

平成 29 年 5 月 17 日現在

機関番号：12601

研究種目：若手研究(B)

研究期間：2015～2016

課題番号：15K20965

研究課題名(和文)大規模ベイジアンネットワーク構造探索並列アルゴリズムの研究

研究課題名(英文) Research on parallel algorithms for large scale Bayesian network structure search

研究代表者

玉田 嘉紀 (Tamada, Yoshinori)

東京大学・医科学研究所・特任講師

研究者番号：80435495

交付決定額(研究期間全体)：(直接経費) 1,800,000円

研究成果の概要(和文)：2次元トラスネットワークを持つスーパーコンピュータ向けの最適DAG探索アルゴリズムを研究・開発し、FX10 4800計算ノードを用いて36変数200サンプルの DAG 探索(ベイジアンネットワーク)に11時間38分で計算させることに成功した。36変数の最適DAG探索が可能であることを示した世界初の成果である。その後、本アルゴリズムを改良し京コンピュータ20768計算ノードを用いて37変数の最適DAG探索に成功し現在論文準備中である。

研究成果の概要(英文)：We have successfully developed the novel parallel algorithm for searching for the optimal DAG structure using the supercomputer with 2D torus interconnect network. Using this algorithm we have succeeded in searching for the optimal DAG structure of the 36 variables and 200 samples using FX10 4800 computation nodes. This is the first example of searching for the optimal DAG structure with 36 variables in the literature. We have then improved this algorithm and succeeded in the optimal 37 variable DAG search using K computer 20768 computation nodes. Currently we are preparing the submission of the paper of this result.

研究分野：バイオインフォマティクス

キーワード：ベイジアンネットワーク グラフ探索アルゴリズム 並列アルゴリズム

### 1. 研究開始当初の背景

非循環有向グラフ (DAG) とは、閉路のない有向グラフで、多変量確率構造を表すベイジアンネットワークと対応づけることができる (図 1)。すなわち多変量のデータが得られた際にその確率構造を DAG で表現することができ、データに最適な確率構造の推定を DAG 探索とみなすことができることを意味する。確率構造の探索は事後確率の最大化に基づいて行う。この事後確率を DAG に対するスコアとみなせば、ベイジアンネットワークの推定とは、最良なスコアとなる DAG の探索問題とみなせる。ベイジアンネットワーク推定の応用の一例として遺伝子ネットワーク推定がある。遺伝子の発現量を確率変数とみなし、遺伝子の発現量を計測した遺伝子発現データから、遺伝子間の発現の確率構造を推測し、それを解析することにより遺伝子間の発現制御の関係や薬剤への細胞システムの応答を解析することが可能である。

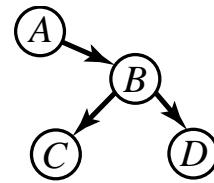
一般にヒトの遺伝子は 2 万程度存在するため、これらすべての遺伝子を含む遺伝子ネットワークが得られることが望ましいが、計測データを最もよく説明できるネットワーク構造 (= 確率構造) を探索すること、つまり最適な DAG を探索することは計算量的に非常に難しい問題として知られている。最適 DAG 探索は解候補となる DAG の中からスコア上で最適なものを選べばよいが、解候補となる DAG の数が膨大なため、それらを列挙しその中から最適なものを探す必要がある。

動的計画法を用いた効率のよいアルゴリズムが発見され、それを用いることにより数十変数程度のベイジアンネットワークの推定が可能になった (Ott et al. 2004)。しかし上で述べたようにヒト遺伝子の数に比べるとはるかに小規模で実用的とは言い難い。したがって、より高速で大規模なベイジアンネットワークが推定可能なアルゴリズムが求められている。

計算時間のかかる問題を解く方法の一つに高速な計算機を使う方法が挙げられる。すなわちスーパーコンピュータを利用して、膨大な解候補を高速に探索すればよい。しかしながら動的計画法によるアルゴリズムの並列アルゴリズムは自明ではなく、スーパーコンピュータがあるだけでは高速に問題が解くことができるわけではない。大規模な DAG 探索問題を解くには、効率良く大規模な並列計算が可能な DAG 探索のための並列アルゴリズムが必要である。

