

スーパーコンピューティング ニュース

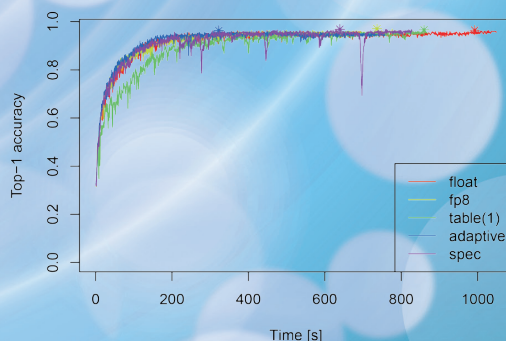
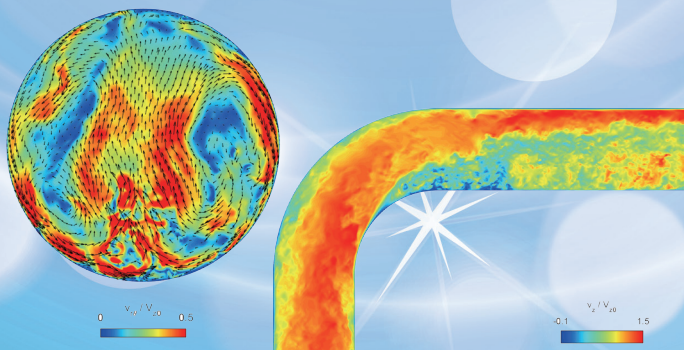
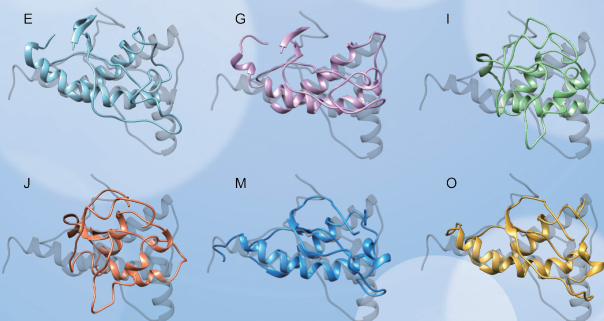
SUPERCOMPUTING NEWS

東京大学情報基盤センター スーパーコンピューティング部門

Vol.20, No.Special Issue 1

2018.2

特集：平成29年度「若手・女性利用者推薦」前期課題 成果報告



表紙の画像は、以下の課題より提供を受けております。

- 大規模並列環境における少精度型を用いたディープラーニングの学習精度の検証
(東京工業大学 情報理工学院 大山 洋介)
- 問題分割と対称性検知を用いた、高速なプランニングアルゴリズムの開発
(東京大学大学院 総合文化研究科 浅井 政太郎)
- **OpenFOAM** を用いた乱流スカラー輸送の大規模数値計算
(名古屋大学 工学研究科 恒吉 達矢)
- 拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析
(長崎大学大学院 医歯薬学総合研究科 大滝 大樹)

目 次

特集：平成 29 年度「若手・女性利用者推薦」前期課題 成果報告

平成 29 年度特集号発行にあたって	1
下川辺隆史（東京大学情報基盤センター）	
拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析.....	3
大滝大樹（長崎大学大学院医歯薬学総合研究科）	
大規模並列環境における少精度型を用いたディープラーニングの学習精度の検証.....	10
大山洋介（東京工業大学情報理工学院）	
OpenFOAM を用いた乱流スカラー輸送の大規模数値計算	17
恒吉達矢（名古屋大学工学研究科）	
第一原理計算に基づく大規模自己組織化ナノ構造における熱電特性の計算.....	23
新屋ひかり（横浜国立大学大学院工学研究院/大阪大学大学院基礎工学研究科 附属スピントロニクス学術連携研究教育センター）	
福島鉄也（大阪大学ナノサイエンスデザイン教育研究センター）	
吉田 博（東京大学スピントロニクス学術連携研究教育センター）	
将来の大規模メモリーコアプロセッサ環境に向けたビッグデータ基盤処理の性能評価.....	28
佐藤 仁（産業技術総合研究所人工知能研究センター）	
酵素反応機構の理解に向けた効率的にタンパク質の揺らぎを取り込む計算手法の開発... 35	
山田健太（理化学研究所杉田理論分子科学研究室）	
問題分割と対称性検知を用いた、高速なプランニングアルゴリズムの開発.....	40
浅井政太郎（東京大学大学院総合文化研究科）	

平成 29 年度特集号発行にあたって

下川辺 隆史

東京大学情報基盤センター

1. はじめに

本特集号は、東京大学情報基盤センター（以下、センター）で実施した平成 29 年度「若手・女性利用者推薦」前期に実施され本期間で終了した課題の成果報告をまとめたものです。

2. 若手・女性利用者推薦

センターでは、概ね 40 歳以下の若手研究者（学生を含む）及び女性研究者（年齢は問わない）による、スーパーコンピュータ、大規模ネットワーク機器などの大型計算資源を使用した研究を対象とした公募型プロジェクトを実施しています。

センターの教員による審査の上、年間で数十件程度の優れた研究提案を採択しております。採択された課題では申請した計算資源を無料で使用することができます。

前期・後期の半年単位で募集を行う一般枠（継続申請と再審査の上、最大で 1 年間の実施が可能）と、学部・大学院生を対象とし、主に夏期における利用を想定したインターン制度があります。一般枠、インターン制度それぞれについて、パーソナルコース（個人単位で実施）、グループコース（グループ単位で実施）があり、様々な研究スタイルに対応しています。

採択された課題のうち、「学際大規模情報基盤共同利用・共同研究拠点（JHPCN）」の萌芽型共同研究課題の条件を満たすものについては、同拠点共同研究課題審査委員会での審査の上、JHPCN の萌芽型共同研究課題としても採択され、更に毎年 7 月に開催される JHPCN のシンポジウムにて発表の機会が与えられる場合があります。

本制度に採択された課題は終了後、得られた成果をもとに、「学際大規模情報基盤共同利用・共同研究拠点（JHPCN）」の公募型共同研究（一般課題、国際課題、企業課題）等へと進展することが大いに期待されます。

本制度の詳細は、以下の Web ページをご覧ください。

<http://www.cc.u-tokyo.ac.jp/service/wakate/>

3. 平成 29 年度「若手・女性利用者推薦」前期 採択課題

平成 29 年度「若手・女性利用者推薦」前期の課題公募では、表 1 に示した 14 件の課題が採択されました。4 件の課題は、後期に継続され実施されています。本特集号では前期で実施期間が終了した 10 件の課題のうち 7 件の課題の成果報告が掲載されています。なお、採択された課題のうち 4 件の課題については、平成 29 年 12 月 6 日に開催された「若手・女性利用者推薦」成果報告会¹で、その成果が報告されました。

表1 平成29年度「若手・女性利用者推薦」前期 採択課題

課題名	代表者名	所属
Computational materials science and theory [‡]	Bhattacharyya Swastibrata	横浜国立大学大学院 工学研究院
拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析	大滝 大樹	長崎大学大学院 医歯 薬学総合研究科
brucite の(001)面における摩擦特性の決定 ^{†‡}	奥田 花也	東京大学 理学部
大規模並列環境における少精度型を用いたディープラーニングの学習精度の検証 [†]	大山 洋介	東京工業大学 情報理 工学院
OpenFOAM を用いた乱流スカラー輸送の大規模数値計算	恒吉 達矢	名古屋大学 工学研究 科
ハイブリッドクラスタシステムにおける通信削減QR分解実装*	高柳 雅俊	山梨大学大学院 総合 研究部
タンパク質-リガンド複合体への共溶媒効果の系統的解析*	山守 優	大阪大学 基礎工学研 究科
超大規模な線形数値計算に対する精度保証付き数値計算法の開発と評価 ^{†‡}	寺尾 剛史	芝浦工業大学大学院 理工学研究科
カスケード型超並列シミュレーションに立脚した遷移経路探索法の開発*	原田 隆平	筑波大学 計算科学研 究センター
第一原理計算に基づく大規模自己組織化ナノ構造における熱電特性の計算	新屋 ひかり	横浜国立大学大学院 工学研究院 (採択時：大阪大学大 学院 基礎工学研究科)
磁気回転不安定性によるブラックホール降着流の角運動量輸送機構の解明*	町田 真美	九州大学大学院 理学 研究院
将来の大規模メニーコアプロセッサ環境に向けたビッグデータ基盤処理の性能評価	佐藤 仁	産業技術総合研究所 人工知能研究センター
酵素反応機構の理解に向けた効率的にタンパク質の揺らぎを取り込む計算手法の開発 [†]	山田 健太	理化学研究所
問題分割と対称性検知を用いた、高速なプランニングアルゴリズムの開発	浅井 政太郎	東京大学 総合文化研 究科

* 後期に継続課題として採択された課題です。

† 「若手・女性利用者推薦」成果報告会で成果報告された課題です。

‡ 論文投稿中などのために、本特集号には成果報告が掲載されていない課題です。これらの成果報告は、次号以降に掲載される予定です。

拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析

大滝 大樹

長崎大学大学院医歯薬学総合研究科

1. はじめに

タンパク質が誤って折り畳まれることで形成される凝集体はアミロイドと呼ばれ、繊維状の構造をなす。これが身体の器官に異常蓄積すると、アルツハイマー病、パーキンソン病、プリオン病など、神経変性疾患を主とする様々な病を引き起こす。これまでの研究により、種々の疾患とアミロイドの関係が明らかになったものの、その凝集様態（構造、プロセスなど）については未だに分かっていない部分が多い。

本研究ではプリオンタンパクを取り上げる。プリオンでは、アミノ酸配列に変異を導入した場合に凝集様態が異なることが実験研究により示されている。我々はこれまでプリオンの分子動力学 (Molecular Dynamics: MD) シミュレーションを行い、変異がプリオンタンパクの構造を大きく揺らがせるという知見を得た。しかし、通常の MD 計算は局所安定構造周りを探索するのみで、タンパク質の複雑な自由エネルギー面の大域的な情報を得ることはできない。そこで本研究では、拡張アンサンブル法の 1 つであるレプリカ交換分子動力学 (Replica Exchange Molecular Dynamics: REMD) 法[1]を用いた分子シミュレーションを行い、プリオンタンパクが取り得る構造を幅広く探索する。野生型に加え、実験で凝集様態の変化が示唆されている代表的な変異体についても計算を行う。自由エネルギー面の差異やプリオンタンパクが取り得る構造について詳細に調べ、実験で見られる凝集様態の差異と関連付けて議論することを目的とする。

本研究は、『平成 29 年度 (前期) 東京大学情報基盤センター「若手・女性利用者推薦」』および『学際大規模共同利用・共同研究拠点 (JHPCN) 萌芽型共同研究課題』における採択課題「拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析」(拠点課題 ID: EX17302) によって行われ、現在も研究を続けている。本稿では野生型の計算結果について述べる。

2. プリオン

プリオンはタンパク質性の感染性因子である。異常型プリオンが正常型プリオンの形状を変化させることで感染・増殖していく。図 1 に正常プリオンタンパクの主要部 (残基番号 119-231) の構造を示す。正常プリオンタンパクの主要部は 3 つの α -helix (H1: 143-151, H2: 171-184, H3: 199-221) と 2 つの β -strand を持ち、H2 と H3 の間にはジスルフィド結合が存在する。異常型プリオンは β シート構造を多く持ち、これらが凝集してアミロイド繊維が形成されることが分かっているが、その正確な 3 次元構造は明らかになっていない。

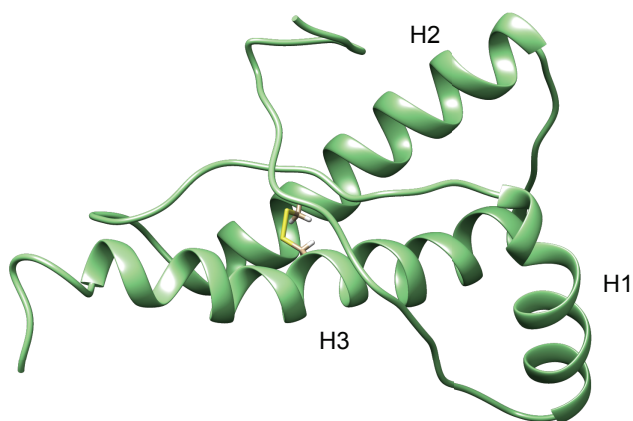


図 1: 正常プリオンタンパク主要部の構造

Protein Data Bank (PDB) ID: 2LSB の 119-231 残基を表示。H2 と H3 の間にはジスルフィド結合 (黄色) が存在する。UCSF Chimera[2]を用いて作成。

3. レプリカ交換分子動力学 (REMD) 法

図 2 に REMD 法の模式図を示す。REMD 法では、温度の異なる複数の系のコピー (レプリカ) の分子動力学計算を独立・並行的に実行し、隣り合うレプリカ間の温度を確率的に交換することで幅広い構造空間のサンプリングを可能とする。その一方、レプリカ交換を適切な確率 (20-30 % が良いとされている) で行うためには、系の大きさや計算条件に応じたレプリカ数を用意する必要があるため[3]、多くの計算資源が必要になる。本研究では正常型プリオンタンパクを初期構造として REMD シミュレーションを行う。

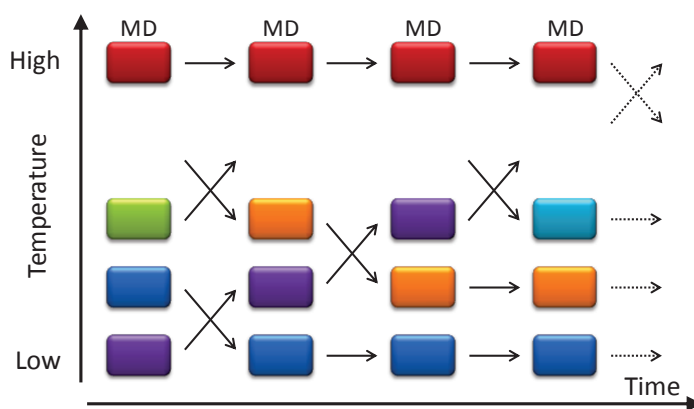


図 2: REMD 法の模式図

横軸は時間、縦軸は温度、四角は系のレプリカを表す。各レプリカでは温度の異なる独立した短い MD 計算が行われ、隣接した温度のレプリカペアについて温度交換の試行が行われる。試行が採択された場合のみ温度を交換する。

4. 計算条件

まず、計算のための系の構築を行った。正常プリオンタンパクの構造データを Protein Data Bank¹ (PDB) から入手し (PDB ID:2LSB), 主要部 (119-231) の構造を取り出した。次に、末端の影響を抑えるために N 末端側, C 末端側をそれぞれアセチル化, N-メチルアミド化することでキャップした。その後、プリオンタンパクの周囲に水分子を配置し、イオン濃度が 150 mM となるようにナトリウムイオンと塩化物イオンをランダムに配置した。最終的な系の原子数は 38,569 であった。

MD 計算は Gromacs[4]を用いて行った。タンパク質の力場は AMBER の ff99SB-ILDN[5], 水分子には TIP3P[6]を用いた。タンパク質の水素以外の原子に拘束をつけて最急降下 (steepest descent) 法で 5,000 ステップ, 共役勾配 (conjugate gradient) 法で 2,000 ステップの最小化を行った。次に、拘束をつけたまま系の温度を 310 K まで昇温し, 1 ns かけて拘束を徐々に緩めた後, 圧力・温度一定の NPT アンサンブルで 500 ps の MD 計算を行った。van der Waals 相互作用のカットオフは 1.2 Å とし, クーロン力による静電相互作用の計算には Particle Mesh Ewald 法を用いた。系の温度は Velocity-rescaling 法[7]により制御し, 圧力は Berendsen 法[8]により 1 atm に制御した。MD 計算の時間刻みは 2 fs とした。以上により得られた構造を初期構造として, 300 K~450 K の間に 84 個のレプリカを用意し, 各温度においてレプリカの交換を伴わない NPT アンサンブルでの平衡化計算を 2 ns 行った。その後, レプリカ当たり 160 ns の Production Run を行った。したがって, 全体で 160 ns * 84 レプリカ = 13.44 μ s 分のサンプリングを行ったことに相当する。レプリカ交換の試行は 4 ps (2,000 ステップ) 毎に行い, その度にスナップショットを取得した。これにより各レプリカについて 40,000 点, 合計 3,360,000 点のスナップショットを得た。このスナップショットを用いて解析を行った。

5. 計算結果

本稿では野生型の計算結果について述べる。図 3 に REMD 計算におけるレプリカ間の交換確率, レプリカ空間の遷移, 温度空間における遷移を示す。交換確率はシミュレーションの温度領域に渡ってほぼ一定であり, 平均交換確率は 30 % であった (図 3(a))。図 3(b) は 300 K のレプリカ番号を時系列でプロットしている。シミュレーション中, あらゆる番号のレプリカが 300 K になっており, レプリカ空間のランダムウォークを確認できる。また, 各レプリカは時間経過と共に

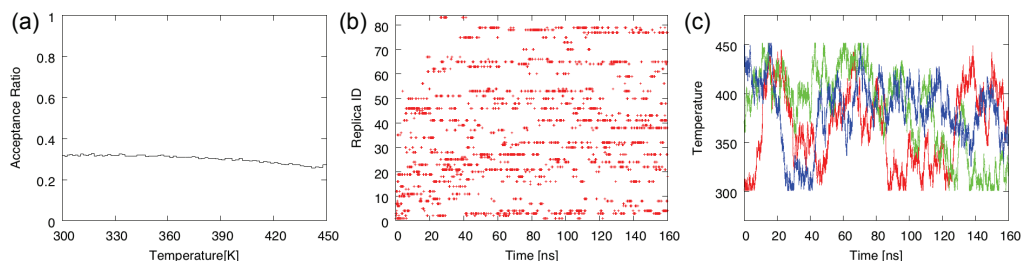


図 3: REMD 計算におけるレプリカ交換の様子

(a) Production Run におけるレプリカの交換確率。(b) 300 K におけるレプリカ番号の時間変化。(c) 代表的な 3 つのレプリカにおける温度変化 (赤: レプリカ 1, 緑: レプリカ 43, 青: レプリカ 83)。

¹ <https://www.rcsb.org/pdb/home/home.do>

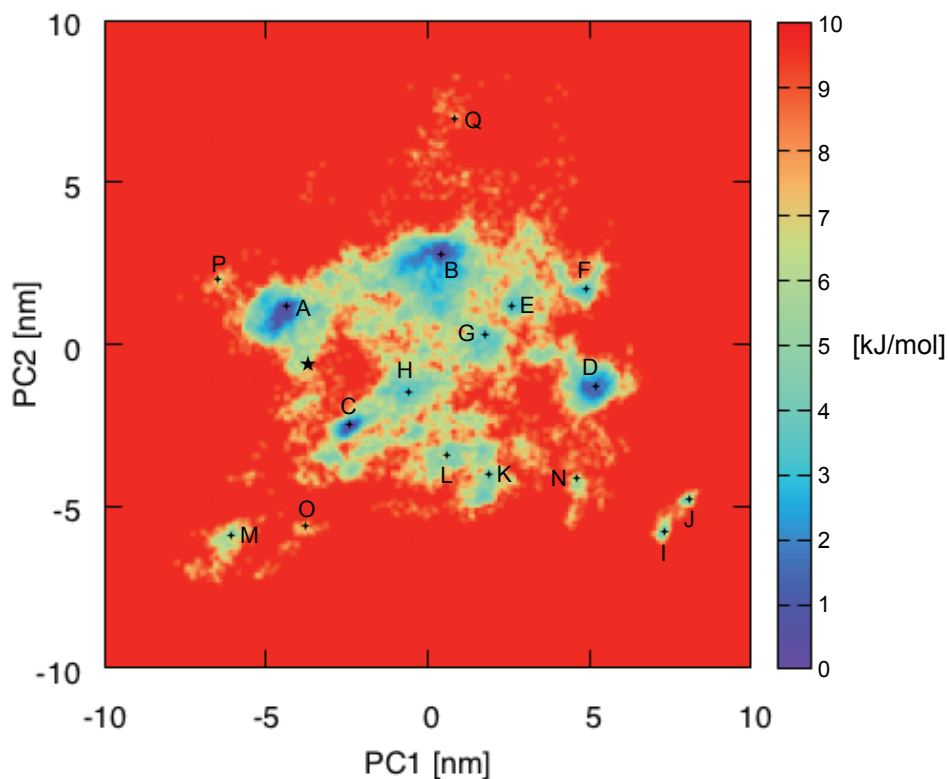


図 4: 300 K における自由エネルギー面

横軸は第一主成分 (PC1), 縦軸は第二主成分 (PC2)。★印は主成分空間におけるオリジナルの PDB 構造の位置, A-Q は代表的な極小点を示す。

最低温度から最高温度を経巡っており (図 3(c)), 温度空間のランダムウォークが実現されていることが分かる。これらにより, REMD 計算が適切に行われていることが確認できる。

次に 300 K のトラジェクトリのスナップショットを用いて主成分解析を行った。図 4 に第一主成分と第二主成分を軸にして得られた自由エネルギー面を示す。PDB の構造 (★印) 以外にも複数のエネルギー極小点を確認できる。これらの極小点に対応する構造をスナップショットから抽出し, DSSP[9, 10]による二次構造解析を行った結果が図 5 である。H2 (171-184) と H3 (199-221) は両者の間に形成されているジスルフィド結合 (図 1) により比較的 helix 構造を保持しているが, 構造 E, G, I, J, M, O では H1 (143-151) の helix 構造を保てていないことが分かる。これらの構造を図 6 に示した。PDB の構造と比較すると明らかに H1 の部分が大きく崩れていることが分かる。構造 I, J では H2 や H3 も大きく変形している。このように, REMD 法を用いることで通常の MD 計算よりも幅広い構造空間のサンプリングに成功した。

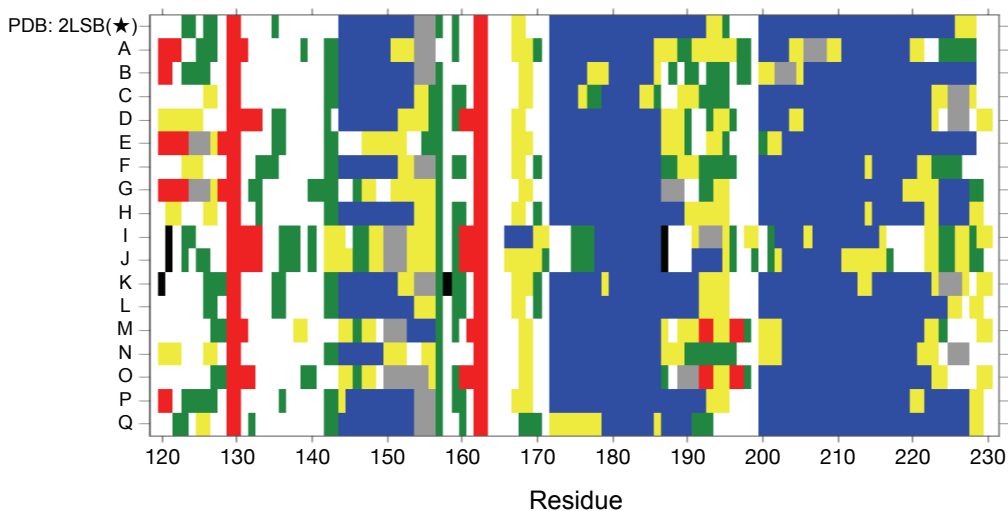


図 5: 各エネルギー極小点におけるプリオンタンパクの二次構造

赤: β -sheet, 黒: β -bridge, 緑: Bend, 黄: Turn, 青: α -helix, グレー: 3_{10} -helix, 白: Coil。横軸は残基番号, 縦軸は構造の ID を表す (★およびA-Qは図4を参照)。二次構造はDSSPを用いて解析を行った。

