングしたノードは，Interest Packet を受け取った通信インタフェース (Face) を Pending Interest Table に記憶する．Name に対応するデータは Data Packet として返信される．Data Packet のルーティングは Pending Interest Table を参照することにより行われる．Pending Interest Table には Interest Packet のたどったルートが記憶されている．そのため，Interest Packet のルートの逆順を通って Interest Packet の送信元まで Data Packet を返送することができる．Forwarding Information Base が正しく設定されていれば，各ノードが自律的に動作して，データを要求するノードとデータを持つノードが中継ノードを通じて通信を行うことができる．ネットワークを中央集中管理するサーバなしで，エンドノードと中継ノードのみによって通信が完了する．エンドノード間が物理的に通信可能であれば情報共有が可能である．ゆえに，災害地域において残された設備が自律的に動作して物理的に通信可能な範囲で局所的なネットワークを構築し，情報共有を行うことが可能となる．

NDN によって，局所的なネットワーク，たとえば，基地局とそれに接続する端末からなるネットワーク内において，情報共有が可能となる．しかしながら，その場合，情報共有が可能な範囲は同一の基地局に接続している端末間のみであり，その効果は限定的である．そこで，分断されたネットワークとその間を移動するユーザを活用して，Delay Torelant Network(DTN) を構成し接続することで，より広範囲での情報共有を目指す．DTN を構成するために NDN 上で動作するマルチユーザチャットシステムの Chronos の切断耐性を利用した [5]．これにより，災害時の不安定なアクセスパスを利用して広範囲で情報共有が可能となる．

また，普段から利用しているサービスやアプリケーションを災害時にもシームレスに利用するためには，ユーザや端末の認証が重要な要素の一つとなる．災害時においても，認証を行うことはなりすましやデマ情報の拡散を防ぐ上でも重要である．NDN では，データの送信元やデータに改竄がないことを認証する仕組みがそのパケット構造内に含まれている．Data Packet はそのデータと Name の領域に電子署名を付与することが必須になっている．署名を検証することによりデータの送信元とデータの信頼性を確認することができる．これによりデータをアプリケーションサーバに集め，データの送信元やその信頼性を確保する必要がなくなる．ゆえに，ユーザ間で直接データの授受を行った場合にもその信頼性を保証することができる．今回の想定環境であるグローバル・ネットワークから分断された局所的なネットワーク内でも，データの信頼性を保ちつつやり取りを行うことができる．本研究では，署名には HIDE を用いる．

HIDE は，ID Based Encryption(IBE) における ID と Key Generation Center(KGC) に階層構造を導入したものである [6]．ID Based Encryption(IBE) では，ユーザの秘密鍵は KGC によって生成される．まず，KGC によって鍵生成のための共通パラメータおよびマスター秘密鍵が生成される．この時，マスター秘密鍵は KGC によって秘匿され，共通パラメータは公開される．ユーザの秘密鍵は，マスター秘密鍵，共通パラメータとユーザの ID から生成される．ユーザはそれぞれの ID に対応したユーザ秘密鍵と共通パラメータを保持する．暗号化は，共通パラメータと

ユーザ ID によって行われ，復号は，ユーザ秘密鍵によって行われる．

IBE の利点の 1 つとして，公開鍵認証局が不要であることがあげられる．

Public Key Infrastructure(PKI) では，公開鍵は一般にユーザの Identity と関連性のないものである．そのため，ユーザが利用している公開鍵の正当性・信頼性を保証するためには，公開鍵認証局によって発行される公開鍵証明書が必要になる．また，暗号化を行う際には相手ユーザの公開鍵の取得が必要になる．それゆえ，災害時の利用を想定すると，公開鍵認証局へのアクセスが保証できないことや，暗号化を行う前に公開鍵を取得するというプロセスが必要になるために，PKI は利用できない可能性が高い．

一方 IBE は，共通パラメータとユーザ ID が公開鍵として扱われる．ユーザの Identity を表すユーザ ID をそのまま公開鍵となるため，その正当性・信頼性の保証は不要となる．共通パラメータは公開鍵のようにユーザ毎に異なるものではなく，すべてのユーザで共通のものである．そのため，共通パラメータの取得は一度きりでよく，災害時に KGC から分断されたとしても，災害前と同様に IBE を利用することができる．また，暗号化は相手の ID と自身が取得した共通パラメータのみによって可能である．そのため，共通パラメータは秘密鍵を取得した時に同時に取得され，ID もまた事前にユーザ間で共有されていることが想定すると，PKI のような公開鍵を取得するというプロセスは簡略化される．署名の検証を行うために頻繁に公開鍵を利用する必要がある本システムにおいて，この特徴は大きなメリットとなる．

