

無線ネットワークエミュレーションにおける 電波伝搬シミュレーションの高度化

橋本 光世¹ 明石 邦夫² 篠田 陽一³

概要: ソフトウェアによる無線ネットワークエミュレータを用いた, インタラクティブな実験が可能なネットワークテストベッドが存在する. インタラクティブな実験環境は, モビリティシミュレーション, 電波伝搬シミュレーション, 無線ネットワークエミュレーションの順に処理を高速に繰り返すことで実現している. このような実験環境では, 構築する無線ネットワーク環境の忠実度を維持するため, 各シミュレーションの計算時間が制限される. そのため, 大規模な実験を可能にするためには, 各シミュレーションの高速化が必要である. 既存の電波伝搬シミュレーションは逐次処理であるが, 計算処理の依存関係がないため, 並列化による高速化が可能である. また, 全てのノードの通信経路を計算を行うが, 無線通信には通信可能な距離が存在するため, 通信エリア外のノードは通信品質の計算から省くことが可能である. 本論文では, 無線ネットワークエミュレーションにおける電波伝搬シミュレーションの計算量の削減, 並列処理を実現する deltaHQ を提案した. deltaHQ は, 通信可能な通信エリア内に存在するノード間のみの通信品質の計算に絞ることで, 計算量を削減する. さらに, 並列処理を行い高速化を実現する. 提案した deltaHQ をリアルタイムに実行されるインタラクティブな実験環境に適用することで大規模化が実現可能となる.

キーワード: 無線ネットワーク, 電波伝搬シミュレーション, ネットワークエミュレーション, 並列分散処理, 最近傍探索

Improving Radio Wave Propagation Simulation for Wireless Network Emulation

KOSEI HASHIMOTO¹ KUNIO AKASHI² YOICHI SHINODA³

1. はじめに

ネットワークやインターネットでの利用を想定して開発されたアプリケーションソフトウェアの大規模な実証実験に, ネットワークテストベッドが利用されている. ネットワークテストベッドは有線ネットワークだけでなく, 無線ネットワークでの利用を想定した実験も可能である. 無線ネットワークでの利用を想定した実験が可能なネットワークテストベッドに ORBIT [1] が挙げられる. また, センサ

ネットワークでの利用を想定した実験が可能なネットワークテストベッドに SensLAB [2] が挙げられる. これらは, 実際に無線通信を行う機器を用いた実験が可能なネットワークテストベッドである. 実験に用いる無線ネットワーク環境は, 無線通信を行う機器を屋内に格子状に配置し, 機器の通信品質を制御することで構築される. したがって, 実際に無線通信を行う機器を用いていることから, 高い忠実度を有した実験が可能である. しかし, 大規模な実験は広大な施設が必要となる.

そこで, 汎用計算機が接続された有線ネットワーク上に大規模な無線ネットワーク環境を構築可能な, ソフトウェアによる無線ネットワークエミュレータが提案されている. 無線ネットワークエミュレータは, 有線ネットワーク上に無線ネットワーク特有の packets 損失や遅延, 帯域幅などの変動を再現する. また, 有線ネットワーク上に無線ネットワーク環境を構築することで, 外界の環境ノイズや電波

¹ 北陸先端科学技術大学院大学先端科学技術研究科
Graduate School of Advanced Science and Technology,
Japan Advanced Institute of Science and Technology

² 情報通信研究機構
National Institute of Information and Communications
Technology

³ 北陸先端科学技術大学院大学情報社会基盤研究センター
Research Center for Advanced Computing Infrastructure,
Japan Advanced Institute of Science and Technology

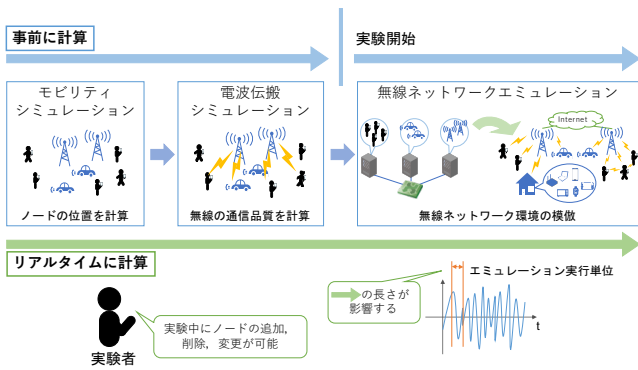


図 1 実験基盤の処理の流れ

の干渉など影響を受けない。そのため、再現性のある実験が可能となる。このような無線ネットワークエミュレータによる大規模な実験が可能で、ネットワークテストベッドの1つにStarBED [3]がある。StarBEDで提案された無線ネットワークエミュレータの実装として、NETorium [4]やQOMET [5,6]が挙げられる。

