

生体モデルにおける数式の類似性に着目した OpenMP による並列シミュレーションの検討

吉川 禎^{†1} 奥山 倫弘^{†1} 置田 真生^{†1}
浅井 義之^{†2} 安部 武志^{†2} 野村 泰伸^{†3}
八木 哲也^{†4} 萩原 兼一^{†1}

1. はじめに

我々は、汎用生体モデル記述言語 *insilicoML*¹⁾ により記述された生体モデルを入力とし、生体機能を表現する微分方程式の初期値問題を解くシミュレータ *insilicoSim*²⁾ を開発している。*insilicoSim* は、モデルを変換したバイトコード (BC) を実行するインタプリタ方式として動作する。また、MPI を用いて SMP クラスタ上で並列に実行できる。

現在の *insilicoSim* では、大規模なモデルに対して実用的な時間内にシミュレーションを完了できない。例えば、4 万以上の数式を含む神経細胞ネットワークモデルのシミュレーションに約 10 時間を要する。この理由は、第 1 にインタプリタ実行の計算コストが大きいためである。加えて、MPI を用いたノード内通信の通信コストが大きいためである。

そこで、計算コスト削減のために、BC からソースコード (SC) を自動生成し、コンパイルにより実行ファイル (EX) を得るコンパイラ方式を開発する。大規模モデルに対してはコンパイル時間が問題となるが、モデルにおける数式の類似性に着目することにより、SC の行数を削減し、コンパイル時間を短縮する。

さらに、通信コスト削減のために、MPI および OpenMP を用いたハイブリッド並列化を検討する。その前段階として、本研究ではノード内通信に OpenMP を用いた場合の並列化効率について評価する。

2. シミュレータの概要

本シミュレータは、まず、モデル内の数式を抽出し、数式間の参照関係に基づく依存グラフを生成する。次

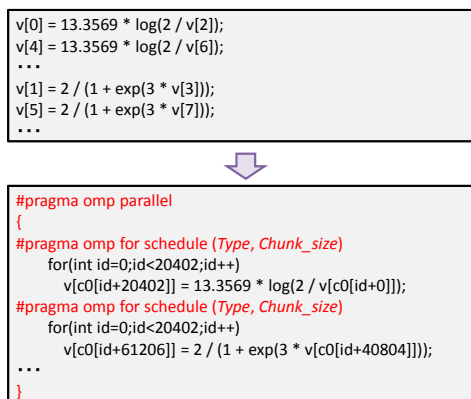


図 1 数式の統合および OpenMP による並列化

に、ASAP スケジューリング³⁾ に基づいて数式の計算順序を決定し、BC に変換する。その後、BC 群から OpenMP を用いた SC を生成し、コンパイルする。最後に得られた EX を実行し、シミュレーション結果を得る。なお、数値解法にはオイラー法を用いた。

ここで、SC の出力前に数式を統合して SC の行数を削減する。一般にモデルは細胞などのより小さい下部組織の集合であり、モデル内には下部組織を表す類似の数式が多数存在する。図 1 のように、同時点で実行できる類似の数式を 1 つの for 文に統合する。

さらに、数式の統合結果を利用して OpenMP による並列化を施す。数式の統合により、SC には多数の for 文が存在する。各 for 文に対し、図 1 のように OpenMP の指示文を挿入する。schedule 指示節における引数は実験的に決定した。

3. 性能評価

実験には 40 コアの共有メモリ計算機 1 台 (Intel Xeon E7-4850 2.0 GHz 10-core × 4, 主記憶 256GB) を用いた。OS は Ubuntu 11.10, gcc のバージョンは

^{†1} 大阪大学大学院情報科学研究科

^{†2} 沖縄科学技術大学院大学オープンバイオロジーユニット

^{†3} 大阪大学大学院基礎工学研究科

^{†4} 大阪大学大学院工学研究科

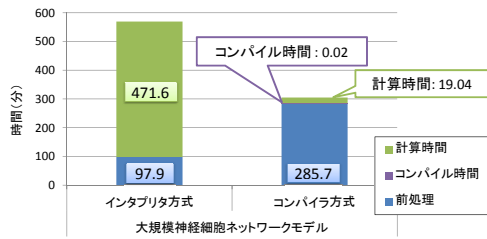


図2 インタプリタ方式との比較 (単位: 分)

4.6.1 を用いた . コンパイル時には最適化オプション-O2 を用いた . 対象モデルは , 三次元神経細胞ネットワークモデル⁴⁾ (細胞数 2 万 , 数式数 180 万) である .