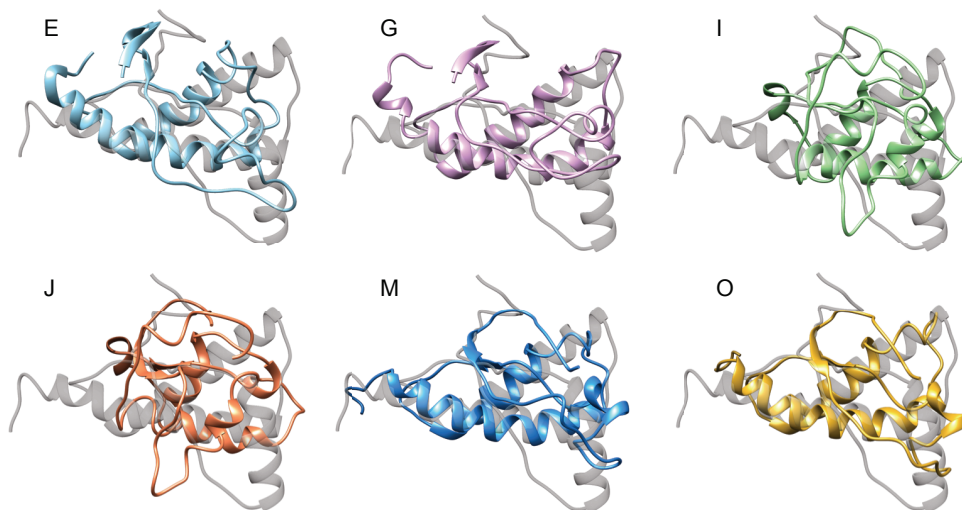


図 6: エネルギー極小点 (E, G, I, J, M, O) におけるプリオンタンパクの構造

PDBの構造 (灰色) と重ねて比較している。UCSF Chimera[2]を用いて作成。

6. まとめと今後の課題

本研究ではプリオンタンパクの構造探索を目的として, REMD法を用いた分子シミュレーションを行った。シミュレーションにより得られたスナップショットについて主成分解析を行い, 主成分軸を用いて描いた自由エネルギー面を参照することで複数の局所安定構造を得た。自由エネ

ルギー面およびDSSPによる二次構造解析の結果が示すように、幾つかの構造はオリジナルのPDB構造から大きく異なるものであり、特にH1の構造変化が最も顕著であった。

本稿では野生型のREMDシミュレーションの計算結果について述べた。変異体の計算は現状サンプリング時間が不十分であるため今後は追加計算を予定している。野生型と変異体の結果を比較することで、変異の導入によりどのような構造変化がもたらされるか、構造変化が顕著な変異は何かを調べる予定である。プリオンタンパクの凝集様態については多くの実験報告があり、幾つかの変異体について凝集様態が大きく変化することが分かっている。変異体についての計算データを蓄積し、計算と実験との比較検証を目指す。

謝 辞

本研究は『平成29年度(前期)東京大学情報基盤センター「若手・女性利用者推薦」』および『学際大規模共同利用・共同研究拠点(JHPCN)萌芽型共同研究課題』における採択課題「拡張アンサンブル法を用いたタンパク質の構造変化と変異が及ぼす影響の解析」(拠点課題ID:EX17302)によって行われた。本研究における数値計算は東京大学情報基盤センターのReedbush-U上で行われた。また、田口謙助教(長崎大学大学院医歯薬学総合研究科)にはプリオンの実験研究者の立場から多くの有益な助言を頂いた。

参 考 文 献

- [1] Y. Sugita, Y. Okamoto, “Replica-exchange molecular dynamics method for protein folding”, *Chem. Phys. Lett.* 314, 141-151 (1999)
- [2] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng, T. E. Ferrin, “UCSF Chimera—a visualization system for exploratory research and analysis”, *J. Comput. Chem.* 25, 1605-1612 (2004)
- [3] A. Patriksson, D. van der Spoel, “A temperature predictor for parallel tempering simulations”, *Phys. Chem. Chem. Phys.* 10, 2073-2077 (2008)
- [4] S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, E. Lindahl, “GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit”, *Bioinformatics* 29, 845-854 (2013)
- [5] K. Lindorff-Larsen, S. Piana, K. Palmo, P. Maragakis, J. L. Klepeis, R. O. Dorr, D. E. Shaw, “Improved side-chain torsion potentials for the AMBER ff99SB protein force field”, *PROTEINS: Struct. Funct. Gen.* 78, 1950-1958 (2010)
- [6] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, M. L. Klein, “Comparison of simple potential functions for simulating liquid water”, *J. Chem. Phys.* 79, 926-935 (1983)
- [7] G. Bussi, D. Donadio, M. Parrinello, “Canonical sampling through velocity rescaling”, *J. Chem. Phys.* 126, 014101 (2007)
- [8] H. J. C. Berendsen, J. P. M. Postma, A. DiNola, J. R. Haak, “Molecular dynamics with coupling to an external bath”, *J. Chem. Phys.* 81, 3684-3690 (1984)
- [9] W. Kabsch, C. Sander, “Dictionary of protein secondary structure: pattern recognition

of hydrogen-bonded and geometrical features” , *Biopolymers* 22, 2577-2637 (1983)

[10] W. G. Touw, C. Baakman, J. Black, T. A. H. te Beek, E. Krieger, R. P. Joosten, G. Vriend, “A series of PDB related databases for everyday needs” , *Nucleic Acids Research* 43, D364-D368 (2015)

大規模並列環境における少精度型を用いた

ディープラーニングの学習精度の検証

大山 洋介

東京工業大学 情報理工学院

1. はじめに

深層学習(ディープラーニング, 以下 DL)とは生物の神経回路を模した計算モデルである Deep Neural Network(以下 DNN)を用いた機械学習の一手法である。DL は ILSVRC(画像認識に関する競技会)で CNN(Convolutional Neural Network)が他の古典的な機械学習アルゴリズムと比較して高い推論精度を達成した[1]ことを一つのきっかけとして機械学習分野における一種のブレイクスルーとして近年非常に注目されている。

DNN や CNN の計算は行列同士の積(General Matrix Multiply, GEMM)や畳み込み演算が計算時間において支配的であると言われていることから, 学習や推論に単体の GPU や GPU クラスタが用いられることが多い。複数の GPU を用いて学習を行う場合, データ並列学習と呼ばれる一回の学習ステップで使用されるデータサンプルの集合(ミニバッチ)を GPU 数に分割して計算を行う手法が広く用いられている。この際, 高いスケーラビリティを維持するためには GPU ごとのサンプル数を一定に保って並列数を増加させる(弱スケールする)必要があるが, サンプル数が過度に増加することによってモデルパラメータの更新量のランダム性が低下し, 推論精度が悪化することが指摘されている。

本研究では, DL が自然科学的なデータセットと冗長なモデルを用いることから計算要素の数値誤差に対してロバストであるという仮定を用いた上で, 通信データを低い精度で表現することで推論精度を損なわずに並列化を行う手法を提案した。特に, 特定の通信精度に対する推論精度の悪化が未知である場合に, 精度を適合的に変更することで常に最適な通信手法を選択する手法の予備評価を行った。

本報告では, 東京大学情報基盤センター「若手・女性利用者推薦」平成 29 年度前期研究課題「大規模並列環境における少精度型を用いたディープラーニングの学習精度の検証」によって得られた研究成果を報告する。

2. データ並列学習

DL におけるデータ並列学習とは, 主に確率的勾配降下法(Stochastic Gradient Descent, 以下 SGD)を用いてモデルパラメータの学習を行う際の並列化手法のひとつである。

SGD では時刻 t におけるパラメータ $W^{(t)}$ をデータセットからランダムに選択された部分集合 D (ミニバッチ)によって次のように反復的に更新する:

$$W^{(t+1)} \leftarrow W^{(t)} - \frac{\eta^{(t)}}{|D|} \sum_{x \in D} \frac{\partial L}{\partial W}(x; W^{(t)})$$

ここで $\partial L/\partial W$ はコスト関数のパラメータについての勾配、 $\eta^{(t)} > 0$ は学習係数と呼ばれる更新度合いを決定するハイパーパラメータである。

データ並列学習では SGD の各データサンプル $x \in D$ についての計算に依存性がないことを利用し、複数の GPU で勾配を並列に計算し、その後勾配の総和を GPU 間通信により計算することで並列化を行う。この際、各 GPU が計算するミニバッチ D の部分集合 D' についての勾配 $\sum_{x \in D'} \frac{\partial L}{\partial W}(x; W^{(t)})$ は GPU 内で計算できることから、並列数にかかわらず各 GPU が送信・受信するデータ量は一定である(パラメータ数に比例する)。この通信には、プロセス間のデータをリダクションし結果を全プロセスが取得する all-reduce が使用される(図 1)。

データ並列学習では 1 回のステップ(図 1 の 1.~3.)で GPU が要する計算時間は $O(\text{ミニバッチサイズ}/\text{GPU 数})$ であるのに対し、通信時間はデータサイズが一定であるという性質から一般に $O(\log(\text{GPU 数}))$ となる。よって、ミニバッチサイズを一定に保ったまま(計算内容を保ったまま)並列数を増加させた場合には通信時間がボトルネックとなる傾向にある[4]。

一方で、並列数の増加にともなってミニバッチサイズを増加させる(弱スケールによる並列化を行う)場合は通信性能による性能低下が起きりづらいが、一般に SGD においてミニバッチサイズを過度に増加させると推論精度が悪化するため、サンプルあたりの計算速度の向上が学習自体の速度向上に必ずしも寄与するわけではない。近年ではモデルパラメータの最適化手法の改良などの理論的な方面からのアプローチにより単一モデルの学習を推論精度の悪化なく 1000 GPU 程度までスケールする成果が報告されているが[2]、これらの手法が多様な DNN のすべてについて適用可能であるかどうかは依然明らかではない。

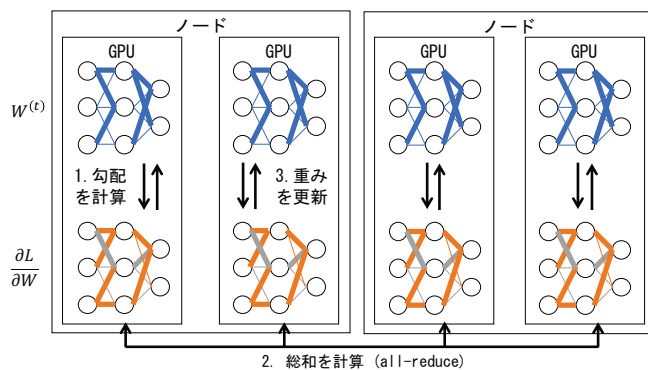


図 1: データ並列学習。

1. コスト関数の勾配計算, 2. GPU 間通信による勾配の総和の計算, 3. 重み(パラメータ)更新を繰り返すことによって学習が進行する。

3. 低精度型を用いた通信手法に関する先行研究

DL は自然科学的なデータセットを用いて学習を行うことが多く、また学習手法としてもランダム性をともなう SGD が用いられることから、厳密な計算精度を要求するような HPC アプリケーションと比較して個々の計算要素の数値誤差に対して比較的ロバストであると言われている。これを反映した例として、NVIDIA 社の GPU では Pascal 世代より半精度浮動小数点数演算がハードウェアレベルでサポートされ、さらに Volta 世代では Tensor Core と呼ばれる 4×4 行列 (半精度) 同士の行列積を計算するユニットが導入されるなど、DL 向けの低精度な演算がサポートされつつある。

このような低精度な演算・データ型を通信部分に導入することにより通信量・通信時間を削減する手法についても研究が行われている。例えば、Seide ら [5] はコスト関数の勾配を符号により 1 bit に符号化する手法 (1-bit SGD) を提案しており、この手法は Microsoft が開発している DL フレームワークである CNTK に実装されている。しかし 1-bit SGD では all-reduce 通信の際に 2 回の符号化・復号化を行う必要があり、また通信中に発生する符号化誤差を次の通信に持ち越す必要があるなど、通信中に (単純な要素ごとの加算と比較して) 複雑な処理が必要となる問題がある。

一方で我々は 8 bit の浮動小数点数型 (以下 fp8) を用いて MPI のユーザ定義型・演算によって低精度な all-reduce 通信を行う手法を提案した [4]。fp8 は半精度浮動小数点数型の上位 8 bit を使用した型であることから、半精度型の演算・変換に対応した CPU・GPU で容易に加算や変換を行うことができる (図 2)。fp8 を用いた加算アルゴリズムは Tsubame-KFC/DL の 16 ノードで 256×10^6 要素の MPI_Allreduce を実行した場合について単精度型を用いた場合と比べて 3.2 倍の高速化を達成したほか、CaffeNet と GoogLeNet という二つの CNN の学習においてミニバッチサイズを固定したまま推論精度を損なわずにそれぞれ 2.7 倍、2.2 倍の高速化を達成した。

一方で、fp8 を用いた場合は MPI プロセス数やオーバーフローを防ぐためのスケールリングを行う回数によって学習後の推論精度が大きく異なる場合があることが明らかになった。しかし、一般にデータ型の精度が推論精度にどの程度影響するかどうかは明らかではなく、ミニバッチサイズの問題と同様に計算速度と学習速度のトレードオフを調整する必要性が依然存在する。

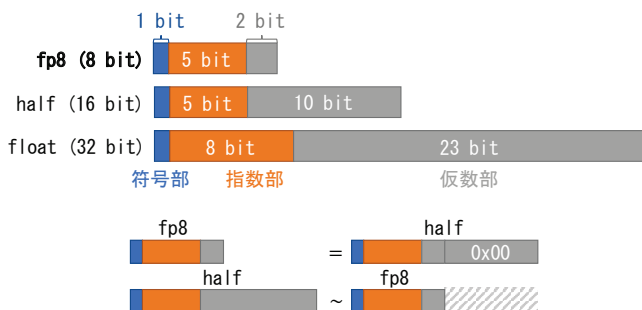


図 2: 浮動小数点数型の構成 (上) および fp8 と半精度浮動小数点数型 (half) の変換 [4]。

fp8 と half は下位 8 bit の追加または無視により容易に変換することができる。

4. 提案手法

本研究では通信に使用する型(以下, 通信手法)を適恰的に選択することにより, 低精度な通信手法による推論精度の悪化を防止しつつ通信量を削減する手法を提案する。提案手法では, 以下の手順によって学習時に周期的に最適な通信手法を選択する(図 3)。

1. 学習のある時点でモデルパラメータや momentum 等の学習の状態に関するデータをコピーし, その時点から異なる通信手法を用いて並行に一定ステップの学習を行う。
2. すべての通信手法について, さらに一定ステップの学習を継続した際に得られる推論精度を学習曲線(推論精度やコスト関数をステップ数や時間を軸としてプロットした曲線)の性能モデル[6]により予測する。
3. 単位時間あたりの最も推論精度の向上が大きい通信手法を選択し, 1. で学習した地点からさらに一定ステップの学習を行う。
4. ユーザが指定したステップ数に達するまで 1. 以降を繰り返す。

今回は提案手法で得られる推論精度に関する予備評価という目的で, 1. の複数の通信手法による学習を逐次的に行い最良の手法を選択する手法を DL フレームワークである Caffe[7]に実装した。学習状態の保存・復元については GPU メモリ上にデータをコピー・復元するよう実装した。

通信手法としては, 単精度(float)と fp8 のほかに, より精度が低く通信量を削減できる手法として 1 つの数値を N bit ($N = 1, 2, 4, 8$) で符号化する手法を実装した(以下 table(N))。この手法では fp8 でスケール係数を決定する手法と同様にテンソルごとに勾配の絶対値をランダムサンプリングし, その内の $99 \times \frac{i}{2^{N-1}}$ パーセンタイル ($i = 1, 2, \dots, 2^{N-1}$) の値を絶対値とした代表値とした符号化テーブルを作成する。通信の際には 1-bit SGD と同様にテーブルを用いて符号化・復号化を行う。

学習曲線のモデルについては Domhan ら [6] により提案されている初等関数 5 種類の線形結合をモデルとし, 過去に選択された通信手法による学習曲線の全サンプルを用いて非線形最小二乗法によりフィッティングを行った。

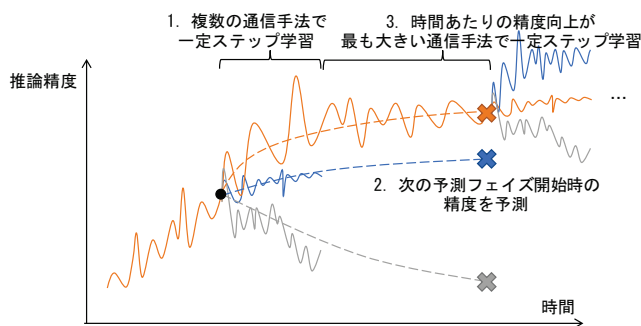


図 3: 適恰な通信手法の選択手法。

1. ~3. を繰り返すことで, その時点で最も時間あたりの推論精度の向上が最も高い通信手法で学習を行う。

5. 評価

本研究では提案手法を DL フレームワークの Caffe[7]に適用し, AlexNet[1]の一部のレイヤーの順番を入れ替えたネットワークである CaffeNet と GoogLeNet[8]の2つの CNN の学習について Reedbush-H 上で評価を行った。データセットとしては ILSVC 2012 データセット[3]のうちランダムに選択した 16 クラスのみを使用し, 学習係数等のハイパーパラメータについては Caffe に付属しているサンプルファイルの設定を変更せずに用いた。また, ミニバッチサイズはどちらの CNN についても 256 とし, それぞれ 90, 120 epoch の学習を行った。

学習に使用する並列数は, ミニバッチサイズ 256 の場合の 1 ステップの実行時間を計測した上で CaffeNet については 2 ノード (P100 GPU×4), GoogLeNet については 8 ノード (P100 GPU×16) とした。また, ノード内 (2 GPU) の勾配の加算については NVIDIA GPU 用の集団通信ライブラリ NCCL を用いて単精度で行い, その後のノード外の加算について前述の単精度・fp8・table(1)による加算を実装した。

提案手法を用いた際の CaffeNet の学習曲線を図 4 に示す。なお, 提案手法の 1. と 3. で用いるステップ数としては, 今回試行した内で特に提案手法の学習時間が短かったそれぞれ 200, 400 ステップを用いた。結果より, fp8 と table(1)を用いた場合は一定 epoch の学習時間について float と比較してそれぞれ 2.3, 2.5 倍の高速化を達成した一方で, それぞれ top-1 accuracy (推論精度) はそれぞれ 1.7%, 13.5% 程度低下した。一方で提案手法 (図 4 の “spec”) は 2.1 倍の高速化を達成し, かつ推論精度については float をわずかに上回る結果となった。提案手法で実際に選択された通信手法としては, 初回のステップのみ float が選択され, その後は fp8, table(1), adaptive のいずれかが選択された。これにより, 通信手法を適格的に変更することで推論精度を保ったまま学習することができたと考えられる。

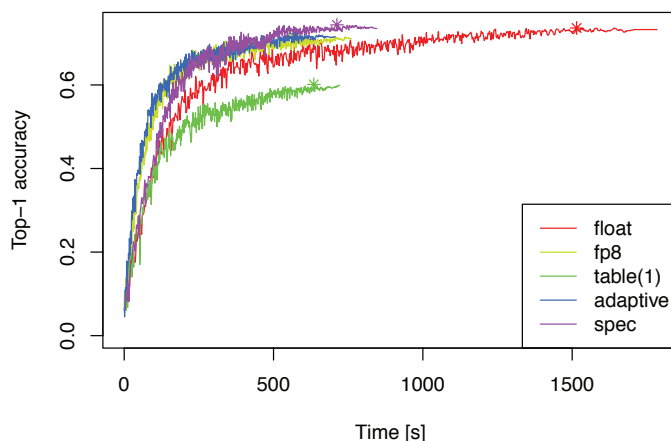


図 4: 提案手法 (“spec”) を用いた CaffeNet の学習結果。

提案手法の学習曲線は各ステップで選択された通信手法のみによる経過時間を用いてプロットした。

“adaptive” はパラメータのテンソルごとに最も高速な通信手法を選択した場合の結果を表す。

各項目の星印は最も Top-1 accuracy が高い点を表す。

提案手法を用いた際の GoogLeNet の学習曲線を図 5 に示す。図 5 では 1. と 3. のステップ数としてそれぞれ 100, 200 を用いた。CaffeNet では通信手法により推論精度が悪化するケースがあった一方で、GoogLeNet を用いた評価ではいずれの通信手法についても最良 top-1 accuracy の差が 1%以内にとまった。この理由としては、CaffeNet と GoogLeNet のレイヤー数や畳み込みフィルタ等のネットワーク構造の違いや並列数の違いによって計算誤差に対する影響度合いが異なったためであると考えられる。

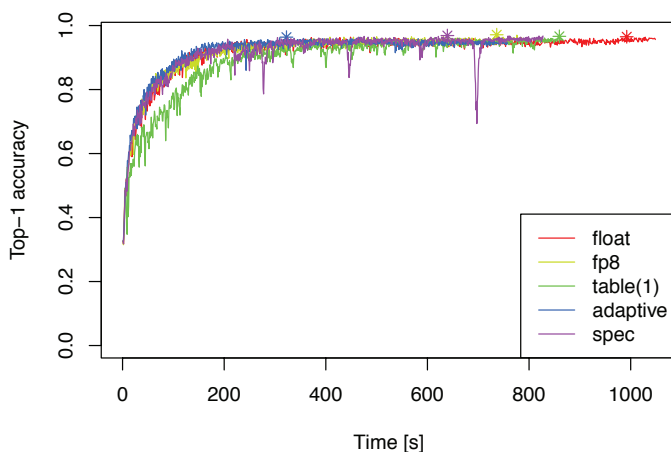


図 5: 提案手法(“spec”)を用いた GoogLeNet の学習結果。

6. まとめと今後の課題

結果より、適恰的に通信手法を選択することによって単一の低精度通信手法を用いる場合に生じる悪化を防止できるケースが確認できた。一方で、本報告では提案手法 1. の部分については逐次的に通信手法の評価を行っており、実質的な学習時間の短縮は実現できていない。よって今後の課題としては、この投機的な学習についてマルチジョブを用いて並行に実行する手法を実装・評価することが挙げられる。

参考文献

- [1] A. Krizhevsky, I. Sutskever, and H. Geoffrey E. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems* 25 (NIPS2012), (11):1-9, 2012.
- [2] T. Akiba, S. Suzuki, and K. Fukuda. Extremely Large Minibatch SGD: Training ResNet-50 on ImageNet in 15 Minutes. *arXiv e-prints*, 2017.
- [3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, (3):211-252, 2015.

