УДК 004.8

А.А. Фильченков, А.Л. Тулупьев

АЛГОРИТМ ВЫЯВЛЕНИЯ АЦИКЛИЧНОСТИ ПЕРВИЧНОЙ СТРУКТУРЫ АЛГЕБРАИЧЕСКОЙ БАЙЕСОВСКОЙ СЕТИ ПО ЕЕ ЧЕТВЕРТИЧНОЙ СТРУКТУРЕ

Фильченков А.А., Тулупьев А.Л. **Алгоритм выявления ацикличности первичной структуры алгебраической байесовской сети по ее четвертичной структуре.**

Аннотация. Алгебраические байесовские сети (АБС) относятся к классу логико-вероятностных графических моделей систем знаний с неопределенностью, которые позволяются использовать интервальные оценки вероятности для представления неопределенности в знаниях. Одним из наиболее важных условий работы АБС является отсутствие циклов в их вторичной структуре. Первичная структура, над которой можно построить ациклическую АБС, называется ациклической. *Цель работы* — предложить алгоритм выявления ацикличности первичной структуры на основе анализа четвертичной структуры АБС, а также оценка сложности этого алгоритма. В работе сформулирован алгоритм выявления ацикличности, доказана его корректность, оценена его сложность и предложен ряд способов, направленных на ускорение работы этого алгоритма.

Ключевые слова: алгебраические байесовские сети, четвертичная структура, вероятностные графические модели систем знаний, глобальная структура, ацикличность первичной структуры.

Filchenkov A.A., Tulupyev A.L. **Algorithm for detection of algebraic Bayesian network primary structure acyclicity based on its quaternary structure.**

Abstract. Algebraic Bayesian networks (ABN) belong to a class of logical and probabilistic graphical models of systems of knowledge with uncertainty. ABN allows to use interval probability estimates to represent uncertainty in knowledge. One of the most important conditions for ABN performance capability is the absence of cycles in its secondary structure. The primary structure, on which an acyclic ABN can be synthesized, is called acyclic primary structure. The goal of the work is to propose an algorithm for detection of the primary structure acyclicity on the basis of analysis of the quaternary structure of the ABN, as well as evaluation of the algorithm complexity. The algorithm for acyclicity detection is formulated, its correctness is proven, its complexity is estimated and a number of improvements for the acceleration of this algorithm are proposed.

Keywords: algebraic Bayesian networks, quaternary structure, machine learning, probabilistic graphical knowledge models, global structure, primary structure acyclicity.

1. Введение. Алгебраические байесовские сети (АБС) относятся к классу логико-вероятностных графических моделей систем знаний с неопределенностью, которые позволяют использовать интервальные оценки вероятности для представления неопределенности в знаниях [3–22]. Одна из актуальных задач, стоящих в теории АБС — это задача обучения АБС. Ее разделяют на две подзадачи: локального обучения, связанного с формированием оценок вероятности для фрагментов знаний, и глобального обучения, которое также подразделяется на две подзадачи. Первая подзадача глобального обучения состоит в формировании набора фрагментов знаний по известным статическим данным, а вторая — в формировании графа смежности над первичной структурой.

Известно [5, 9, 13], что для быстрой и корректной работы алгоритмов АБС граф смежности должен быть ациклическим, а также то, что ациклические графы смежности возможны не над любой первичной структурой. Первичная структура, над которой возможно построить ациклический граф смежности, называется ациклической, и над этой структурой множество ациклически графов смежности совпадает с множеством минимальных графов смежности.

Выявление ацикличности первичной структуры является актуальным по ряду причин, наиболее значимой из которых является требование формировать именно ациклические первичные структуры при обучении первичной структуры (или, иначе говоря, синтезировать на основе исходных данных АБС, представимую ациклическим графом).

Известны алгоритмы построения как всего множества минимальных графов смежности, так и случайного минимального графа смежности [1, 2, 23–27, 30], однако выявление ацикличности первичной структуры через построение минимального графа смежности и поиск циклов в нем является достаточно длительным процессом и может быть улучшено за счет привлечения критериев ацикличности минимального графа смежности, предложенных в работе [29].

Эти критерии основаны на анализе циклов в семействе особых графов, построенных на подграфах максимального графа смежности, которое носит название четвертичной структуры АБС (формальное определение будет дано ниже).

Цель работы — предложить алгоритм выявления ацикличности первичной структуры на основе анализа четвертичной структуры, а также оценка сложности этого алгоритма.

2. Основные определения и обозначения.

Определение 1. *Граф* — пара $\langle V, E \rangle$, где V — множество вершин графа, а E — множество ребер, каждое из которых является неупорядоченной парой (v_i, v_j), $i \neq j, v_i,\ v_j \in V$. Для удобства будем через V и E обозначать функции от графа, возвращающие множество его вершин и множество его ребер соответственно: $V(G') = V'$; $E(G') = E'$, где $G' = \langle V', E' \rangle$. Также V от множества ребер будет возвращать множество концов этих ребер: $V(E) = \{v | \exists e \in E, \exists u: e = (v, u)\}$.

