**УДК** 004.8

А.А. Фильченков

# АЛГОРИТМ ПОСТРОЕНИЯ МНОЖЕСТВА МИНИМАЛЬНЫХ ГРАФОВ СМЕЖНОСТИ ПРИ ПОМОЩИ САМОУПРАВЛЯЕМЫХ КЛИК-СОБСТВЕННИКОВ

*Фильченков А.А.* **Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников.**
**Аннотация.** Существует эффективный алгоритм построения множества минимальных графов смежности по заданному набору максимальных фрагментов (при помощи самоуправляемых клик), однако он может быть улучшен путем привлечения результатов активно разрабатывающейся теории глобальной структуры алгебраической байесовской сети. Целью данной работы является разработать улучшенную версию этого алгоритма за счет усовершенствованного построения множества вершин, входящих в клики: вместо полного перебора всех весов клик и вершин производить поиск для каждой клики ее потомков среди других клик. Предложенное улучшение легко в основу нового алгоритма построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников, корректность которого также была доказана.
**Ключевые слова:** алгебраические байесовские сети, вторичная структура, машинное обучение, вероятностно-графические модели систем знаний, глобальная структура.

*Filchenkov A.A.* **Minimal join graph set synthesis self-managed possession cliques algorithm.**
**Abstract.** The effective minimal join graph set synthesis algorithm (of self-managing cliques) from a given maximal knowledge pattern set exists, however it can be improved by engaging results of theory of algebraic Bayesian network global structure that is under development. The goal of the work is to improve the known minimal join graph set algorithm by perfecting designing set of vertices that cliques include: instead of full search of all the pairs of a vertex and a clique weight, particular search for each clique its descendants is suggested. A new algorithm of self-managed possessions cliques implementing suggested improvement is designed.
**Keywords:** algebraic Bayesian networks, secondary structure, machine learning, probabilistic graphical models, global structure.

**1. Введение.** Алгебраическая байесовская сеть (АБС) — логико-вероятностная графическая модель баз фрагментов знаний (ФЗ) с неопределенностью [1–2, 15, 26]. У АБС выделяют две структуры: первичную, которая представляет собой максимальные фрагменты знаний (МФЗ), и вторичную, которая является графом на этих МФЗ. [7, 15, 17]. Такой граф называется графом смежности, и алгоритмизация операций обработки АБС во многом опирается на его структуру [11–14, 16, 18]: именно от вида графа смежности зависит эффективность, а зачастую и сама возможность осуществления логико-вероятностных глобальных априорного и апостериорного выводов, а также поддержания глобальной непротиворечивости.

Как и в байесовских сетях доверия (БСД), более известного родственника АБС, от которых последние выгодно отличаются в первую очередь возможностью работы с интервальными оценками [1–2, 15], в АБС с вычислительной точки зрения наиболее «удобной» вторичной структурой являются деревья смежности [10], однако они возможны лишь для ограниченного множества первичных структур.

В общем случае поиска «эффективных» вторичных структур для произвольного набора фрагментов знаний особый интерес представляет множество минимальных графов смежности, в котором эти структуры содержатся (в том числе это множество содержит все деревья смежности, если они существуют) [4–8].

К сожалению, множество минимальных графов смежности характеризуется недостаточной по сравнению с его важностью изученностью. Именно поэтому актуальной задачей является построение этого множества. В этом направлении был сделан ряд шагов: так, первый алгоритм построения этого множества был сформулирован (в виде схемы) в работе [21], в которой также содержались основное множество используемых в работе с минимальными графами смежности терминов, значительные теоретические результаты и фундаментальная теорема о представлении множества минимальных графов смежности в виде декартового произведения множеств особых подграфов минимальных графов смежности, которая является основой для алгоритмов синтеза указанного множества. Следует также выделить теоретические работы [3, 23, 25], в которых развивались результаты, полученные в статье [21]. В работах [19, 20] были предложены улучшения и улучшенные алгоритмы построения множества минимальных графов смежности, которые опирались на компаративный анализ клик, предложенный в статье [22]. Работа [24] дополняет этот анализ исследованием ребер графов смежности, формируя теоретическую базу для новых улучшения алгоритмов построения множества минимальных графов смежности.

*Цель* данной работы — предложить улучшение существующего алгоритма построения множества минимальных графов смежности [21] на основе построения множества вершин для особых клик не прямым перебором всех вершин после построения всех весов, а параллельным построением весов и вместе с тем множеств вершин этих клик; а также реализовать данное улучшение, разработав новый алгоритм построения этого множества.

**2. Основные определения и обозначения.** В данном разделе приводятся определения так, как они были введены в статье [24], содержащейся в настоящем выпуске журнала. Это позволило привести определения максимально сжато, опустить формализм, иллюстрации и пояснения.

