A-009

ベイジアンネットワークの確率計算のための ZDD 分解法に関する実験と考察

Experiments and Considerations on ZDD Decomposition for Representing Bayesian Networks

高　サン　　湊　真一
Shan Gao　　Shi-ichi Minato
北海道大学　大学院　情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

1　まえがき

ベイジアンネットワークの確率分布を計算するための方法の一つとして，Multi-Linear Function（MLF）と呼ばれる数式を生成する方法が知られている．MLF 式に基づく確率計算に要する時間は，MLF 式の長さに比例するが，一般に MLF 式は元のベイジアンネットワークのサイズに対して指数関数的に大きくなるため，その計算は容易ではない．湊らは，Zero-Suppressed BDD（ZDD）を用いて，ベイジアンネットワークを表現し，効率よく確率計算を行う手法を提案した[1]．この手法は，場合によっては従来手法よりもずっと高速に確率計算を行う事が出来る．本稿では，この高速計算法を更に改善するために，ZDD を因数分解して，サイズを削減する方法及び d-分離に基づく分解法を検討し，実験と考察を行う．

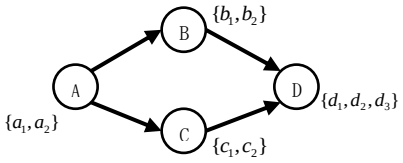


図 1　BN の例

2　ベイジアンネットワークを表現する ZDD

ベイジアンネットワーク（以下 BN と呼ぶ）は図 1 に示すような DAG（非巡回有向グラフ）による確率モデルの表現方法の 1 つである．各ノード（以下 BN ノードと呼ぶ）は，それぞれ独立した個別の確率変数 X を持っている．また各 BN ノードは，上流側の BN ノードの確率変数の値に依存する条件付き確率テーブル（Conditional Probability Table; CPT）を持ち，確率変数の確率分布が表現されている．BN の確率分布を計算するための方法の 1 つとして，Multi-Linear Function（MLF）と呼ぶ数式を構成する方法が知られている．MLF 式は 1 つの確率変数 x に対して，X の確率分布の数値を記号的に表現する λ 変数と，X の値を表現する θ 変数の 2 種類の変数を使って表現する．以下に MLF の例を示す．

$$\begin{aligned}
&\lambda_{a_1}\lambda_{b_1}\lambda_{c_1}\lambda_{d_1}\theta_{a_1}\theta_{b_1|a_1}\theta_{c_1|a_1}\theta_{d_1|b_1c_1}\\
&+\lambda_{a_1}\lambda_{b_1}\lambda_{c_1}\lambda_{d_2}\theta_{a_1}\theta_{b_1|a_1}\theta_{c_1|a_1}\theta_{d_2|b_1c_1}\\
&+\lambda_{a_1}\lambda_{b_1}\lambda_{c_1}\lambda_{d_3}\theta_{a_1}\theta_{b_1|a_1}\theta_{c_1|a_1}\theta_{d_3|b_1c_1}\\
&\cdots\\
&+\lambda_{a_2}\lambda_{b_2}\lambda_{c_2}\lambda_{d_3}\theta_{a_2}\theta_{b_2|a_2}\theta_{c_2|a_2}\theta_{d_3|b_2c_2}
\end{aligned}$$

与えられた BN の MLF 式を生成することで，ある観測データに対する確率変数の確率分布を機械的に求めることができる．この確率計算に要する時間は，MLF 式の長さに比例するが，一般に MLF 式は元の BN サイズに対して指数的に増大するため，計算するのに非常に多くの時間がかかる．ただし，MLF 式を因数分解して，コンパクトな算術式として表現することで，確率計算を高速化することができ，この因数分解を行う方法の 1 つとして ZDD を用いる手法が湊らにより提案されている[1]．ZDD は多数のアイテム組合せからなる集合データを二分木状のグラフで場合分けして表現し，これを DAG に圧縮したものである．

MLF 式は λ 変数と θ 変数からなる一種の組合せ集合と見なすことができる．よって ZDD によりコンパクトに表現することができる．例えば図 1 のノード B に対する MLF 式は

$$\begin{aligned}
MLF_B = &\lambda_{a_1}\lambda_{b_1}\theta_{a(0.4)}\theta_{b(0.2)} + \lambda_{a_1}\lambda_{b_2}\theta_{a(0.4)}\theta_{b(0.8)}\\
&+\lambda_{a_2}\lambda_{b_1}\theta_{a(0.6)}\theta_{b(0.8)} + \lambda_{a_2}\lambda_{b_2}\theta_{a(0.6)}\theta_{b(0.2)}
\end{aligned}$$

となるが，これに対応する ZDD は，MLF 式を因数分解した多段の算術式と解釈することができ，個々の節点は，下位の節点の計算結果を用いた算術乗算と加算を表している．つまり，MLF 式の ZDD を生成した後に，ZDD のサイズに比例する回数の算術演算により確率計算を実行できるということを意味している．現実の BN に対して ZDD を生成すると，因数分解前の MLF 式の長さに対して 1 万分の 1 以下に圧縮できる場合もあり，顕著な効果がある[1]．しかし，それでも ZDD が大きくなり過ぎる場合もあるので，更にコンパクトな表現が求められている．

3　ZDD 分解法

3.1　ZDD 因数分解

湊らが提案した Fast Weak Division[2]により，分割された ZDD を因数分解する．MLF 式の各項は，同じ変数が何度も出現することが多いので，それらをまとめて，くくり出すことにより，MLF 式のサイズを削減することができる．一般に因数分解の仕方は一意ではなく，よい因数を見つけることが必要である．今回は以前に湊が示した発見的手法[2]を用いた．その方法を以下に示す．

```
Divisor(MLF){
    v ← MLF に 2 回以上を現れる文字;
    if(v が存在)   return Divisor(MLF/v);
    else return MLF;
    }
```

