

OpenMP を用いた電波伝搬シミュレーションの高速化

1810562 藤本 翔太 成見研究室

1 はじめに

近年では、目まぐるしい無線通信技術の進歩によりその利用が増えてきてシステム設計の際に伝播特性を解析したり、モデル化したりすることが頻繁に必要とされている。解析、モデル化の手法として実験的アプローチ、理論的アプローチの 2 種が存在する。そのうち理論的アプローチは、コスト、時間、労力において優れているためよく用いられる [1]。本研究においても理論的アプローチを用いていく。

荒生はレイトレーシング法による電波伝搬シミュレーションを GPU にて高速化し、CPU に対して最大 5,000 倍高速化したと述べている [2]。しかし、CPU 側の並列化が行われておらず正しい比較とは言えない。計算量の多い共通鍵暗号 AES の並列処理には OpenMP が用いられている事例 [3] もあることから、本研究でも OpenMP を使用し高速化を行う。

2 目的

本研究の目的は電波伝搬シミュレーションを CPU において並列化することである。

3 既存研究

3.1 イメージ法を用いた電波伝搬シミュレーションの高速化

高橋、石田、吉浦はレイトレース法を用いた構内無線伝搬シミュレータを作成した [4]。最後のイメージ点とその反射物体との位置関係により反射波の存在する領域をあらかじめ求め、その領域内の評価点のみ計算を行う手法を考察した。本手法を従来のレイトレース法に付加することにより、2 次反射まで考えたとき、計算時間を従来法のその約半分に削減することができた。

3.2 Xeon Phi プロセッサ上での共通鍵暗号 AES の並列処理

小梁川、吉田は OpenMP による AES-CTR の高速化手法を行った [3]。大規模なデータを暗号化および復号化するためには多くの時間を要するため、高速化の需要が高まっていると考えた。評価の結果、16MB データの暗号化および復号化の並列処理時間が大幅に短縮されており、提案手法の有効性が確認された。

3.3 GPU を用いた電波伝搬シミュレーションの改良

植田はレイトレーシング法に特化した GPU である RTX を用いた電波伝搬シミュレーションの更なる高速化を行った [5]。元のプログラムとの比較の結果、特に変更を行ったレイトレーシング部分について、高負荷条件で約 5 倍に、全体ではおよそ 1.3 倍に高速化できた。また、レイトレーシング部分の計算時間を測定したところ、レイの数

が多い場合のシミュレーションにおいてソートの処理時間が大半を占めることが分かった。また、RTX が搭載している RT コアにより高速にレイトレーシング法の計算が行えることを確認した。

4 システム実装

4.1 OpenMP の概要

OpenMP を使用すると、マルチコア CPU の複数のコアを活用し、複数のスレッドで処理を行う。本研究では、各スレッドは並列領域が開始される for 文と同時に生成され (fork)、並列領域が終了すると、逐次処理はマスタースレッドのみに制限される (join) 図 1。並列領域の開始と終了はディレクティブで指定される。

4.2 高速化

下記の 4 つの項目をプログラムに実装した。

- #pragma: 並列化を行うためのディレクティブ
- reduction: 本研究では複素数の計算を行うためのディレクティブ
- private: 他スレッドと変数を共有しないためのディレクティブ
- ループ変数の整数化
- 出力用の一時変数追加

5 評価

5.1 評価方法

シミュレーション空間の大きさや反射回数を変えた場合に並列度によって計算時間がどのように変わるかを評価する。

5.2 実験条件

データの測定環境について述べる。使用する環境を表 1、実験のシミュレーション条件を表 2 に示す。

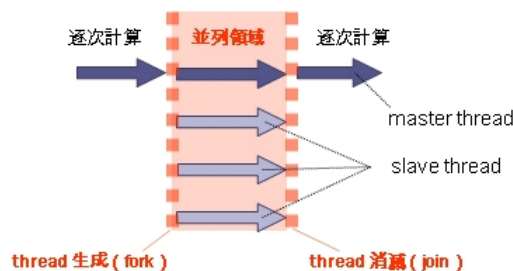


図 1 プログラムの実行方法 (参考文献 [6] より引用)

表 1 実行環境

OS	Ubuntu 20.04.6
CPU	Intel Core i7-13700F
コア数	16
p-core	8
e-core	8
CPU メモリ	16GB DDR4