*Граф* — пара $\langle V, E\rangle$, где $V$ — множество вершин графа, а $E$ — множество ребер, каждое из которых является неупорядоченной парой $(v_i, v_j),\ i \neq j, v_i,\ v_j \in V$.

*Алфавит* — множество атомарных пропозициональных формул $A = \{x_1, \dots, x_n\}$, над которым будут заданы максимальные фрагменты знаний. *Слово* $V$ — подмножество алфавита.

*Множество главных конъюнктов максимальных фрагментов знаний (МФЗ), вошедших в АБС,* — это такое множество слов $V^* = \{V_i \subseteq A\}_{i=1}^{i=m}$, что:

1) оно не содержит несобственное подмножество алфавита.
2) никакое слово полностью не содержит никакое другое слово.

*Граф максимальных фрагментов знаний* — ненаправленный граф $G$, вершины которого соответствуют элементам множества главных конъюнктов МФЗ, вошедших в АБС, а ребра удовлетворяют условию

$$(V_i, V_j) \in E(G) \Rightarrow V_i \cap V_j \neq \emptyset.$$

*Вес* $W(V_i)$ *вершины* $V_i \in V(G)$ — множество атомов алфавита, вошедших в $V_i$. *Вес* $W(\{V_i, V_j\})$ *ребра* $\{V_i, V_j\} \in E(G)$ графа $G$ определяется как пересечение весов тех вершин, которые соединены этим ребром. *Вес* $W(H)$ *подграфа* $H \subseteq G$ — наибольшее по включению слово, которое входит в веса всех его вершин.

*Магистральный путь* $B{:}\, V_b \dashrightarrow V_e$ от вершины $V_b$ до вершины $V_e$, пересечение весов которых непусто, — это такой путь от вершины $V_b$ до вершины $V_e$, что вес любой принадлежащей ему вершины содержит пересечение весов начальной и конечной вершин. $B{:}\, V_b \dashrightarrow V_e = P{:}\ V_b \rightsquigarrow V_e$, такой, что

$$\forall V_i \in B W(V_b) \cap W(V_e) \subset W(V_i).$$

Граф *магистрально связен*, если между каждой парой несовпадающих вершин, веса которых содержат общие элементы, существует магистральный путь. Будем обозначать множество магистрально связных графов через **BCG.**

*Граф смежности* — магистрально связный граф МФЗ. *Минимальный граф смежности* — граф смежности, число ребер в котором минимально. Будем обозначать множество минимальных графов смежности через **MJG**. *Максимальный граф смежности* $G_{\max}$ — наибольший по числу ребер граф смежности.

*Сужение* $G \downarrow U$ ненаправленного графа $G$ на слово $U$ — это ненаправленный граф, в который входят только те вершины и ребра исходного графа $G$, веса которых содержат или равны $U$:

Любое слово, являющееся весом какого-либо ребра графа $G$, будем называть *значимым словом графа* $G$, а сужение графа $G$ на такое слово — *значимым сужением*.

*Клика* $U$ — значимое сужение максимального графа смежности на вес $U$. Любая клика является полным подграфом графа $G_{\max}$. Множество всехтаких клик будем обозначать как $\mathrm{Clique}$.

*Граф клик* — направленный граф, вершинами которого являются клики из множества $\mathrm{Clique}$. Ребро из вершины $P$ в вершину $Q$ проведено, если клика $P$ содержит клику $Q$, и не существует клики $R$, такой, что клика $P$ содержит клику $R$ и клика $R$ содержит клику $Q$.

*Сильное сужение* $G \not\downarrow U$ — значимое сужение $G \downarrow U$, из которого удалили все ребра веса $U$:

$$G \not\downarrow U = \langle \{V_i | V_i \in V(G), U \subseteq W(V_i)\}, \{E_i | E_i \in E(G), U \subset W(E_i)\} \rangle.$$

*Сильное сужение* графа $G_{\max} \not\downarrow U$ представляет собой компоненты связности, на которые разбивается сужение $G_{\max} \downarrow U$ удалением ребер веса $U$. *Владение* $P_U^i$ — множество вершин $i$-й компоненты связности сильного сужения $G_{\max} \not\downarrow U$.

*Доменная вершина* $D_U$ клики $U$ — вершина, принадлежащая клике $U$ и не принадлежащая ни одному из ее сыновей.

*Вассал* $V_U$ клики $U$ — множество вершин, входящих в какого-либо сына клики $U$.