- [4] 大山洋介, 野村哲弘, 佐藤育郎, 松岡聡, “ディープラーニングのデータ並列学習における少精度浮動小数点数を用いた通信量の削減,” 情報処理学会研究報告, Vol. 2017-HPC-158, No. 30, pp. 1-10, 2017.
- [5] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. In Proceedings of the Annual Conference of the International Speech Communication Association (Interspeech 2014), pp. 1058-1062, 2014.
- [6] T. Domhan, J. T. Springenberg, and F. Hutter. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2015), pp. 3460-3468, 2015.
- [7] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. arXiv e-prints, 2014.
- [8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2015), 2015.

OpenFOAM を用いた乱流スカラー輸送の大規模数値計算

恒吉達矢

名古屋大学 工学研究科

1. はじめに

火力・原子力発電プラントの炭素鋼配管における減肉事象の一因として、流れ加速型腐食が挙げられる(Dooley and Chexal, 2000)。流れ加速型腐食は基本的には腐食による減肉 (corrosion) であり、気液二相流条件下で生じる液滴や気泡の配管内壁への衝突による機械的な作用による減肉 (erosion) とは区別される。流れ加速型腐食の本質は、配管内壁から流体中への鉄イオンの物質伝達現象であり、流体力学因子として壁面の物質移行係数 k_c を正確に算出することが求められている。著者を含む研究グループでは、電気化学的手法による物質移行係数の直接的な測定実験とともに、数値計算により物質移行係数の算出と乱流スカラー輸送の解析を行ってきた。流れ加速型腐食が発生する個所は、オリフィスやエルボのように、流れが剥離や再循環、旋回といった3次元的に複雑なパターンであるとともに、非定常性が強い場合が多い。そのため、数値計算においては、Large Eddy Simulation (LES) による解析が有効である。LES ではグリッドスケール以下の渦の影響のみをモデル化することにより、計算負荷を低減しつつ、流動場の特徴的な構造を高い精度で解析することを可能としている。

著者らは、LES を用いた流動場計算とともに、鉄イオンの濃度場を模擬したパッシブスカラーとして温度場を計算し、壁面の物質移行係数 k_c に相当する熱伝達率 h を求めた(恒吉他, 2017)。浮力や密度変化を考慮しないパッシブスカラーとして温度場を計算する限り、温度場と濃度場の支配方程式の差異はない。ただし、計算条件として、水の熱伝達における無次元拡散係数であるプラントル数 Pr は 5.847 であり、流れ加速型腐食の物質伝達における無次元拡散係数であるシュミット数 Sc (100~1000)とは値が異なっている。流動場中の最小渦スケールであるコルモゴロフ・スケール η に対して、熱や物質のスカラー輸送では $Pr > 1$ あるいは $Sc > 1$ の条件において、変動の最小スケールはバチェラー・スケール $\eta_s = \eta/\sqrt{Pr}$ 、 $\eta_s = \eta/\sqrt{Sc}$ となる。LES では必ずしも、最小スケールまで直接的に計算するわけではないが、高プラントル数条件において計算精度を維持するためには相応にメッシュ数を増加させた大規模計算が必要となる。

本研究では、数億セルのメッシュを設定し、レイノルズ数が 15000 のエルボ管流れを対象に、プラントル数が 1、5.847、25、64 の 4 条件における温度場計算を行い、プラントル数の違いが温度場の形成や壁面の熱伝達に及ぼす影響を明らかにすることを目的とした。

2. 数値計算方法

計算にはオープンソースコードである OpenFOAM(version 2.2.1)を用いた(OpenCFD, 2015)。計算条件を表 1 に示す。鉄イオンの濃度場を模擬したスカラー場として温度場の計算を行うため、OpenFOAM の標準ソルバのひとつである pimpleFoam をベースとしてコードの追加および修正を行った。ひとつの流動場計算に対して、異なるプラントル数条件の温度場計算を同時に行うようにしている。これにより、実験的測定では不可能な、完全に同一な流動場における、異なる条件のスカラー場の解析を可能としている。

表 1 LES の計算条件

Software	OpenFOAM (2.2.1)	
Solver	Modified pimpleFoam	
Turbulent model	Dynamic Smagorinsky model	
Fluid	Kinematic viscosity	$\nu=8.5735\times 10^{-7}$ m ² /s
	Prandtl number	$Pr=1, 5.847, 25, 64$
	Turbulent Prandtl number	$Pr_t=0.9$
Time integration	Implicit unsteady	
Transient term	2nd-order backward difference scheme	
Convection term	2nd-order total variation diminishing scheme	
Diffusion term	2nd-order central difference scheme	
Boundary conditions	Inlet	Fully developed
	Outlet	Constant total pressure
	Wall	Non slip
Reynolds number	$Re=V_{z0}D/\nu=1.5\times 10^4$ (V_{z0} : Cross sectional mean streamwise velocity)	

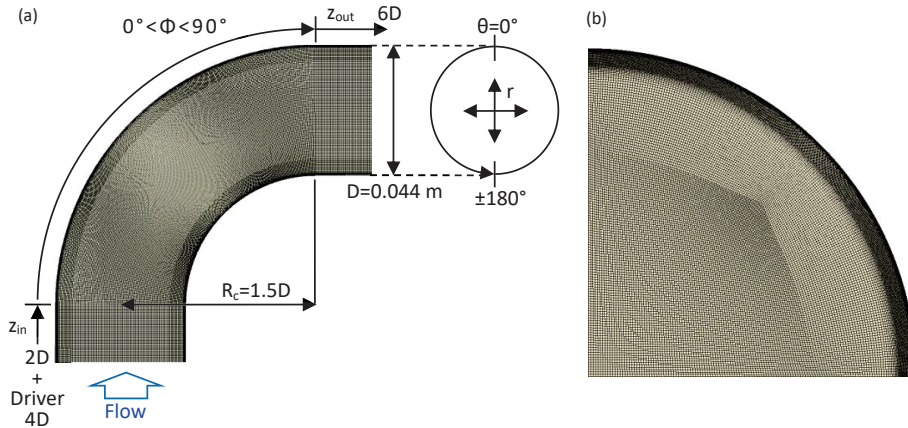


図 1 (a)計算領域とメッシュの概要図、(b)円管断面におけるメッシュ概要。

LES の乱流モデルは動的スマゴリンスキーモデル(Germano et al., 1991)を用いた。OpenFOAM の標準ライブラリにおける動的スマゴリンスキーモデル(homogeneousDyn Smagorinsky)は計算対象として一様乱流を想定している。そこで、壁乱流の計算で使用するための修正(OpenFOAM Foundation, 2015)を行った。サブグリッドスケールの乱流熱拡散の算出に用いる乱流プラントル数 Pr_t は 0.9 とした。管直径 D および断面平均流速 V_{z0} を用いて定義されるレイノルズ数 Re は 15000 とした。時間発展は 2 次精度後退差分を用いた陰解法である。移流項は制限関数に superbee を用いた TVD スキームを適用した。拡散項は 2 次精度中心差分である。

図 1 に計算領域の概要を示す。エルボ管の形状は、曲り中心から管軸中心までの距離で定義される曲率半径が $R_c = 1.5D$ の 90 度エルボである。計算領域は、エルボの上流側および下流側に $6D$ の長さを設定した。入口には mappedPatch という境界条件を用いて、下流方向長さ $4D$ の位置の断面における物理量を参照して入口境界の値を定めている。これにより、入口から下流方向長さ

4Dの区間は発達乱流を生成するドライバーとして機能する。温度場計算の境界条件は、プラントル数の違いによらず、入り口での断面平均温度を 300 K、壁面を 310 K の等温加熱条件とした。統計処理時間 t は、乱流の初期発達を除いた 2 秒から 12 秒までの 10 秒間とした。これは、 $V_{z0}t/D = 66$ であり、計算領域を断面平均流速で 5 回程度流下するのに要する時間である。

メッシュはヘキサメッシュ(6面体セル)を用いた。セルサイズ L は円管発達乱流における摩擦速度 u_τ を用いた無次元数 $L^+(=u_\tau L/\nu)$ で表した場合に、壁垂直方向が 0.12 から 1.1、周方向が 1.2、流れ方向が 3.7 である。これは、プラントル数が最も大きな $Pr = 64$ の温度場計算において、相対的なセルサイズ $L^{++}(=\sqrt{Pr}L^+)$ で表した場合に、壁垂直方向が 1.0 から 8.8、周方向が 10、流れ方向が 30 となるようにしている。メッシュ数は、円管断面において約 14 万セル、計算領域全体で約 4.7 億セルである。

本計算には東京大学の Oakforest-PACS を用いた。64 ノードを使用し、4096 並列での計算を実施した。Xeon Phi の物理 68 コアのうち、OS ジッタを避けるため最初の 2 コア (0, 1) 以外から 64 コアを使用し、Hyper-Threading は使用しなかった。プリポスト処理の一部には、名古屋大学の大規模共有メモリシステムである SGI UV2000 を用いた。

3. 結果および考察

3.1 エルボ管の流動場

図 2 (a)に管軸を含む垂直断面における瞬時の流れ方向流速のコンター、図 2 (b)にエルボ出口の円管断面における瞬時の 2 次流れをコンターとベクトルで示す。垂直断面のコンターをみると、エルボの腹側から流れの剥離が生じていることが確認できる。本計算条件の $Re = 15000$ では、エルボの曲がり角がおよそ $\phi = 45^\circ$ において、流れの剥離が生じている。エルボへの流入時には、主流部の高速領域は腹側に偏っている。エルボ内腹側からの流れの剥離とともに、高速領域は管軸中心へ移り、さらにエルボを通過した後は背側への偏流となる。剥離流れの後流の腹側出口周辺では、流れ方向の流速が小さくなっている。瞬時場で見るとわずかに逆流も生じている。ただし、平均流速において逆流は生じていない。エルボ出口の円管断面でもわかるとおり、円管中心の主流部は、定常的に背側へ向かっている。壁面に近い領域では、管壁に沿って背側から腹側へ回り込む旋回流が発生している。この旋回流は非定常性が強く、旋回流が左右から流れ込むエルボ出口周辺の腹側では、細かな擾乱が生じており、流れ方向成分と 2 次流れ成分ともに流速変動が大きい。

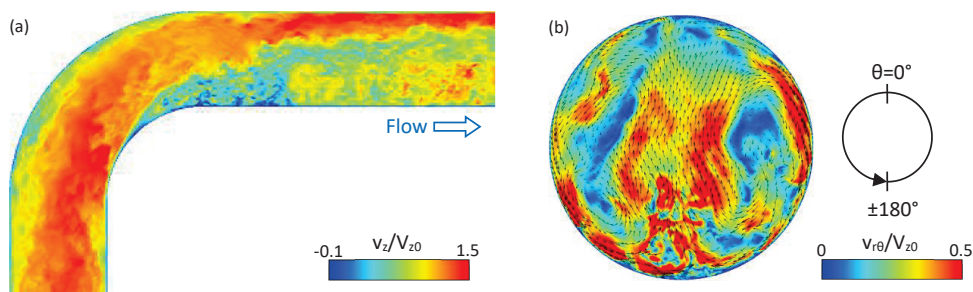


図 2 (a)垂直断面における瞬時の流れ方向流速、(b)エルボ出口における瞬時の 2 次流れ。

3.2 異なるプラントル数条件における温度場

図3に垂直断面における瞬時の温度場を示す。時刻は図2に示した流動場と同一である。瞬時温度 θ_t は時間平均したバルク温度 θ_{tb} との差を取り、壁面とバルクの温度差($\theta_{tw} - \theta_{tb}$)により無次元化している。プラントル数の違いにより、温度の変化量が大きく異なることがわかる。 $Pr = 1$ の場合は、エルボ腹側の剥離流れの後流において、温度が大きく上昇している。また、エルボ出口から下流にかけて腹側の広い範囲で温度の上昇が確認できる。 $Pr = 5.847$ の場合は、 $Pr = 1$ の場合と比較して、エルボ腹側における温度の上昇が抑えられていることがわかる。エルボ出口における温度分布をみても、腹側の温度変化は $Pr = 1$ の場合の半分程度まで減少している。プラントル数の増加により、エルボ腹側の温度変化はさらに小さくなっていく。 $Pr = 25$ と $Pr = 64$ の場合は、コンターにおいて、剥離点の下流側の比較的剥離点に近い範囲では温度の上昇が確認できるが、エルボ出口周辺の温度変化はわずかである。ただし、プラントル数の増加により温度上昇は減少するが、流速変動によって生じた高温と低温の温度変動の分布傾向はほとんど同一である。剥離せん断層や剥離の後流の流速変動が大きな領域で顕著であるが、図2に示した流動場の擾乱に対応した温度変動が確認できる。温度の変化量はプラントル数に依存するが、温度の分布傾向は流動場によって一意に定まっている。

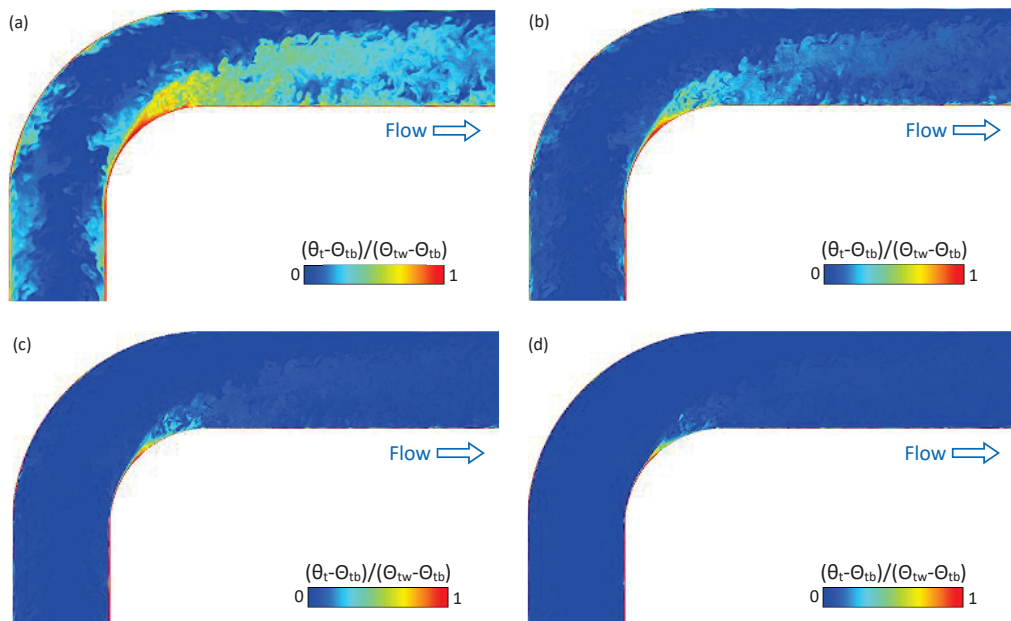


図3 垂直断面における瞬時の温度分布。(a) $Pr=1$, (b) $Pr=5.847$, (c) $Pr=25$, (d) $Pr=64$ 。

3.3 エルボ管壁面におけるヌセルト数のプラントル数依存性

図4に壁面の熱伝達率 h を無次元化したヌセルト数 Nu として示す。壁面は展開図で示しており、横軸が流れ方向位置、縦軸が周方向位置である。流れ方向位置は、エルボ入り口を $z = 0$ としており、 $z/D = 2.36$ がエルボ出口である。プラントル数を大きくすると、ヌセルト数の値はおおよそプラントル数の0.4乗に比例して大きくなる。異なるプラントル数において、エルボ管壁面のヌセルト数の分布傾向は概ね一致している。エルボ入口のわずかに下流側からエルボ出口にか

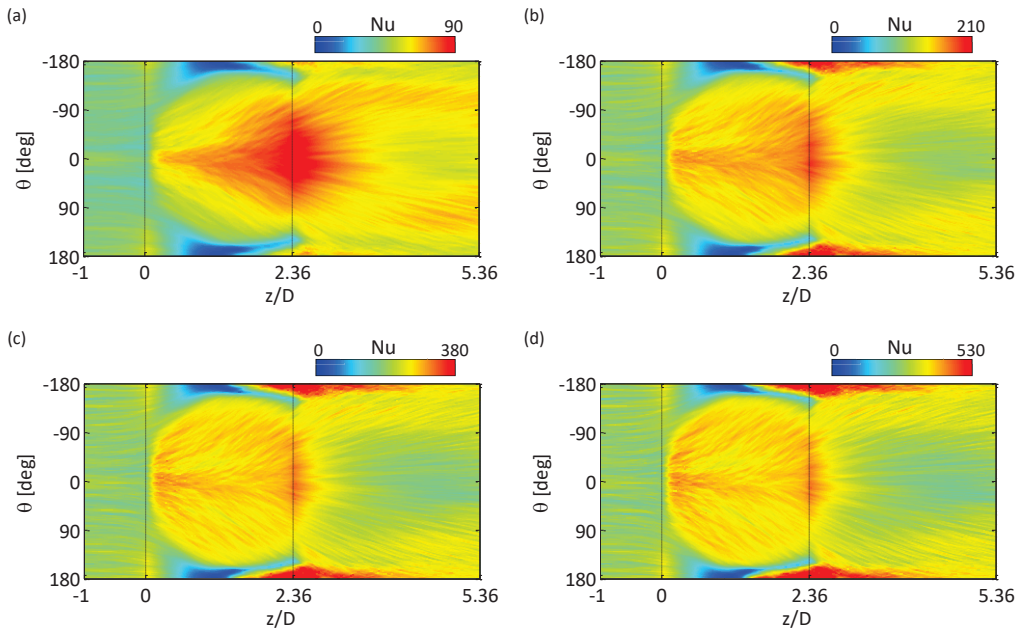


図4 壁面展開図におけるヌセルト数の分布。(a) $Pr=1$, (b) $Pr=5.847$, (c) $Pr=25$, (d) $Pr=64$ 。

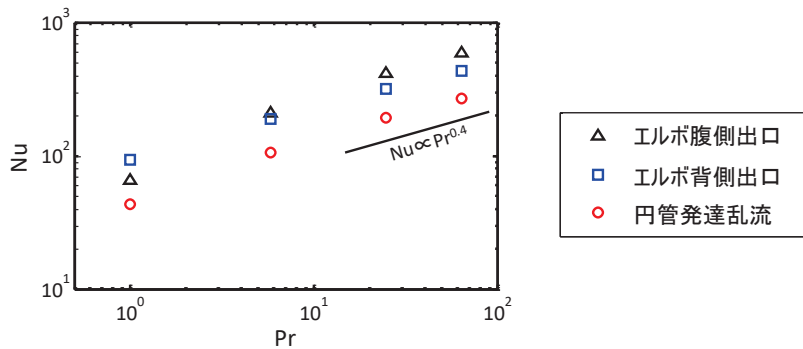


図5 エルボ管流れにおけるヌセルト数のプラントル数依存性。

けて、背側を中心に広く値の高い領域が分布している。また、腹側のエルボ出口周辺においてもヌセルト数が鋭く増加している。ただし、プラントル数が高い $Pr = 25$ と $Pr = 64$ のヌセルト数分布が定性的によく一致しているのに対して、 $Pr = 1$ の場合のヌセルト数分布はやや異なっている。他の条件と異なり、 $Pr = 1$ の場合は、エルボ出口において腹側よりも背側の方がヌセルト数の値が大きい。図5にエルボ出口の背側と腹側におけるヌセルト数のプラントル数依存性を示す。また、ドライバー部の円管発達乱流における値もあわせて示す。円管発達乱流では、熱伝達の経験式として Dittus-Boelter の式 (Dittus and Boelter, 1930) が広く用いられており、ヌセルト数のプラントル数依存性としては $Nu \propto Pr^{0.4}$ とされている。本計算においても、円管発達乱流におけるヌセルト数は $Nu \propto Pr^{0.4}$ に近い依存性を示している。しかし、エルボ出口におけるヌセルト数は腹側と背側でそれぞれ異なるプラントル数依存性を示している。特に、プラントル数が低い領域ではその差異が顕著である。エルボ背側出口は、主流部の偏流が強く、壁面近傍の流速変動

は比較的小さい。一方、エルボ腹側出口は、平均流速は小さいが、非定常性の強い旋回流の影響により、壁面近傍においても流速変動が大きい。この複雑流動との複合的な効果により、円管発達乱流とは異なるヌセルト数のプラントル数依存性が表れたものと考えられる。

4. まとめ

OpenFOAM を用いて、レイノルズ数が 15000 のエルボ管流れを対象に流動場および温度場の LES 計算を実施した。同一のエルボ管流動場に対して、温度場の無次元拡散係数であるプラントル数が 1、5.847、25、64 の 4 条件において温度場計算を行い、プラントル数の違いが温度場の形成や壁面の熱伝達に及ぼす影響について考察した。

エルボ管において形成される流動場は、曲がりの背側と腹側で大きく異なる。背側は管壁への偏流が強く、時間的な擾乱は少ない。腹側では、背側から管壁に沿って回り込む旋回流の影響により、非定常性が強い流れ場となっている。本計算の結果、このエルボ管流動場における熱伝達現象において、プラントル数の違いによる複雑な効果があらわれることを明らかにした。円管発達乱流におけるヌセルト数はプラントル数 Pr の 0.4 乗に比例することが知られているが、エルボ管の壁面のヌセルト数は必ずしも $Nu \propto Pr^{0.4}$ とはならない。特に、プラントル数が低い領域ではその差異が顕著である。この流動場の状況による異なるプラントル数効果について、今後より詳細な解析を行う予定である。

謝 辞

本研究は東京大学情報基盤センター「若手・女性利用者推薦」平成 29 年度（前期）の課題として行われたものである。ここに記して謝意を表する。

参 考 文 献

- Dittus, F. W. and Boelter, L. M. K., Heat transfer in automobile radiators of the tubular type, University of California Publications in Engineering, Vol.2, No.13 (1930), pp.443-461.
- Dooley, R. B. and Chexal, V. K., Flow-accelerated corrosion of pressure vessels in fossil plants, International Journal of Pressure Vessels and Piping, Vol.77, No.2-3 (2000), pp.85-90.
- Germano, M., Piomelli, U., Moin, P. and Cabot, W. H., A dynamic subgrid-scale eddy viscosity model, Physics of Fluids A, Vol.3, No.7 (1991), pp.1760-1765.
- OpenCFD, OpenFOAM: The open source computational fluid dynamics(CFD) toolbox, available from <<http://www.openfoam.com>>, (accessed on 26 May, 2015).
- OpenFOAM Foundation, Bug Reporting (issue #99): Dynamic Smagorinsky model with local coefficient values, available from <<http://www.openfoam.org>>, (accessed on 26 May, 2015).
- 恒吉達矢, 伊藤高啓, 歌野原陽一, 辻義之, 形状係数測定における壁面伝達境界条件が及ぼす影響に関する研究, 日本機械学会論文集, Vol.83, No.847 (2017), DOI:10.1299/transjsme.16-00417.