Определение 2. *Алфавит* — множество атомарных пропозициональных формул $A = \{x_1, \dots, x_n\}$.

Определение 3. *Слово* V — подмножество алфавита: $V \subseteq A$.

2.1. Первичная и вторичная структуры АБС. Будем следовать обозначениям, введенным в работах [20, 22, 32].

С точки зрения рассуждения о вторичной структуре АБС (а также всех прочих структур АБС, отличных от первичной, т. е. структур, не хранящих вероятностную информацию), фрагмент знаний представляется намного в более простом виде, чем он определяется для первичной структуры.

Определение 4. *Фрагмент знаний* — слово над заданным алфавитом.

Определение 5. *Набор максимальных фрагментов знаний* (МФЗ) — это набор фрагментов знаний, таких, что ни один фрагмент знаний не содержится ни в каком другом фрагменте знаний. Набор МФЗ — это *первичная структура* АБС. Множество первичных структур АБС будет обозначаться как **PS.**

Определение 6. *Вес* $W(K)$ *МФЗ* K — слово, которое состоит из атомов, вошедших в K.

Определение 7. *Сепаратор* $W(K_i, K_j)$ двух МФЗ K_i и K_j — пересечение весов соответствующих МФЗ:

$$W\left(K_i, K_j\right) = W(K_i) \cap W\left(K_j\right).$$

Определение 8. Два МФЗ называются *сепарированными*, если их сепаратор непуст.

Определение 9. *Граф МФЗ* — граф, построенный над набором МФЗ, такой, что ребра в нем возможны только между сепарированными вершинами. Вес ребра определяется как соответствующий сепаратор.

Определение 10. *Магистральный путь* $M(K_i, K_j)$ между двумя сепарированными МФЗ K_i и K_j в графе МФЗ — это такой путь от K_i до вершины K_j, что вес любой принадлежащей этому пути вершины содержит сепаратор K_i и K_j.

Определение 11. Граф *магистрально связен,* если между каждой парой сепарированных вершин существует магистральный путь.

Определение 12. *Граф смежности* — магистрально связный граф МФЗ. Граф смежности — это *вторичная структура* АБС. Над одной и той же первичной структурой можно построить множество графов смежности.

Определение 13. *Минимальный граф смежности* — граф смежности, число ребер которого минимально. Известно [2], что минимальные графы смежности и только они являются *нередуцируемыми графами смежности*, то есть такими графами смежности, что любые графы, полученные удалением из них любого числа ребер, не будут являться графами смежности. Множество минимальных графов смежности будем обозначать как $\mathbf{MJG}$.

Определение 14. *Максимальный граф смежности* $G_{\max}$ — наибольший по числу ребер граф смежности. Так как в графе МФЗ возможны не все ребра, а только те, которые соединяют вершины, пересечение весов которых непусто, то максимальный граф смежности вовсе необязательно совпадает с полным подграфом. Для заданного множества вершин существует и при этом единственный максимальный граф смежности [30].

2.2. Третичная полиструктура АБС. Будем следовать обозначениям, введенным в работе [31].

Определение 15. Сужение на вес U — подграф максимального графа смежности, веса всех вершин и ребер которого содержат вес U.

Определение 16. *Значимая клика веса* U — сужение на вес U, совпадающий с каким-либо сепаратором максимального графа смежности. Множество всех значимых клик будем обозначать как Clique. Каждая значимая клика является полным графом.

Определение 17. *Праклика* PraC — ненаправленный граф, являющийся сужением на пустой вес (т. е. фактически совпадающий с ним).

Определение 18. *Замкнутое сверху множество значимых клик* — множество значимых клик, к которому добавлена праклика.

Для краткости изложения будем далее под «значимой кликой» понимать элемент замкнутого сверху множества значимых клик, т. е. либо собственно значимую клику, либо праклику.

Определение 19. Вес значимой клики P будем обозначать как $W(P)$. Будем говорить, что значимая клика P *содержит* значимую клику Q, если $W(Q)$ содержит $W(P)$ или, что то же самое, множество вершин значимой клики P содержит множество вершин значимой клики Q. Следует отметить, что любой значимый вес будет являться весом какой-либо значимой клики.

Определение 20. Если значимая клика P содержит значимую клику Q, то значимая клика P называется *предком* значимой клики Q, а значимая клика Q — *потомком* значимой клики P.

Определение 21. Если значимая клика P является предком значимой клики Q, и не существует такой значимой клики R, что R является одновременно потомком P и предком Q, то значимая клика P называется *родителем* клики Q, а значимая клика Q — *сыном* значимой клики P.

Определение 22. *Родительский граф* над подмножеством A множества сужений — направленный граф, вершинами которого являются значимые клики из множества A. В частности, *родительский граф над замкнутым сверху множеством значимых клик* — это родительский граф, построенный над указанным множеством. Ребро из вершины P в вершину Q проведено, если значимая клика P является родителем значимой клики Q. Будем считать, что сыновья значимой клики упорядочены по индексу их веса.