Два вассала $V_U^i$ и $V_U^j$ называются *братьями*, если их пересечение непусто:

$$V_U^i \leftrightarrow V_U^j \Leftrightarrow V_U^i \cap V_U^j \neq \emptyset.$$

Два вассала $V_U^i$ и $V_U^j$ называются *полусиблингами*, если существует такой упорядоченный набор вассалов $\{V_U^{w_1}, \dots, V_U^{w_n}\}$, что $V_U^i$ — брат $V_U^{w_1}$, $V_U^{w_1}$ — брат $V_U^{w_2}$, …, $V_U^{w_{n-1}}$ — брат $V_U^{w_n}$, а $V_U^{w_n}$ — брат $V_U^j$:

$$V_U^i \leftrightarrow V_U^j \Leftrightarrow \exists \{V_U^{w_1}, \dots, V_U^{w_n}\}: V_U^b \leftrightarrow V_U^{w_1}, V_U^{w_n} \leftrightarrow V_U^e$$
$$\text{и } \forall i < n \;\; V_U^{w_i} \leftrightarrow V_U^{w_{i+1}}.$$

*Полусиблинговый путь* между двумя родственными вассалами $V_U^i$ и $V_U^j$ — такой набор вассалов $\{V_U^{w_1}, \dots, V_U^{w_n}\}$, что $V_U^b \leftrightarrow V_U^{w_1}$; $V_U^{w_n} \leftrightarrow V_U^e$ и $\forall i < n \; V_U^{w_i} \leftrightarrow V_U^{w_{i+1}}$.

*Братство* $B_U$ клики $U$ — непустой набор вассалов $\{V_U^1, \dots, V_U^l\}$ клики $U$, такой, что с каждым вассалом в братство входят все его полусиблинги и только они:

$$B_U = \{V_U^i | (V_U^i \in B_U) \& (V_U^i \nleftrightarrow V_U^j) \Rightarrow V_U^j \in B_U; V_U^i, V_U^j \in B_U \Rightarrow \Rightarrow V_U^i \nleftrightarrow V_U^j\}.$$

***Теорема о классификации владений [22].*** Любое владение любого сильного сужения $G \Downarrow U$ является либо доменной вершиной, либо вассалом, либо братством $U$

*Сжатие* $\sigma_U$ компоненты связности $P_U^i \subseteq G \Downarrow U$ в вершину $f_i$ — отображение на множестве подмножеств вершин, сопоставляющее множеству вершин $P_U^i$ вершину $f_i$.

*Сжатие* $\sigma_U$ множества ребер

$$E_{i,j} \subset (G \downarrow U) \backslash (G \Downarrow U),$$

соединяющих владения $P_U^i$ и $P_U^j$ в ребро $e_{i,j}$ — отображение на множестве подмножеств ребер, сопоставляющее множеству ребер $E_{i,j}$ ребро $e_{i,j}$, соединяющее вершины $f_i = \sigma_U(P_U^i)$ и $f_j = \sigma_U(P_U^j)$ и имеющее кратность, равную $|E_{i,j}|$.

*Сжатие* $\sigma_U$ графа смежности $G$ в граф $K_U$ — отображение на множестве графов, сопоставляющий графу $G$, являющему графом смежности, граф $K_U$, вершинами которого являются владения сильного сужения $G \Downarrow U$, а ребро между двумя вершинами $f_1$ и $f_2$ графа $K_U$ существует, если существует ребро в графе $G$ между вершинами, принадлежащими соответствующим $f_1$ и $f_2$ владениям $P_U^1$ и $P_U^2$. Кратность такого ребра $(f_1, f_2)$ равна числу всех ребер, соединяющих вершины из $P_U^1$ и $P_U^2$.

*Феод* — $f_i$ — вершина, получившаяся путем сжатия какого-то владения.

*Курия веса* $U$ — $K_U$ — ненаправленный граф с кратными ребрами, полученный путем сжатия сужения $G \downarrow U$.

*Оммаж* $H_U$ — курия $K_U$, являющаяся деревом, все ребра которой имеют кратность, равную единице. Любое сжатие минимального графа смежности является оммажем.

Жила $S_U$ — множество ребер графа смежности G, такое, что $E_u = \{\sigma_U(e) | e \in S_U\}$ является множеством ребер оммажа сжатия $\sigma_U$. В жилу $S_U$ входят те и только те ребра минимального графа смежности, вес которых равен $U$.

*Пучок* — граф, построенный на исходном наборе вершин, множество ребер которого равно объединению жил, выбранных по одной для каждого значимого слова.

***Теорема (о множестве минимальных графов смежности) [21, 23].*** Множество минимальных графов смежности совпадает с множеством пучков.

***Следствие 1 [21].*** Число ребер в минимальных графах смежности одинаково.

***Следствие 2 [21].*** Множество минимальных графов смежности совпадает с множеством пучков, которое равно декартовому произведению множеств жил каждой клики.

***Следствие 3 [21].*** Мощность множества графов смежности равна произведению мощностей множеств жил каждой клики.

***Следствие 4 [21].*** Согласно следствию 2, чтобы построить множество графов смежности, достаточно для каждой клики построить множество соответствующей ей жил.

***3. Классификация клик и алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик.*** *Собственное ребро клики* — ребро, принадлежащее клике, вес которого совпадает с весом клики.