しかしながら，ユーザが同一の秘密鍵を使い続けることは鍵の漏洩の危険性を高めるため，セキュリティ上の問題がある．また，ユーザが頻繁に鍵を更新するために，KGC へアクセスすることは KGC への負荷が高く困難である．そこで本研究では，特に HIDE を用いることによってユーザ自身によって秘密鍵を生成できるようにすることを提案する．HIDE は IBE における ID および KGC を階層構造にしたものである．これにより，ユーザ自身に KGC の役割を割り当てることが可能となり，ユーザ自身のみによって秘密鍵の生成が可能となる．ユーザが目的ごとに異なる秘密鍵を生成し利用することで，それぞれの秘密鍵の漏洩の可能性は大きく低減され，漏洩した場合のリスクも分離することができる．

以上の機構により，被災地域における孤立したネットワーク内で各ノードが自律的に動作して情報共有を行う．次節で詳細な機構について述べる．

3. システム構成

今回のシステムの概要について述べる．図 1 は被災地域におけるシステムの想定利用環境である．被災地において，基地局は被災によりコアネットワークから分断され，それぞれの基地局で孤立したネットワークを構成している．各ユーザが発信したメッセージは，そのユーザが接続している基地局内の他のすべてのユーザと共有される．各基地局内のメッセージを受信しているユーザが他の基地局の圏内へと移動することにより基地局間の情報共有を目指す．

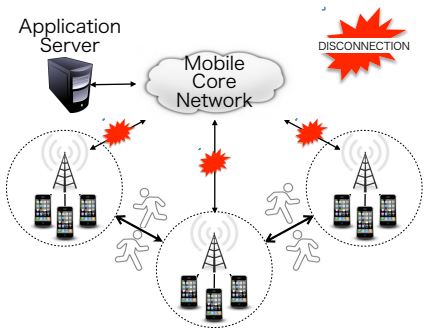


図 1　災害時の想定環境

まず，本システムで用いる HIDE の階層構造について述べる．次に，情報共有に利用するアプリケーションのプロトコルについて述べる．最後に，コアネットワークから分断された基地局が端末を認証するプロトコルについて述べる．

3.1 HIDE の階層構造

図 2 に今回のシステムで利用する HIDE の階層構造を示す．

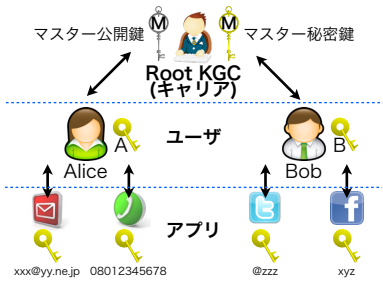


図 2　HIDE における階層構造

root KGC にはインフラやネットワークを提供するキャリアを想定する．キャリア (root KGC) は，そのサービスを利用しているユーザにユーザの ID を使ってユーザ秘密鍵を生成する．ユーザは，生成されたユーザ秘密鍵と共通パラメータを受け取る．同一のキャリアに所属するユーザ間では，共通パラメータも同一であり，KGC にアクセスすることなく自律的に暗号化および復号が可能となる．異なるキャリアに所属するユーザ間でも，相手キャリアの共通

パラメータを取得することにより，そのキャリアに所属する全ユーザと暗号化および復号のやりとりが可能となる．公開鍵毎に公開鍵証明書を取得して公開鍵の信頼性を確かめる必要のある PKI と比べて大きく処理が簡略化される．

二層目の KGC はユーザが担う．ユーザは KGC から受け取った鍵をマスター秘密鍵としてさらに鍵を生成する．ユーザは利用目的ごとに ID を設定し使い分けることができる．例えば，電話を利用するときは，KGC から受け取った鍵と電話番号からアプリケーションごとに対応するユーザ秘密鍵を生成し，通信の暗号化に用いる．通信の目的ごとに鍵を使い分けることができ，ユーザが鍵生成局から受け取った鍵を使用する頻度も減るため，鍵が漏洩する危険性が低くなる．

本システムでは，ユーザは，アプリケーションごとに秘密鍵を生成するために ID を設定する必要がある．現行の Facebook や Twitter といったアプリケーションを利用する場合，ユーザ毎に ID が割り当てられているため，ID の設定は現在利用されているものをそのまま利用することが出来る．また，ユーザの認証には ID とパスワードが用いられる．ユーザの ID と秘密鍵によって認証を行う本システムは，現行のシステムからスムーズな移行が可能である．