Определение 23. *Третичной полиструктурой* АБС называется семейство направленных графов, построенных над некоторыми подмножествами множества сужений. В частности элементом третичной полиструктуры является и родительский граф над замкнутым сверху множеством значимых клик.

2.3. Четвертичная структура АБС. Будем следовать обозначениям, введенным в работах [29].

Определение 24. *Сильное сужение веса* U — значимая клика веса U, из которой удалили все ребра веса U.

Сильное сужение представляет собой компоненты связности, на которые разбивается соответствующая значимая клика.

Определение 25. Компоненты связности сильного сужения называются *владениями.*

Определение 26. *Доменной вершиной* называется вершина, принадлежащая значимой клике U и не принадлежащая ни одному из ее сыновей.

Определение 27. *Вассалом* V_U^i называется множество вершин, совпадающих с множеством вершин какого-либо сына значимой клики U.

Определение 28. Два вассала V_U^i и V_U^j значимой клики U называются *братьями*, если их пересечение непусто.

Определение 29. *Полусиблинговый путь* между двумя вассалами V_U^i и V_U^j значимой клики U — такая последовательность вассалов значимой клики U без самопересечений, первым элементов которой является V_U^i, а последним — V_U^j, причем каждые два соседних вассала этой последовательности являются братьями.

Определение 30. Если полусиблинговый путь между V_U^i и V_U^j существует, то они называются *полусиблингами* для значимой клики U.

Определение 31. *Братство* значимой клики U — непустой набор вассалов этой клики, такой, что с каждым вассалом в братство входят все его полусиблинги и только они.

Теорема 1 (о классификации владений) [33]. Любое владение любой значимой клики является либо ее доменной вершиной, либо ее вассалом, либо ее братством.

Введем понятие полусиблингового графа.

Определение 32. *Полусиблинговый граф* HSG_U для значимой клики веса U — это граф, построенный над множеством сыновей этой значимой клики, ребро между двумя вершинами которого проведено тогда и только тогда, когда соответствующие сыновья пересекаются по их вершинам (МФЗ):

$$\mathrm{HSG}_U = \langle \{S_i | S_i \in \mathrm{Sons}(C_U)\}, \{(S_i, S_j) | S_i \cap S_j\} \rangle.$$

Замечание 1. Вершинами компонент связности полусиблингового графа являются либо вассалы, либо братства соответствующей клик.

Определение 33. *Четвертичная структура* АБС — семейство полусиблинговых графов, задаваемых каждой значимой кликой.

Определение 34. *Полусиблинговый цикл* — цикл полусиблингового графа.

Определение 35. *Братский цикл* — полусиблинговый цикл $S_1, S_2, \dots, S_n$, такой, что пересечение всех этих вассалов непусто.

Определение 36. *Небратский цикл* — полусиблинговый цикл, не являющийся братским.

Замечание 2. В братском цикле все вассалы, в него входящие, приходятся друг другу братьями, тогда как в небратском цикле это условие может как выполняться, так и не выполняться.

Теорема 2 (о циклах минимальных графов смежности) [30, 32]. Любой минимальный граф смежности, построенный над заданным набором вершин, является ациклическим тогда и только тогда, когда не существует полусиблинговых небратских циклов.

Следствие 1. Все графы смежности, построенные над данным набором вершин, являются циклическими или ациклическим одновременно.

2.4. Соглашения об оформлении алгоритмов. Будем считать, что на вход алгоритмов поступает неупорядоченное множество МЗФ $\mathrm{Weights}$, элементы которого представляются битовыми последовательностями фиксированной длины. Последнее гарантирует нам, что объединение и пересечение двух весов, а также проверка включения одного веса в другой выполняются за $O(1)$.

В записи алгоритмов

- запись «$a \leftarrow b$» обозначает присвоение значения элемента b элементу a;
- запись «$S \rightarrow e\colon \mathrm{Cond}(e)$» обозначает извлечение произвольного элемента, удовлетворяющего условиям Cond, из множества S в переменную e. $\mathrm{Cond}(e)$ может отсутствовать, тогда речь идет об извлечении произвольного элемента множества S.

CYCLES — функция, которая по заданной матрице инцидентности графа возвращает множество его циклов (без учета порядка вершин). Будем считать, что CYCLES работает за $O(n^2)$, где n — размер матрицы инцидентности.

PARENTTREEOVERUPEXTENDEDUSEFULCLIQUES — функция, которая по заданному набору МФЗ строит родительский граф над расширенным вверх множеством значимых клик. При этом для каждого узла построенного дерева определены следующие функции:

SONS — возвращает множество сыновей соответствующего элемента (т. е. значимой клики или праклики);

VERTICES — возвращает множество вершин, принадлежащих соответствующему элементу (т.е. значимой клике или праклике). При этом подобное множество будет представляться индексом бинарной последовательности, которая характеризует вхождение или не вхождение соответствующих вершин в набор, что позволит осуществлять операцию пересечения, используя побитовые операторы за константу.