По числу собственных ребер множество клик можно поделить на:

- *безреберные* — $C_0$ — клики (сужения), у которых нет собственных ребер;
- *однореберные* — $C_1$ — клики, у которых ровно одно собственное ребро;
- *многореберные* — $C_n$ — клики, у которых более одного собственного ребра.

По наличию детей множество клик можно поделить на:

- *бездетные* — $C^-$ — клики, у которых нет детей;
- *родительские* — $C^+$ —клики, у которых есть дети.

Таблица 1. **Классификация клик**

| Наличие ребер | Бездетные | Родительские |
|---|---|---|
| Безреберные | $C_0^-$ | $C_0^+$ |
| Однореберные | $C_1^-$ | $C_1^+$ |
| Многореберные | $C_n^-$ | $C_n^+$ |

По числу феодов, в которые сжимаются клики, они делятся на:

- *моноклики* — клики, сжимающиеся до одного феода.
- *стереоклики* — клики, сжимающиеся до более чем одного феода.

*Псевдоклика* — $C_0^-$ — сужение, не имеющее собственных ребер и не имеющее детей.

*Моноклика-0* — $C_0^+$ — сужение, не имеющее собственных ребер, но имеющее детей.

*Биклика* — $C_1^-$ — клика, имеющее ровно одно собственное ребро и не имеющее детей. Ее единственное ребро является *обязательным* — оно входит во все минимальные графы смежности [24].

*Моноклика-1* — $C_1^+$ — клика, имеющая ровно одно собственное ребро и имеющая детей. Ребро это клики является *избыточным* — оно не входит ни в один из минимальных графов смежности [24].

*Бездетная поликлика* (возможно также название *бездетная стереоклика*) — $C_n^-$ —клика, имеющая более одного собственного ребра, но не имеющая детей.

*Родительская поликлика* — $C_n^+$ — клика, имеющая более одного собственного ребра и имеющая детей.

*Моноклика-n* — $mC_n^+$ — клика, имеющая более одного собственного ребра и имеющая детей, но состоящая ровно из одного феода.

*Родительская стереоклика* — $pC_n^+$ —клика, имеющая более одного собственного ребра, имеющая детей, и состоящая более чем из одного феода.

Полученные в работе [22] результаты удобно расположить в таблице (табл. 2).

*Основное множество вершин клики* — множество вершин, являющихся концами ее собственных ребер.

***Утверждение 1 [24].*** Основное множество вершин стереоклики совпадает с множеством ее вершин.

*Базовая схема алгоритма построения множества минимальных графов смежности* [21].

Над множеством вершин $V$ строится максимальный граф смежности $G_{\max}$. По этому графу строится граф клик $\mathrm{Clique}$. Каждой клике сопоставляется множество вершин, которые в нее попали. На графе клик задается порядок.

По заданному порядку перебираются все клики из упорядоченного дерева клик. Для каждой клики при помощи алгоритма Прюфера перебираются все оммажи, поскольку оммаж является деревом — и для каждого оммажа перебираются все соответствующие ему жилы. Таким образом, для каждой клики перебираются все жилы, соответствующие ее весу. Все такие жилы записываются в массив жил для данной клики.

Таблица 2. **Характеристики различных клик**

| Сужение | Обозначение | Является ли кликой | Есть ли дети | Число собственных ребер | Число вершин | Число феодов | Число жил |
|---|---|---|---|---|---|---|---|
| Псевдоклика | $C_0^-$ | Нет | Нет | 0 | 1 | 0 | 0 |
| Моноклика-0 | $C_0^+$ | Нет | Да | 0 | $> 2$ | 1 | 1 |
| Биклика | $C_1^-$ | Да | Нет | 1 | 2 | 2 | 1 |
| Моноклика-1 | $C_1^+$ | Да | Да | 1 | $> 2$ | 1 | 1 |
| Бездетная стереоклика | $C_n^-$ | Да | Нет | $> 1$ | $> 2$ | $> 2$ | $> 1$ |
| Родительская поликлика | $C_n^+$ | Да | Да | $> 1$ | $> 2$ | $\forall$ | $\forall$ |
| Моноклика-$n$ | $mC_n^+$ | Да | Да | $> 1$ | $> 2$ | 1 | 1 |
| Родительская стереоклика | $pC_n^+$ | Да | Да | $> 1$ | $> 2$ | $> 1$ | $> 1$ |

В [19] было предложено три улучшения базового алгоритма (схему базового алгоритма смотри в [21]), а также сам алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик.

***Улучшение 1 (исключение незначимых сужений) [19].*** Вместо того чтобы рассматривать все возможные клики, мы можем исключить из рассмотрения моноклики-0, так как они не создают никаких жил и перебор оставшихся клик будет сведен к перебору весов ребер максимального графа смежности.

***Улучшение 2 (исключение клик с единственным владением) [19].*** Вместо того чтобы строить жилы для моноклик-1 и моноклик-$n$, мы можем исключить их из множества клик, для которых строятся жилы, что ускорит вывод минимальных графов смежности, так как нужно будет перебирать меньшее число клик.