無線ネットワークエミュレータを用いた実験は、図1の流れで行われる。実験者が実験を行う手法には、事前に通信品質を計算する手法と実験中にリアルタイムに通信品質を計算する手法が提案されている。

リアルタイムに計算を行う手法では、モビリティシミュレーション、電波伝搬シミュレーション、ネットワークエミュレーションの3つの処理が高速に繰り返行われる。そのため、実験者によるノードの追加、削除といった実験内容の変更をリアルタイムに実験に反映可能となる。このようにインタラクティブな実験が可能となる。

この手法で構築する無線ネットワーク環境の忠実度は、電波伝搬シミュレータの計算時間が大きく影響する。そのため、大規模な実験環境を構築する場合、電波伝搬シミュレーションによる計算時間が膨大となり、無線ネットワーク環境の忠実度を維持できないという問題がある。QOMETにリアルタイム性を持たせた実装であるDyamiQ [7]では、100台程度の規模での実験が上限とされる。無線端末は増加し続けており、これらを利用したアプリケーションソフトウェアの実験を可能にする必要がある。そのため、リアルタイムに計算する手法での大規模な実験を実現するためには、限られた時間でより多くの計算を行う必要がある。したがって、高速化を行う対象は規模の向上による計算量の増加が著しい電波伝搬シミュレーションとする。

本論文では、リアルタイムに実行される手法を用いた検証基盤における電波伝搬シミュレーションの高速化を実現するdeltaHQを提案する。提案したdeltaHQをリアルタイムに計算する手法を用いた検証基盤に取り入れることで、インタラクティブな実験環境における無線ネットワークアプリケーションの大規模な実験が可能になる。

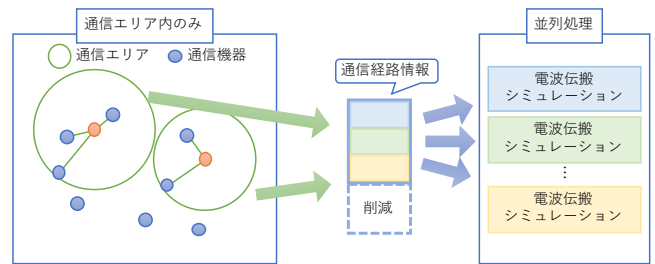


図 2 deltaHQ の概要

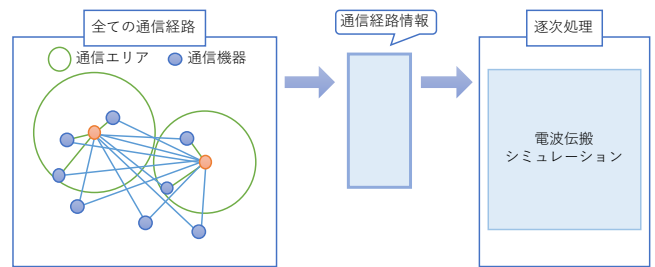


図 3 deltaQ の概要

2. 電波伝搬シミュレーションの高度化

既存の電波伝搬シミュレーションであるdeltaQ [5,6]は、統計に基づいた数理モデルを利用している。これは、建造物や周りの無線端末による電波の干渉や環境ノイズなど実験環境に合わせて定義された定数を用いており、計算時間が他の計算モデルに比べて短い。現在の電波伝搬シミュレーションは逐次処理であるが、シミュレーションに用いられる数理モデルは各通信経路間での計算処理の依存関係がない。そのため、並列化が有効であり、GPGPUやMulti-Core CPUを利用することで高速化が可能である。また、既存手法では全ての通信経路について無線通信品質を計算しているため、ノード数で計算量が決定する。電波は、減衰の性質を持つため伝搬距離が長くなったり障害物に反射したりすると電波強度は低下していく。そして、電波強度がある一定値を下回ると無線端末は通信不可能になる。これらから、事前に通信不可能となる電波の伝搬距離に存在する無線端末を探索し、通信品質の計算から省くことで計算量が削減可能になる。これら並列化と計算量の削減を電波伝搬シミュレーションに取り入れることで高速化し、精度を保ったまま規模を上げることが可能になる。