3.2 通信プロトコル

本システムで利用する Chronos を模したアプリケーションの処理の流れを示す．以下の図 4 から 6 は処理の流れを示すフローチャートである．図 4 アプリケーション全体のフローチャート，図 5 は Interest Packet を受信したときのアプリケーションの処理，図 6 は Data Packet を受信した時のアプリケーションの処理である．

アプリケーションの処理の流れを述べる上で重要となる Digest Log について先に述べる．Digest Log はメッセージリストとメッセージの Name から計算した Digest 値を紐付けるための機構である．図 3 に Digest Log の一例を示す．

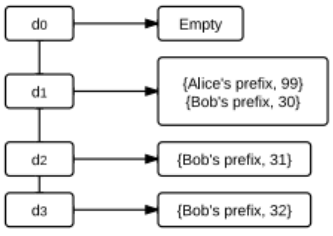


図 3 Digest Log の例 (from [5])

Digest Log にはメッセージの Name とそれから計算された Digest 値が記録される．初期状態では，Digest 値 d0=0 とメッセージが空の状態である．新たなメッセージが見つかるたびにそのメッセージの Name と Name から計算された Digest 値が追記される．Digest 値を Digest Log から参照することで，現在どのメッセージが取得されているか知ることができる．以下ではこの Digest Log を利用してメッセージ共有を行うアプリケーションの処理の流れについて説明する．

図 4 について述べる．

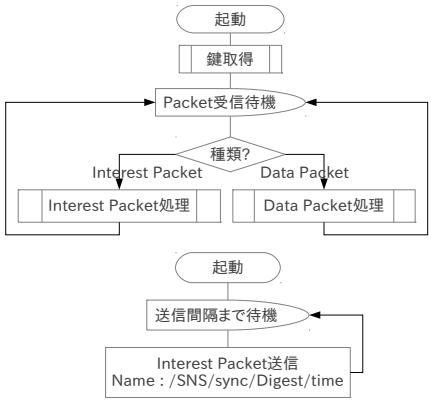


図 4 アプリケーションの処理

まずアプリケーションは，起動時にアプリケーションで使用する秘密鍵を取得する．提案アプリケーションでは HIDE の導入を考えており，アプリケーションを起動しているノード自身も KGC の 1 つとして振る舞う．ノードはノード内で使用する鍵を用途ごとに生成する．アプリケーション内で生成された Data Packet の署名にはこの秘密鍵を利用する．

図 4 に示すように，鍵取得後のアプリケーションの全体の処理としては主に二種類ある．1 つには定期的に sync Interest Packet を送信するものである．sync Interest は現在自身が保存している情報を周囲に知らせるものであり，その詳細は図 5,6 で述べる．この Packet により周囲のノードと保存している情報の同期をとり共有をはかる．もう 1 つはパケットが送られてくるまで待機し，受信した時にパケットを処理するものである．NDN では主に Interest Packet と Data Packet の二種類が存在する．それぞれの処理について以下で述べる．

図 5 は Interest Packet を受信した時の処理の流れである．本アプリケーションでは，chronos を模して，Interest Packet として sync Interest Packet,recovery sync Interest Packet およびメッセージに対する Interest Packet を用意した．

まず，sync Interest を受信した時，アプリケーションは Digest Log を参照し，Name に含まれるダイジェスト値が既知のものであるかを判断する．既知のものである場合，それ以降に受信した新規メッセージの Name を Data Packet として返し処理を終了する．未知のダイジェスト値であった場合，Data Packet として 0Byte のデータを返

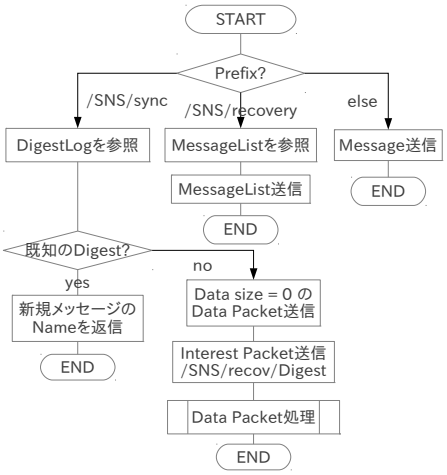


図 5　Interest Packet の処理

し，新規メッセージがわからないことを表す．また，未知のダイジェスト値の解決をはかるために，recovery sync Interest を送信する．

recovery sync Interest を受信した時，アプリケーションは自身が保持している Digest tree から各ユーザの状況を取得し Data Packet として返信する．ユーザによって発信されたメッセージの Name の構成は/Application ID/User ID/Sequence Nomber であるから，User ID と今までに受け取ったメッセージの Sequence Nomber の最大値をユーザの状況として返す．