***Улучшение 3 (априорный учет ребер однореберных бездетных клик) [19].*** Вместо того чтобы строить жилы для биклик, мы можем исключить их из множества клик, для которых строятся жил, добавив их ребра в граф обязательных ребер, что ускорит вывод минимальных графов смежности, так как нужно будет перебирать меньшее число клик.

*Алгоритм построения минимальных графов смежности при помощи самоуправляемых клик (self-managing cliques algorithm)* [19] строит по заданному набору вершин $V$, соответствующих множеству главных конъюнктов максимальных ФЗ, все возможные минимальные графы смежности.

```
Require: V
Ensure: MJG = {⟨V, E_i⟩}
Weights = ∅
for all u, v ∈ V, (u ≠ v)&(W(u)⋂W(v) ≠ ∅) do
   Weighs ← W(u)⋂W(v)
end for
Cliques = ∅
for all v ∈ V do
   for all w ∈ Weights
      if w ⊂ W(v) do
         Cliques → C_w : W(C_w) = w
         C_w ⟵ u
         Cliques ⟵ C_w
      end if
   end for
end for
NecessaryEdges = ∅
StereoIndex = ∅
for all C_w ∈ Cliques do
   if |C_w| = 2 then
      C_w → v
      C_w → u : u ≠ v
      NecessaryEdges ⟵ (v, u)
   end if
   else
      P = компоненты связности строго сужения
      клики C_w
      if |P| > 1 then
         StereoIndex ⟵ w
         S_w = ∅
         for all t is_a дерево на P do
            for all s is_a жила для t do
               S_w ⟵ s
```

**end for**
**end for**
**end if**
**end else**
**end for**
$MJG = \emptyset$
**for all** $\{s_w^i\}$: $\{s_w^i\}$ —индексирующая последовательность жил стереоклик $w \in StereoIndex, s_w^i \in S_w$ **do**
$E = NecessaryEdges$
**for all** $s \in \{s_w^i\}$ **do**
$E = E \cup s.edges$
**end for**
$MJG \longleftarrow \langle V, E \rangle$
**end for**
**return** $MJG$

Листинг 1. Алгоритм построения минимального графа смежности при помощи самоуправляемых клик.
Здесь и далее $S \leftarrow e$ обозначает добавление элемента $e$ к множеству $S$, а $S \rightarrow e: Cond(e)$ обозначает извлечение элемента, удовлетворяющего условиям $Cond$, из множества $S$ в переменную $e$.

В работе [20] предложено еще одно улучшение. К формулируемому в этой статье алгоритму это улучшение никак не относится и указано здесь для сохранения нумерации.

***Улучшение 4 (построение владений через сыновей) [20].*** Вместо того, чтобы строить владения клик путем поиска компонент связности соответствующего строго сужения, мы можем строить владения как компоненты связности относительно родства вассалов.

**4. Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик (в нашей терминологии — улучшение 5 — учет основного множества ребер).** Вместо того, чтобы сначала строить множество весов ребер, а потом сужать на каждый вес набор вершин, мы можем для каждого веса ребра построить основное множество вершин соответствующей ему клики. Благодаря этому мы можем сразу выделить клики с единственным собственным ребром (у них основное множество вершин состоит всего из двух вершин) и, проверив, кто из них является, а кто не является родителем, можем сразу построить биклики и исключить из рассмотрения моноклики-1.

***Пояснение.*** В алгоритме, использующем самоуправляемые клики, множество вершин для каждой клики строится перебором всех пар вершин, а затем перебором всех вершин для каждого значимого веса. Алгоритм клик владений несколько сокращает перебор, позволяя для каждого значимого слова перебирать вершины одного из его родителей. Улучшение 5 позволяет строить множества вершин стереоклик одновременно с построением множества значимых весов (это следует из утверждения 1). Далее, благодаря тому, что у моноклик-1 и биклик мощность множества основных вершин равна двум, на начальном этапе алгоритма мы можем сразу исключить первые и добавить ребра вторых в граф обязательных ребер.

На текущем этапе сложно говорить о том, насколько улучшение 5 ускоряет работу алгоритма. В рамках данной статьи мы оставим этот вопрос открытым.

*Алгоритм построения минимальных графов смежности при помощи самоуправляемых клик-собственников (proprietor self-managing cliques algorithm)* строит по заданному набору вершин $V$, соответствующих множеству главных конъюнктов максимальных ФЗ, все возможные минимальные деревья смежности графа МФЗ.

Над множеством вершин $V$ строится максимальный граф смежности $G_{\max}$. Перебираются все его ребра, и концы каждого ребра добавляются в множество основных вершин соответствующего веса. Перебираются все построенные клики и запоминаются те, у которых нет детей (бездетные клики). Выбираются веса, которым соответствуют множества основных вершин, состоящие ровно из двух вершин. Ребра тех из них, у которых нет детей, добавляются в граф обязательных клик $necessaryEdgesGraph$.