本論文ではインタラクティブな実験環境の大規模化を対象とした電波伝搬シミュレーションの計算量の削減、並列化による高速化を行うことを高度化とする。

3. deltaHQ の設計

3.1 全体設計

本論文で提案したdeltaHQの概要を図2に示す。図3はdeltaQの概要を表した図である。図2と図3は、それぞれ2つのノードを送信ノードとする場合の計算リストに

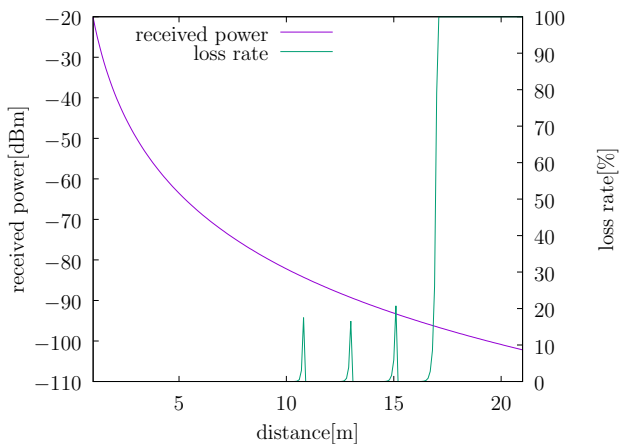


図 4 パケット損失率とノード間の距離の関係

加える経路を表した図である。図 3 に示す deltaQ では、フィールド全体に存在するノードの全ての通信経路に対して無線通信品質を逐次処理で計算している。

それに対し deltaHQ では、送信ノードの送信エリア内に存在する受信ノードとの通信経路に対して、無線通信品質を計算することで計算量を削減している。さらに並列処理で無線通信品質を計算することで高速化を実現している。deltaHQ の処置の流れは、最初に、モビリティシミュレーションの結果から送信ノードを中心とした通信エリア内に存在する受信ノードを探索する。探索後、送信ノードと通信エリア内に存在する受信ノードとの経路情報を計算する経路リストに加える。全てのノードを送信ノードとした時の探索を終えたあと、完成した経路リストの大きさと並列数からプロセッサ 1 コアあたりの処理単位を計算し、各コアに処理を割り当てる。その後、それぞれのコアは割り当てられた処理を行うことで経路リストの通信品質を全て計算する。このように電波伝搬シミュレーションの計算量の削減と並列化を行うことで高速化を実現する。

3.2 計算量の削減

電波は伝搬距離が長くなるほど弱くなる性質を持っており、送信ノードと受信ノードの距離が長くなるほど受信ノードの受信電力は低下する。それに加えて、受信電力が弱くなるほど通信品質は低下する。図 4 は deltaQ を用いてノード間の距離におけるパケット損失率と受信電力を算出したグラフである。図 4 の結果では、送受信ノード間の距離が長くなるほど受信電力が低下していくことが確認できる。また、送受信ノード間の距離が約 10 m からパケットの損失が始まり、次第に上昇して約 17 m でパケット損失率が 100% になり通信できなくなる。このように通信エリアは受信電力と電波の距離から決定でき、パケット損失率が 100% になる距離を通信エリアと定義する。

実験開始前の事前計算として、通信可能だと推測する通

表 1 実験環境

Model	FUJITSU PRIMERGY RX200 S6
Chipset	FUJITSU D3031
CPU	Intel Xeon CPU E5620 2.40GHz 4 core × 2
Memory	48GB

信エリアを各ノードごとに計算する。実験中、送信ノードの受信エリア内に存在するノードの探索は送受信ノード間の距離と通信エリアの半径を比較することで行う。通信エリア内であれば、通信品質を求める経路リストに追加する。

3.3 並列処理

並列処理は分散メモリ型と共有メモリ型に分類される。分散メモリ型は、ネットワークで接続された複数の計算機上、または同一計算機上で実行される複数プロセスがメッセージ交換によるデータ交換を行う。共有メモリ型は、複数プロセッサを搭載した 1 台の計算機上で実行される複数スレッドが共有メモリ空間を用いたデータ交換を行う。また、これらを併用した並列処理はハイブリッド型と呼ばれる。