メッセージに対する Interest Packet を受信した時，Application 内に保存されているメッセージを参照し，対応するメッセージを保存していれば，そのメッセージを転送する．

図 6 は Data Packet を受信した時の処理の流れである．

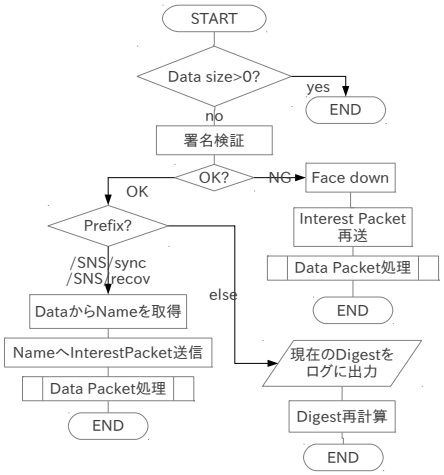


図 6　Data Packet の処理

Data Packet を受け取った時，そのサイズが 0Byte であれば，通信可能な範囲に新規メッセージがないことを意味するため，処理を終了する．Data Packet のサイズが 0Byte より大きければ，まずその署名の検証を行う．署名の検証に失敗した場合，信頼できないデータであるため，そのデータの取得を諦め処理を終了する．署名の検証に成功した場合，データの Name 毎に処理を行う．

sync もしくは recov を Name の要素として含む場合，そのデータには新規メッセージに関する情報が含まれている．この時，新規メッセージの取得のために，それぞれのメッセージに対する Interest Packet を送る．メッセージを取得した後は，Digest Log を更新し処理を終了する．

sync もしくは recov 以外のデータはメッセージとして扱う．メッセージを受信した時，Digest tree を更新しその前のダイジェスト値を Digest Log に記録する．

以上の流れにより，災害時の分断された局所的ネットワーク内においても，サーバに頼らず所属するユーザ間で情報の共有を行う．

3.3　認証プロトコル

基地局とユーザ間の相互認証は，3G/LTE で HLR/AuC 間で相互認証および鍵生成に用いられる Authentification and Key Agreement(AKA) を踏襲したものを実行することにより行う [7]．今回は相互認証のみを目的とするため，認証部分の処理のみを示す．以下に詳細な処理の手順を示す．

(1) ユーザの端末から基地局に対して Authentication data request として Interest Packet を送信する (Name:/hoge/hoge/SQN)
(2) 基地局は秘密情報 K_{eNB} とシーケンス番号 SQN から認証リクエストを生成する．
- ランダムチャレンジ RAND
- 予想応答 XRES=F2(K_{eNB},RAND)

(3) K_{eNB} と RAND をユーザの公開鍵で暗号化しユーザに Data Packet として送信する．
(4) User authentication response を要求する Interest Packet を送信する．(Name:/hoge/hoge)
(5) ユーザは Data Packet の署名情報から送信元の基地局を認証する．
(6) ユーザは秘密鍵により K_{eNB} および RAND を入手し，応答 RES を計算する．
(7) ユーザは応答 RES を基地局の公開鍵により暗号化し Data Packet として送信する．
(8) 基地局は秘密鍵により RES を入手し予想応答 XRES と比較することによりユーザを認証する．

以上によりユーザと基地局間の相互認証が完了する．両者の公開鍵と秘密鍵はともにキャリアによって生成されたものであり，マスター公開鍵は共通であるため，同一のキャリアのサービスを受領・提供する関係にあることが認証される．

アプリケーションを利用するユーザ間の認証について述べる．アプリケーションの利用では，データの送信者の認証のみを行い，データの受信者 (Interest Packet の送信者) の認証は行わないものとする．NDN では，データはどこのノードにキャッシュされているものでも同一のものとして扱うため，データがどのノードに受信されても問題はない．データの送信元はなりすましや誤情報の氾濫を防ぐために必要となる．送信元の認証は NDN の枠組み内で，Data Packet に付与された署名により可能となる．正しい署名のなされていない，送信元不明のデータについては，受信したユーザによって破棄される．