Алгоритмы построения родительского графа над множеством значимых клик (не расширенным) предложены в работе [28]. Сложность этих алгоритмов в значительной степени определяется структурой сети (а именно численными показателями, представленными в соответствующих формулах). Оба алгоритма работают не больше, чем за $O(|V|^5)$, где V — первичная структура, по которой строится родительский граф, однако она не достигается даже в худшем случае, а в среднем алгоритм работает значительно быстрее (уточнение оценки сложности алгоритма на сегодняшний день является открытым вопросом).

Алгоритм построения интересующего нас родительского графа отличается от рассматриваемого только добавлением в родительский граф над множеством значимых клик вершины, которая будет соединена со всеми теми вершинами исходного родительского графа, у которых нет родителей. Сложность этого действия также зависит от реализации алгоритма, однако, по всей видимости, пренебрежимо мала.

Исходя из изложенных выше соображений, мы не будем явно выражать сложность алгоритма построения родительского графа над расширенным вверх множеством значимых клик; вместо этого введем функцию $\mathrm{PTOUEUC}(V)$, где V — первичная структура АБС, обозначающая указанную сложность.

3. Алгоритм выявления ацикличности первичной структуры АБС по ее четвертичной структуре. Алгоритм выявления ацикличности первичной структуры по ее четвертичной структуре для заданного набора МФЗ определяет, можно ли над данной первичной структурой построить ациклический граф смежности, и если можно, то возвращает значение TRUE, иначе возвращает значение FALSE.

```
Require: V
Ensure: V ∈ PS
ParentTree ← ParentTreeOverUpExtendedUsefulCliques(V)
hasCycles ← FALSE
for all c ∈ ParentTree do
  for all s_i, s_j ∈ Sons(c) do
    if Vertices(s_i) ∩ Vertices(s_j) ≠ ∅
      HSGIncM[i, j] ← 1
    else
      HSGIncM[i, j] ← 0
    end if
  end for
  Cycles ← Cycles(HSGIncM)
  if Cycles ≠ ∅
    for all l ∈ Cycles
      l → f
      ComVerts ← Vertices(f)
      hasLocalCycles ← FALSE
      for all g ∈ l, g ≠ f
        ComVerts ← ComVerts ∩ Vertices(g)
        if ComVerts = ∅
          hasLocalCycles ← TRUE
          break
        end if
      end for
```

```
24:         if hasLocalCycles = TRUE
25:           hasCycles ← TRUE
26:           break
27:         end if
28:       end for
29:       if hasCycles = TRUE
30:         break
31:       end if
32:     end if
32: end for
33: return hasCycles
```

Листинг 1. Алгоритм выявления цикличности первичной структуры АБС по ее четвертичной структуре.

На шаге (1) строится родительский граф над замкнутым сверху множеством значимых клик.

В цикле (3–32) перебираются все значимые клики из множества значимых клик, а также праклика, и для каждого рассматриваемого объекта выясняется, существуют ли небратские циклы в соответствующем полусиблинговом графе.

В цикле (4–10) путем перебора всех пар несовпадающих сыновей рассматриваемой значимой клики (или праклики) строится матрица инцидентности соответствующего полусиблингового графа HSGIncM.

На шаге (11) строятся все циклы рассматриваемого полусиблингового графа.

В случае если множество таких циклов непусто (проверка на шаге (12)), то в цикле (13–28) перебираются все полусиблинговые циклы.

На шаге (14) выбирается произвольная вершина (вассал) рассматриваемого полусиблингового цикла и в переменную ComVerts сохраняется множество вершин, входящих в соответствующего вассала. Далее в цикле (17–23) последовательно перебираются все прочие вершины рассматриваемого полусиблингового цикла (вассалы) и в ComVerts остаются только те вершины, которые входят также и в этот вассал (шаг (18)). Таким образом, после окончания цикла (17–23) ComVerts будет хранить вершины, входящие во всех вассалов рассматриваемого полусиблингового цикла. Если в какой-то момент времени ComVerts оказывается пусто (проверка на шаге (19)), то это означает, что цикл является небратским (шаг 25), потому что если не существует вершин, входящих в каждого вассала из некого подмножества вассалов полусиблингового цикла, то и не существует вершин, входящих в каждого вассала полусиблингового цикла. Если существование небратского цикла обнаружилось, то можно не продолжать поиски как среди небратских циклов для данной значимой клики (или праклики) (оператор на шаге (16)), так для других значимых клик (условный оператор (29–31)).

Утверждение 1. Алгоритм выявления цикличности первичной структуры АБС по ее четвертичной структуре для ациклической первичной структуры вернет значение FALSE, а для циклической — значение TRUE.

Доказательство. Будем доказывать, что алгоритм вернет значение FALSE тогда и только тогда, когда не существует небратских полусиблинговых циклов, что эквивалентно доказываемому утверждению согласно теореме 2.