本論文で提案した deltaHQ は、電波伝搬シミュレーションの高速化に重点を置いている。よって、ネットワークを経由しない共有メモリ空間でのデータ交換を用いる共有メモリ型の並列処理を行う。

3.3.1 並列処理の粒度

共有メモリ型を用いる並列処理はスレッドを用いる。スレッドには、スレッド生成時、同期処理、スレッドの処理量の違い、スケジューリングなどによるオーバーヘッドが存在する。そのため処理時間が短い場合は並列化の効果が薄い。

したがって、本論文では事前計測として無線通信品質の計算にかかる時間を計測し、その結果からスレッドにどのような処理を割り当てるか検討する。計測するための実験環境は表 1 の通りである。

deltaQ を用いてノード数を 1024 として 5 回計測し、1 経路の計算にかかる時間の中央値、四分位点を算出した。その結果を表 2 に示す。計測した結果から 1 経路の計算時間の中央値は約 4.46 μ s であった。そのため、本論文では、1 スレッド 1 経路ではなく複数にまとめたブロックとしてスレッドに割り当てることとする。

並列処理においてオーバーヘッドとなりうる要因として、排他制御、同期処理、計算量の違いがあげられる。そのため、これらを考慮した並列処理を行う。各スレッドが担当する計算量を均等にするため、ブロックの大きさは配列の大きさをコアの数で割った値とする。割り切れない場合は

表 2 測定結果

parameter	median[μ s]	lo quartile[μ s]	up quartile[μ s]
1 connection	4.460	4.428	4.517