第一原理計算に基づく大規模自己組織化ナノ構造における熱電特性 の計算

新屋ひかり

横浜国立大学大学院工学研究院

大阪大学大学院基礎工学研究科附属スピントロニクス学術連携研究教育センター

福島鉄也

大阪大学ナノサイエンスデザイン教育研究センター

吉田博

東京大学スピントロニクス学術連携研究教育センター

1. はじめに

熱電材料は高いゼーベック係数と電気伝導度、低い熱伝導率によって効率化される。これら3つの物理量は全てトレードオフの関係にあるため通常のバルク材料での最適化は困難である。近年ではキャリアとフォノンの平均自由行程の違いを考慮した界面や不純物クラスターの形状・サイズ制御により、「ガラスのように低い熱伝導率を持つが電子はよく通す」、いわゆるフォノンガラスの実現に向けた研究が進められている。ここで不純物クラスターの低次元化を行うと、状態密度が急峻となりゼーベック係数の向上も期待されるので、まさに「一度で三度美味しい」手法である。

本稿では $(\text{GeTe})_{1-x}(\text{AgSbTe}_2)_x$ (以下、TAGS と記載する)を対象物質として扱った。TAGS は GeTe と AgSbTe_2 に相分離する傾向にあり、さらに AgSbTe_2 中で不純物クラスター $[2V_{\text{Ag}}+\text{Sb}_{\text{Ag}}]$ が自発的に析出することが第一原理計算に基づく先行研究で得られている[1-3]。TAGS を用いて上記で述べたフォノンガラスを実現させるためには不純物クラスターの形状やサイズ制御が必須となる。第一原理計算に基づく形状・サイズ制御のシミュレーションはいくつかの例で成功を収めている[4]。このようなシミュレーションを行う際には個々の原子間の相互作用を知る必要がある。TAGS は四元合金であるため局所環境効果を獲得するためのモデルは煩雑となり、従来の電子状態計算手法ではオーダー N^3 程度の計算量になるため現実的ではない。

そこで、本稿ではオーダー N 遮蔽 Korringa-Kohn-Rostoker (KKR) グリーン関数法のパッケージである KKRnano を Oakforest-PACS スーパーコンピュータシステム上で最適化を施し、TAGS の大規模全電子計算を行なった。KKRnano は一般的な電子状態計算手法では困難とされているナノスケールサイズの配置・構造・スピン不規則系の電子状態計算をオーダー $N(N^2)$ レベルで計算可能である。以下、KKRnano プログラムの簡単な説明と局所環境効果が与える電子状態計算結果について述べる。

2. オーダー N 遮蔽 Korringa-Kohn-Rostoker (KKR) グリーン関数法

KKR 法は Korringa, Kohn, Rostoker によって考案された固体中の電子の状態を計算する手法の一つである[5, 6]。一般的な固有値問題を解くような電子状態計算手法とは異なり、電子系を多重散乱理論により記述する。当初は結晶のグリーン関数の極から電子の固有値計算を行なっていたため、計算量の大きさから実用上困難を伴うことが指摘されていた。しかし、計算機性能の向上や数値計算手法の発展により密度汎関数理論[7, 8]と組み合わせることでコーン・シャム方程式に対する一電子グリーン関数を直接計算することが可能となり、KKR グリーン関数法として近年盛んに利用されている。KKR グリーン関数法は全電子計算であるため擬ポテンシャルを選択する必要もなく、Shape function を利用し異方的なポテンシャル部分をボルン近似により球対称部分

からの摂動として取り扱うことで、フルポテンシャルへの拡張も可能である。また、得られた一電子グリーン関数から線型応答理論を用いることで輸送現象、磁気光学効果、フォノン励起等が比較的容易に議論することができるのも KKR グリーン関数法の注目すべき長所の一つと言える。

フルポテンシャル KKR グリーン関数法では結晶に対して図1のようにポロノイ分割を行う。

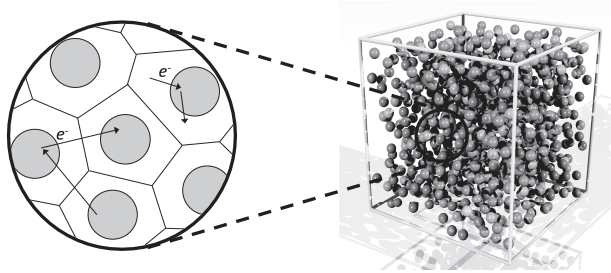


図 1 : 結晶のポロノイ分割

左図は結晶中に存在する原子の拡大図であり、 e^- と黒矢印は電子が結晶ポテンシャルによって散乱されている様子を表している。

分割された各ポロノイセルに対して、次のようにグリーン関数が定義される。

$$G(\mathbf{r} + \mathbf{R}^n, \mathbf{r}' + \mathbf{R}^{n'}; E) = \delta^{nn'} G_s^n(\mathbf{r}, \mathbf{r}'; E) + \sum_{LL'} R_L^n(\mathbf{r}; E) G_{LL'}^{nn'}(E) R_L^{n'}(\mathbf{r}'; E)$$

r, R はそれぞれ原子と電子の位置、 E は虚数を含むエネルギー、 $R_L^n(\mathbf{r}; E)$ は動径方向のシュレディンガー方程式の原点で正則な解である。右辺第1項は単一ポテンシャルによる散乱を表し、多重散乱の効果は第2項に含まれている。単一ポテンシャルによる散乱項はオーダー N レベルで計算可能であるので、大規模系においても計算コストの観点からは問題にはならない。しかし、多重散乱項の取り扱いが単純ではなく、係数 $G_{LL'}^{nn'}(E)$ (KKRの構造定数と呼ばれる)を得るためには、下記のダイソン方程式を解く必要がある。

$$G_{LL'}^{nn'}(E) = G_{LL'}^{r,nn'}(E) + \sum_{n''L''L'''} G_{LL''}^{r,nn''}(E) t_{L''L'''}^{n''}(E) G_{L''L'}^{n''n'}(E)$$

ここで、 $L = (l, m)$ 、 $G_{LL'}^{r,nn'}(E)$ は参照系のグリーン関数、 $t_{L''L'''}^{n''}$ はシングルサイトの t 行列である。KKR グリーン関数法はコーン・シヤム方程式を解くわけではなく、あらわに基底関数を必要としないためカットオフエネルギー等のパラメーターを必要としないが、グリーン関数の部分波展開を行う際の $L (L_{max})$ がパラメーターとなる。実際の計算では有限の L_{max} を用いるため、グリーン関数の虚数部分から導かれる電子密度にずれが生じるが、Lloyd 公式により求められる状態密度を利用し電子密度の補正を行う。

ダイソン方程式を解く部分が KKR グリーン関数法において最も計算コストが必要な箇所となる。一般的な KKR グリーン関数法の場合は、参照系として解析的に解くことができる自由空間のグリーン関数を利用する。この場合、上記方程式を解くためには逆行列の計算が必要となり、計算コストがオーダー N^3 となる。遮蔽 KKR グリーン関数法ではオーダー N (またはオーダー N^2) を実現するために、参照系として斥力ポテンシャルが周期的に並べられた系を使用する。この参照系のグリーン関数は距離に対して指数関数的に減衰するため、疎行列を扱うことになり計算コストが大幅に減少することとなる。この遮蔽化に加えて逆行列をあらわに計算することなく逐次的にダイソン方程式を解くことで大規模並列計算の実現が可能である。ダイソン方程式を形式的に書き下すと

$$G = -(\Delta t)^{-1} + (\Delta t)^{-1} X (\Delta t)^{-1}$$

となる。 $(\Delta t)^{-1}$ はサイト対角な量で簡単に計算可能なので、ボトルネックとなる箇所は $X = [(\Delta t)^{-1} - G^r]^{-1}$ で定義される量 (KKR 行列もしくは散乱積分演算子と呼ばれることもある) である。これは、 $X^{(i+1)} = \Delta t + \Delta t G^r X^{(i)}$ のように逐次方程式に変換し、下記の線型連立方程式を解く問題へと帰着できる。

$$(\Delta t G^r - 1)X = -\Delta t$$

通常の電子状態計算で扱う固有値問題とは異なり、このような線型連立方程式は行列 X の L 列成分をMPIにより各プロセスに分配することでプロセス間のコミュニケーションなく独立で逐次処理を実行でき、高効率の原子数 N に対する並列計算が可能である。KKRnano パッケージでは上記の線型連立方程式をクリノフ部分空間法の一つである擬似的最小残差法 (TFQMR 法) を上記の線型連立方程式に適用しており、さらにブロック巡回行列を用いた前処理により収束性を高めている [9-11]。

図2にKKRnanoのフローチャートを示す。密度汎関数理論に基づきセルフコンシステント計算を行うが、平面波法や他の手法で見られるコーン・シャム方程式の固有値問題を扱うことはしない。まず、参照系ポテンシャルを設置し、単一ポテンシャルによる散乱問題を解く。得られた原点正則な解 $R_L^t(r; E)$ と t 行列を用いてダイソン方程式からグリーン関数を求め、その虚部から電子密度を算出する。最終的に次のステップのための有効ポテンシャルを構築し、セルフコンシステントループを回すことで基底状態の探索を行う。KKRnanoは上記で述べた原子数並列に加えて、エネルギー並列とスピン分極をしている系にはスピン並列の計3段階のMPI並列に対応している。エネルギー並列に関してはフェルミ準位付近のダイソン方程式の収束性が遅く、計算時間的に支配的になるため3~4の並列数で頭打ちになる。これらに加え、疎行列計算の取り扱いにおいてOpenMPに対応しているので、合計4段階の大規模並列計算が可能となっている。KKRnanoの並列化効率等の情報はホームページ (http://www.fz-juelich.de/ias/jsc/EN/Expertise/High-Q-Club/_node.html) で知ることが可能である。またKKRnanoは既に希薄磁性半導体、強磁性相変化物質、高エントロピー合金等の構造・配置・スピン不規則性を含む大規模系にも応用されているので、興味がある方は参考文献を参照して頂きたい [12-15]。

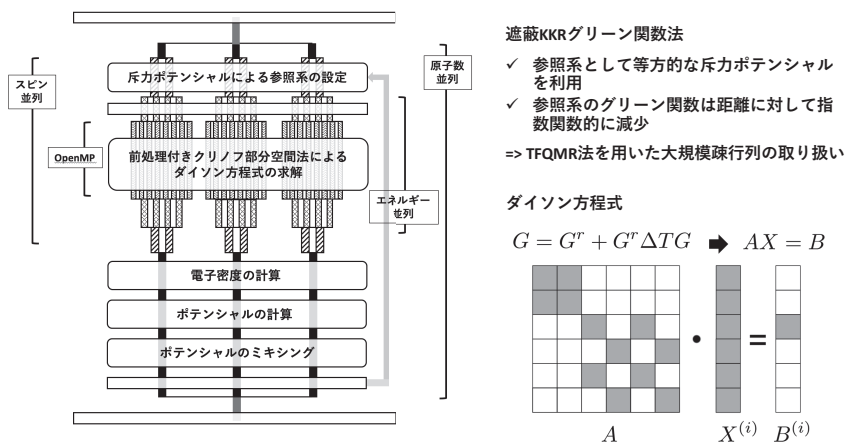


図 2: KKRnano プログラムのフローチャートと MPI/OpenMP ハイブリッド並列の概要

KKRnano では原子数、スピン、エネルギー、OpenMP の4段階でのハイブリッド並列が可能である。また、ダイソン方程式を解く際には計算速度と並列化効率を向上させるために、前処理付きのクリノフ部分間法 (TFQMR 法) を利用する。

3. 大規模全電子計算による $(\text{GeTe})_{0.8}(\text{AgSbTe}_2)_{0.2}$ の電子状態の解析

KKRnano プログラムパッケージを用いた大規模全電子計算により $(\text{GeTe})_{0.8}(\text{AgSbTe}_2)_{0.2}$ の電子状態を調べた。TAGS の結晶構造は2つの FCC 副格子を持つ NaCl 型結晶構造であり、1つの副格子には Ge、Ag、Sb がランダムに分布し、もう一方の副格子を Te が完全に占有していると実験的に報告されている。我々は TAGS の構造モデルとして NaCl 構造を基とした $6 \times 6 \times 6$ の合計 1,728 原子を含んだスーパーセル 1,728 原子を採用した (組成的には $\text{Ge}_{692}\text{Ag}_{36}\text{Sb}_{36}\text{Te}_{864}$ で表される)。このスーパーセルの大きさは陽イオンサイトの原子不規則性を十分に記述できる大きさである。計算には Oakforest-PACS スーパーコンピュータシステムを使用し、Hyper-Threading を用いた合計 13,824 プロセッサ (原子数:1,728 × エネルギー数:4 × OpenMP:2) による OpenMP/MPI ハイブリッド並列計算を行った

図3は $\text{Ge}_{692}\text{Ag}_{36}\text{Sb}_{36}\text{Te}_{864}$ における各原子の部分状態密度を示している。エネルギーの原点はフェルミ準位にとっている。Ge の 4s 状態、Sb と Te 原子の 5s 状態が、それぞれエネルギー的に深い -9 から -8 eV、 -10 から -9 eV、 -12 から -11 eV の位置に形成されている。Ag の 4d 電子が価電子の底に局在した状態を作っており、価電子帯は Te の 5p 状態が支配的である。この価電子状態は Ge の 4s 状態と Sb の 5s 状態が Te の 5p 状態と混成した結果、反結合軌道成分がフェルミ準位付近に押し出されたと解釈すると理解しやすい。各原子の部分状態密度はかなりの幅を持って分布していることが見てとれる。これは Ge、Ag、Sb 原子はランダムに配置されているため原子は異なる局所環境を反映するからであり、同じ原子種であったとしても異なる状態密度を持つ (局所環境効果)。本稿には掲載していないが、KKRnano では局所環境効果を反映した局所エネルギーを効率よく計算可能であるので、一回の大規模なセルフコンシステント計算から局所構造の安定性を議論することも可能である。不規則系を取り扱う他の有名な手法として平均場近似がある。平均場近似では不均一なポテンシャル散乱の効果を有効媒質に置き換えることで問題の簡略化を行う。特にコヒーレントポテンシャル近似 (CPA) は多重散乱理論に基づいた KKR 法と効率よく組み合わせることが可能で KKR-CPA 法として幅広く不純物系・合金等の電子状態や磁気特性の研究に使用されている。計算コストが低く物性のケミカルトレンドを調べるのに最適な手法であるが、シングルサイト近似であるので、このような局所環境効果の議論に応用することはできない。図3には配置平均を取った各原子の部分状態密度 (黒線) も示されているが、これらの値は KKR-CPA 法で計算した部分状態密度と矛盾しないことが確認されている。それゆえ、今回、 $\text{Ge}_{692}\text{Ag}_{36}\text{Sb}_{36}\text{Te}_{864}$ の計算で使用したスーパーセルは原子配置のランダム性を考慮できるほど十分に大きく、スーパーセル法で一般に問題になる周期性から生じる人工的なクローン相互作用の影響を排除できていると結論できる。

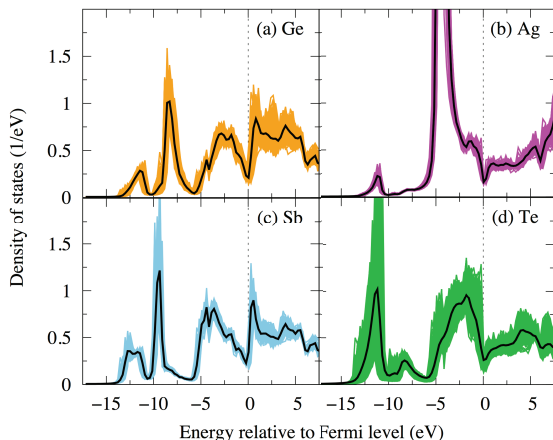


図 3: $\text{Ge}_{692}\text{Ag}_{36}\text{Sb}_{36}\text{Te}_{864}$ の部分状態密度

(a)Ge、(b)Ag、(c)Sb、(d)Te の部分状態密度を示しており横軸はフェルミ準位から測ったエネルギーである。黒線は各原子サイトの平均部分状態密度を表している。

4. まとめ

本稿ではOakforest-PACS スーパーコンピュータシステムを利用した、遮蔽 KKR グリーン関数法による熱電物質 $(\text{GeTe})_x(\text{AgSbTe}_2)_{1-x}$ の大規模電子状態計算について紹介した。今回用いた KKRnano は大規模配置・構造・スピン不規則系の電子状態計算に有効な第一原理計算プログラムパッケージであり、スーパーコンピュータと併用すれば数万原子の全電子計算が可能である。今後は線型応答理論に基づいた伝導特性・格子特性を定量的に評価するアプリケーションを KKRnano に実装することにより、ナノ・サブミクロンスケールレベルで新規熱電変換材料のデザインを行う予定である。

謝 辞

本研究は、東京大学情報基盤センター「若手・女性利用者推薦」平成 29 年度（前期）の課題によって行われた。また、JST CREST「量子状態の高度な制御に基づく革新的量子技術基盤の創出」強磁性量子ヘテロ構造による物性機能の創出と不揮発・低消費電力スピンドバイスへの応用（Grant No. JPMJCR1777）の支援に感謝する。

参考文献

- [1] H. Shinya, A. Masago, T. Fukushima, H. Funashima, and H. Katayama-Yoshida, *Jpn. J. Appl. Phys.* **53**, 111201 (2014).
- [2] H. Shinya, A. Masago, T. Fukushima, and H. Katayama-Yoshida, *Jpn. J. Appl. Phys.* **55**, 041801 (2016).
- [3] H. Shinya, A. Masago, T. Fukushima, and H. Katayama-Yoshida, *Jpn. J. Appl. Phys.* **56**, 081201 (2017).
- [4] H. Shinya, T. Fukushima, A. Masago, K. Sato, and H. Katayama-Yoshida, *Phys. Rev. B* **96**, 104415 (2017).
- [5] J. Korringa, *Physica*, **13**, 392 (1947).
- [6] W. Kohn and N. Rostoker, *Phys. Rev.* **94**, 1111 (1954).
- [7] P. Hohenberg and W. Kohn, *Phys. Rev.* **136**, B864 (1964).
- [8] W. Kohn and L. J. Sham, *Phys. Rev.* **140**, A1133 (1965).
- [9] R. Zeller, *J. Phys.: Condens. Matter* **20**, 294215 (2008).
- [10] A. Thiess, R. Zeller, M. Bolten, P. H. Dederichs, and S. Bluegel, *Phys. Rev. B* **85**, 235103 (2012).
- [11] M. Bolten, A. Thiess, I. Yavneh, and R. Zeller, *Linear Algebra Appl.* **436**, 436 (2012).
- [12] A. Thiess, P. H. Dederichs, R. Zeller, S. Bluegel, and W. R. L. Lambrecht, *Phys. Rev. B* **86**, 180401(R) (2012).
- [13] T. Fukushima, H. Katayama-Yoshida, K. Sato, H. Fujii, E. Rabel, R. Zeller, P. H. Dederichs, W. Zhang, and R. Mazzarello, *Phys. Rev. B* **90**, 144417 (2014).
- [14] T. Fukushima, H. Katayama-Yoshida, K. Sato, M. Ogura, R. Zeller, and P. H. Dederichs, *J. Phys. Soc. Jpn.* **86**, 114704 (2017).
- [15] M. Ogura, T. Fukushima, R. Zeller, and P. H. Dederichs, *J. Alloys Compd.* **715**, 454 (2017).

将来の大規模メニーコアプロセッサ環境に向けたビッグデータ基盤処理の性能評価

佐藤 仁

産業技術総合研究所 人工知能研究センター

1. はじめに

近年、ビッグデータ処理が注目されており、SNS、道路の経路探索、スマートグリッド、創薬、遺伝子解析などペタ～ヨタバイト級のデータに対する高速処理の需要が高まっている。現状のビッグデータ処理は、数台～数百台の安価な Web サーバ由来のクラスタやクラウド上の HDD にファイルとして分散し、Hadoop 等の I/O やネットワーク性能の効率の悪いプラットフォーム上でのリニアスキャンなどの処理が行われている。一方、最新のスループットプロセッサや大容量なメモリ、バーストバッファなどの不揮発性メモリを活用した I/O アクセラレータを備え、Infiniband や OmniPath などの広帯域で低遅延な高速ネットワークが接続されたスーパーコンピュータは、超高速な演算性能に加えて強力な I/O 性能を提供できるため、ビッグデータ処理基盤として有望であると考えられ、実際、米国 NERSC Cori (2016 年)、東京大学 Reedbush や東京工業大学 TSUBAME3 (2017 年)をはじめ、米国 SDSC Gordon (2012 年)、米国 LLNL Cataryst (2013 年)などビッグデータ処理にも適したスーパーコンピュータが登場しはじめている。一方で、このような先進的な環境を活用したデータ処理のソフトウェア基盤は少なく、大規模実行の事例もまだまだ少ないのが現状である。本稿では、将来のメニーコアプロセッサ環境に向けたビッグデータ処理の研究開発の動向として、Graph500、メニーコアプロセッサを考慮した MapReduce や大規模分散ソートなどの具体的な事例について紹介する。

2. Graph500

グラフは、辺と頂点で記述された連結されたオブジェクトを表現するための基本的な数学的表現である。ヘルスケア、システム生物学、ソーシャル・ネットワーク、ビジネスインテリジェンス、電力網などの様々な重要アプリケーションがグラフを用いてモデル化されている。

(図 1) さらに、近年、このようなアプリケーション分野において様々なデータが大量に生成されるようになったため、大規模なグラフに対する高速処理の要求が非常に大きくなっており、高性能計算での重要なカーネルのひとつとなってきた。実際、従来の Linpack によるスーパーコンピュータの計算処理の性能を競う Top500 リストに加え、近年、大規模なグラフ処理を行うことでスーパーコンピュータのビッグデータ処理能力を競う Graph500 リストが登場してはじめている [1]。図 2 に Graph500 ベンチマークの概要を示す。現在のベンチマークではスケールフリーや直径の小ささなどの現実のネットワークをモデル化したクロネッカーグラフに対して幅優先探索 (BFS) を行った際の実行時間を計測する。幅優先探索とは、ある頂点から隣接している頂点への探索を繰り返し、グラフ内で隣接した頂点を全探索するアルゴリズムである。しかし、現状では、分散メモリで構成されたスーパーコンピュータ上での実行に適した幅優先探索のアルゴリズムには多くの最適化の余地がある。そこで我々は、大規模計算環境でのビッグデータ処理のハードウェア・ソフトウェアに関連する問題点を明らかにし将来のスーパーコンピュータの設計へ活かすことを目的とし、Graph500 の開発を進めている。