Прежде всего, заметим, что алгоритм вернет значение FALSE в том и только том случае, если не будет достигнут шаг (25). В силу условия (24) это эквивалентно тому, что не будет достигнут шаг (20). Поскольку операторы прерывания (21, 26 и 30) срабатывают только после достижения указанных шагов, то из этого следует, что если алгоритм вернет FALSE, то переберет все элементы циклов.

Докажем, что если алгоритм вернул FALSE, то ни для какой клики и для праклики не существует небратских полусиблинговых циклов. Как уже было отмечено, в этом случае алгоритм переберет все значимые клики и праклику и для каждой построит циклы, которые также переберет. Корректность построения циклов по матрице инцидентности следует из корректности метода CYCLES, корректность построения матрицы инцидентности следует из построения (шаги (5)–(8)).

Таким образом, будут рассмотрены все полусиблинговые циклы четвертичной структуры. Для каждого такого цикла будет рассмотрено множество ComVerts, которое по построению (шаги (15) и (18)) содержит множество вершин, входящих в каждый из вассалов, которые входят в динамически расширяемое множество путем добавления к множеству, состоящему из произвольного вассала, входящего в цикл, вассалов, которые входят в этот цикл, но еще не содержатся в этом множестве. Иначе говоря, ComVerts строится последовательным пересечением вершин очередного вассала и общих вершин для вассалов, рассмотренных до этого. После перебора всех вассалов, входящих в рассматриваемый полусиблинговый цикл, ComVerts будет содержать вершины, входящие в каждый из этих вассалов.

Шаг (20) достигается только тогда, когда ComVerts оказалось пусто для какого-то цикла, что, как было замечено выше, эквивалентно тому, что цикл является не братским. Следовательно, если он не был достигнут, то не существует небратских циклов, а значит первичная структура ациклична.

Теперь докажем, что если первичная структура ациклична, то алгоритмом будет возвращено значение TRUE. Пускай для некоторой значимой клики (или праклики) веса U в соответствующем ей полусиблинговом графе существует цикл, который не является братским. Либо эта клика в какой-то момент будет выбрана на шаге (3), либо цикл прервется раньше, что, как было показано выше, возможно только тогда, когда будет достигнут оператор прерывания и возвращено значение TRUE. Когда соответствующая клика будет выбрана, для нее будет построена матрица инцидентности (шаги (5–8)), которая будет содержать циклы (по теореме 2).

При переборе (шаг 13) полусиблинговых циклов рассматриваемый небратский цикл либо будет достигнут, либо соответствующий цикл алгоритма будет прерван ранее, что эквивалентно тому, что алгоритм возвратит TRUE.

Поскольку не существует вершин, входящих в каждого вассала этого цикла, то при одном из выполнений шага (18) ComVerts окажется пустым. Это приведет к выполнению шага (20), что, как было сказано ранее, эквивалентно тому, что алгоритм возвратит TRUE.

Из этого следует, что алгоритм возвратит значение FALSE тогда и только тогда, когда первичная структура ациклична.

Утверждение 2. Сложность алгоритма выявления ацикличности первичной структуры АБС по ее четвертичной структуре в худшем случае равна

$$O\left(\mathrm{PTOUEUC}(V) + \sum\nolimits_{u \in U^{\uparrow}} |S_u|^2 + \sum\nolimits_{l \in L} |l|\right),$$

где $U^{\uparrow}$ — множество значимых весов, объединенное с пустым весом, S_u — множество сыновей значимой клики веса u; L — множество полусиблинговых циклов, $|l|$ — число вершин в полусиблинговом ле l.

Доказательство. Сложность выполнения шагов (3–10) равна

$$O\left(\sum\nolimits_{u \in U^{\uparrow}} |S_u|^2\right) \quad (1)$$

поскольку будет рассмотрено пересечение каждой пары сыновей каждой значимой клики. Поскольку пересечение сыновей будет выполняться за константу, как и присвоение на шаге (6) или (8), то сложность будет равна сумме мощностей множеств сыновей значимых клик, возведенных в квадрат.

Такая же сложность будет и на шаге (11), поскольку для каждой значимой клик выявляются циклы в графе, построенном над $|S_u|$ вершинами.

Сложность выполнения шагов (12–32) в худшем случае (когда ни один цикл не прервется), будет равна

$$O\left(\sum_{l \in L} |l|\right), \tag{2}$$

потому что для каждого полусиблингового цикла будут перебираться все его вершины и будет строиться пересечение этих вершин (т. е. вассалов) за константу.

Сложив выражения (1)–(2) и добавив сложность построения родительского графа над расширенным вверх множеством значимых клик, получим искомое выражение.

4. Возможности ускорения алгоритма. Алгоритм выявления ацикличности первичной структуры АБС по ее четвертичной структуре может быть ускорен благодаря применению нескольких улучшений.

