

大規模ワイヤレスネットワーク構築のための ネットワークシミュレータns3の高速化



● 研究の目的・意義

【技術背景】

高性能計算(HPC)とネットワークを専門とする異なる専門分野の研究者が連携し、モバイルワイヤレスネットワーク性能評価の計算機シミュレーション技術と高速並列計算技術という2つの技術を融合する学際的研究を行う。

【課題】

モバイルワイヤレス技術開発では汎用のネットワークシミュレータであるns3シミュレータを用いた技術開発が盛んである。しかしながら、近年モバイル端末の増加によるシステムの大規模化が進み、シミュレータが大規模計算を扱う。このことで、計算時間の長大化が問題になっている。すでに、シミュレータで扱えない限界が見えてきており、対策が急務である。

【研究目的】

ns3シミュレータおよびシミュレーションの高速化を行う。スーパーコンピュータ環境を用いて、大規模ワイヤレスネットワークのns3シミュレーションを極めて短時間に行い、研究業界および産業界に、大きな利益をもたらすことを目的としている。

● 参加者

1. 代表者：村瀬勉 (名古屋大学 情報基盤センター)
2. 副代表者：片桐孝洋 (名古屋大学 情報基盤センター)
3. 副代表者：Eng Keong Lua (ケンブリッジ大学)
4. 共同研究者：秋葉康佑、前田 和輝、楊 智赫、黄 靖誠 (名古屋大学 大学院情報学研究科 大学院生)

● 研究計画

【1】 ns3シミュレータ駆動部分の並列高速化

①分析：ns3のシミュレータ駆動処理のうち、並列化可能な部分とシーケンシャルにならざるをえない部分とを切り出す。通信でのパケット送信部分のほとんどが並列化出来るが、連続するパケットについては、前のパケットの通信成功の可否により、あとのパケットの処理が決まるといった部分を判別する。なお、この判別によりもたらされる高速化は、ns3シミュレータにおいて汎用部分であるため、今回対象のモバイルワイヤレス通信の大規模化のみならず、ns3を用いるすべてのシミュレーションの高速化をかける。

②高速化：実際のスパコンコアにns3を多スレッドとして実装し、最適な分割数を求める。

【2】 モバイル端末が移動する軌跡上での分割並列計算

①分析：モバイル端末が移動する軌跡を予め計算しておき、その軌跡上をいくつか分割して、それぞれの分割したシミュレーションを並列に行う。この操作においては、ステートフルな連続したシミュレーションを独立の複数に分割するため、近似計算になるが、ns3シミュレータは、疑似乱数を用いたモンテカルロシミュレーションであるため、分割数がそれほど多くなければ、全体にそれほど大きな影響はないと思われる。

②高速化：実際のスパコンコアにns3を多スレッドとして実装し、最適な分割数を求める。

● 予備評価

- 2023年度から課題の抽出を行っており、2024年度からは、フィージブルソリューションの検討に入った
- 高速化を適用するネットワークモデルとして、AMR(自律走行ロボット)の最適経路計算アルゴリズムであるVDR法と、中点追従法を開発した。このVDR法と中点追従法へ、本申請での研究成果を本格適用する。

【スパコン「不老」の80コアを使った高スレッド並列実行】

- VDR法を適用した。VDR法は潜在的な並列性がある。
- スパコン「不老」の80コアを使った高スレッド並列実行のプロダクトランの性能評価を行った(図1)。
- VDR法のスクリプトレベルの並列化をした。PCで逐次に動作していたns3の処理をスレッド並列化し、「不老」クラウドシステムで性能測定を行った。
- プロダクションレベルの処理の逐次実行で、約34時間(推定)実行時間が約20分で実行できることが明らかとなった。

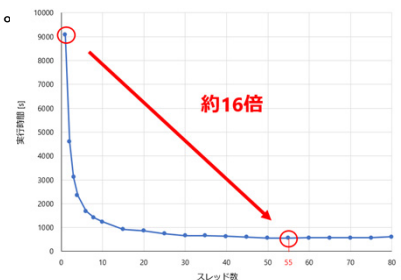


図1:VDR法の「不老」での高スレッド並列化

【Miyabi-CでのMPI実行】

- 中点追従法の適用を想定する。中点追従法は依存があり並列化できない。
- Miyabi-Cの1ノードを利用。MPI版のns3をインストールした。
- ns3のMPI評価用ベンチマークであるnms-p2p-nix-distributed
 - ・大学キャンパスのようなネットワークを複数作り、それらを広域リンクでリング状につなぎ、あるキャンパスの端末から別キャンパスの端末へUDPパケットを送る。
 - ・複数のDARPA NMS風キャンパスネットワークをリング接続し、各キャンパスのLAN端末をsink、次キャンパスのNet1ノードをsourceとしてUDPトラフィックを流し、MPI分散実行とNix-vector/global routingの挙動・スケーリングを見る。

表1:Miyabi-CでのMPI実行
(campuses=8, clients=20, packets=20)

並列数	実行時間	並列化効率	台数効果
1	6.25	1.00	1.00x
2	3.68	0.84	1.69x
4	2.71	0.57	2.30x
8	2.48	0.31	2.57x