Перебираются все остальные веса. Для каждого веса строится клика и для нее строятся все ее феоды путем поиска компонент связности на множестве основных вершин. Если клика является монокликой (т.е. содержит ровно один феод), то она не рассматривается. Если же клика является стереокликой, то добавляется в массив стереоклик.

При помощи алгоритма Прюфера перебираются все оммажи и для каждого оммажа перебираются все соответствующие ему жилы. Таким образом, для каждой стереоклики перебираются все жилы, соответствующие ее весу. Все такие жилы записываются в массив жил для данной стереоклики.

Согласно теореме о множестве минимальных графов смежности, нам достаточно построить все возможные пучки, чтобы получить все минимальные графы. Это можно сделать, перебрав всевозможные комбинации жил для каждой стереоклики, объединяя такие жилы и граф $necessaryEdgesGraph$.

**Require**: $V$
**Ensure**: $MJG = \{\langle V, E_i \rangle\}$
$Cliques = \emptyset$
**for all** $u, v \in V, (u \neq v) \& (W(u) \cap W(v) \neq \emptyset)$ **do**
$w = W(u) \cap W(v)$
$Cliques \rightarrow C_w : W(C_w) = w$
$C_w \longleftarrow u$
$C_w \longleftarrow v$
$Cliques \longleftarrow C_w$
**end for**
$Childless = \emptyset$
**for all** $C \in Cliques$ **do**
$D =$клика с последний по порядку весом
**while** $\big(W(D) \not\subset W(C)\big) \& (W(D) > W(C))$ **do**
$D =$клика с предыдущим по порядку весом
**end while**
**if** $W(D) \not\subset W(C)$ **then**
$Childless \leftarrow C$
**end for**
$NecessaryEdges = \emptyset$
$StereoIndex = \emptyset$
**for all** $C_w \in Cliques$ **do**
**if** $|C_w| = 2$ **then**
**if** $C_w \in Childless$ **then**
$C_w \rightarrow v$
$C_w \rightarrow u : u \neq v$
$NecessaryEdges \longleftarrow (v, u)$
**end if**
**end if**
**else**
$P =$ компоненты связности строго сужения клики $C_w$
**if** $|P| > 1$ **then**
$StereoIndex \longleftarrow w$
$S_w = \emptyset$
**for all** $t$ **is_a** дерево на $P$ **do**

34:         **for all** $s$ **is_a** жила для $t$ **do**
35:           $S_w \longleftarrow s$
36:         **end for**
37:       **end for**
38:     **end if**
39:   **end else**
40: **end for**
41: $MJG = \emptyset$
42: **for all** $\{s_w^i\}: \{s_w^i\}$ –индексирующая последовательность жил стереоклик $w \in StereoIndex, s_w^i \in S_w$ **do**
43:   $E = NecessaryEdges$
44:   **for all** $s \in \{s_w^i\}$ **do**
45:     $E = E \cup s.edges$
46:   **end for**
47:   $MJG \longleftarrow \langle V, E \rangle$
48: **end for**
49: **return** $MJG$

Листинг 2. Алгоритм построения минимального графа смежности при помощи клик владений.

В цикле (2–8) строится основное множество вершин для каждого значимого сужения.

В цикле (10–17) строится множество бездетных клик $Childless$ путем перебора всех клик и поиска сыновей для выбранной клики.

В цикле (12–14) ищется клика-потомок для рассматриваемой клики до тех пор, пока она не найдется или вес перебираемых клик не совпадет с весом рассматриваемой клики.

В цикле (20–40) алгоритм строит множество жил для каждой стереоклики.

Условный оператор (21–27) обрабатывает клики, у которых мощность основного множества вершин равна двум (т.е. биклики и моноклики-1).

Условный оператор (22–26) обрабатывает биклики, добавляя их единственные ребра к множеству обязательных ребер $NecessaryEdges$.

Условный оператор (28–39) при условии, что мощность основного множества вершин клики больше двух, строит множество жил для данной клики.

Условный оператор (30–38) исключает из рассмотрения клики, сжимаемые до одного феода (т.е. имеющие ровно одно владение).

В цикле (33–37) перебираются все деревья, построенные на феодах клики (т.е. оммажи). Для этого, как правило, используется алгоритм Прюфера.

В цикле (34–36) в массив жил для данной клики добавляются все жилы, соответствующие данному оммажу.

В цикле (42–48) алгоритм перебирает все кортежи, индексирующие жилы для каждой клики. Каждый кортеж индексирует набор жил, выбранных по одной для каждой клики. В цикле происходит объединение всех жил одного кортежа с ребрами из множества обязательных ребер. Граф, содержащий эти ребра, добавляется к множеству минимальных графов смежности.

***Утверждение 2.*** Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников строит множество минимальных графов смежности.