1. *Перебор только простых циклов.* Можно потребовать перебора не всех циклов, а только пустых (т. е. циклов без внутренних ребер). Очевидно, что таким образом в среднем требуемое время поиска уменьшится. Однако в худшем случае оба алгоритма будут работать одинаково.

2. *Построение родительского графа одновременно с выявлением ацикличности.* При использовании любого из алгоритмов построения родительского графа над замкнутым сверху множеством значимых клик, можно искать небратские полусиблинговые циклы сразу, как только будет построено соответствующее сужение. Это позволит в случае существования небратских полусиблинговых циклов затратить меньшее время на их выявление, чем в исходном алгоритме, однако в худшем случае оба алгоритма будут работать одинаково.

5. Заключение. В работе предложен алгоритм выявления цикличности первичной структуры АБС на основе анализа ее четвертичной структуры. Этот алгоритм позволяет отвечать на вопрос «Возможно ли построить над данной первичной структурой ациклический граф смежности» без непосредственного построения графов смежности.

Сложность работы алгоритма во многом зависит от способа представления набора МФЗ, а также от сложности работы алгоритма построения родительского графа (над произвольным множеством). Следовательно, одним из способов ускорения выявления ацикличности первичной структуры является ускорение работы указанного алгоритма.

В работе также предложены два варианта ускорения предложенного алгоритма, связанные с определенными эмпириками. Однако это ускорение осуществляется лишь в среднем, в худшем же случае (когда не существует небратских полусиблинговых циклов, и, следовательно, необходимо перебрать все полусиблинговые циклы для всех значимых клик), алгоритмы работают одинаково.

Важно отметить, что выявление ацикличности может осуществляться также и при использовании других структур. Один из способов опирается на подсчет числа ребер в минимальном графе смежности [2]. Другим предполагаемым способом может являться анализ дополнительной структуры, которая представляется набором гиперграфов над атомами алфавита.

Литература

1. *Опарин В.В., Тулупьев А.Л.* Синтез графа смежности с минимальным числом ребер: формализация алгоритма и анализ его корректности // Труды СПИИРАН. СПб.: Наука, 2009. Вып. 11. С. 142–157.
2. *Опарин В.В., Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Матроидное представление семейства графов смежности над набором фрагментов знаний // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики. 2010. Вып. 4. С. 73–76.
3. *Сироткин А.В.* Модели, алгоритмы и вычислительная сложность синтеза согласованных оценок истинности в алгебраических байесовских сетях // Информационно-измерительные и управляющие системы. 2009. №11. С. 32–37.
4. Сироткин А.В., Тулупьев А.Л. Моделирование знаний и рассуждений в условиях неопределенности: матрично-векторная формализация локального синтеза согласованных оценок истинности // Труды СПИИРАН. 2011. Вып. 18. [в печати].
5. *Тулупьев А.Л.* Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 40 с. (Сер. Элементы мягких вычислений).
6. *Тулупьев А.Л.* Алгебраические байесовские сети. Логико-вероятностный подход к моделированию баз знаний с неопределенностью. СПб.: СПИИРАН, 2000. 292 с.
7. *Тулупьев А.Л.* Алгебраические байесовские сети: локальный логико-вероятностный вывод: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 80 с. (Сер. Элементы мягких вычислений).
8. *Тулупьев А.Л.* Алгебраические байесовские сети: система операций локального логико-вероятностного вывода // Информационно-измерительные и управляющие системы. 2009. №4. С. 41–44.
9. *Тулупьев А.Л.* Алгебраические байесовские сети: система операций глобального логико-вероятностного вывода // Информационно-измерительные и управляющие системы. 2010. №11. С. 65–72.
10. *Тулупьев А.Л.* Алгебраические байесовские сети. Теоретические основы и непротиворечивость. СПб.: СПИИРАН, 1995. 76 с.