### 2. 研究の目的

本研究では、スーパーコンピュータを用いることにより、より大規模な最適な遺伝子ネットワーク探索、すなわちベイジアンネットワーク推定、あるいは最適 DAG 探索を実現することにある。これまで著者らは 256 並列で 32 変数の DAG 探索が可能なアルゴリズムを考案していたが (Tamada et al. 2011)、現



$$\begin{aligned} \Pr(A, B, C, D) = \\ \Pr(A)\Pr(B|A) \\ \Pr(C|B)\Pr(D|B) \end{aligned}$$

図 1. DAG と対応する確率構造の例

代の最新スーパーコンピュータの並列度は百万を超えるレベルになっており、これまでの著者のアルゴリズムでは最新スーパーコンピュータの性能を十分に生かすことができない。Nikolova ら (2011) は 2048 並列まで可能なアルゴリズムを示し、33 変数の DAG 探索を 1 時間程度で計算可能なことを実証している。これはそれまでの最大規模の DAG 構造探索であるが、実証した並列度は 2048 並列までであり、著者の既存アルゴリズムと同様、現代のスーパーコンピュータの並列度と比較すると十分な並列性能とはいえない。

著者らの既存アルゴリズムはスーパーコンピュータの計算ノード全体で行う通信が頻繁に必要なことから、その並列性能に限界が生じていた。そこで著者らは京コンピュータの持つトラス構造の内部通信ネットワークの構造に着目し、この内部通信ネットワーク上で高速に通信可能な計算可能方式を用いることによって、より大規模な並列化が可能になるのではないかと考えた。すなわち、トラス構造を用いた京コンピュータでは隣接通信が非常に高速に行える構造となっている。また局所的な隣接通信は全体の通信ネットワークに影響を与えない。したがって、この隣接通信だけを行うアルゴリズムを用いることによって DAG 探索が可能になれば非常に大規模に並列化が可能になり、最適 DAG 探索もより大規模なネットワークに対して行うことが可能になると考えた。

隣接通信だけによる最適 DAG 探索の基本的な考えは以下の通りである。内部通信ネットワークを 1 次元のトラスネットワーク、すなわちリング状のネットワークと仮定する。既存アルゴリズムでは全対全通信によって必要なデータを必要な計算ノードに一斉に送信していた。それをバケツリレー式に隣接通信のみによって計算ノードにあるデータを丸ごと、データの必要性の有無にかかわらず次々に送る。1 つの計算ノードにあるデータを両方向の隣接計算ノードに同時に送信すれば計算ノード総数  $N$  に対し  $\lfloor N/2 \rfloor$  回の通信ステップで全体のデータの交換が完了する。しかしこのままではすべての計算ノ

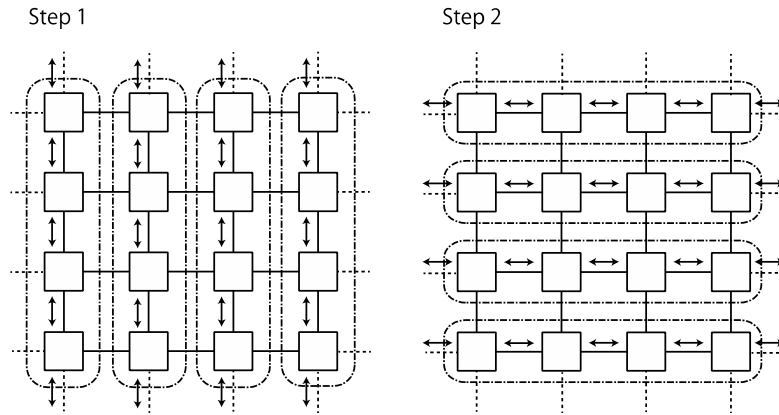


図2．2次元トラスネットワークを利用した2フェーズによるデータ交換方式。Step 1 は縦方向のみの通信を行い、データを交換する。同一列の計算ノードはこのフェーズによりすべて同じデータを共有する。Step 2 で、横方向の計算ノードと通信を行い、共有したデータをバケツリレー式に配送・交換する。