***Доказательство.*** В цикле (2–8) будет построено основное множество вершин для каждой клики, содержащей более двух вершин, поскольку к соответствующим множествам вершин для каждого ребра добавляются его концы.

В цикле (12–14) выясняется, является ли клика бездетной. Если у клики есть потомок, то у нее есть и дети. Поиск клик-сыновей ведется только среди клик, вес которых имеет больший лексико-графический порядок, чем вес рассматриваемой клики, потому что вес потомка идет по порядку после веса родителя.

В цикле (10–17) строится множество бездетных клик $Childless$ потому что перебираются все клики, и для каждой клики определяется, является ли она бездетной.

Условный оператор (22–26) добавит к множеству обязательных ребер ребро биклики, поскольку именно биклика имеет мощность основного множества вершин, равную двум, и не имеет детей.

Условный оператор (21–27) будет обрабатывать только биклики и моноклики-1, поскольку, как это было доказано в [24], только они имеют мощность основного множества вершин, равную двум, а моноклики-1 не участвуют в построении множества минимальных графов смежности (согласно улучшению 2).

В цикле (34–36) будут построены жилы для по данному оммажу, а в цикле (33–37) будут построены все жилы для данной клики, что следует из определения жил и оммажей.

Условный оператор (28–39) при условии, что мощность основного множества вершин клики больше двух, строит множество жил для данной клики, поскольку такие клики являются стереокликами, а для них множество вершин совпадает с основным множеством вершин, которые мы построили в цикле (2–8).

В цикле (20–40) будут рассмотрены все клики, для стереоклик будет построено множество жил, собственные ребра биклик будут добавлены в граф обязательных ребер, а моноклики будут проигнорированы.

Цикле (42–48) перебирает все наборы жилы по одной для каждой клики стереоклики и объединяет их с множеством обязательных ребер. Согласно теореме о минимальных графах смежности, мы перебираем таким образом все возможные множества ребер минимальных графов смежности.

**5. Заключение.** В статье были рассмотрены результаты анализа клик и вторичных структур, проведенные в статьях [22–25]. На основании утверждений, сформулированных в работе [24] было предложено улучшение известного алгоритма построения множества минимальных графов смежности при помощи самоуправляемых клик.

Улучшение состоит в оптимизации построения множеств вершин клик. Вместо построения множества вершин для стереоклик предлагается строить *основное множество вершин*, т.е. множество вершин, которые являются концами *собственных ребер* (ребер, вес которых совпадает с весом клики). Как было доказано в работе [24], эти два множества совпадают для стереоклик. В то же время построение основного множества для стереоклик значительно проще, чем построение просто множества вершин — достаточно просто перебрать все ребра и их концы добавить в множества вершин соответствующего веса. Отдельно обрабатываются не относящиеся к стереокликам биклики и моноклики-1 — клики, у которых число собственных ребер равно одному, а мощность основного множества ребер равна двум. Как было доказано в [24], ребра первых являются обязательными, т.е. содержатся в каждом минимальном графе клик, тогда как ребра вторых являются избыточными — они не входят ни в один минимальный граф смежности. Отсюда моноклики-1 удаляются из рассмотрения, а ребра биклик добавляются к множеству обязательных ребер, которое в дальнейшем объединяется с наборами жил в пучки.

Предложен алгоритм построения множества минимальных графов смежности, реализующий указанное улучшение, а также три улучшения, предложенные в [20], который будет выполняться быстрее, чем базовый алгоритм, предложенный в статье [21]. Доказана корректность алгоритма.

Дальнейший анализ глобальной структуры алгебраических байесовских сетей позволит обособить и структурировать новые элементы, специальная работа с которым ляжет в основу дальнейших улучшений алгоритма построения множества минимальных графов смежности. Кроме того, открытым остается вопрос оценки сложности алгоритма.