11. *Тулупьев А.Л.* Апостериорные оценки вероятностей в идеале конъюнктов // Вестник СПбГУ. 2010. Серия 10. Вып. 1. С. 95–104.
12. *Тулупьев А.Л.* Ациклические алгебраические байесовские сети: логико-вероятностный вывод // Нечеткие системы и мягкие вычисления: Научный журнал Российской ассоциации нечетких систем и мягких вычислений. 2006. Том 1, № 1. С. 57–93.
13. *Тулупьев А.Л.* Байесовские сети: логико-вероятностный вывод в циклах. СПб.: Изд-во С.-Петербургского ун-та, 2008. 140 с. (Элементы мягких вычислений).
14. *Тулупьев А.Л.* Генерация множества ограничений на распределение оценок вероятности над идеалом цепочек конъюнкций // Вестник молодых учёных. 2004. № 4. Серия: Прикладная математика и механика. 2004. № 1. С. 35--43.
15. *Тулупьев А.Л.* Непротиворечивость оценок вероятностей в алгебраических байесовских сетях. Вестник СПбГУ. Сер. 10. 2009. Вып. 3. С. 144–151.
16. *Тулупьев А.Л.* Непротиворечивость оценок вероятностей в идеалах конъюнктов и дизъюнктов. Вестник СПбГУ. Сер. 10. 2009. Вып. 2. С. 121–131.
17. *Тулупьев А.Л.* Основы теории алгебраических байесовских сетей: программа спецкурса для студентов старших курсов и аспирантов. СПб.: СПбГУ, 2007. 7 с.
18. *Тулупьев А.Л.* Оценка чувствительности результата априорного логико-вероятностного вывода в интеллектуальных информационных системах // Известия высших учебных заведений: Приборостроение. 2009. №9. С. 3–6.
19. *Тулупьев А.Л.* Согласованность данных и оценка вероятности альтернатив в цикле стохастических предпочтений // Известия высших учебных заведений: Приборостроение. 2009. № 7. С. 3–8.
20. *Тулупьев А.Л., Николенко С.И., Сироткин А.В.* Байесовские сети: логико-вероятностный подход. СПб.: Наука, 2006. 607 с.
21. *Тулупьев А.Л., Сироткин А.В.* Алгебраические байесовские сети: принцип декомпозиции и логико-вероятностный вывод в условиях неопределенности // Информационно-измерительные и управляющие системы. 2008. № 10. т. 6. С. 85–87.
22. *Тулупьев А.Л., Сироткин А.В., Николенко С.И.* Байесовские сети доверия: логико-вероятностный вывод в ациклических направленных графах. СПб.: Изд-во С.-Петерб. ун-та, 2009, 400 с.
23. *Тулупьев А.Л., Столяров Д.М., Ментюков М.В.* Представление локальной и глобальной структуры алгебраической байесовской сети в Java-приложениях // Труды СПИИРАН. 2007. Вып. 5. СПб.: Наука, 2007. С. 71–99.
24. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик // Труды СПИИРАН. 2010. Вып. 1 (12). С. 119–133.
25. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников // Труды СПИИРАН. 2010. Вып. 3 (14) С. 150–169.
26. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи клик владений // Труды СПИИРАН. 2010. Вып. 2 (13). С. 119–133.
27. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи клик-собственников владений // Труды СПИИРАН. 2010. Вып. 4 (15). С. 193–212.
28. *Фильченков А.А.* Алгоритмы построения третичной структуры алгебраической байесовской сети // Труды СПИИРАН. 2011. Вып. 17. С. 197–218.
29. *Фильченков А.А., Тулупьев А.Л.* Анализ циклов в минимальных графах смежности алгебраических байесовских сетей // Труды СПИИРАН. 2011. Вып. 17. С. 151–173.

30. *Фильченков А.А., Тулупьев А.Л.* Структурный анализ систем минимальных графов смежности Труды СПИИРАН. 2009. Вып. 11. С. 104–127.
31. *Фильченков А.А., Тулупьев А.Л.* Третичная структура алгебраической байесовской сети // Труды СПИИРАН. 2011. Вып. 18. С. 164–187.
32. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Особенности анализа вторичной структуры алгебраической байесовской сети // Труды СПИИРАН. 2010. Вып. 1 (12). С. 97–118.
33. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Структурный анализ клик минимальных графов смежности // Вестн. Тверск. гос. ун-та. Сер.: Прикладная математика. 2011. №20. С. 139–151.

Фильченков Андрей Александрович — аспирант кафедры информатики математико-механического факультета С.-Петербургского государственного университета (СПбГУ), младший научный сотрудник лаборатории теоретических и междисциплинарных проблем информатики Учреждения Российской академии наук Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: автоматическое обучение вероятностных графических моделей. Число научных публикаций —32. aaafil@mail.ru, СПИИРАН, 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178, РФ; р.т. +7(812)328-3337, факс +7(812)328-4450. Научный руководитель — А.Л. Тулупьев.

Filchenkov Andrey Alexandrovich — PhD student of Computer Science Department, SPbGU, junior researcher, Theoretical and Interdisciplinary Computer Science Laboratory, SPIIRAS Research area: machine learning of probabilistic graphical models. The number of publications — 32. aaafil@mail.ru, SPIIRAS, 14-th line V.O., 39, St. Petersburg, 199178, Russia; office phone +7(812)328-3337, fax +7(812)328-4450. Scientific advisor — A.L. Tulupyev.

Тулупьев Александр Львович — д.ф.-м.н., доцент; заведующего лабораторией теоретических и междисциплинарных проблем информатики Учреждения Российской академии наук Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН), профессор кафедры информатики математико-механического факультета С.-Петербургского государственного университета (СПбГУ). Область научных интересов: представление и обработка данных и знаний с неопределенностью, применение методов математики и информатики в социокультурных исследованиях, применение методов биостатистики и математического моделирования в эпидемиологии, технология разработки программных комплексов с СУБД. Число научных публикаций — 250. ALT@iias.spb.su, www.tulupyev.spb.ru; СПИИРАН, 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178, РФ; р.т. +7(812)328-3337, факс +7(812)328-4450.