4. 情報伝達性能評価

提案システムの情報伝達性能についてシミュレーションにより評価を行なった．シミュレーションソフトウェアは ns-3 上に実装された NDN のシミュレータである ndnSIM を利用した [8]．ns-3 は ns-3.16 を利用し，ndnSIM は 2013 年 1 月 7 日時点で最新のものを用いた．実行環境は，CPU:Intel(R) Xeon(R) CPU E5540 @ 2.53GHz，メモリ容量 23.6GB のマシンである．Name は次のように定義した．

- sync Interest： /SNS/sync/RootDigest
- recovery sync Interest： /SNS/recov/UnknownDigest
- message： /SNS/UserID/SeqNo

4.1 シミュレーションモデル

モデルとして図 7 の世田谷区の経堂・桜上水付近の避難所を考え，避難所の間を移動するユーザを想定してシミュレーションを行った．

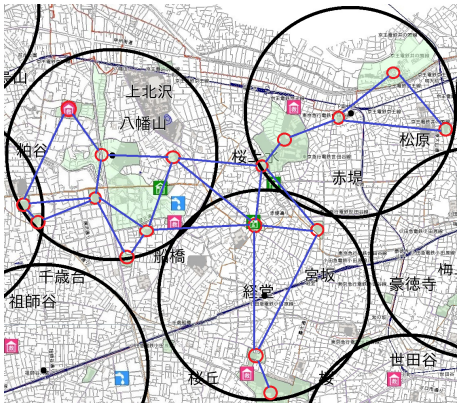


図 7 トポロジーモデル

黒い円は基地局を介して通信可能な範囲であり，半径 1km である．赤い円で避難所を示し，それぞれにモバイル端末を所持したユーザが存在する．初期状態では各避難所に 100 人のモバイル端末を持ったユーザが避難している．ユーザは，一定の平均移動間隔をもつポアソン過程に従って避難所間を移動する．移動する避難所先は，現在いる避難所と青い線で結ばれた避難所のいずれかにランダムに移動する．ユーザの移動は徒歩を想定し，Google マップによる推奨ルートを時速 4 km で移動するものとした．ユーザが基地局の通信可能範囲にいる限りは，同一の基地局に所属するユーザと基地局を介して通信可能である．平成 24 年度版情報通信白書によると平成 22 年度において，加入者による総通信回数 608.7 億回，契約数 11953.5 万加入であるため，一契約あたりの平均発信間隔は約 1.3 時間である [2]．またドコモのデータによると東日本大震災直後には通常時の約 60 倍の発信があったと推定されることから，今回の各端末のメッセージの平均送信間隔は約 17 分とした [9]．

4.2 シミュレーション結果

シミュレーションの結果を以下に示す．前述のパラメータを用いて，その他のパラメータとメッセージ到達時間の関係について調査する．

メッセージ到達時間を以下の通り定義する．まず，メッセージ到着率を，メッセージを受信したノード数/全ノード数により定める．すべてのメッセージについてメッセージ到着率を算出する．メッセージ到着率が 95%を初めて超えた時間をメッセージ配送時間とし，その平均を平均配送時間と定める．

Digest Log のサイズはメッセージ到達までの時間にほとんど影響を与えないと考えられる．Digest Log のサイズが大きいほど，受信した Digest 値を解決できる可能性が高まる．しかしながら，受信した Digest 値が Digest Log に見つからなかった場合でも，recov Interest Packet を送信することにより新規メッセージを探すことができる．そのため，大きな Digest Log の領域を確保しても，伝達性能の向上は期待できない．以下のシミュレーションでは Digest Log のサイズを 50 としてシミュレーションを行う．

Content Store のサイズはメッセージ到達時間にほとんど影響を与えないと考えられる．アプリケーションでは，Content Store とは別にアプリケーション内にメッセージを保存している．これによって，Content Store のメッセージが破棄されたとしても，アプリケーション内から Content Store に必要なメッセージを提供できる．ゆえに．Content Store のサイズが小さく，メッセージが破棄されやすくても，伝達性能には影響はない．

図 8 に平均配送時間と sync Interest Packet の送信間隔との関係を示す．図 9 に平均配送時間とユーザの移動間隔の関係を示す．

図 8 から，平均配送時間は sync Interest Packet の送信間隔に対してほぼ線形に増加することがわかる．また，図 9 からは，平均配送時間はユーザの移動間隔に対して指数関数的な増加傾向を見せることがわかる．ゆえに，図 9