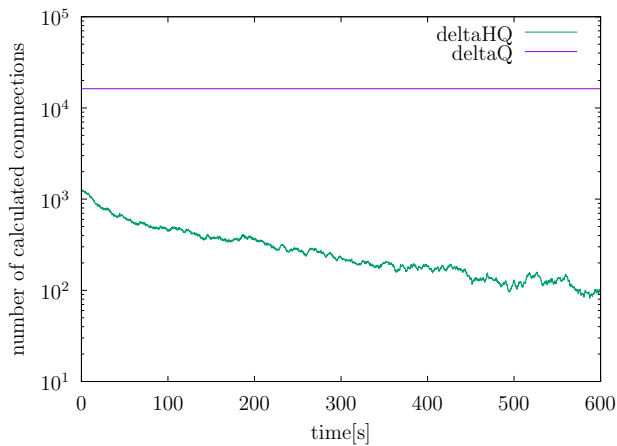


図 5 ランダムウォーク時の計算量の変化

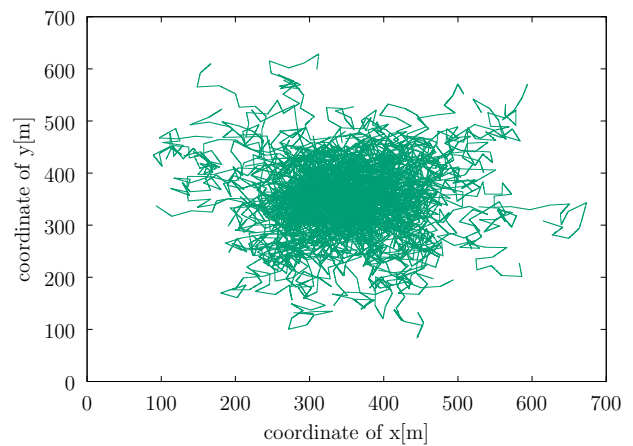


図 7 ランダムウォークの移動軌跡

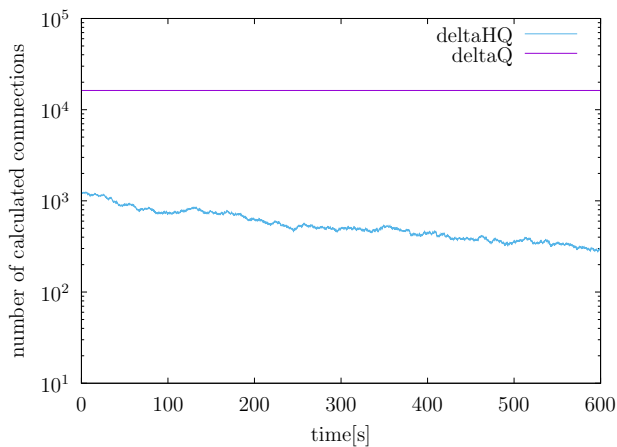


図 6 Levy walk 時の計算量の変化

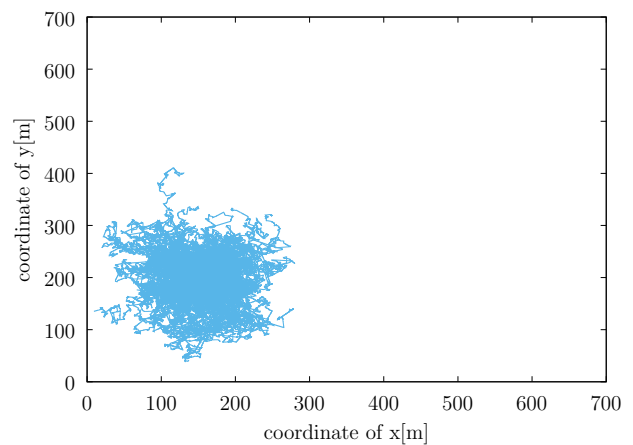


図 8 Levy Walk の移動軌跡

各ブロックに 1 経路ずつ追加する。その結果、各スレッドが担当する計算量の差は 1 経路分のみとなる。

4. 評価

4.1 計算量の削減

提案した計算量の削減手法が、どのような実験環境でどの程度削減できるかを確認するため、無線端末の移動を対象にいくつかの移動モデルを用いて評価した。移動モデルは線形移動やランダムウォークが一般的である。しかし、本論文で対象としている無線端末は、人と共に移動する場合と乗り物に搭載された状態で移動する場合が考えられる。そのため、評価には人の行動に即した Levy Walk [8] とランダムウォークを用いた。Levy Walk は、短い直線移動を頻繁に行い、稀に長い直線移動をする行動パターンである。したがって、ランダムウォークに比べ行動エリアの重複を軽減できる移動モデルである。Levy Walk を用いた評価には、式 1、2 を利用した。式 1 は、一度に移動する直線距離 L の分布がべき乗則に従うことを表している。式 2 の Min 、 Max はそれぞれ移動距離の最小値と最大値を表し

ており、 $rand()$ は 0~1 までの一様分布乱数を生成する関数とする。

$$P(L) \propto L^{-1+\beta}, (0 < \beta < 2) \quad (1)$$

$$L = (Min^{-\beta} + (rand() \times (Max^{-\beta} - Min^{-\beta})))^{-\frac{1}{\beta}} \quad (2)$$

評価はランダムウォーク、Levy Walk 共に以下の条件で行なった。ノード数を 128、ノードの速度を 1 m/s、初期位置は 100 m 四方の中にランダム配置とし、0.1 s 刻みで 10 min 間のシミュレーションを行った。また、全てのノードの送信電力は 20 dBm とし、通信エリアの範囲は 17 m とした。評価を行う際に Levy Walk に設定した各パラメータを表 3 に示す。

ランダムウォークを用いた評価の計測結果を図 5、7 に示す。図 5 は、シミュレーション時間の経過と計算する必

表 3 Levy Walk の設定パラメータ

parameter	value
β	1.2
Max length[m]	20
Min length[m]	1

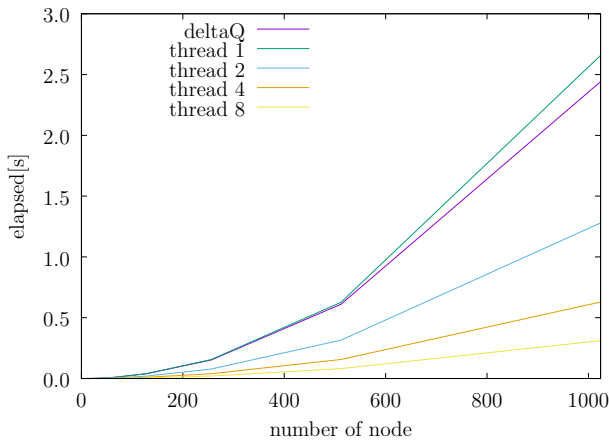


図 9 全経路の計算にかかる時間

要のある通信経路数の変化を計測したグラフである。図 7 は、ランダムウォークによる全ノードの軌跡を表している。図 5 では、計算する必要がある通信経路の最大数は 1252 経路、最小値は 82 経路であった。