3.1 コンパイラ方式による高速化

まず , 数式の統合により , SC の行数を 480 万行から 271 行まで削減した . 統合前のコンパイル時間は二日以上を要したが , 統合後は 1.2 秒まで短縮できた .

次に , 図 2 にシミュレーションの総所要時間をインタプリタ方式と比較した結果を示す . コンパイラ方式を用いることにより , 計算において 24.8 倍の高速化を達成した . 一方 , 前処理に要する時間は 2.9 倍に増加した . この理由は , 前処理における数式のスケジューリング処理を , インタプリタ方式は並列処理しているが , コンパイラ方式は逐次処理するためである .

3.2 OpenMP による並列化効率

まず , 効率のよい並列化のため , 最も性能のよい OpenMP の schedule 指示節における引数を予備実験で求めた . 選択可能なスケジューリング方法は , static , dynamic および guided の三種類である .⁵⁾ これらに対し , ループの分割数 d (統合した数式の数 i / チャンクサイズ s) を変化させ , EX の実行時間を測定した . 具体的には , d を 40 (ブロック分割) から最大の i (サイクリック分割) まで変化させた .

測定の結果 , static を用いてブロック分割した際に最もよい性能を示した (図 3) . この理由は , 各 for 文において , 全数式の計算量が等しいためと考えられる . この場合 , 静的に割り当てた際に負荷を均等に分散でき , 割り当てのオーバーヘッドが小さい . また , 大規模なモデルでは統合できる数式が多いと考えられる . そのため static によるブロック分割時に , 均等に割り当てられない式が相対的に減少し , 性能が向上する .

図 4 に , static によりブロック分割した際に , スレッド数を変化させた時の速度向上率を示す . 結果 , 40 スレッドを用いて , 14.7 倍の速度向上を達成した . 速度向上率が 14.7 倍に留まっている理由の一つは , スレッド数に対して i が小さい for 文を十分並列化できていないためである .

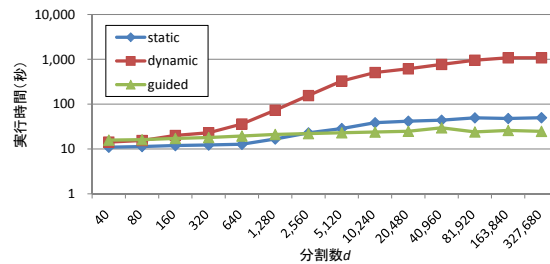


図3 schedule 指示節と実行時間

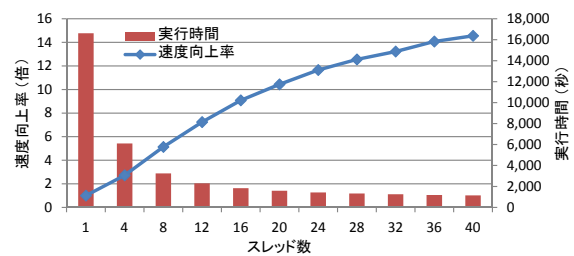


図4 速度向上率

4. ま と め

insilicoML で記述された大規模な生体モデルに対して , OpenMP で並列化したシミュレーションコードを自動的に生成する手法を提案した . 今後の課題は , 統合した数式群に対する SSE を用いた SIMD 実行 , OpenMP および MPI のハイブリッド並列化である .

謝辞 本研究の一部は , 科学研究費補助金 (基盤研究 (B) 23300007 および 若手 (B) 23700036) ならびに大阪大学グローバル COE プロジェクト *in silico medicine* の支援による .

参 考 文 献

- 1) Y. Asai et al. Specifications of *insilicoML* 1.0: A multilevel biophysical model description language. *J. Physiological Sciences*, Vol. 58(7), pp. 447-458, 2008.
- 2) E. Heien et al. *insilicoSim*: an extendable engine for parallel heterogeneous biophysical simulations. In *Proc. 3rd Int'l Conf. Simulation Tools and Techniques*, Mar. 2010.
- 3) R.A. Walker et al. Introduction to the scheduling problem. *IEEE Design & Test of Computers*, Vol. 12(2), pp. 60-69, 1995.
- 4) 津崎圭亮ら. *in silico* 1 次視覚野神経回路. 第 24 回バイオエンジニアリング講演会論文集, Jan. 2012.
- 5) R. Chandra et al. *Parallel Programming in OpenMP*. Morgan Kaufmann, San Mateo, CA, Oct. 2000.