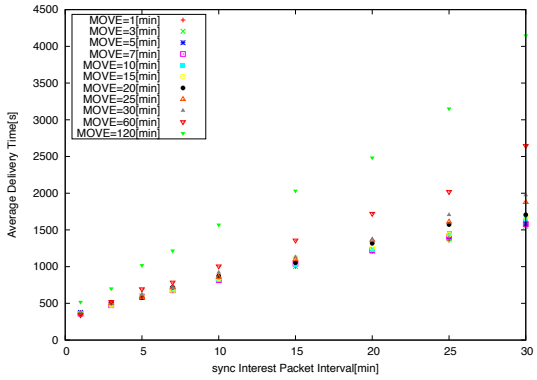


図 **8**　sync Interest Packet の送信間隔と平均配送時間

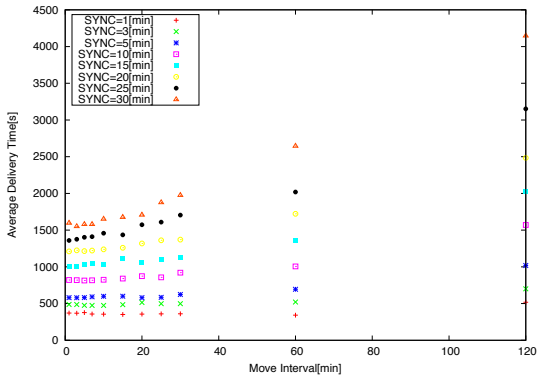
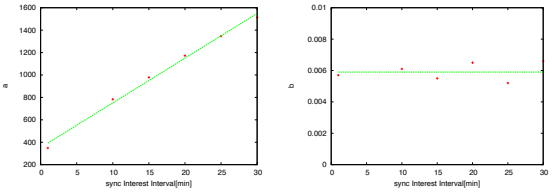


図 **9**　移動間隔と平均配送時間

は $y = ae^{bx}$ という形の式で近似可能である．図 10(a) および図 10(b) に推定したパラメータ a，b と sync Interest Packet の送信間隔との関係を示す



(a) 定数 a　　(b) 定数 b

図 **10**　近似値

図 10(a) から，定数 a は sync Interest Packet の送信間隔に対して線形に増加することがわかる．線形近似を行うと，$y = 39.89x + 352.3$ がえられた．図 10(b) から，定数 b はほぼ一定の値をとっており，その平均値は 0.0059 であった．

以上から，すべてのユーザが避難所間を移動する場合，その平均配送時間 $T_{100\%}$ は sync Interest Packet の送信間隔 t_1 とユーザの移動間隔 t_2 を変数として，次の式で表すことができる．

$$T_{100\%} = f_{100\%}(t_1, t_2) = (39.89t_1 + 352.3)\mathrm{e}^{0.0059t_2}$$

これは，平均配送時間が，sync Interest Packet の送信間隔に対して線形に増加，ユーザ移動間隔に対して指数関数的に増加をしたことに一致する．図 11 に近似関数とシミュレーションの結果の関係を示す．

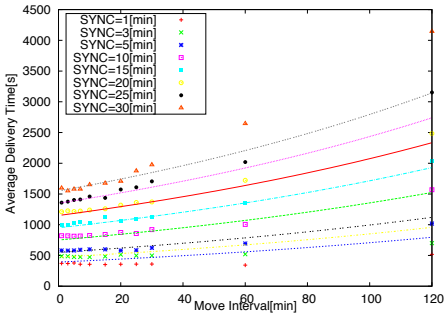
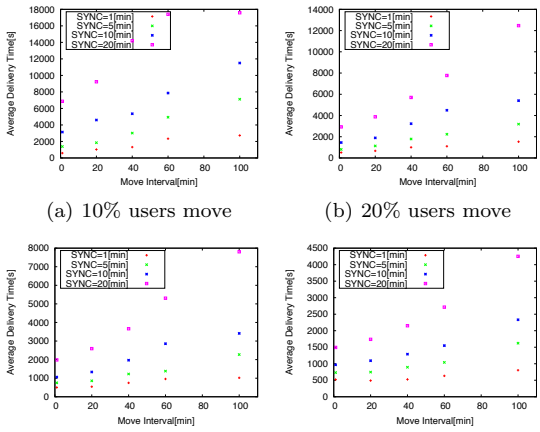


図 **11**　シミュレーション結果と近似関数

配送時間の長い領域では近似関数に一致しない点も存在するが，近似関数はほぼシミュレーションの結果に合致する．

上記の結果ではすべてのユーザが移動する．実際には移動するユーザは一部であり，避難所に留まったままのユーザも多いはずである．そこで移動するユーザの割合を変えて上記と同様のシミュレーションを行った．図 12 にその結果を示す．