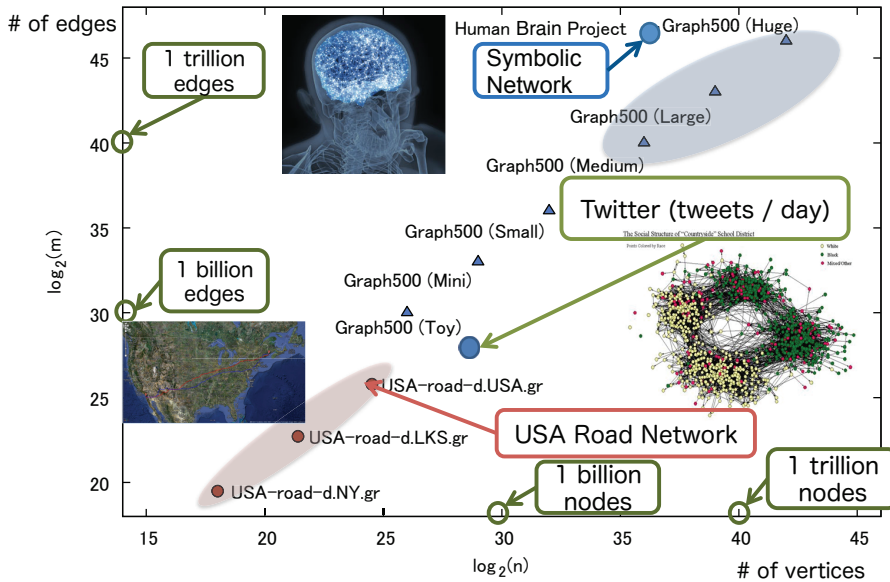


図 1 様々な大規模グラフ

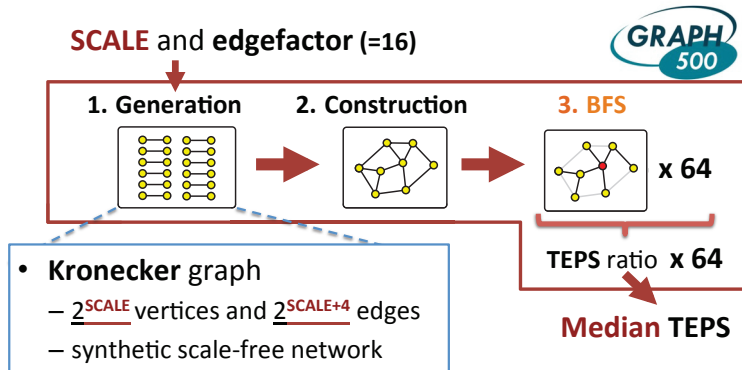


図 2 Graph500 の概要

スーパーコンピュータでは、数千～数万台の計算ノードを使った並列分散アルゴリズムが求められるが、BFSの並列分散アルゴリズムを設計するのは容易なことではない。単純なデータ構造や探索アルゴリズムでは、計算ノードの台数が数百～数千台規模に増えたときネットワーク性能に律速されて全体の性能が頭打ちになってしまう。計算ノード数が数千～数万台規模においても計算ノード数に比例した性能を得て、かつ、効率よく処理を行うためには、以下の条件を満たすアルゴリズム・データ構造が必要となる。

- 大規模に分割したときでも、メモリ使用量とアクセスコストをともに小さくできるグラフ探索向け疎行列のデータ構造
- 通信ノード数と通信データ量がともに低く抑えられる通信アルゴリズム
- グラフの分割数が増えても探索エッジ数を小さく保つ探索アルゴリズム

Graph500の参照実装では、データ構造や通信アルゴリズムなどの問題により、数百台の計算ノード規模で性能が頭打ちになってしまう。大規模並列分散向けBFSの最新の研究として、2次元分割のWaveを使った手法があるが、この方法は、高速化に有効で他の実装でも広く用いられているHybrid BFSの手法を使っておらず、最適な方法ではない。また、Hybrid BFSを使った分散並列アルゴリズムも提案されているが、これもデータ構造や通信アルゴリズムの問題により、千台の計算ノード程度で性能が頭打ちになっている。

今回、将来の大規模メニーコアプロセッサ環境に向けたビッグデータ基盤処理の性能評価の一環として、Oakforest-PACSでGraph500を実行させた。今回の実行では、基本的には、GPU版Graph500上野実装v1.2をベースにCPUモードで動作させ、`-xMIC-AVX512 -gopenmp`などのコンパイラオプション、`numactl, taskset`などのコンパイラオプション、`I_MPI_XXX, KMP_XXX, OMP_XXX`などの環境変数など標準的な動作方法で実行したところ、Scale32で平均次数16のグラフ(2^{32} 頂点 $\sim 2^{36}$ 辺)を、計算ノード64台を用いて、 3.94×10^9 TEPS程度達成できることを確認した。技術的な内容を広範囲に検討した結果、更なる性能向上を達成するためには、やはり、ソースコードレベルでメモリ確保の際のアライメントを揃えた上で、AVX512などのSIMD命令を活用する最適化を導入する必要があることがわかった。今回は、工数の関係からここまでの最適化は実現せず今後の課題となったが、同様のこのような最適化手法の考え方はIntel Skylake世代以降の標準的なCPUでも応用できそうな見込みである。

3. メニーコアプロセッサを考慮したMapReduce

大規模データ処理の分野では、MapReduceプログラミングモデル[4]が登場し、数千台規模のクラウド上のクラスタ型計算機上で局所性と耐故障性を考慮したスケーラブルなペタバイト～ヨットバイト級のデータ並列処理を可能にしている。MapReduceでは、分散したkey-valueのペアデータに対して局所性を考慮した統一的な操作を並列で適用するために、並列データ処理のプロセスをMap, Shuffle, Reduceの3つの処理に分解する。そして、Map処理で入力データとなるkey-valueペアから中間データとなるkey-valueペアを生成し、Shuffle処理で同じkeyに対してvalueのリストを生成し、最後にReduce処理で中間データをShuffleすることにより得られたkeyとvalueのリストから最終出力となるkey-valueのペアデータを生成する。MapReduceは局所性を考慮したデータ並列処理を提供するという点でメモリ空間が階層的に分かれているGPUアクセラレータへも有効な手法であると考えられる。一方、メニーコアプロセッサを搭載した大規模なスーパーコンピュータへMapReduce処理を適用した事例は少ない。

そこで、我々は、現在、メニーコアプロセッサや不揮発性メモリデバイスを搭載したスーパーコンピュータ上でスケーラブルなデータ並列処理を目指したソフトウェア基盤としてHAMAR(Highly Accelerated Data Parallel Processing Framework for Deep Memory Hierarchy Machines)の開発を進めている。現在のHAMARの実装では、処理する対象のデータを動的にいくつかのチャンクに分割し、CPUとGPU間のデータ転送とGPUアクセラレータ上でのMap, Shuffle, Reduce処理をできるだけオーバーラップさせることで、GPUアクセラレータに搭載されたメモリの容量を超えるデータセットに対しても効率のよい処理が可能になっている。また、GPUベースの外部ソートなどOut-of-coreな基本アルゴリズムの実装を多く取り入れている。

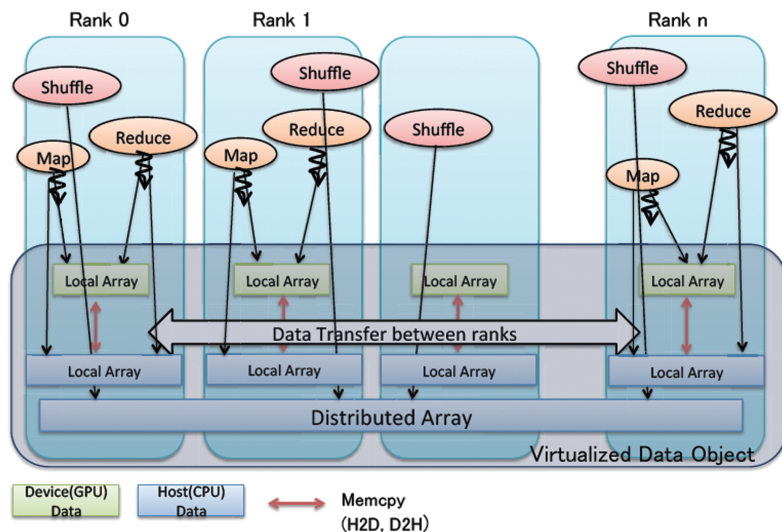


図 3 HAMAR の概要

HAMAR を用いた実装のケーススタディとして、MapReduce 型の大規模グラフ処理である GIM-V (Generalized Iterative Matrix-Vector multiplication) と呼ばれる反復行列ベクトル積の一般化されたアルゴリズム [5] を実装している. いま、 i, j が $\{1, \dots, n\}$ の値をとるとし、 $M = (m_{i,j})$ を $n \times n$ の行列、 $v = (v_i)$ を大きさ n のベクトルとする. ここで、オペレータ x_G を導入することにより、GIM-V アルゴリズムを次のように定義できる.

$$v' = M x_G v$$

where $v'_i = \text{assign}(v_i, \text{combineAll}_i(x_j \mid j = 1 \dots n, \text{and } x_j = \text{combine2}(m_{i,j}, v_j)))$

上式は、 combine2 、 combineAll 、 assign という 3 つのオペレータを用いることにより表現される (図 4). 上の操作を、PageRank, Random Walk with Restart, Connected Component といったグラフアルゴリズムの実装によって定義された収束条件を満たすまで反復処理する.

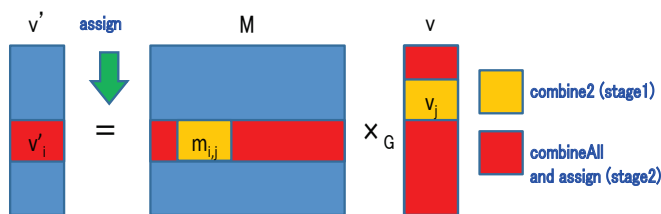


図 4 GIM-V の概要

Hamar を用いて、行列ベクトル積の一般化を MapReduce へ適用した GIM-V により PageRank アルゴリズムを実装し、TSUBAME2.5 の 1024 ノード (12288 CPU コア, 3072 GPU) を用いて大規模な実証実験を行った. デバイスメモリの容量を超えるグラフデータ (171.8 億頂点、2749 億枝からなる大規模グラフ) を処理する場合に、1 ノード当たり 3GPU を使用した場合、2.81 GEdges/sec (1 秒当たり処理した枝数, 47.7GB/sec) の性能となり、CPU 上での実行に対して 2.10 倍の高速化を確認した. また、ウィークスケーリングの性能を計測した結果、1024 ノード (3072GPU) を使用した場合に、1 ノード (3GPU) を使用した場合に対して 686 倍の性能向上を

示し、良好なスケーラビリティを示すことを確認した (図 5) [6].

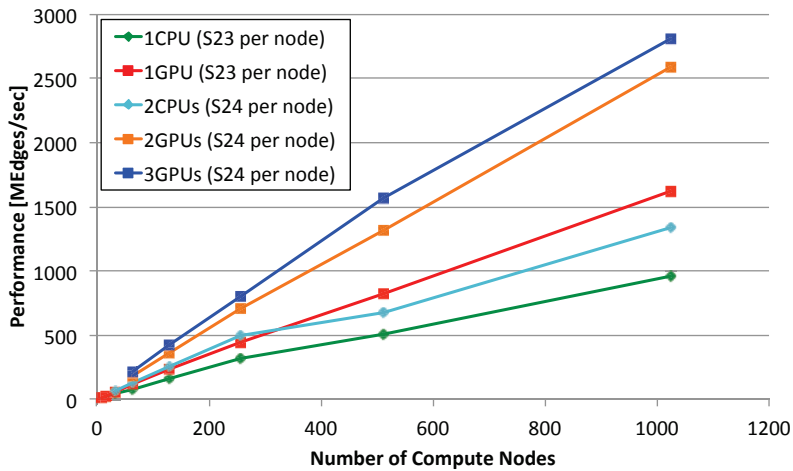


図 5 GIM-V のスケーラビリティ

4. メニーコアプロセッサを考慮した大規模分散ソート

ソートアルゴリズムは最も基本的なアルゴリズムの一つであり、ゲノム解析や多体問題など様々なアプリケーションのカーネルに使用されている。近年、このようなアプリケーションでの急激なデータの増加に伴い、膨大な量のデータをソートするために分散システム向けのソートアルゴリズムが数多く提案されてきた。その中でも、データの通信量を減らす事によって、高速にソートできるアルゴリズムとして、Splitter-based parallel sorting algorithm というアルゴリズムが広く知られている。Splitter-based parallel sorting algorithm では、スプリッターを導入することでデータの通信コストを減少させ、高速化に成功したが、相対的に計算処理のコストが高まり、全体の処理時間の大部分を占めるようになることが問題となっている。

Splitter-based parallel sorting algorithm では、以下のような手順でソートが行われる (図 6)。

1. それぞれのプロセスに配置されたデータを独立にソートする。
2. その後、データの転送時の基準となるスプリッターを選択する。
3. 選択したスプリッターに合わせてプロセス間で通信を行い、データの転送を行う。
4. 最後に、マージ操作でプロセス上のデータをソートされた状態にする。

分散システム用のマージソートや基数ソートではデータ転送フェイズが数多く行われるが、このアルゴリズムではスプリッターを元にしたデータ転送によって通信回数を減らすことで通信コストを下げている。

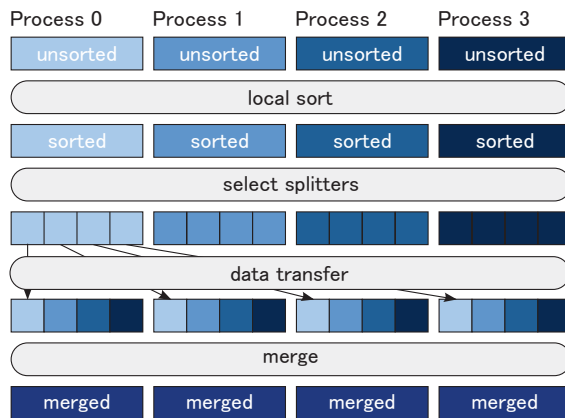


図 6 Splitter-based Parallel Sorting Algorithm

我々は, Splitter-based parallel sorting algorithm の一つである HykSort [7] を拡張し, メニーコアプロセッサの代表的な類型として GPU アクセラレータによる高速化を行った. 実装では, 最もコストの大きい計算処理である各プロセス上でのソート処理を GPU 上で行うように変更した. GPU メモリは DRAM と比べると非常に小さいため, GPU メモリを溢れるようなサイズのデータをソートする時には, 複数個にチャンクを分けてソートを行い, 最後にソートされたチャンクをマージすることで対応している.

以下は, メニーコアプロセッサ版の hyksort の性能を GPU スパコン (TSUBAME2.5) 上最大 1024 ノードを使用して行ったものである. 各ノード上ではソケットあたりに 1 つのプロセスが配置され, 各プロセスが 2GB の 64 bit 整数のデータを所有している状況で開始する. 弱スケーリングの実験では, 2048 プロセス (1024 ノード, 2048 GPU) の時に 0.25TB/s 程のスループットが得られた. これは CPU 1 スレッドのみの実装と比べると 3.61 倍, CPU 6 スレッド並列のものとは比べると 1.40 倍の性能となっている (図 7) [8].

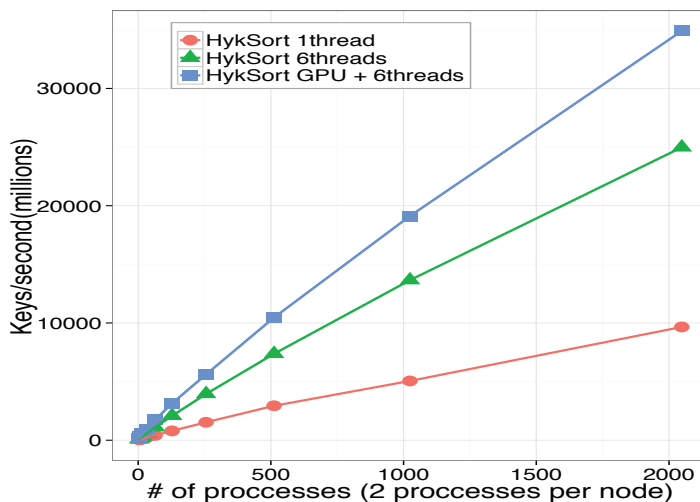


図 7 hyksort のスケーラビリティ

5. おわりに

本稿では、将来の大規模メニーコアプロセッサ環境に向けたビッグデータ処理の研究開発の動向として、Graph500、メニーコアプロセッサを考慮した MapReduce 処理や大規模分散ソートなどの具体的な事例についての紹介を行った。これらの事例は 2018 年頃に産総研に導入される ABCI (AI Bridging Cloud Infrastructure) [9] などのエクストリームなビッグデータ処理を指向した大規模計算基盤アーキテクチャの設計などに活かされる予定である。

参考文献

- [1] Graph500: <http://www.graph500.org>.
- [2] Fabio Checconi, Fabrizio Petrini, Jeremiah Willcock, Andrew Lumsdaine, Anamitra Roy Choudhury, and Yogish Sabharwal, “Breaking the speed and scalability barriers for graph exploration on distributed-memory machines”, Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (SC’12), Article No.13, p13:1–13:12, 2013.
- [3] Scott Beamer, Krste Asanović, and David Patterson, “Direction-optimizing breadth-first search”, Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (SC’12), Article No.12, p12:1–12:10, 2012.
- [4] Jeffrey Dean and Sanjay Ghemawat, “MapReduce: simplified data processing on large clusters”, Communications of the ACM, Vol. 51, Issue 1, p107–113, 2008.
- [5] U. Kang, Charalampos E. Tsourakakis, and Christos Faloutsos, “PEGASUS: A Peta-Scale Graph Mining System Implementation and Observations”, Proceedings of the 9th IEEE International Conference on Data Mining (ICDM ’09), p229–238, 2009.
- [6] Koichi Shirahata, Hitoshi Sato, Satoshi Matsuoka, “Out-of-core GPU Memory Management for MapReduce-based Large-scale Graph Processing”, 2014 IEEE International Conference on Cluster Computing (Cluster2014), p221–229, 2014.
- [7] Hari Sundar, Dhairya Malhotra, and George Biros, “HykSort: a new variant of hypercube quicksort on distributed memory architectures”, Proceedings of the 27th international ACM conference on International conference on supercomputing, p293–302, 2013.
- [8] Hideyuki Shamoto, Koichi Shirahata, Aleksandr Drozd, Hitoshi Sato, Satoshi Matsuoka, “GPU-Accelerated Large-Scale Distributed Sorting Coping with Device Memory Capacity”, IEEE Transactions on BigData, Vol.2, Issue 1, p57–59, 2016.
- [9] 小川宏高, 松岡聡, 佐藤仁, 高野了成, 滝澤真一郎, 谷村勇輔, 三浦信一, 関口智嗣, “AI 橋渡しクラウド-AI Bridging Cloud Infrastructure (ABCI)-の構想”, 情報処理学会研究報告ハイパフォーマンソコンピューティング (HPC) 2017-HPC-160(28), p1–7, 2017.

酵素反応機構の理解に向けた効率的にタンパク質の揺らぎを取り込む計算手法の開発

山田 健太

理化学研究所 杉田理論分子科学研究室

1. はじめに

生体内分子システムが長い時間を掛けて創り上げてきた、酵素タンパク質の触媒する反応機構を解明することは、近年のエネルギー問題や環境問題を解決する新規触媒の開発や、反応条件が緩やかで環境負荷の低い生体触媒の工業利用に直結する、人類にとって重要な課題である。このため、生物学だけでなく化学や工学分野でも注目され、実験と理論の両立場から、微生物や植物細胞が有するタンパク質を対象にした研究が盛んに行われている。

計算機を用いて理論化学の問題に取り組む分野である計算化学では、量子力学的手法 (QM 法) を、実験で得られた酵素の立体構造から取り出した活性中心に適用し、触媒反応の解析が行われてきた。この手法は、電子を露わに扱うため、化学結合の生成・開裂を記述することができる一方で、計算コストが高い。近年では、活性中心周囲のタンパク質を露わに扱うために、低計算コストの分子力場を用いた分子力学的手法 (MM 法) と組み合わせた QM/MM 混合法によって反応解析を行うのが主流となっている。しかし、この混合法に基づいたダイナミクスの計算によってタンパク質の揺らぎを記述することは依然現実的ではなく、また基礎とする立体構造が、酵素のある瞬間的な状態から求められたものである。このため、従来手法では生体内では起こり得ない反応機構を提案してしまう可能性がある。一方、生体内におけるタンパク質の揺らぎを調べるために、古典力学に基づいて分子力場における原子・分子系のダイナミクスを追跡できる分子動力学(MD)シミュレーションが用いられている。しかし、分子力場では化学結合の生成・開裂を記述することは困難である。そこで、生体内における酵素反応機構の理解を目指して、上記手法を組み合わせた新規計算手法の開発に取り組んだ。

2. 計算対象：一酸化窒素還元酵素の反応解析

地球上の窒素循環を維持するために不可欠な過程である脱窒は、自然界では主に細菌 (脱窒菌) によって営まれる。脱窒菌においては、4種の金属酵素によって、硝酸塩(NO_3^-)や亜硝酸塩(NO_2^-)が窒素分子(N_2)まで還元される($\text{NO}_3^- \rightarrow \text{NO}_2^- \rightarrow \text{NO} \rightarrow \text{N}_2\text{O} \rightarrow \text{N}_2$)と考えられている。亜硝酸還元酵素(Nitrite Reductase, NiR)によって作られる一酸化窒素(NO)は、一酸化窒素還元酵素(Nitric Oxide Reductase, NOR)により亜酸化窒素(N_2O)に変換される。不対電子を有し細胞毒であるNOを N_2O へと無毒化する還元反応は、脱窒菌の生命維持に重要な機能であること、化学の本質である結合の生成と開裂が逐次的に生じること、 N_2O が強力な温室効果ガスであること、さらに嫌気呼吸から好気呼吸へという膜タンパク質の分子進化を理解する糸口となる可能性をもつことから、分子生物学をはじめとして、さまざまな分野から注目されている。

NORの活性中心はヘム b_3 、非ヘム鉄とこれらを取り囲む側鎖によって構成される (図1)。この部位において、NO二分子が反応

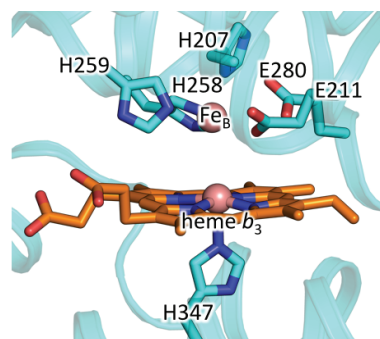


図1: NOR (PDBID: 3o0r)の活性中心

Fe_B と $\text{heme } b_3$ の鉄原子は二価で、それぞれの配位数は5である。

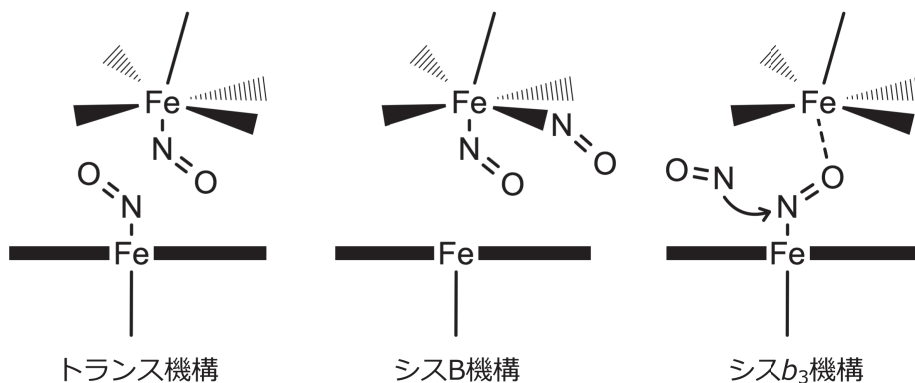


図2: NOR 活性中心における NO 還元反応機構モデル

NO 二分子が配位する鉄原子の違いと、生成すると考えられているハイポナイトライド($\text{N}_2\text{O}_2^{2-}$)がとる配座によって区別されて呼ばれている。