表 2 想定空間

x 方向の大きさ (m)	5
y 方向の大きさ (m)	4
z 方向の大きさ (m)	3, 30, 300
送信点	(3, 3, 2)
周波数	2000[Hz]
x,y 方向の反射回数	2, 5, 10,20
z 方向の反射回数	0
並列度	1, 2, 4, 8, 16

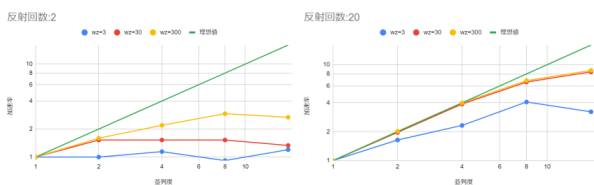


図 2 計算速度の speed-up

5.3 実験結果

図 2 は反射回数 2 回 (左) と反射回数 20 回 (右) のときの、並列度を変えた時の速度上昇率 (speed-up) を表す。これより並列度が小さいうちは理想緑線と近い値を示しており、大きくなりすぎると計算時間が多くなる傾向がある。反射回数とトンネルの奥行が大きくなるほど理想に近づくことが確認できた。つまり計算量が多いことが並列化の効率を上げる条件であることがわかった。また計算量によっては並列度の頭打ちが存在し、ピークとなる並列度合いが存在する。

図 3 は、CPU 内で使うコアの種類と数を変えた時の速度上昇率を表す。Intel の第 12 世代以降は性能の高い P-core と電力効率の高い E-core のハイブリッドアーキテクチャとなっている。P-core の偶数スレッド、P-core の 0 から n のスレッド、E-core のスレッドを使用する実験を行った。特に p-core は早くピークを迎える点、理想直線に他のスレッドを使用したときよりも近づいている点を確認できた。反射回数 20 回においては、P-core 偶数スレッド、P-core 0-n スレッド、E-core の順で優れていることが本実験より確認できた。

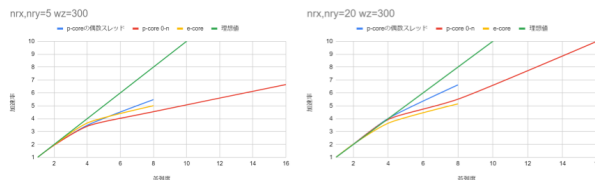


図 3 トンネルの奥行 300 における各反射回数の speed-up

6 おわりに

6.1 まとめ

本研究ではイメージング法を OpenMP を利用して並列計算を行うことで電波伝搬シミュレーションの高速化を行った。元のプログラムとの比較の結果、計算量が膨大である場合には、理論値に近い結果を得ることができた。また、P-core や E-core などの異なるコアを比較することで P-core の偶数スレッドが本実験では最適である結果を確認できた。

6.2 今後の課題

6.2.1 複雑な空間における計算

本研究では、単純なトンネル構造における電波の送受信のシミュレーションを行った。更に発展させるためには、複雑なシミュレーションにおける高速化でも OpenMP が有用であるか評価する必要がある。

6.2.2 他の並列化手法との比較

今回の実験では OpenMP のみでの高速化を行ったが、他にも AVX 命令を用いた並列化も存在する。

6.2.3 条件の増加による妥当性

本実験では 1 種類のみ CPU で行った。よりコア数の多い CPU を使用することで更なる高速化を見込める。

参考文献

- [1] 今井哲朗, 電波伝搬解析のためのレイトレーシング法-基礎から応用まで-, コロナ社, 2016 年
- [2] 荒生太一, 成見哲. レイトレーシング法に特化した GPU による電波伝搬シミュレーションの高速化, 電気通信大学情報ネットワーク工学専攻修士論文, 2021
- [3] 小梁川 龍, 吉田 明正. Xeon Phi プロセッサ上での共通鍵暗号 AES の並列処理. 第 83 回全国大会講演論文集 2021 (1), 71-72, 2021-03-04
- [4] 高橋 賢, 石田 和人, 吉浦 裕. イメージ法を用いた電波伝搬シミュレーションの高速化. 電子情報通信学会技術研究報告. SSE, 交換システム. 94 (363), 49-54, 1994-11-25
- [5] 植田 武, 成見 哲. GPU を用いた電波伝搬シミュレーションの改良, 電気通信大学情報理工学域 I 類卒業論文 2023
- [6] 計算力学研究センター RCCM, OpenMP とは, <https://www.rccm.co.jp/development/parallel/openmp.html>(アクセス日 2024-1-30)