ドが同期的に動作し、並列化効率が大規模な並列数では悪化することが容易に予想される。したがって、これを、2次元トラスネットワークを仮定し、以下の様に2フェーズの通信方式にする(図2)。

まず最初のフェーズでは縦方向の計算ノードとのみ通信を行い、同一列で同じデータを共有する。次に、横方向のみに通信を行い、共有したデータを各計算ノードに送信する。この方式では計算ノード数を  $N = \sqrt{N} \times \sqrt{N}$  と仮定すると、それぞれ  $\lfloor \sqrt{N}/2 \rfloor$  回の通信で計算が終了し、また同期的に動作する計算ノード数もわずか  $\sqrt{N}$  個となる。したがって高並列時においても効率的な動作が期待できる。また通信中の待ち時間を利用して受信したデータを用いた計算を行う通信遅延隠蔽を行い、通信時間による効率の悪化を防ぐ。

本研究の目的はこの2次元トラスを用いた隣接通信による最適 DAG 探索アルゴリズムの実証が目的である。

### 3. 研究の方法

本研究は東京大学情報基盤センターの京互換スーパーコンピュータ FX10 を用いた。FX10 は16コア1.8GH駆動の富士通製 SPARC64 IXfx を計算ノードあたり1基搭載したスーパーコンピュータで計4800ノードが導入されている。搭載メモリは計算ノードあたり32GiBである。計算ノードは6次元Tofuネットワークによって接続され、隣接ノード間の通信速度は5.0GB/sである。この6次元Tofuネットワークからユーザビュー3次元のトラスネットワークがユーザからは利用可能である。一般利用枠では全4800ノードのうち1440ノードが利用可能であった。アルゴリズムの実装はC言語を用いており、並列化はMPI及びOpenMPを用いた。

まずFX10を利用して1次元トラスネットワーク、すなわちリング状ネットワークのアルゴリズムを実装し、バケツリレー式の通信により最適 DAG 探索が可能であることを確認し

た。次に、2次元トラスネットワーク用に実装したアルゴリズムの拡張を行い、既存アルゴリズム、1次元アルゴリズム、2次元アルゴリズムそれぞれで性能を測定し、評価を行った。初期の構想では必要であればさらに3次元へと拡張し、さらなる高効率実行を構想していたが、それは行わなかった。

### 4. 研究成果

表1に28変数200サンプルでの既存アルゴリズム、1次元アルゴリズム、2次元アルゴリズムの結果を示す。81ノード(1296コア)利用時、既存アルゴリズムが2922.8秒かかっていた計算が1次元アルゴリズムでは1450.6秒になり、隣接通信による高速化が実証できた。一方、並列化効率に目を移すと既存アルゴリズムが81ノード利用時と1296ノード(20,736コア)利用時の比較、すなわち16倍の計算ノード数に対し、2.1倍程度(効率0.13)の高速化と非常に効率が悪いのに対し、1次元アルゴリズムは7.2倍(効率0.45)と大幅に改善しているものの理想値である16倍(効率1.0)との比較では良いとは言えない。324ノード利用時、1次元アルゴリズムの並列化効率は0.85となっていることを考えると、予想された大規模並列化による1次元アルゴリズムの非効率性が確認できたと言える。

一方、2次元アルゴリズムでは81ノード利用時1169.0秒と比較的低並列においても1次元アルゴリズムからさらに高速化することがわかり、さらに1296ノード利用時には並列化効率が0.81と大幅に改善されることがわかった。既存アルゴリズムで1387.2秒かかっていたものが、同一の計算リソースを用いて2次元アルゴリズムでは90.7秒で終了しており、15倍という大幅な高速化を達成することができた。以上より従来のアルゴリズムによる2048並列と比較して20,736並列という大幅な高効率大規模並列を実証できた。