し、N-N 結合の生成と N-O 結合の開裂によって、 N_2O が生成される($2\text{NO} + 2\text{H}^+ + 2\text{e}^- \rightarrow \text{N}_2\text{O} + \text{H}_2\text{O}$)。この還元反応について、さまざまな実験研究から 3 種類 (トランス機構、シス B 機構とシス b_3 機構) の反応機構がこれまでに提案されてきた (図 2)。トランス機構ではヘム b_3 の鉄原子と非ヘム鉄のそれぞれに NO が配位した構造、シス B 機構では非ヘム鉄に NO 二分子が配位した構造、そしてシス b_3 機構ではヘム b_3 の鉄原子に配位した NO に対する NO の攻撃を介して、反応が進行すると想定されている。

脱窒菌の一種である緑膿菌由来の NOR(PDBID: 3o0r)を用いた NO 還元反応の解析に対して、実験ではトランス機構を示唆する急速冷凍電子スピン共鳴法の結果が報告されている[1]。理論計算では QM 法と QMMM 混合法を用いて既に行われており[2,3]、両手法の結果ともシス b_3 機構を提案している。また、この NOR 活性中心を変異によってミミックするように創成されたミオグロビン(EngMb)に対しても、QMMM 混合法による解析ではシス b_3 機構が有利だという結果が得られている[4]。しかしながら、振動分光実験により、EngMb では NO 二分子が異なる鉄原子に配位した構造の存在、つまり、トランス機構が強く示唆されている[5]。こういった理

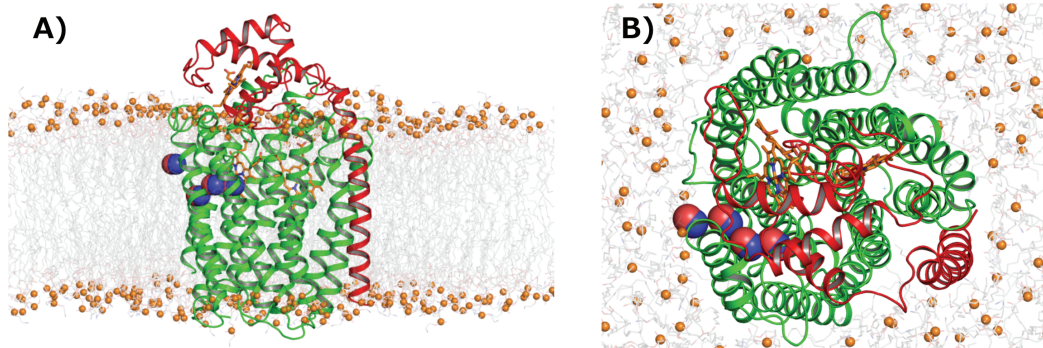


図3: NOR+4NO 系の初期構造 A) サイドビュー B) トップビュー

NOR 系の全原子 MD シミュレーションのスナップショットに対して、その NOR の疎水性チャネルに NO 4 分子を配置することによって、NOR+4NO 系の初期構造をセットアップした。また、橙色の球は脂質分子の P 原子を示している。

論と実験が導く結論の不一致をもたらす原因の1つとして、タンパク質が本来有する揺らぎを理論計算では考慮していないことが挙げられる。

3. 研究目的と方法

活性中心に NO 分子が存在するときのスナップショットに基づいて、タンパク質の揺らぎによる活性中心への影響を考慮できる場をつくり、その場の存在下において反応機構を QM 法によって決定するという一連の手法を開発することが本研究の目的である。このために、まず、膜タンパク質である NOR を脂質二重膜に埋め込み、周囲に十分な水分子、および、150mM になるようにイオンを配置することによってモデル系をセットアップした。次に、この系の平衡化、および、100 ns の全原子 MD シミュレーションを実施した。NOR の構造が結晶構造と似ているスナップショットに対して、NO 4 分子を NOR の疎水性チャネルとそれと膜との境界付近に適宜配置した(NOR+4NO)系を作成した

(図3)。そして、この系のシミュレーションを行い、100ns のトラジェクトリを2本得た。また、同様の手続きによって、緑膿菌のもつ生体環境をモデリングした NiR と NOR の酵素複合体 (NiR:NOR 複合体, 図4) 系の全原子 MD シミュレーションを用いて、この複合体の構造安定性に関する調査を併せて実施した。

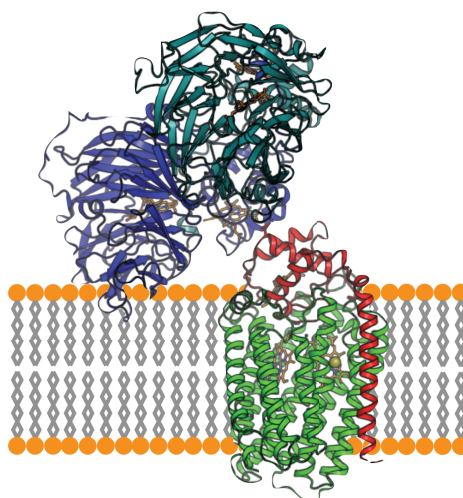


図4: 緑膿菌で発見された NiR:NOR 複合体 [6]
青色と青緑色で表したホモダイマーが NiR で、黄緑色と赤色で表したヘテロダイマーが NOR である。また、脂質二重膜は模式的に描いたものである。

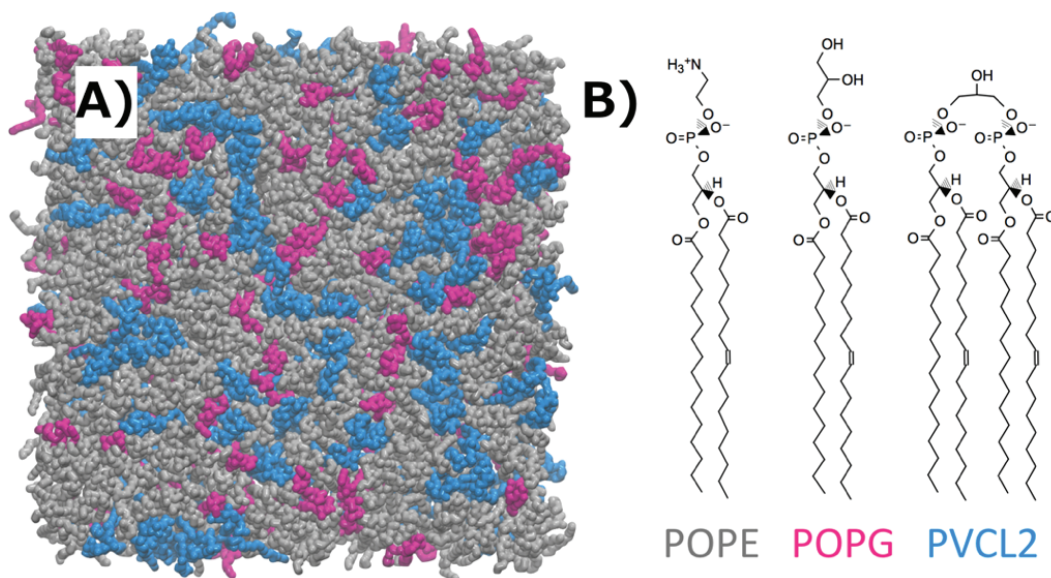


図5:A)作成した混合脂質二重膜とB)その構成分子

生体環境における NiR:NOR 複合体の構造安定性を調べるために使用した3種の分子で構成される脂質二重膜

NOR 系で使用する POPE 分子から構成される脂質二重膜と, NiR:NOR 複合体系で使用する POPE 分子, POPG 分子と PVCL2 分子から成る脂質二重膜は CHARMM-GUI[7]によって作成した(図5)。NiR:NOR 複合体系で混合膜を採用した理由は, 緑膿菌の形質膜がもつ脂質組成に似せることによって, 重要だと考えられる膜表面と NiR の相互作用を適切に考慮するためである。また, この相互作用を議論するために, 脂質二重膜の代わりに直径が約 86 Å と小さいナノディスクを用いた複合体系も作成した。力場パラメータは CHARMM C36 を一部改変したものを使用した[8]。静電相互作用の計算は PME 法によって行い, 温度と圧力の制御には Bussi 法を使った。MD ソフトウェアは理化学研究所の杉田グループで開発しているプログラムである GENESIS を用いた[9]。

4. 結果と考察

A. NOR+4NO 系

今回得られた 2 本の 100 ns のシミュレーションでは, NO 分子は活性中心に到達せず, 本研究の方法開発に必要な, 活性中心に NO 分子が存在するスナップショットを得ることができなかった。NO 分子の動きを吟味すると, 活性中心の傍に存在する残基 5 つ(Phe136, Val206, Trp209, Val210, Phe352)でつくられている「ポケット」に停留することが分かった(図6)。シミュレーション時間を延ばす, または, ラジカル分子である NO 分子の力場を再検討することによって, この分子が活性中心に行き着くのではないかと期待される。

B. NiR:NOR 複合体系

3 本の 300 ns のシミュレーションを実施した結果, NiR の解離は起こらず, 複合体構造は壊れなかった。得られたトラジェクトリを解析したところ, NiR-NOR 間の結合部位では, 結晶構造で見出された Glu119-Arg71 間の塩橋はシミュレーション時間にわたってよく維持されること, NiR の Lys61 や NOR の Lys100 をはじめとした塩橋周囲の残基と heme c によって, NiR-NOR 間だけではなく NiR 内・NOR 内の水素結合が頻繁に生成されることが分かった(図7)。塩橋を含めた, これらの水素結合ネットワークが, NiR:NOR 複合体の維持に寄与していると示唆された。

一方, NiR と膜のコンタクト部位では, 脂質二重膜から構成される脂質分子と NiR の残基が水素結合を生成することが分かった(図8)。平均して NiR-膜間で生じる水素結合ペアの数は約 10.1 であり, この部位が膜に対して船のアンカーのように働くことによって, NiR の動きを抑制し, NOR からの解離が困難になっていると

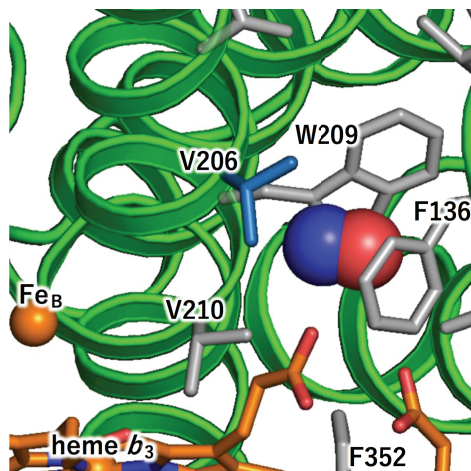


図6: NO 分子が停留するポケット

Phe136, Val206, Trp209, Val210, Phe352 で構成されているポケットで NO 分子(青と赤色の球)はスタックしてしまい, 活性中心には到達しなかった。

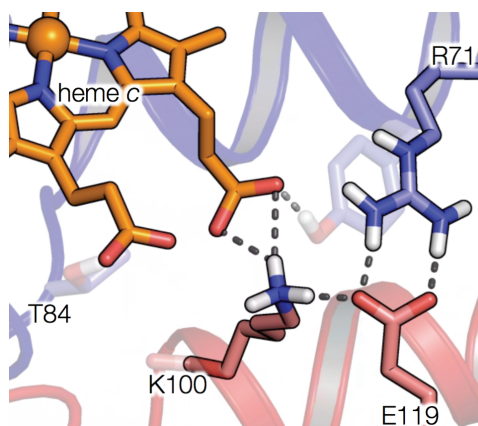


図7: NiR-NOR 結合部位のスナップショット

青色は NiR, 赤色は NOR, 点線は水素結合を表している。NiR の Lys61 は省略している。

考えられる。ナノディスク系ではこのペア数は約1.4となり、脂質二重膜系よりも水素結合の生成が非常に少なくなるという結果が得られた。しかし、今回実施したシミュレーション時間ではナノディスク系でもNiR:NOR複合体が壊れることはなかった。したがって、生体環境における複合体の構造安定性に対する、NiR-膜間のコンタクト部位の親和的相互作用がもたらす寄与はあくまで副次的であり、NiR-NOR間の結合部位における親和的相互作用の寄与が主要であると考えられる。

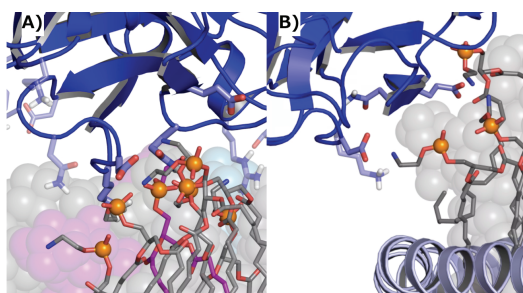


図8: NiR-膜コンタクト部位のスナップショット
A) NiR:NOR 複合体+脂質二重膜系と B) NiR:NOR 複合体+ナノディスク系。残基と水素結合している脂質分子はスティックで、そうではない脂質分子は球で表している。

5. 結論

生体内における酵素反応機構の理解を目指して、QM法、QM/MM混合法とMDシミュレーションを組み合わせた新規計算手法の開発に取り組んだ。計算対象として膜タンパク質NORの活性中心におけるNO還元反応を採用した。脂質二重膜にNORを埋め込み、その疎水性チャネルに4NO分子を配置した系のシミュレーションを実施したが、活性中心にNO分子が存在するスナップショットは得られなかった。同時に、NiR:NOR複合体系のMDシミュレーションも実施した。その生体環境における構造安定性に対して、NiR-NOR結合部位とNiR-膜のコンタクト部位で生成される水素結合が寄与しており、特に前者における親和的相互作用が重要であることが示唆された。

参考文献

- [1] Kumita, H., Matsuura, K., Hino, T., Takahashi, S., *et al.*, *J. Biol. Chem.* **2004**, 279 55247.
- [2] Blomberg, M. R. A., Siegbahn, P. E. M., *Biochemistry* **2012**, 51, 5186.
- [3] Shoji, M., Hanaoka, K., Kondo, D., Sato, A., Umeda, H., Kamiya, K., Shiraishi, K., *Mol. Simul.* **2014**, 112, 393.
- [4] Attia, A. A. A., Silaghi-Dumitrescu, R., *J. Mol. Model.* **2015**, 21, 130.
- [5] Matsumura, H., Hayashi, T., Chakraborty, S., Lu, Y., Moënne-Loccoz, P., *J. Am. Chem. Soc.* **2014**, 136, 2420.
- [6] Terasaka, E., Yamada, K., Wang, P.-H., *et al.*, *Prot. Natl. Acad. Sci. USA* **2017**, 114, 9888.
- [7] Lee, J., Cheng, X., Swails, J. M., *et al.*, *J. Chem. Theory, Comput.* **2016**, 12, 405.
- [8] Best, R. B., Zhu, X., Shim, J., *et al.*, *J. Chem. Theory, Comput.* **2012**, 8, 3257.
- [9] Jung, J., Mori, T., Kobayashi, C., *WIREs Comput. Mol. Sci.* **2015**, 5, 310.

問題分割と対称性検知を用いた、高速なプランニングアルゴリズムの開発

浅井政太郎

東京大学大学院 総合文化研究科

1. はじめに

本報告は、平成 29 年度スーパーコンピューター若手・女性利用者推進制度の前記課題「問題分割と対称性検知を用いた、高速なプランニングアルゴリズムの開発」の成果を報告する。内容は、国際会議 AAAI-2018 に採択された論文[Asai and Fukunaga, 2018] の日本語訳・要旨である。

2. Introduction

近年、ドメイン非依存プランニングソルバの性能は大幅に改善した。しかしプランニングソルバは、入力として、人間が環境を PDDL [McDermott, 2000] のようなモデリング言語を用いて離散表現に分割した **記号表現** を必要とし、そのような記号表現は、PDDL などのモデリング言語で人が直接入力するか、あるいは人力で作った他の記号表現を PDDL に変換する必要がある。その結果、プランニングの枠組みには、「知識獲得のボトルネック」と呼ばれる、現実世界を人間が理解しシンボルに翻訳するステップが存在する。

記号表現の要求は実応用における大きな障害である。自動プランニングは与えられたモデルで記述できる状況にしか対応できない。従って例えば、想定外の状況に事欠かず、かつ (モデル変更のための) 通信が容易でない環境 (例:火星探査ロボット) には適用できない。この問題を解決するためには、生のセンサー入力からいかにして記号モデルを生成するかという問題、すなわちシンボル接地問題[Steels, 2008]の解決が必要である。

近年、物体認識[Ren *et al.*, 2015]、音声認識[Deng *et al.*, 2013]、問題解決システム[Mnih *et al.*, 2015; Graves *et al.*, 2016]など、様々なニューラルネットワーク (NN) ベースの手法が成功を収めつつある。しかし、純粋な NN ベースの問題解決システムは、記号的システムの持つ理論的性質 (アルゴリズムの完全性や解の最適性など) を保証しない。

この両者の欠点を補完するため筆者らは、NN ベースの知覚システムを用いて、ドメイン非依存プランニングの記号的入力モデルを画像から自動的に生成し、解く手法を開発した。このことで本研究は、自動プランニング技術の適用性を大幅に拡大し、両方のパラダイムに貢献する。今回主題となるのは、**記号的システムと NN ベースシステムをいかにして繋げるか**である。なぜなら、NN ベースシステムの出力は通常、記号的システムには扱えない連続値だからである。

Fig. 1 左の写真は、右にある写真を 3x3 スライディングパズル (8 パズル) としてランダムに動かした例である。我々は、このようなパズルをドメイン非依存に自動で最適に解くことの出来る**潜在空間プランナ (LatPlan)** を制作した。システムの入力は、許される動作の前後を写している多数の画像ペアであり、それ以上のラベルは与えられていない。通常記号的ソルバにとってこのような 8 パズルは (記号表現がある限り) 自明な問題である。しかし今回扱うような、画像のみを入力として「動く物体」「格子状」「タイル」などの事前知識を一切持たずにドメイン非依存に問題を解決するシステムは、全く自明ではない。

LatPlan は以下の 3 つの要素からなる: (1) NN ベースの **State AutoEncoder (SAE)** はセンサー入力と記号表現の双方向の変換を担う。SAE の実装は Gumbel-Softmax[Jang *et al.*, 2017] 活性化関数を用いた Variational AutoEncoder [Kingma and Welling, 2013]で、その隠れ層の活性化値は離散値 (カテゴリカル 1-hot 表現) に収束する。ここでカテゴリ数を 2 に制限することで、1-hot 表現を命題の真偽値に対応させることが出来る。SAE を画像データを用いて入力と出力が一致するよう訓練すると、SAE は 画像入力→命



図1 画像によって与えられた8パズル。

題列→画像出力 (入力と一致) を達成するネットワークとして訓練される。結果、ネットワークの前段 (エンコーダ) および後段 (デコーダ) を用いて、ノイズのある実数値センサー入力を命題列に変換、および逆変換する関数が得られる。(2) **Action Model Acquisition (AMA)** は SAE が画像から生成した命題列から PDDL モデルを生成する。(3) 記号的ソルバが PDDL モデルを解き、記号的プランを探索結果として得る。このプランは未知の記号列で表現されているが、この命題列を SAE の逆変換により人間に理解可能な画像に戻すことができる。実験では、LatPlan を 8-puzzle, LightsOut, およびハノイの塔の画像版を用いて評価した。

3. 背景

3.1. 古典プランニング

古典プランニングソルバの性能は、知識あり前方探索探索の下界関数の進歩によって近年劇的に進歩している。ソルバ (プランナ) の入力は $\Pi = \langle P, O, I, G, A \rangle$ 、ここで P は一階述語論理の述語、 O は述語の引数となるオブジェクト、 I は初期状態、 G はゴール条件、そして A は探索空間で許される遷移を表している **アクションスキーマ集合** である。状態とは、命題変数に対する真偽値の割当てであり、条件はそのうち一部を指定する部分的割当てとなる。それぞれの命題変数は、述語にオブジェクトを適用したものである。アクションスキーマ $a \in A$ はさらに 5-tuple $\langle params, pre, e^+, e^-, c \rangle$ で表される。これは順にパラメータ、前提条件 (precondition)、追加効果、削除効果、コストを表す。アクションスキーマのパラメータを O 中のオブジェクトで代入すると、**アクション** が得られる。 c が指定されない場合は、通常 $c = 1$ が想定される。これらの入力は PDDL [Bacchus, 2000] のようなモデリング言語で記述される。

Fig. 2 は、3x3 パズル (8 パズル) を一階述語論理、および対応する PDDL で表現した一例である。

```

Empty(x0, y0)          (empty x0 y0)
^At(x1, y0, panel6)   (at   x1 y0 panel6)
^Up(y0, y1)           (up   y0 y1)
^Down(y1, y0)         (down y1 y0)
^Right(x0, x1)        (right x0 x1)
^Left(x1, x0) ...     (left x1 x0) ...