Levy Walk を用いた評価の計測結果を図 6, 8 に示す。図 6 は、シミュレーション時間の経過と計算する必要がある通信経路数の変化を計測したグラフである。図 8 は、ランダムウォークによる全ノードの軌跡を表している。図 6 では、計算する必要がある通信経路の最大数は 1248 経路であり、最小値は 278 経路であった。

図 7 と図 8 を比較し、計算量の削減量評価する。図 7 は図 8 に比べて、軌跡が広く拡散している。そのため、図 7 は、通信エリア内のノードが少なく Levy Walk に比べて計算量が大きく減少している。

既存の電波伝搬シミュレーションでは全ての経路を計算するため、 $128 \times 127 = 16256$ 経路の計算を何度も行う必要があった。しかし、本論文で提案した計算量の削減手法を取り入れることで、ランダムウォーク、Levy Walk 共に 92% 以上の計算量を削減することを確認した。

4.2 並列処理

並列処理を評価するために、シナリオの 1 ステップにおける全ての通信経路のシミュレーションが完了するのにかかった時間を計測した。ノード数を 2~1024 まで 2^n ずつ変化させた時の、全てのノード間の通信経路を計算するための計算間の変化を図 9 に示す。

deltaQ と deltaHQ を 1 スレッドで実行した場合を比較すると、スレッドによるオーバーヘッドのため deltaHQ の方が計算に時間がかかっている。また、deltaHQ のスレッド数を 2 倍ずつ変化させると計算時間は半減していくことがわかる。これらから並列処理は効果的だと言える。

5. 議論

5.1 通信エリアに存在するノード探索の高速化

提案した通信エリアに存在するノードの探索は、無線の通信品質を計算する度に全てのノードの通信経路の距離を算出する。そして、通信エリアの半径を超えているかを比較する。比較したのちに計算リストに加える処理を行っている。全てのノードの通信経路を計算する時の計算量オーダーは $O(n^2)$ であり、ノードの数が増えると計算時間が膨大になる。そこで、空間充填曲線の 1 つである Z order curve を用いて 2 次元、3 次元座標を 1 次元に写像し、電波伝搬シミュレーションのノードの位置関係を 1 つの数直線で表現する。その結果、送信ノードの通信エリアに存在するノードは数直線上の近隣に存在し、距離を計算することなく高速に通信エリア内のノードが探索可能になる。

5.2 並列処理のオートチューニング

deltaHQ の並列化は、一般的な計算機に搭載されている CPU を用いた場合のものであった。CPU は 1GHz 帯のクロックで 100 個近いコアを持つものから、4GHz 帯のクロックで十数個のコアを持つものまでである。また、GPGPU のような数千個ものコアをもつようなものもある。GPGPU は、コア数が多く並列処理に有効であるがデータのサイズ次第でデータのコピーによるオーバーヘッドも考慮すべきである。

実験に用いる計算機は、環境によって異なり、実験内容によって規模も異なるため、最適な並列化を行うには、機器構成によって並列処理の粒度、データを割り当てる単位を変える必要がある。また、シミュレーションに用いる数理モデルの変更によりリアルタイム性を維持できない場合は、1 つの通信経路を計算する処理の中で並列化を行う必要がある。

これらを考慮して実験を行う度に、電波伝搬シミュレーションに変更を加えることは実験の準備コストが発生することになる。そのため、実験内容や機器構成の情報から並列化の粒度、データの割り当て単位を事前計算し、実験前のコンパイル時に適応する方法を検討する必要があると考えている。

5.3 分散処理

deltaHQ はある時間の情報を計算するために 1 つ前の時間の情報を用いており、全ての経路をメモリに保持し続ける必要がある。ノード数が数千、数万規模になると 1 台の計算機では全ての情報を保持できないため、複数の計算機に情報を分散する必要がある。複数の計算機に情報を分散して配置する方法の 1 つに、超並列計算機のように複数の計算機をネットワークで接続し、全てのメモリを 1 つの統

合されたメモリとして利用する方法がある。本論文で対象にしているテストベッドは、計算時間が制限されているため、ネットワークを経由してメモリにアクセスする上記の仕組みは適切ではない。計算機間で通信することなく情報を分散させる方法は、今後の課題である。