データセット名 (BN 変数 ID)	因数分解前			因数分解後			計算時間
	ZDD 節点数	MLF 式の項数	MLF 式の文字数	ZDD 節点数	MLF 式の項数	MLF 式の文字数	
alarm(36)	4551	1 百億以上	5 千億以上	6784	3500	13512	32m49.767s
d-分離で分割した alarm(36)	2545	127650816	4581992448	2834	1567	5588	0m11.365s
Water(26)	16975	280593	9196679	13847	8546	31533	1m25.444s
carpo(13)	219	1536	33280	164	72	243	0m0.011s

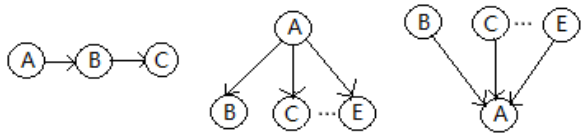
表 1 実験結果

例えば，$F = abd + abe + cd + ce + acd$ の場合，Divisor(MLF)により，F を $abd + abe + cd + ce + acd = (ab + c)(d + e) + acd$ のように変換できる．$p = ab + c$，$q = d + e$ とすると，F は $pq + acd$ になる．F の文字数は 13 から 10 まで削減される．

本研究では，BN の MLF 式における Divisor を見つけて因数分解することにより，ZDD サイズの削減を試みた．

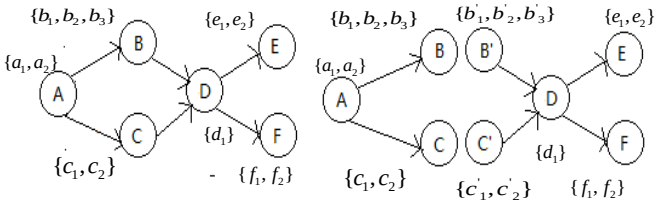
3.2 BN の d-分離による分解

d-分離は BN の確率変数の条件付独立性を決定するために用いられる．d -分離とは，有向グラフにおいて，図 2 に示すように 3 種類の結合の構造として定義することができる[3]．(a)逐次結合：A は B に影響を与え，B は C に影響をもつ．B の状態が分かってしまったら，A と C は独立になってしまう．(b)分岐結合：A の状態が分かってしまったら，A のすべての子は独立になってしまう．(c)合流結合：A もしくはその子孫の状態が分からない限り，B，C，…E は独立と見なせる．d-分離のルールを用いて，BN を幾つかの部分に分けられる．例えば，図 3(a)において，B と C の状態が分かってしまったら，A と D は独立になるので，図 3(b)のように 2 つ部分を分けて，それぞれを ZDD で表現できる．しかし，新しい変数 $\{c'_1, c'_2\}$ と $\{b'_1, b'_2, b'_3\}$ を導入しなければならないので，BN に対する確率推論の計算時間が増加してしまう．そこで本稿では Tarjan のアルゴリズム[4]でグラフの連結成分の数が増える切断点を見つけ，BN を分割する．そのような切断点で BN を分割すれば ZDD サイズが大きく削減される可能性が高い．図 3(a)の場合，B と C の代わりに D を見つける．細かい方法については，まず，BN を無向化する[5]．そして無向グラフに対して Tarjan のアルゴリズムで切断点を探す．適切な切断点が見つけられたらその切断点で BN を分割する．見つけられなかったら分割しない．



(a)逐次結合　(b)分岐結合　(c)合流結合

図 2　d-分離の例



(a)　(b)

図 3　d-分離による分割の例

4　実験の結果と考察

本実験において使用した PC は Intel Core(TM)2 Quad CPU Q9550 @ 2.83GHz ×4，Ubuntu 12.04LTS，主記憶 3.8GiB で ZDD の最大節点数は 1，000 万個としている．

表 1 は BN Benchmark[6]のデータセットの alarm，Water 及び carpo に対する実験結果である．各例題において一番 ZDD サイズが大きくなっている BN 変数を取り出して実験を行った．表 1 に示す通り alarm の例では，適切な d-分離で BN を分割した後で因数分解を行うことにより，ZDD のサイズを更に削減できた．しかし，Water と carpo の例では，適切な d-分離が見つけられなかったが，因数分解で ZDD のサイズがやや小さくなった．alarm の場合は因数分解で ZDD のサイズが増えてしまうが，Divisor(MLF)を改良して，よい Divisor を見つければ ZDD のサイズを削減できる可能性がある．ZDD サイズを考えずに MLF 式の項数と文字数だけを考えると，因数分解により大幅に減少しており，BN を表現する MLF 式を短く表現できていると言える．

参考文献

[1] 湊真一，大規模な離散構造データを扱うための GPU 利用法の検討，電子情報通信学会 2011 ソサイエティ大会, AI-1-5, Sep. 2011.

[2] Minato Shi-ichi, Multi-Level Logic Synthesis Using ZBDDs, Binary Decision Diagrams and Applications for VLSI CAD The Kluwer International Series in engineering and Computer Science Volume 342, pp. 81-94, 1996

[3] 植野真臣，ベイジアンネットワーク，ISBN：978-4-339-06103-1，コロナ社，pp.55-65, 2013

[4] Bridge (graph theory), http:/ /en.wikipedia.org/wiki/Bridge _(graph_theory)

[5] Silvia Acid, Luis M. de Campos, An Algorithm for Finding Munimum d-Separating Sets in Belief Networks, Depto. de Ciencias de la Comutatacion e I.A. Universidad de Granada 18071-Granada, Spain, 1996

[6] Bayesian network Repository, http://www.cs.huji.ac.il/labs/ compbio/Repository