(a) 10% users move　　(b) 20% users move

(c) 30% users move　　(d) 50% users move

図 **12**　Average Delivery TIme v.s. Move Interval

それぞれのグラフは，10%(図 12(a))，20%(図 12(b))，30% (図 12(c))，50%(図 12(d)) のユーザが移動した場合の，ユーザの移動間隔と平均配送時間との関係である．それぞれの系列は sync Interest Packet の送信間隔によって異なる．すべてのユーザが移動する場合と同様に，到着時間は移動間隔に対して指数関数的に増加する傾向を示している．これらの結果について，すべてのユーザが移動する場合と同様に近似式の推定を行った．以下がその近似式である．

$$T_{10\%} = f_{10\%}(t_1, t_2) = (332.02t_1 + 76.23)\mathrm{e}^{0.0126t_2}$$

$$T_{20\%} = f_{20\%}(t_1, t_2) = (130.69t_1 + 322.65)\mathrm{e}^{0.0135t_2}$$

$$T_{30\%} = f_{30\%}(t_1, t_2) = (76.471t_1 + 401.49)\mathrm{e}^{0.0125t_2}$$

$$T_{50\%} = f_{50\%}(t_1, t_2) = (50.721t_1 + 416.95)\mathrm{e}^{0.00823t_2}$$

移動ユーザの割合を x%として，メッセージの到達時間 $T_{x\%}$ は以下の式で近似できる．

$$T_{x\%} = f_{x\%}(t_1, t_2) = (\alpha(x)t_1 + \beta(x))\mathrm{e}^{\gamma(x)t_2}$$

α, β, γ と移動ユーザの割合の関係を図 13 に示す．

図 13 に移動ユーザの割合と係数 α, β, γ の関係を示す．α, β は 10%から 20%で大きく変化し，それ以上では大きな変化は見られない．移動するユーザの割合が大きくなると，t_1 が到着時間に与える影響が小さくなることが伺える．一方，γ は 30%から 50%の間で大きく変化している．50%を超えるユーザが移動したとしても，50%のユーザが移動する場合と，到着時間は大きく異なることはないことを示している．

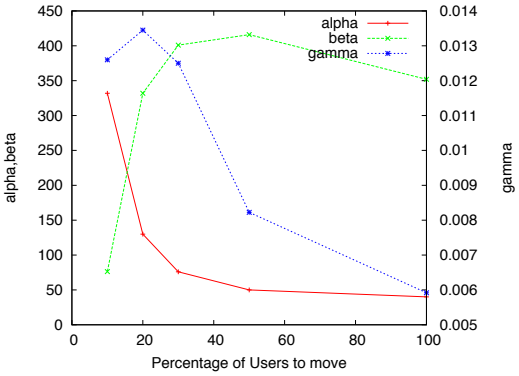


図 **13** 近似値

災害時には，電源が確保できず，ユーザは利用している端末の電力をなるべく消費しないようにすることが予想される．そこで，ユーザが移動している間は電源を OFF にし，避難所に到着しとどまっている間のみ電源を ON にするような行動をとった場合のメッセージ配送時間についてシミュレーションを行った．移動頻度と移動している時間，すなわち，ユーザが電源を切っている時間の割合の関係を図 14 に示す．

次に，移動中に電源を切る行動をとった場合，電源を入れたままで移動する場合と比較してどれほど到達時間が増加するかを図 15 に示す．

図 14 と図 15 を比較すると，電源を切ることによる消費電力の削減効果よりも，配送時間の増加率のほうが大きい．消費電力の削減により稼働時間がのびたとしても，それ以上に配送時間が伸びてしまっては無意味である．移動中には電源を切らないほうが，アプリケーションとしての性能は高いことがわかる．

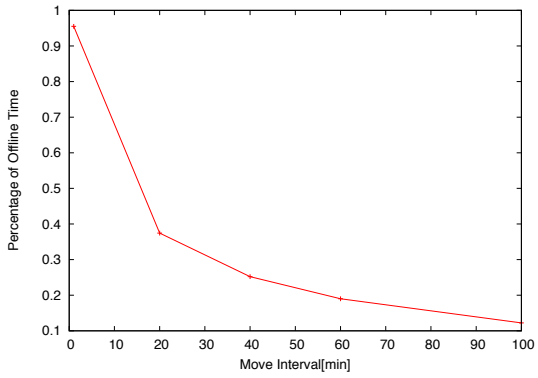
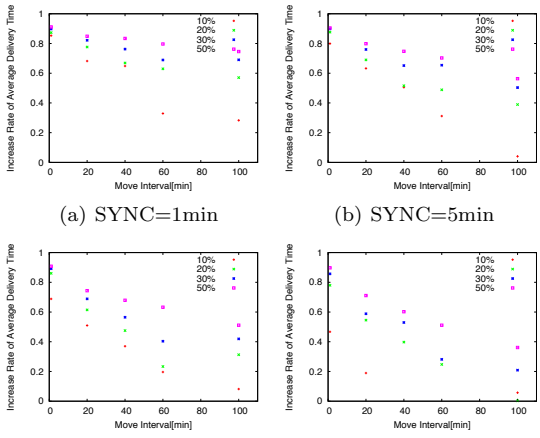