Tulupyev Alexander Lvovich — PhD in Computer Science, Dr. of Sc., Associate Professor; Head of Theoretical and Interdisciplinary Computer Science Laboratory, SPIIRAS, Professor of Computer Science Department, SPbSU. Research area: uncertain data and knowledge representation and processing, mathematics and computer science applications in socio-cultural studies, biostatistics, simulation, and mathematical modeling applications in epidemiology, data intensive software systems development technology. Number of publications — 250. ALT@iias.spb.su, www.tulupyev.spb.ru; SPIIRAS, 14-th line V.O., 39, St. Petersburg, 199178, Russia; office phone +7(812)328-3337, fax +7(812)328-4450.

Поддержка исследования. Работа выполнена при финансовой поддержке РФФИ, проект № **09-01-00861**-а «Методология построения интеллектуальных систем поддержки принятия решений на основе баз фрагментов знаний с вероятностной неопределенностью», а также гранта правительства Санкт-Петербурга для победителей конкурса грантов Санкт-Петербурга для студентов, аспирантов, молодых ученых, молодых кандидатов наук 2010 г., грант № 2.1/03–06/018.

Рекомендовано ТиМПИ СПИИРАН, зав. лаб. А.Л. Тулупьев, д.ф.-м.н., проф.
Работа поступила в редакцию 05.11.2011.

РЕФЕРАТ

Фильченков А.А., Тулупьев А.Л. **Алгоритм выявления ацикличности первичной структуры алгебраической байесовской сети по ее четвертичной структуре.**

Работа глобальных алгоритмов алгебраических байесовских сетей (АБС) опирается на ее вторичную структуру, представляемую графом смежности, и особенную трудность для применения этих алгоритмов создает наличие циклов во вторичной структуре. Не над любой первичной структурой возможно построить ациклический граф смежности (если это возможно, то первичная структура называется *ациклической*). Непосредственное построение даже произвольного графа смежности над данной первичной структурой и определение его цикличности является чрезвычайно затратным, поэтому необходимо уметь определять ацикличности первичной структуры без непосредственного построения вторичной структуры для того, чтобы выбирать подходящую первичную структуру сети (последнее представляет собой задачу глобального обучения первичной структуры).

В работе предложен алгоритм выявления ацикличности первичной структуры АБС без непосредственного построения вторичной структуры, а именно, на основе анализа ее четвертичной структуры. Четвертичная структура АБС — это семейство графов, в которых вершинами выступают сыновья значимых клик в родительском дереве над замкнутым сверху множеством значимых клик, а ребра между двумя сыновьями проведены тогда и только тогда, когда эти сыновья пересекаются по вершинам.

Работа алгоритма опирается на теорему о циклах, которая утверждает, что первичная структура является ациклической тогда и только тогда, когда в любом цикле четвертичной структуры все вершины цикла имеют общий элемент. Алгоритм предполагает построение родительского графа над замкнутым сверху множеством значимых клик и последовательное исследование всех циклов, возникающих между сыновьями значимых клик. Доказана корректность работы алгоритма.

Дана оценка сложности работы алгоритма, которая зависит также от сложности работы алгоритма построения родительского графа.

Предложены улучшения алгоритма выявления ацикличности, основанные на определенных эмпириках, которые, однако, улучшают работу в среднем, но в худшем случае (при ацикличности первичной структуры) не меняют время работы.

ABSTRACT

Filchenkov A.A., Tulupyev A.L. **Algorithm for detection of algebraic Bayesian network primary structure acyclicity based on its quaternary structure.**

The performance of algebraic Bayesian networks (ABN) global algorithms is based on its secondary structure that is represented as a join graph, and a special difficulty for the performance of these algorithms is caused by the existence of cycles in the secondary structure. Acyclic join graph can be synthesized not over any primary structure (if it is possible, such a primary structure is called acyclic). Direct synthesizing of even arbitrary join graph over given primary structure and detecting its acyclicity is very expensive, so it is important to be able to detect the acyclicity of the primary structure without the direct synthesizing of the secondary structure in order to be able to choose a suitable primary structure of the network (this is a problem of the global learning of the primary structure).

The algorithm to detecting of the acyclicity of the ABN primary structure without the direct synthesizing its secondary structure, namely, on the basis of an analysis of its quaternary structure is proposed in the paper. The ABN quaternary structure is a family of graphs in which the vertices are the sons of useful cliques in parent tree over useful clique set closed from above, and the edges between two sons exists if and only if these sons intersects as vertices sets.

The algorithm is based on a theorem of cycles, which proves that the primary structure is acyclic if and only if in any cycle of the quaternary structure all the vertices of the cycle have a common element. The algorithm requires the synthesis of the parent tree over useful clique set closed from above and consistent study of all the cycles in the quaternary structure. The correctness of the algorithm is proved.

The estimation of the algorithm complexity, which also depends on the complexity of the algorithm for synthesizing of the parent graph, is given.

The algorithm improvements based on certain empiricism are proposed. However, the acceleration of the algorithm in average can be achieved, but not in the worst case (when given primary structure is acyclic).