```

	6	8
7	3	2
5	1	4

図2 人間の作った、3x3 スライディングタイルパズル (8 パズル) の一階述語論理表現と PDDL 表現の例。これには、**述語記号** empty, up, down, left, right, at および**オブジェクト記号**, 例えば $x_i, y_i, panel_j$ ($i \in \{0..3\}, j \in \{1..8\}$) が含まれている。

古典プランニング問題を解くことは、許されたアクションを用いて初期状態からゴール条件を満たす状

```

When
Empty(x, yold)  ∧  (:action slide-up ...
at(x, ynew, p)  ∧  :precondition
up(ynew, yold) ;  (and (empty ?x ?y-old)
then              (at ?x ?y-new ?p) ...)
-Empty(x, yold) ∧  :effects
Empty(x, ynew) ∧  (and (not (empty ?x ?y-old))
-at(x, ynew, p)  ∧  (empty ?x ?y-new)
at(x, yold, p)   ∧  (not (at ?x ?y-new ?p))
                  (at ?x ?y-old ?p)))

```

	6	8
7	3	2
5	1	4

図3 同様に、タイル7を上をスライドさせるアクションの一階述語論理表現と PDDL 表現。ここではアクション記号 `slide-up` が用いられている。

態へ至る経路を発見することである。この際、状態 s にアクション a を適用して状態 t に遷移することを $t = a(s)$ と書き、 a は状態遷移関数と呼ばれる。この状態遷移は $s \supseteq pre$ が満たされたときのみ許され、結果として $t = (s \setminus e^-) \cup e^+$ が得られる [Bacchus, 2000]。

State-of-the-Art プランナはこの問題をグラフ上の経路探索問題として解く。定義からわかるようにグラフの形状は (例えば隣接行列などで) 明示的には与えられてはおらず、初期状態から指数関数的に大きな、メモリには収まらないサイズのグラフが生成される。したがって、これらの問題を解くためには知識あり探索を用いて高度な枝刈りを行いながら探索を行う必要がある。代表的なアルゴリズムとしては、最短経路と求める場合には A^* 、そうでない場合には Greedy Best-First Search がある。様々な下界関数のおかげで [Helmert and Domshlak, 2009; Sievers *et al.*, 2012; Helmert *et al.*, 2007; Bonet, 2013; Hoffmann and Nebel, 2001; Helmert, 2004; Richter *et al.*, 2008]、現在の state-of-the-art プランナは、最小でも数千ステップのプランを必要とする巨大な問題を解くことができる [Asai and Fukunaga, 2015]。

4. 知識獲得のボトルネック

理想的には、前節で見たような記号的表現を自動的に学習できる仕組みがあれば好ましいが、そのタスクを人間の助けなくまったく自動で行えるシステムはいままで実現できておらず、実用的なシステムは人間の手によるモデリングに依存している。そのため、記号的 AI システムは **知識獲得のボトルネック** [Cullen and Bryman, 1988]、すなわち、**人間が現実世界を記号表現にモデリングするのにかかるコスト** を避けられない。

古典プランニングのためのモデルを完全自動で生成するためには、**記号接地** と **Action Model Acquisition (AMA)** が必要である。**記号接地** とは、「ノイズで、連続値の、構造を持たない巨大な入力から、コンパクトで、ノイズがなく、離散値で、分離された個別の要素すなわち記号への変換関数を、教師無しで学習する」操作である。PDDL は 6 種の記号を持つ: オブジェクト、述語、命題、アクション、問題、問題ドメインである (Table 1)。異なる種類の記号はそれぞれ別の方法で接地される。たとえば、画像認識のコミュニティでは画像中の物体 (例:顔)、その属性 (例:男女)、あるいは動画中の動き (例:料理) を検出する手法がそれぞれ提案されている。これらはそれぞれオブジェクト、述語、アクションの記号を接地していると考えられる。

一方、**アクションモデル** とは、環境の状態遷移を表すなんらかのデータ構造である (記号的であるとは限らない)。PDDL でのアクションモデルは、それぞれのアクション記号に付随する前提条件と追加・削除効果に相当する (Fig. 2)。

この研究では、命題記号とアクション記号の接地、およびアクションモデルの生成に焦点を絞る。

記号の種類	
オブジェクト記号	panel7, x₀, y₀ ...
述語記号	(empty ?x ?y) (up ?y₀ ?y₁)
命題記号	empty₅ = (empty x₂ y₁) (6th application)
アクション記号	(slide-up panel₇ x₀ y₁)
問題記号	eight-puzzle-instance1504, etc.
ドメイン記号	eight-puzzle, hanoi

表 1 PDDL 中の 6 種の記号。

4.1. 既存の AMA 手法

既存の AMA 手法は、すでに接地された記号からアクションモデルを生成することを仮定している。したがって、その入力記号は記号的、あるいは「ほぼ」記号的な入力である。ARMS [Yang *et al.*, 2007], LOCM [Cresswell *et al.*, 2013], Mourão *et al.* (2012) はアクション記号、オブジェクト記号、述語記号を仮定している。Framer [Lindsay *et al.*, 2017] システムは自然言語を入力として PDDL を出力するが、用いられる自然言語入力には一定の文法構造や一貫した語の使用などが仮定されており、ほぼ記号的な入力を扱っている。

Konidaris *et al.* は PDDL を semi-MDP (2014) と呼ばれる連続値モデルから生成する。しかし、semi-MDP は *move*, *interact* など人間によって与えられたアクション記号を持つ **別の形式のモデル**に過ぎず、実際に行われたことはモデルからモデルへの変換であり、生の現実世界から自動で記号を抽出したとはいえない。また、状態を表す入力も、画像などの非構造データではなく、物体への x/y 軸距離、ライトの光量、スイッチの状態など、それぞれ区別可能な構造にすでに分解された記号的な入力である。

4.2. オートエンコーダと隠れ表現

オートエンコーダ (AE) は、出力が入力と一致するように学習を行う前方伝播ニューラルネットである [Hinton and Salakhutdinov, 2006]。その中間レイヤは入出力よりも小さく、入力の**圧縮された隠れ表現**を保持していると捉えられている。AE は入力からの誤差を最小化するように逆誤差伝播法 (Backpropagation) により訓練される。ニューラルネットのニューロンは一般に連続値の活性化値を持つため、これらのシステムを記号的なシステムと接続する方法は自明ではない。

5. LatPlan: システム構成

この章は、LatPlan の全体構造を俯瞰する (Fig. 4)。LatPlan は 2 つの入力を持つ。第 1 の入力は、生データの対のセットである**遷移入力** Tr である。各対 $tr_i = (pre_i, post_i) \in Tr$ は何らかのアクションが実行された前後の環境の変化を表している。第二の入力は **プランニング入力** (i, g) (生データの対) であり、これは環境の初期状態とゴールに対応する。LatPlan の出力は、 i から g へのプラン実行の様子を示すデータ列である。本研究ではこの「データ」の種類として画像を用いる実装を制作したが、アーキテクチャ自身にはそのような仮定は置いていない。したがって、このアーキテクチャは音声・テキストなど他の種類に拡張されることが予想される。

LatPlan は 3 段階にかけて動作する。ステップ 1 では、**State Autoencoder (SAE)** が、環境を写した生の状態データ (画像) とその命題表現の双方向変換関数を教師無しで学習する。SAE の *Encode* 関数は画像を命題表現に変換し、*Decode* 関数は命題表現を画像に戻す。遷移入力 $Tr = \{pre_i, post_i \dots\}$ を用いて SAE を訓練した後、LatPlan は Tr をすべて記号表現 $(Encode(pre_i), Encode(post_i)) = (s_i, t_i) = \overline{tr}_i \in \overline{Tr}$ に変換する。

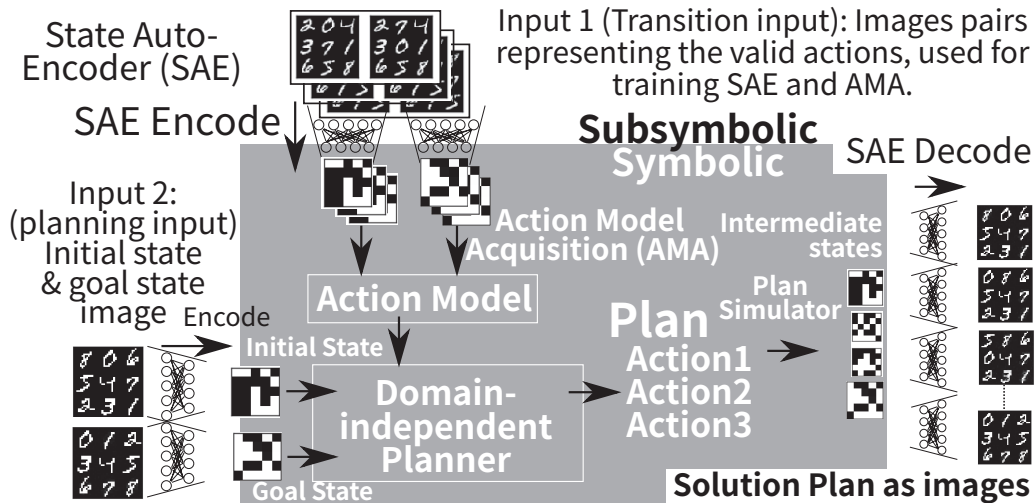


図 4 隠れ空間での古典プランニング: 学習済みの State Autoencoder (Sec.6) を使い、画像ペア (*pre, post*) を記号的遷移に変換する。次に、記号的状態遷移からアクションモデルを学習する。次に、初期状態・ゴール状態を記号的初期状態・ゴール状態に変換する。次に、古典プランナが、記号的空間でゴールへ遷移するプランを求める。最後に、プランの中間状態が State Autoencoder より画像へとデコードされ可視化される。

ステップ 2 では、AMA 手法が \overline{Tr} から アクション記号とアクションモデルを同時に教師なし学習する。本研究では 2 つのアプローチを提案する: AMA_1 は PDDL を直接生成し、一方 AMA_2 は PDDL は生成しないがグラフ探索に用いる 後者関数を生成する。各々に利点と欠点がある。 AMA_1 は 自明な AMA であり、その目的は SAE が生成した命題表現がプランナによる論理推論と互換性があることを証明することである。 AMA_1 は少数例から汎化する能力を持たず、したがって許可された全遷移の集合を入力として必要とする。したがって AMA_1 は AMA として実用的な手法とは言えないが、直接 PDDL を生成できることから、SAE を用いることで生データを既存の古典プランナに与える手法の実現可能性を示すことができる。 AMA_2 は、本論文で示す新しいニューラルネットであり、少ない状態遷移データを汎化しつつアクション記号とアクションモデルを同時に教師無しで学習する。既存の AMA 手法と異なり、 AMA_2 はアクション記号を必要としない。しかし、PDDL を出力しないので、一般的な PDDL ソルバは使えず、それ専用の探索アルゴリズム実装 (A* など) を必要とする。

ステップ 3 では、記号的プランニング問題インスタンスがプランニング入力 (i, g) から生成される。 (i, g) は SAE によって記号表現に変換され、記号的ソルバがこの探索問題を解く。

最後に、生成されたプランが画像にデコードされ可視化される。SAE の出力する命題表現は人間の作った表現ではないので、それぞれの命題の「意味」は人間の観測者にとって自明ではない。しかし、SAE が *Decode* 関数を持つ双方向関数であることから、命題表現で表された探索結果を画像に戻し、可視化することができる (例: Fig. 6)。

6. SAE as a Gumbel-Softmax VAE

画像から命題への 1 対 1 のマッピング自体は、自明に求めることができる。単に離散化された画素値の配列や、または画像ハッシュ関数を用いればよい。しかし、そのような SAE には、一般化 — すなわち、始めて観測した世界の状態を同じシンボルで表す能力、ロバストネス — 同じ状態を表す 2 つの類似した観測を同じ命題表現にマップする能力、双方向性 — 命題表現をもとの表現に戻す能力、が欠けている。Latplan を実

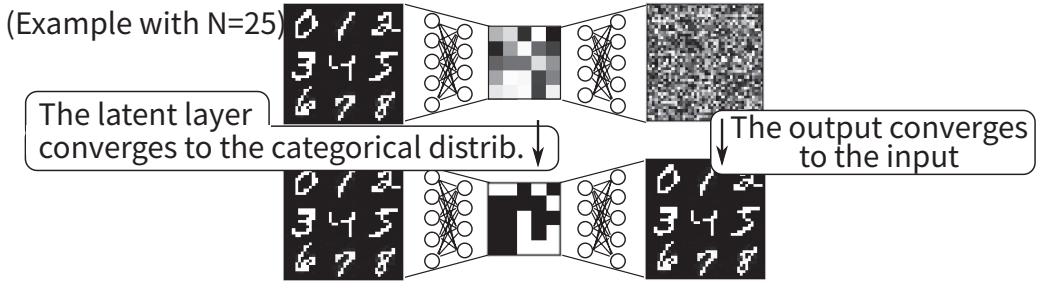


図5 Step 1: 入力との誤差を最小化すると同時に離散値とのKLダイバージェンスを最小化することで、State Autoencoder を訓練する。訓練が進むに連れ、出力は入力に収束し、かつ中間層は離散値に収束する。

現するためには双方向関数が必要であり、Latplan の SAE は生の画素ベクトルのような命題表現ではなく、画像の「エッセンス」をキャプチャした命題表現を得なくてはならない。

本論文の第一の貢献は、SAE を Gumbel-Softmax (GS) [Jang *et al.*, 2017] 活性化関数を用いた Variational Autoencoder [Kingma *et al.*, 2014] として実装する手法である (Fig. 5)。

Variational Autoencoder (VAE) [Kingma and Welling, 2013] は、隠れ層の活性化の分布をある一定の分布 (例:正規分布) に収束させられる特殊なオートエンコーダである。正規分布のようなランダム変数は微分不可能なので、VAE は **reparametrization trick**、すなわち対象となる分布をランダム変数と学習可能な変数に分割する操作を行う。例えば、正規分布 $N(\sigma, \mu)$ は $\mu + \sigma N(1, 0)$ に分解できる。

Gumbel-Softmax (GS) は近年提案された reparametrization trick [Jang *et al.*, 2017] で、活性化値の分布を離散カテゴリカル分布に収束させることが出来る。GS は Gumbel-Max [Maddison *et al.*, 2014] というサンプル手法の微分可能な近似手法と捉えられる。出力 z が one-hot 表現、例えば値域が $D = \{a, b, c\}$ であるとき $\langle 0, 1, 0 \rangle$ は “b” を表す。Gumbel-Max の入力クラスに属する確率を表すベクトル π 、例えば $\langle .1, .1, .8 \rangle$ である。このとき Gumbel-Max は D から π に従って以下のように離散値をサンプリングする: $z_i \equiv [i = \arg \max_j (g_j + \log \pi_j) ? 1 : 0]$ ただし、 g_j は Gumbel(0, 1) [Gumbel and Lieblein, 1954] から相互に独立に得られたサンプルである。Gumbel-Softmax は $\arg \max$ を softmax で近似することにより全体を微分可能にする: $z_i = \text{Softmax}((g_i + \log \pi_i)/\tau)$ 。ここで、「温度」 τ は近似の度合いを決定し、0 へと冷却される。温度が 0 に近づく時、GS の出力は Gumbel-Max の離散サンプルに収束する。

ここで鍵となるのは、これらの離散変数が命題として直接使用できるということである。すなわち、これは Latplan アーキテクチャの命題記号接地に対する解を与える。SAE では、潜在層に GS を使用する。その入力はエンコーダネットワークに接続される。GS の出力は (N, M) 行列である。ここで N はカテゴリ変数の数、 M はカテゴリの数を表す。ここで $M = 2$ と指定すれば、結果として得られる表現は N 個の命題変数として捉えることが可能になる。

訓練後の SAE は、生の入力と命題表現の双方向マッピングを提供する:

- $b = \text{Encode}(r)$ は画像 r を Bool 配列 b に変換する。
- $\tilde{r} = \text{Decode}(b)$ は Bool 配列 b を画像 \tilde{r} に変換する。

$\text{Encode}(r)$ は、生の入力 r をエンコーダネットワークに入力し、GS 層の出力を取り出し、 $N \times 2$ 配列の一行目を取り出すことで長さ N の bool 配列を得る。 $\text{Decode}(b)$ は bool 配列 b をその補数 \bar{b} と結合して $N \times 2$ 配列を得、これをデコーダネットワークに入力することで画像を取り出す。

通常の活性化関数を丸めることで離散値を得る手法では、デコーダネットワークが 0/1 の値を用いて訓練されていないので、命題表現を正しくデコードすることが出来ない。SAE の実装はディープラーニングコミュニ

ニティの様々な技術革新から直接恩恵を受けることが出来る。実際、筆者らが用いた実装は Denoising AE [Vincent *et al.*, 2008] によって AE にノイズ耐性をつけている。

7. AMA₁: Oracular PDDL Generator

最初の AMA 方式である AMA₁ は、PDDL を直接出力する。AMA₁ は、環境で許された全ての遷移を入力として動作する **神託機械的** 戦略であり、理論的に最適な AMA 方式をシミュレートしている。従って、その手法自体はスケーラビリティの観点からは実用的ではない。ただし、AMA₁ が全ての遷移 (すなわち全ての状態を含む) を入力とする一方、SAE は少数の画像データから訓練されたことを明記しておきたい。繰り返すが、AMA は理想的には少数の訓練サンプルから学習を行うべきだが、AMA₁ の目的は SAE を用いた命題表現を用いてプランナが推論を行えるかを示すために意図的に制作されたものである。

AMA₁ は \overline{Tr} を以下のようにして PDDL に変換する。それぞれの遷移 $(s_i, t_i) \in \overline{Tr}$ は各々個別のアクション a_i に対応する。Bool 配列 s_i および t_i 中のそれぞれのビット $b_j (1 \leq j \leq N)$ は、1 のときには命題 (b_j -true)、0 のときには (b_j -false) に変換される。 s_i はアクション a_i の前提条件として直接用いられる。追加・削除効果は s_i と t_i のビット毎の差分から作られる。例えば、 b_j が 1 から 0 になるとき、対応する効果は ($\text{and } (b_j\text{-false})$ ($\text{not } (b_j\text{-true}))$) である。初期状態とゴールも同様に変換される。

AMA₁ の生成した PDDL インスタンスは既存のプランナによって解かれる。我々は Fast Downward [Helmert, 2006] ソルバを用いた。従って、LatPlan は用いるソルバの探索アルゴリズムの理論的性質を全て引き継ぐ。例えば、プランナが完全かつ最適であれば、LatPlan は解が存在すれば必ずそれを返却し、また返却される解は最短の解である。

7.1. AMA₁ の評価

我々は LatPlan + AMA₁ を複数のパズルドメインで評価した。結果として生成されたプランを Fig. 6-7 に示す。

MNIST 8-puzzle は 8 パズルの画像ベースのバージョンで、タイルには MNIST データベース [LeCun *et al.*, 1998] の数字が書かれている。このドメインの有効なアクションは、「0」タイルを隣接するタイルと交換する操作である。すなわち、「0」は通常の「空」タイルとして機能する。マンドリル、スパイダー 8 パズルは、一般の店で売っているような写真を持ちいた 8 パズルである。これらは、MNIST の 8 パズルとは違って各「タイル」が黒色の領域できれいに分離されていない。これは、LatPlan には「格子状の領域」といったような概念を必要としないことを示している。**ハノイの塔 (ToH)** では、円盤 4 つのインスタンスを解き、15 ステップの最適プランを発見している。LightsOut は小型の電子ゲームをベースとしたドメインで、ライトのグリッドがオン/オフ設定 (+ : オン) になっていて、ライトを押すとそのライトおよびその隣接ライトの状態が反転する。ゴールはすべて消灯された状態である。以前とは異なり 1 つのアクションが 5/16 の位置を一度に反転し、またいくつかの「オブジェクト」(ライト) を取り除くため、このドメインは、LatPlan が「局所的な変化」や「一定のオブジェクトが安定して存在すること」などを仮定していないことを表す。最後に、**Twisted LightsOut** は元の LightsOut の画像をスワール効果で歪ませて入力としたもので、これは LatPlan が長方形の「オブジェクト」/領域に限定されないことを示している。

7.2. ノイズ耐性

Fig. 8 は学習済みの SAE にノイズを含んだ入力を与えたときのデコード結果である。SAE が Denoising AE であることから、ノイズを含まない出力を返す。この SAE を用いることで、初期状態やゴール状態がノイズを含んでいても、同様にプランを生成できる。

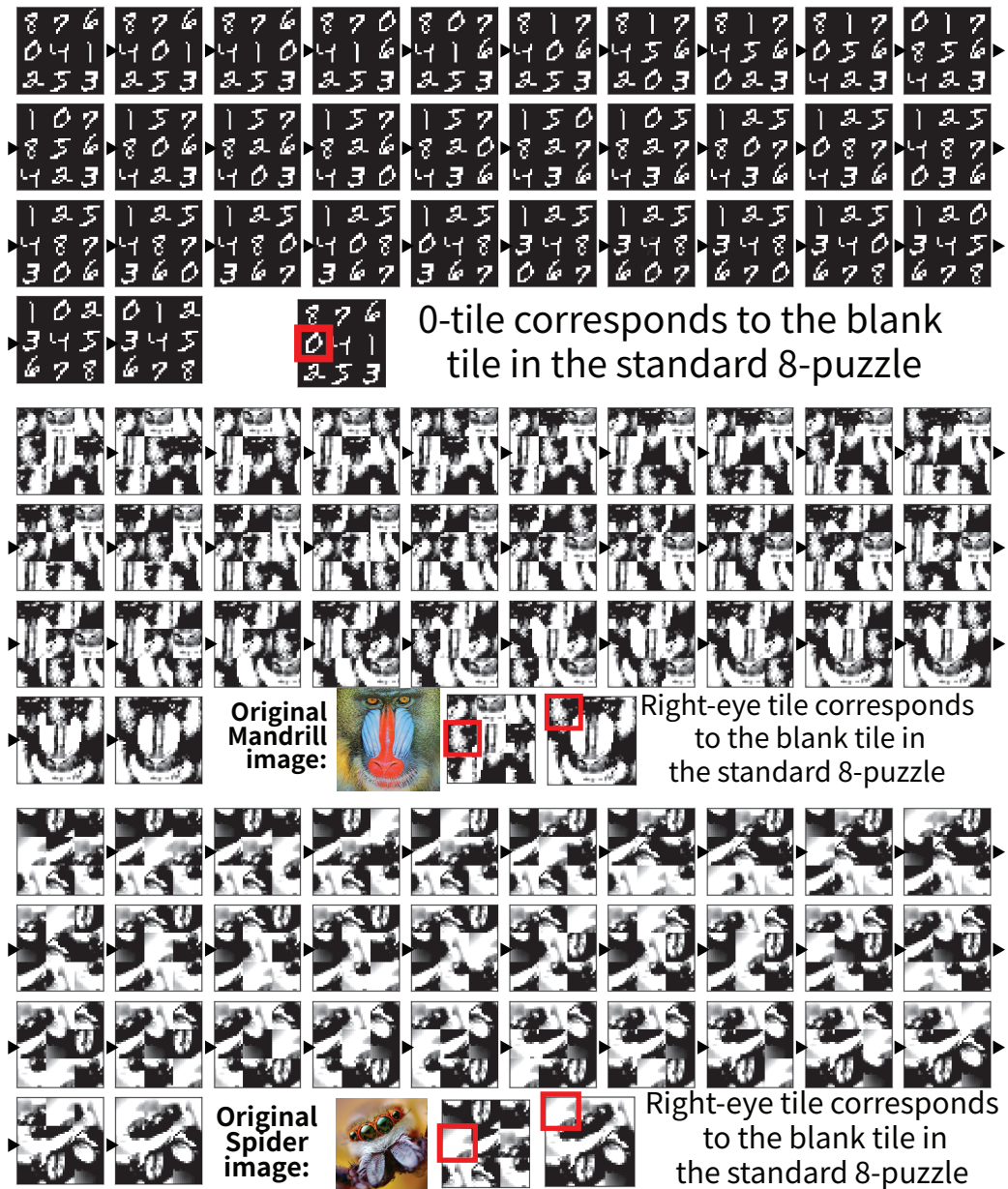


図6 LatPlan + AMA₁ が MNIST/Mandrill/Spider 8 パズルを解いた例。このインスタンスは、8 パズル中最も長い解を持つ (31 ステップ) 問題例であり、LatPlan は最適解を見つけている (Reinefeld 1993)。これは、神託的 AMA₁ を与えられれば、LatPlan が SAE による命題変数を用いて最適解を正しく探索できることを示している。これにより、LatPlan は「スライド」「タイル」などの事前知識は全く与えられること無しに記号の推論を行えることが示された。