図 **14** Move Interval v.s. Percentage of Offline Time



(a) SYNC=1min (b) SYNC=5min

(c) SYNC=10min (d) SYNC=20min

図 **15** Move Interval v.s. Increase Rate of Average Delivery Time

5. おわりに

本稿では，自律性の高いネットワークを構成する NDN に対して，認証センターの不要な自律的認証を可能とする IBE，特に HIDE を適用することによって，ネットワークが分断された場合についても，そのネットワーク内において平常時と同様の通信サービスを同一のユーザインタフェースで可能とする局所的なネットワークを検討した．

HIDE における KGC の階層構造をキャリアとユーザの二階層に展開し，ユーザの秘密鍵の漏洩リスクを低減するとともに，ユーザによるアプリケーションに対する秘密鍵の生成を容易なものとした．アプリケーションにおける認証はユーザ ID とアプリケーションごとの秘密鍵を利用して行うことができ，これは，現在の多くのアプリで行われているユーザ ID とパスワードによる認証の自然な拡張となる．

また，情報伝達には NDN 上で動作するサーバレスマル

チユーザチャットシステムである Chronos のプロトコルを利用し，直接通信ができないネットワーク間においても情報の伝達を可能とした．

これらを利用して，世田谷区の経堂・桜上水周辺を被災地としてモデル化してシミュレーションを行い，移動するユーザの割合や移動間隔などが情報伝達性能に与える影響について調査した．しかしながら，この近似式は今回の経堂・桜上水周辺をモデルとした場合に限られるものである．実際の被災地域としてはより大規模なものも想定される．そのような広範囲でアプリケーションを利用した場合の情報伝達性能についても調査する必要がある．同時に，今回のシミュレーションでは基地局がすべて稼働しているものとしたが，実際の災害時には，物理的に破損していた基地局も存在した．東日本大震災時のように 4 割の基地局が故障した場合についても検討すべきである．

また，今回は被災時の利用について考えたが，通常時に利用できるかについても考慮する必要がある．

参考文献

[1] 情報処理学会論文誌 (IPSJ Journal) 原稿執筆案内，入手先 ⟨http://www.ipsj.or.jp/journal/submit/ronbun_j_prms.html⟩ (2013.03.28).

[2] 平成 24 年度版情報通信白書，入手先 ⟨http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h24/⟩, 2012.

[3] V.Jacobson, D.K.Smetters, N.Briggs, M.Plass, and P.Stewart, “Network- ing Named Content,” in CoNext 2009, Rome, Italy, Dec. 2009.

[4] C. Gentry and A. Silverberg, “Hierarchical id-based cryptography,” in Proceedings of Asiacrypto 2002, vol. 2501 of LNCS, 2002, pp. 548—66. [4] Z.Zhu, C.Bian, A. Afanasyev, V.Jacobson, and L. Zhang, “Chronos:Serverless Multi-User Chat Over NDN,” NDN, Tech. Rep. Technical Report NDN-0008, Oct. 2012.

[5] Z.Zhu, C.Bian, A. Afanasyev, V.Jacobson, and L. Zhang, “Chronos:Serverless Multi-User Chat Over NDN,” NDN, Tech. Rep. Technical Report NDN-0008, Oct. 2012.

[6] ID ベース暗号に関する調査報告書,” 入手先 ⟨http://www.cryptrec.go.jp/report/c08_idb2008.pdf⟩, March 2011.

[7] 3GPP, “3G security;Security architecture,” 3rd Generation Partnership Project(3GPP), TS 33.102, June 2008.

[8] A. Afanasyev, I. Moiseenko, and L. Zhang, “ndnSIM,” 入手先 ⟨http://ndnsim.net/⟩, 2012.

[9] 大規模災害等緊急事態における通信確保の在り方に関する検討会，入手先 ⟨http://www.soumu.go.jp/main_content/000117676.pdf⟩ , June 2011.