表 1 . 既存アルゴリズム、1次元アルゴリズム、2次元アルゴリズムの比較。

previous method ( $n = 28, m = 200$ )					
No. nodes	81	162	324	648	1296
	2922.8	2498.8	1917.2	1426.6	1387.2
1D algorithm ( $n = 28, m = 200$ )					
No. nodes	81	162	324	648	1296
	1450.6	760.3	428.6	298.4	201.2
2D algorithm ( $n = 28, m = 200$ )					
	width	9	18	36	
height	9	1169.0	591.8	305.5	
	18	595.6	303.8	161.4	
	36	308.5	161.5	90.7	

高効率化による高速化が実証できた一方で、2次元アルゴリズムの問題点も明らかになった。2次元アルゴリズムでは縦通信フェーズにおいてすべての同一列計算ノードが同一データを共有する。そのため、共有データを保存するために必要なメモリが増えてしまう。大規模な計算ではメモリ使用量がボトルネックとなってしまう計算がそもそも実行できないことがわかった。そのため、メモリが不足する場合は、通信を複数回に分けてメモリの使用量を抑える方式を実装した。すなわち、縦・横の通信フェーズを複数回反復する。これによりメモリ不足の際にも実行が可能になった。ただし、反復により実行時間の増加が予想される。表 2 は 26 変数  $9 \times 9 = 81$  ノードで計算を行った際の、使用メモリを制限した場合の反復回数である。1.6GB では反復回数は 1 回と追加の反復は必要ない状況からメモリ使用の制約を与えた場合、反復回数の増加に伴って計算時間が大幅に上昇することがわかった。以上のことから 2次元アルゴリズムより効率的な 3次元トラスを用いたアルゴリズムは以上のことからメモリ使用量がさらに増加し、現在の並列度の範囲では効果が少ないと判断し実装は行わなかった。

東京大学情報基盤センターでは FX10 全 4800 ノードを 24 時間独占的に利用可能な大規模 HPC チャレンジという利用枠を年に 4 回設定しており、著者は審査の結果利用を許可された。4800 ノード 76,800 コアを用いて 36 変数 200 サンプルの計算に挑戦したところ 11 時間 38 分で計算を無事終了することができた。36 変数の最適 DAG 探索の実証は世界初であり文献上探索された最大の問題サイズである。

現在、1次元アルゴリズムをベースとしたメモリ使用量を低減する新しいアルゴリズムの研究を進めており良好な結果が得られている。京コンピュータ 20,763 計算ノードを用いて 37 変数の計算に成功している。現在この成果について論文執筆中である。アルゴリズム自体の研究では離散データ向けのスコア関数を用いていたが、遺伝子ネットワークへ適用可能なノンパラメトリック回帰

表 2 . メモリ制限による計算速度への影響

	100MB	200	400	800	1600
$a = 1 \sim 8$	1	1	1	1	1
9	3	2	1	1	1
10	5	3	2	2	1
11	8	4	2	2	1
12	11	6	3	3	1
13	13	7	4	4	1
14	14	7	4	4	1
15	13	7	4	4	1
16	10	5	3	3	1
17	7	4	2	2	1
18	4	2	1	1	1
19	2	1	1	1	1
20 ~ 26	1	1	1	1	1
Exec. time (sec)	802.86	555.80	438.37	344.17	256.43

モデルを用いたスコアを組み込み、京交換機向け及び東京大学医科学研究所ヒトゲノム解析センターのスーパーコンピュータ向けに公開している。

(謝辞)

本研究の実装及び計算実験は主として本多弘和氏によるものであり、この場をお借りして感謝いたします。

## 5 . 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文](計 1 件)

Honda, H., Tamada, Y., Suda, R., Efficient Parallel Algorithm for Optimal DAG Structure Search on Parallel Computer with Torus Network, In Proceedings of the 16th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP 2016), Lecture Notes in Computer Science 10048, 483-502, 2016.

[その他]

ホームページ等

<http://sign.hgc.jp/signbn/>

## 6 . 研究組織

### (1) 研究代表者

玉田 嘉紀 (TAMADA, Yoshinori)

東京大学・医科学研究所・特任講師

研究者番号：80435495