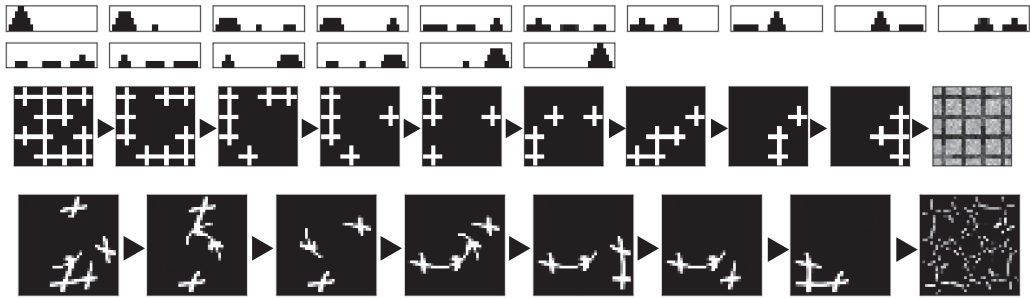


図7 (1) ハノイの塔の例。(2-3) LightsOut と Twisted LightsOut の例。ゴール状態に見える残渣は、小さなノイズが可視化ライブラリによって正規化・強調されたもの。

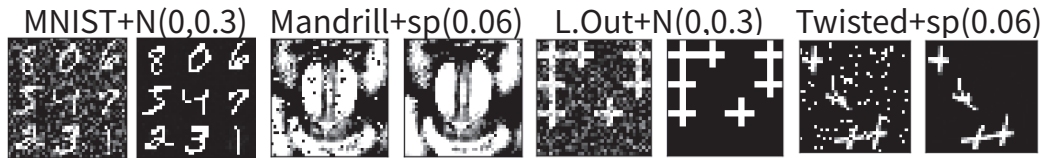


図8 SAE のノイズ耐性: ノイズを追加された初期状態 r とその復号結果 $Decode(Encode(r))$ 。ノイズにはガウシアンノイズと ごま塩ノイズを用いた。LatPlan はこれらに対してもプランを返却した。

8. AMA₂: Action Symbol Grounding

LatPlan + AMA₁ は、(1) SAE が画像 \leftrightarrow 命題ベクトルを学習できること、および (2) 全ての有効な画像から画像への遷移 (つまり状態空間全体) を与えられれば、LatPlan が最適な計画を正しく生成できることを示した。しかし、AMA₁ は状態空間全体を入力として必要とするために実用的ではなく、実際には少数の状態遷移の例からアクションモデルを学習/一般化する能力が必要である。そこで、AMA₂ を提案する。これは、少ない状態遷移データを汎化しつつアクション記号とアクションモデルを同時に教師無しで学習する新たなニューラルアーキテクチャである。

AMA₂ は、Action Autoencoder (AAE) と Action Discriminator (AD) という2つのネットワークで構成されている。AAE は同時にアクション記号と効果を学習し、現在の状態から遷移可能な状態の候補を列挙する能力を提供する。AD は、候補のうちどの遷移が実際に有効であるか、すなわち前提条件を学習する。後者状態 (Successor states) の候補を列挙・フィルタリングすることで、AMA₂ は正しい後者関数を提供する。

8.1. Action Autoencoder

仮に、枝分かれのない線形な探索空間を考えよう。この場合、アクション記号は不要であり、AMA の目的は、現在の状態 s の唯一の後者状態 t を 予測 することと等価になる。従って NN a' は、損失関数 $|t - a'(s)|$ を最小化することで後者関数 $a(s) = t$ を近似するように訓練することができる。このような単純化は ビデオからのシーン予測 [Srivastava *et al.*, 2015] など多くの既存研究で行われている。

しかし、現在の状態がプランニング問題のように複数の後者状態を持つ場合、そのような単純化は適用できない。1つのアクションごとに別々の NN をトレーニングすることも考えられるが、(1) 何個のアクションがあるかもわからない、(2) 状態によって許されるアクションの数が変わる、(3) そもそもどの状態遷移がそのタイプのアクションに属するかも解らない、ということから、訓練のしようがない。単一の NN がマルチモーダル分布を学習することは可能かもしれないが、それは探索アルゴリズムの重要な必要条件である後

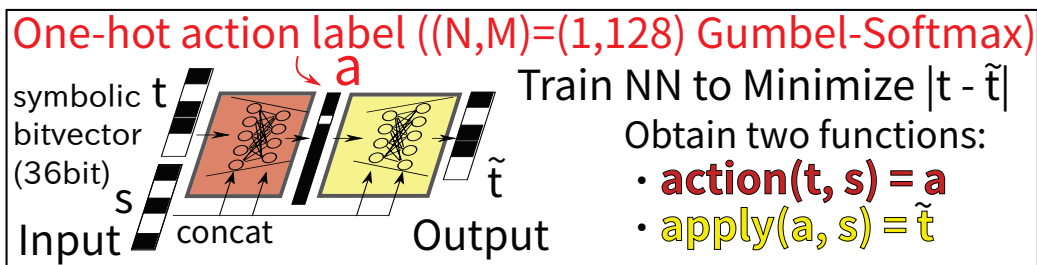


図9 Action Autoencoder.

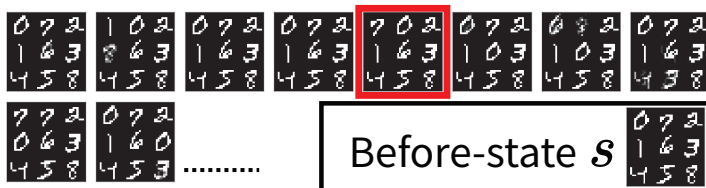


図10 右下の状態の後者状態候補をAAEが列挙したもの。このうち正しいものは赤枠のものだけである。

者状態の列挙が不可能である。

これを解決するために、Action Autoencoder(AAE, Fig.9)を提案する。AAEの重要なアイデアは、遷移を $apply(a, s) = t$ と再定式化することである。この定式化から、アクション記号は訓練されるべき変数であり、また s は状態遷移の背景情報であることが分かる。そこでAAEは、 s, t を入力とし、 t を \tilde{t} と自己符号化するよう誤差 $|t - \tilde{t}|$ を最小化する。典型的なAEとの主な違いは次のとおりである。(1) 潜在層はGumbel-Softmax one-hot 表現であり、これがアクションラベルを示すベクトル a である。(2) 全ての層がそれぞれ s と concatenate される。これは、ネットワークの表現全体が s で条件付けられることを意味する。これにより、128個のアクション記号(7ビット)が、「 s を与えられた場合に t を再構成する」ために必要な条件付き情報(差分)のみを保持することになる。一方、通常のAEは入力の全情報をエンコードする。

結果、AAEは t と a の双方向変換関数を学習するが、変換関数は共に s によって条件付けられている:

- $Action(t, s) = a$ は t からアクション記号 a を返す。
- $Apply(a, s) = \tilde{t}$ はアクション a を s に適用し、後者状態 \tilde{t} を返す。

アクション記号の数(128)は、実際に必要な記号の数の上界となっており、学習後には、どの入力も対応しないような記号が存在しうる。少数のアクション記号を得たことから、LatPlanは現在の状態の後者状態の候補を定数時間で返却することが出来る。仮にAAEが無ければ、 2^N 個の全ての状態を列挙・枝刈りする必要があるが、これはあきらかに現実的ではない。

8.2. Action Discriminator

AAEはアクションの効果(差分)を学習しているが、それぞれのアクションがいつ許されるかという前提条件は学習していない。前提条件による制限がなければ、実際に例えば、3つのタイルを同時に移動したり、同じ番号のタイルを複製したりという推論を行ってしまう Fig.10。そこで、Action Discriminator (AD, Fig.11)が、それぞれの遷移に対して正しいか否かの確率を返すよう訓練される。これは通常の教師あり二値学習機であり、 s, t を入力としてそれが正しい確率を返す。

ここで問題になるのは、二値分類機を訓練するためには「間違い」とラベル付された遷移が必要だが、手

Action Discriminator

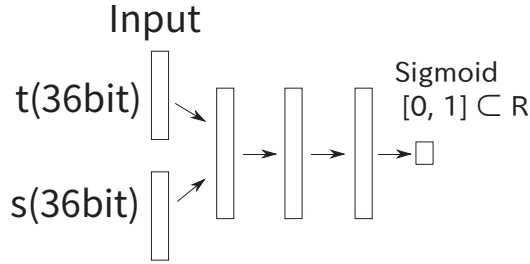


図 11 Action Discriminator.

元にそのような負データがないということである。ここで注意したいのは、**現実世界では、「間違っただけ」遷移は観測することができない**ということである。例えば、テレポーテーションや念力などといった非現実的アクションは「起こり得ず」従って「データは存在しない」。物理環境で動作する現実的ロボットには、従って、明示的な負データを集めることが出来ない。負データを機械的に生成することも考えられるが、そもそも何が正データなのかすら解らないので、負データに対する生成のルールも存在しない。

この問題を解決するために、 AMA_2 は PU-Learning [Elkan and Noto, 2008] を用いる。これは、正負の二値分類機を 正例と 未分類例 から学習するための手法である。ここで、 \overline{Tr} は観測データであるから正例である。未分類例 (正例と負例の混ざったデータ) は、AAE が生成した後者状態の候補である。

8.3. AMA_2 を用いたプランニング

AMA_2 は PDDL を出力しないので、PDDL ベースの既存プランナを用いることが出来ない。しかし、AAE と AD を用いれば後者関数を得られ、これによって A^* [Hart *et al.*, 1968] を用いた記号的探索が可能になる。AAE は (間違っただけを含む) 後者状態候補を返し、AD はその候補を分類する:

$$\begin{aligned} Succ(s) = \{ & t = apply(a, s) \mid a \in \{0 \dots 127\}, \\ & \wedge AD(s, t) \geq 0.5 \} \end{aligned}$$

我々は、状態を bit 配列とし、上記の $Succ$ 関数を後者関数とする A^* を実装した。下界関数には goal-count heuristic (ゴールと異なる bit の数) を用いた。Goal-count heuristics は許容的ではないため結果は非最適だが、この実験の目的はそもそも AMA_2 が正しいプランを生成できるのかを示すことであり、最適性は重要ではない。

8.4. 評価

我々は、AAE によって学習されたアクション記号とアクションモデルの正しさを評価した。評価するドメインには 8 パズル (mnist, mandrill, spider)、LightsOut (+ Twisted) を用いた。それぞれのドメインごとに 200 インスタンスを生成し、正しいプランが返された数を計測した。それぞれのインスタンスにはガウシアン・ごま塩ノイズを加えたバージョンも加えた。200 インスタンスのうち 100 題はゴールから 7 歩のランダムウォークで生成し、残りは 14 歩で生成した。プランナは 3 分の時間制限で評価した。生成されたプランの正しさは、8 パズルなどドメインごとにバリデータを書いて用い、また適宜目視も行った。

Table 2 からは、LatPlan が十分に高い成功率を保っていることが分かる。また、解かれなかった場合の多くは時間切れに依るものである (通常のプランニングと異なり、後者状態の生成のたびに重いニューラルネットの計算が行われるため)。

domain	A:step=7			B:step=14			AD error (in %)	
	std	G	s/p	std	G	s/p	type1	type2
MNIST	72	64	64	6	4	3	1.55	6.15
Mandrill	100	100	100	9	14	14	1.10	2.93
Spider	94	99	98	29	36	38	1.22	4.97
L. Out	100	99	100	59	60	51	0.03	1.64
Twisted	96	65	98	75	68	72	0.02	1.82

表 2 AMA₂: (左) 3 分間の間に正しく解けた問題の数。第二/第三列は G(aussian) / ごま塩 (s/p) ノイズを加えた結果。7 歩のランダムウォークで生成されたベンチマーク A では、LatPlan は過半数の問題を解けている。より難しい 14 歩のベンチマークでも、依然として多くの問題が解かれている。(右) AD の type-1/2 エラー (%): (type-1) 全ての正しい遷移を生成し、AD が否定したものの割合。(type-2) まず、1000 個の状態を生成し、AAE によって後者状態候補を作り、バリデータによって正しいものを取り除いて負例を生成する。このうち、AD が正例と判定したものの割合。一番エラー率の高い MNIST で、失敗したインスタンスが多くなっている。

次に、人工的に作成したデータで AD の正確さを検証した (Table 2)。具体的には、全て正例/負例のデータセットに対して AD の type-1/2 エラーを計測した。全体に渡ってあるていど低いエラー率が観測され、またエラー率がおおよそ失敗の数に対応する傾向があることが分かる。

9. Related Work

ニューラルネットを探索と組み合わせる既存手法には、まず NN を組合せ最適化問題の探索に直接使う手法、たとえば有名な NN ベース巡回セールスマンソルバ [Hopfield and Tank, 1985] がある。Neurosolver は探索ノードを NN のノードに対応させてハノイの塔などを解く [Bieszcza and Kuchar, 2015] いっぽう、これらのシステムの入力はもともと記号表現である。

その他の手法には、知識あり探索の下界関数を得るために NN を探索の内側で用いる手法がある。例えば、スライディングタイルパズルやルービックキューブ [Arfaee *et al.*, 2011]、古典プランニング [Satzger and Kramer, 2013]、あるいは囲碁 [Silver *et al.*, 2016] などに適用されている。

深層強化学習 (DRL) は、ビデオゲームなど複雑な問題を解くことができる [Mnih *et al.*, 2015, DQN]。ただし、これらのシステムは訓練と実行の際にシミュレータに自由にアクセスできることを仮定している。また、DRL システムは特定の初期状態に強く条件付けられているということが知られており、かつ長い時間がたった後に得られる報酬に弱いことが知られている。一方 LatPlan は、限られた数のラベルなしの状態遷移例を与えられるだけで自ら環境のモデル (\approx シミュレータ) を生成し、そのなかで探索を行ってプランを生成するため、外部シミュレータを必要としないばかりか、DRL のように報酬関数を必要とせず、かつ一回の訓練で複数の初期状態・ゴールの問題を解け、また AlphaGo のように専門家による例も必要としない。さらには、DQN や AlphaGo は「石の置き場所」「ボタン」「コントロールレバー (上下左右)」など、決められたセットのアクションの存在を仮定しているが、LatPlan はこれらも必要としない。

10. 結論

我々は LatPlan を提案した。LatPlan は、学習と計画の統合アーキテクチャであり、ラベルなし画像のセットから事前知識なしに古典的な計画問題を生成し、これを記号的ソルバで解き、その計画を人間が理解できる画像列として提示する。本論文では、画像ベースの探索問題 (8 パズル、ハノイの塔、LightsOut) を用いてこのようなアプローチが実現可能であることを示した。本論文の主な技術的貢献は以下のとおりであ

る。(1) **Gumbel-Softmax** を用いて生データと記号表現の双方向マッピングを学習する **SAE** 実装。8 パズルでは、42x42 の訓練画像の「要点」が命題表現に圧縮され、プランナはその命題表現上で動作する。(2) アクション記号とアクションモデルを同時に学習する **AMA₂** システム。これは少数の状態遷移例から、状態遷移の分類ラベルであるアクション記号と、それらの表す効果、および前提条件を学習する。

このシステムがもつ唯一の仮定は、対象となる環境が (1) 完全情報・決定的環境であり、かつ (2) ニューラルネットが与えられたデータから学習できることのみである。従って、我々は、同じシステムが、様々な種類の問題を、一行もコードに変更を加えること無く等しく解くことが出来ることを示した。言い換えるならば、**LatPlan** は、**ドメイン非依存の画像ベース古典プランニングソルバ**である。様々な問題に対して **記号的システムと直接互換性のある** 論理的表現をラベルなしデータから直接生成できるこのようなシステムは、我々の知る限りでは **LatPlan** が初めてである。

我々はシミュレータ無し、エキスパートデータなし、報酬関数無し、アクション前後の状態を表す画像対のみが与えられたときに、深層学習を活用して A* のような古典的な探索アルゴリズムを使用する記号的プランニングを実現可能であることを示した。このアプローチのスケラビリティを判断するためにはより多くの検証が必要だが、これは記号的・非記号的推論のギャップを橋渡しする重要な第一歩であり、将来の研究のための多くの道を開くと考えている。

参考文献

- [Arfaee *et al.*, 2011] Shahab Jabbari Arfaee, Sandra Zilles, and Robert C. Holte. Learning Heuristic Functions for Large State Spaces. *Artificial Intelligence*, 175(16-17):2075–2098, 2011.
- [Asai and Fukunaga, 2015] Masataro Asai and Alex Fukunaga. Solving Large-Scale Planning Problems by Decomposition and Macro Generation. In *ICAPS*, Jerusalem, Israel, June 2015.
- [Asai and Fukunaga, 2018] Masataro Asai and Alex Fukunaga. Classical Planning in Deep Latent Space: Bridging the Subsymbolic-Symbolic Boundary. In *AAAI*, New Orleans, USA, February 2018.
- [Bacchus, 2000] Fahiem Bacchus. Subset of PDDL for the AIPS2000 Planning Competition. In *IPC*, 2000.
- [Bieszczad and Kuchar, 2015] Andrzej Bieszczad and Skyler Kuchar. Neurosolver Learning to Solve Towers of Hanoi Puzzles. In *IJCCI*, volume 3, pages 28–38. IEEE, 2015.
- [Bonet, 2013] Blai Bonet. An Admissible Heuristic for SAS+ Planning Obtained from the State Equation. In *IJCAI*, 2013.
- [Cresswell *et al.*, 2013] Stephen Cresswell, Thomas Leo McCluskey, and Margaret Mary West. Acquiring planning domain models using **LOCM**. *Knowledge Eng. Review*, 28(2):195–213, 2013.
- [Cullen and Bryman, 1988] J Cullen and A Bryman. The knowledge acquisition bottleneck: Time for reassessment? *Expert Systems*, 5(3), August 1988.
- [Deng *et al.*, 2013] Li Deng, Geoffrey Hinton, and Brian Kingsbury. New Types of Deep Neural Network Learning for Speech Recognition and Related Applications: An Overview. In *ICASSP*, pages 8599–8603. IEEE, 2013.
- [Elkan and Noto, 2008] Charles Elkan and Keith Noto. Learning Classifiers from Only Positive and Unlabeled Data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 213–220. ACM, 2008.
- [Graves *et al.*, 2016] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid Computing using a Neural Network with Dynamic External Memory. *Nature*, 538(7626):471–476, 2016.
- [Gumbel and Lieblein, 1954] Emil Julius Gumbel and Julius Lieblein. Statistical theory of extreme values and some practical applications: a series of lectures. 1954.
- [Hart *et al.*, 1968] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2):100–107, 1968.
- [Helmert and Domshlak, 2009] Malte Helmert and Carmel Domshlak. Landmarks, Critical Paths and Abstractions: What’s the Difference Anyway? In *ICAPS*, 2009.
- [Helmert *et al.*, 2007] Malte Helmert, Patrik Haslum, and Jörg Hoffmann. Flexible Abstraction Heuristics for Optimal Sequential Planning. In *ICAPS*, pages 176–183, 2007.
- [Helmert, 2004] Malte Helmert. A Planning Heuristic Based on Causal Graph Analysis. In *ICAPS*, pages 161–170, 2004.
- [Helmert, 2006] Malte Helmert. The Fast Downward Planning System. *J. Artif. Intell. Res.(JAIR)*, 26:191–246, 2006.
- [Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507, 2006.

- [Hoffmann and Nebel, 2001] Jörg Hoffmann and Bernhard Nebel. The FF Planning System: Fast Plan Generation through Heuristic Search. *J. Artif. Intell. Res.(JAIR)*, 14:253–302, 2001.
- [Hopfield and Tank, 1985] John J Hopfield and David W Tank. "Neural" Computation of Decisions in Optimization Problems. *Biological Cybernetics*, 52(3):141–152, 1985.
- [Jang *et al.*, 2017] Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. In *ICLR*, 2017.
- [Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. In *ICLR*, 2013.
- [Kingma *et al.*, 2014] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-Supervised Learning with Deep Generative Models. In *NIPS*, pages 3581–3589, 2014.
- [Konidaris *et al.*, 2014] George Konidaris, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Constructing Symbolic Representations for High-Level Planning. In *AAAI*, pages 1932–1938, 2014.
- [LeCun *et al.*, 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-Based Learning Applied to Document Recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998.
- [Lindsay *et al.*, 2017] Alan Lindsay, Jonathon Read, Joao F Ferreira, Thomas Hayton, Julie Porteous, and Peter J Gregory. Framer: Planning Models from Natural Language Action Descriptions. In *ICAPS*, 2017.
- [Maddison *et al.*, 2014] Chris J Maddison, Daniel Tarlow, and Tom Minka. A* sampling. In *NIPS*, pages 3086–3094, 2014.
- [McDermott, 2000] Drew V. McDermott. The 1998 AI Planning Systems Competition. *AI Magazine*, 21(2):35–55, 2000.
- [Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-Level Control through Deep Reinforcement Learning. *Nature*, 518(7540):529–533, 2015.
- [Mourão *et al.*, 2012] Kira Mourão, Luke S. Zettlemoyer, Ronald P. A. Petrick, and Mark Steedman. Learning STRIPS Operators from Noisy and Incomplete Observations. In *UAI*, pages 614–623, 2012.
- [Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. In *NIPS*, pages 91–99, 2015.
- [Richter *et al.*, 2008] Silvia Richter, Malte Helmert, and Matthias Westphal. Landmarks Revisited. In *AAAI*, 2008.
- [Satzger and Kramer, 2013] Benjamin Satzger and Oliver Kramer. Goal Distance Estimation for Automated Planning using Neural Networks and Support Vector Machines. *Natural Computing*, 12(1):87–100, 2013.
- [Sievers *et al.*, 2012] Silvan Sievers, Manuela Ortlieb, and Malte Helmert. Efficient Implementation of Pattern Database Heuristics for Classical Planning. In *SOCS*, 2012.
- [Silver *et al.*, 2016] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature*, 529(7587):484–489, 2016.
- [Srivastava *et al.*, 2015] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised Learning of Video Representations using LSTMs. In *ICML*, pages 843–852, 2015.
- [Steels, 2008] Luc Steels. The Symbol Grounding Problem has been Solved. So What's Next? In Manuel de Vega, Arthur Glenberg, and Arthur Graesser, editors, *Symbols and Embodiment*. Oxford University

Press, 2008.

[Vincent *et al.*, 2008] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and Composing Robust Features with Denoising Autoencoders. In *ICML*, pages 1096–1103. ACM, 2008.

[Yang *et al.*, 2007] Qiang Yang, Kangheng Wu, and Yunfei Jiang. Learning Action Models from Plan Examples using Weighted MAX-SAT. *Artificial Intelligence*, 171(2-3):107–143, 2007.

東京大学情報基盤センター・スーパーコンピューティングニュース
Vol. 20 No. Special Issue 1 (2018. 2)

スーパーコンピューティングニュース編集スタッフ

編集長 中島研吾

編集幹事 古田智嗣

編集スタッフ 田浦健次郎, 佐藤周行, 埴敏博, 下川辺隆史, 近藤正章, 星野哲也,
三木洋平, 伊田明弘, 河合直聡, 有間英志, 宮寄洋, 田川善教,
佐島浩之, 山本和男, 坂井朱美, 安部達巳, 中張遼太郎, 阿曾義浩,
金子伸一

編集・発行 東京大学情報基盤センター

スーパーコンピューティング部門

〒113-8658 東京都文京区弥生 2-11-16

(電話) 03-5841-2717 (ダイヤルイン)

(FAX) 03-5841-2708