6. おわりに

無線ネットワークエミュレータを用いたインタラクティブな実験が可能なネットワークテストベッドが存在する。インタラクティブな実験は、リアルタイムにモビリティシミュレーション、電波伝搬シミュレーション、無線ネットワークエミュレーションを繰り返し実行することで実現している。構築する無線ネットワーク環境の忠実度は、電波伝搬シミュレータの計算時間が大きく影響する。そのため、大規模な実験環境を構築する場合、電波伝搬シミュレーションによる計算時間が膨大となり、無線ネットワーク環境の忠実度を維持できないという問題がある。

そこで本論文では、インタラクティブな実験が可能な実験基盤における電波伝搬シミュレーションの高速化を実現する deltaHQ を提案し設計と実装を行なった。

提案した deltaHQ は計算量の削減と並列処理を行う。既存の電波伝搬シミュレータは、全てのノードの通信経路に対して無線通信品質の計算を行っていた。しかし、deltaHQ では通信エリアに存在するノードとの経路に計算を絞ることで計算量の削減を可能とした。移動モデルであるランダムウォークと Levy Walk を用いて計算量の削減について評価を行なった。ランダムウォークでは、16256 経路から最大 1252 経路に、最小 82 経路に削減した。Levy Walk では、16256 経路から最大 1248 経路に、最小 278 経路に削減した。これらより、各移動モデルにおいて、計算量を 10 分の 1 以下に削減可能であることを示した。

また、既存研究では逐次処理であった電波伝搬シミュレーションを並列化することで、通信品質の計算に必要な時間を削減した。並列数を 2 倍ずつ変化させた時の計算時間の評価を行なった結果、計算時間は半減していくことを示した。

この 2 つの手法を実現した deltaHQ をインタラクティブな実験環境に適用することで、無線ネットワークエミュレーションのさらなる大規模化が可能となる。

謝辞

本論文を執筆するにあたり、有益な助言をいただいた北陸先端科学技術大学院大学知念賢一准教授、NICT 北陸 StarBED 技術センター井上朋哉博士に深く感謝します。

参考文献

[1] Raychaudhuri, R., Seskar, I., Ott, M., Ganu, S., Ramachandran, K., Kremono, H., Siracusa, R., Liu, H. and

- Singh, M.: Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols. In *Wireless Communications and Networking Conference, 2005 IEEE*, Vol. 3, pp. 1664–1669. IEEE, 2005.
- [2] Des Rosiers, C.B., Chelius, G., Fleury, E., Fraboulet, A., Gallais, A., Mitton, N. and Noël, T.: Senslab. In *International Conference on Testbeds and Research Infrastructures*, pp. 239–254. Springer, 2011.
- [3] Toshiyuki, M., Ken-ichi, C. and Yoichi, S.: Starbed and springos: Large-scale general purpose network testbed and supporting software. In *Proceedings of the 1st international conference on Performance evaluation methodologies and tools*, p. 30. ACM, 2006.
- [4] Kunio, A., Tomoya, I., Shingo, Y., Yuuki, T. and Yoichi, S.: Netorium: high-fidelity scalable wireless network emulator. In *Proceedings of the 12th Asian Internet Engineering Conference*, pp. 25–32. ACM, 2016.
- [5] Razvan, B., Lan Tien, N., Khin Thida, L., Junya, N. and Yoichi S.: Qomet: A versatile wlan emulator. In *Advanced Information Networking and Applications, 2007. AINA'07. 21st International Conference on*, pp. 348–353. IEEE, 2007.
- [6] Razvan, B., Junya, N., Takashi, O., Lan Tien, N., Yasuo, T. and Yoichi Shinoda. A multi-purpose wireless network emulator: Qomet. In *Advanced Information Networking and Applications-Workshops, 2008. AINAW 2008. 22nd International Conference on*, pp. 223–228. IEEE, 2008.
- [7] Razvan, B., Shingo, Y., Tomoya, I., Yuuki, T., Toshiyuki, M. and Yoichi Shinoda.: Dynamiq: A tool for dynamic emulation of networks. *EAI Endorsed Trans. Self-Adaptive Systems*, Vol. 1, No. 2, p. e1, 2015.
- [8] Injong, R., Minsu, S., Seongik, H., Kyunghan, L., Seong Joon, K. and Song C.: On the levy-walk nature of human mobility. *IEEE/ACM transactions on networking (TON)*, Vol. 19, No. 3, pp. 630–643, 2011.