## Литература


1. *Городецкий В.И.* Алгебраические байесовские сети — новая парадигма экспертно-вычислительных систем // Юбилейный сб. тр. институтов Отделения информатики, вычислительной техники и автоматизации РАН. Т. 2. М., 1993, С. 120–141.
2. *Городецкий В.И., Тулупьев А.Л.* Формирование непротиворечивых баз знаний с неопределенностью // Известия РАН. Сер. Теория и системы управления. 1997. № 5. С. 33–42.
3. *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.
4. *Павельчук А.В., Тулупьев А.Л., Тотмянина С.А.* Подход к объектно-ориентированному представлению данных алгебраических байесовских сетей в java-коде и реляционных СУБД // Материалы XI Санкт-Петербургской междунар. конф. «Региональная информатика-2008 (РИ-2008). Санкт-Петербург, 22–24 октября, 2008 г. СПб., 2009. С. 68–76.
5. *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.
6. *Тотмянина С.А., Павельчук А.В., Тулупьев А.Л.* Алгебраические байесовские сети: структуры данных в СУБД и Java- коде // Докл. науч.-практич. конф. студентов, аспирантов, молодых ученых и специалистов «Интегрированные модели, мягкие вычисления, вероятностные системы и комплексы программ в искусственном интеллекте. Коломна, 26–27 мая 2009 г. Т. 2. М., 2009. С. 123–131.
7. *Тулупьев А.Л.* Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности: Учеб. пособие. СПб.: СПбГУ; ООО «Издательство «Анатолия», 2007. 40 с.
8. *Тулупьев А.Л.* Алгебраические байесовские сети: реализация логико-вероятностного вывода в комплексе java-программ // Тр. СПИИРАН. 2009. Вып. 8. С. 191–232.
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.* Ациклические алгебраические байесовские сети: логико-вероятностный вывод // Нечеткие системы и мягкие вычисления: Научный журнал Российской ассоциации нечетких систем и мягких вычислений. 2006. Т. 1, № 1. С. 57–93.
11. *Тулупьев А.Л.* Байесовские сети: логико-вероятностный вывод в циклах. СПб.: Изд-во СПбГУ, 2008. 140 с.

12. *Тулупьев А.Л.* Непротиворечивость оценок вероятностей в алгебраических байесовских сетях. Вестник СПбГУ. Сер. 10. 2009. Вып. 3. С. 144–151.
13. *Тулупьев А.Л.* Преобразование ациклических байесовских сетей доверия в алгебраические байесовские сети // Известия высших учебных заведений: Приборостроение. 2009. № 3. С. 21–23.
14. *Тулупьев А.Л.* Согласованность данных и оценка вероятности альтернатив в цикле стохастических предпочтений // Известия высших учебных заведений: Приборостроение. 2009. № 7. С. 3–8.
15. *Тулупьев А.Л., Николенко С.И., Сироткин А.В.* Байесовские сети: логико-вероятностный подход. СПб.: Наука, 2006. 607 с.
16. *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.
17. *Тулупьев А.Л., Сироткин А.В., Николенко С.И.* Байесовские сети доверия: логико-вероятностный вывод в ациклических направленных графах. СПб.: Изд-во Санкт-Петерб. ун-та, 2009, 400 с.
18. *Тулупьев А.Л., Столяров Д.М., Ментюков М.В.* Представление локальной и глобальной структуры алгебраической байесовской сети в Java-приложениях. // Тр. СПИИРАН. 2007. Вып. 5. СПб.: Наука, 2007. С. 71–99.
19. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик // Тр. СПИИРАН. 2010. Вып. 1 (12). С. 119–133.
20. *Фильченков А.А.* Алгоритм построения множества минимальных графов смежности при помощи клик владений // Тр. СПИИРАН. 2010. Вып. 2 (13).
21. *Фильченков А.А., Тулупьев А.Л.* Структурный анализ систем минимальных графов смежности Тр. СПИИРАН. 2009. Вып. 11. С. 104–127.
22. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Компаративный анализ клик минимальных графов смежности алгебраических байесовских сетей // Тр. СПИИРАН. 2010. Вып. 2 (13).
23. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Особенности анализа вторичной структуры алгебраической байесовской сети // Тр. СПИИРАН. 2010. Вып. 1 (12). С. 97–118.
24. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Ребра графов смежности в контексте компаративного анализа клик минимальных графов смежности алгебраических байесовских сетей // Тр. СПИИРАН. 2010. Вып. 3 (14).
25. *Фильченков А.А., Тулупьев А.Л., Сироткин А.В.* Структурный анализ клик минимальных графов смежности // Вестник Тверского гос. Ун-та. Сер. Прикладная математика. 2010.
26. *Gorodetsky V.I., Drozdgin V.V., Jusupov R.M.* Application of Attributed Grammar and Algorithmic Sensitivity Model for Knowledge Representation and Estimation // Artificial Intelligence and Information, Control Systems of ROBOTSA. Amsterdam: Elsevier Science Publishers B. V., 1984. P. 232–237.

**Фильченков Андрей Александрович** — аспирант кафедры информатики математико-механического факультета Санкт-Петербургского государственного университета (СПбГУ), м. н. с. лаборатории теоретических и междисциплинарных проблем информатики СПИИРАН. Область научных интересов: автоматическое обучение вероятностных графических моделей. Число научных публикаций — 9. aaafil@mail.ru, СПИИРАН, 14-я линия В.О., д. 39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-3337, факс +7(812)328-4450. Научный руководитель — д-р физ.-мат. наук, доцент 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 — 9. aaafil@mail.ru, SPIIRAS, 14th line V.O., 39, St. Petersburg, 199178, Russia; office phone +7(812)328-3337, fax +7(812)328-4450. Scientific advisor — PhD in Computer Science, Dr. of Sc. Associate Professor A.L. Tulupyev.

Рекомендовано ТиМПИ СПИИРАН, зав. лаб., д-р физ.-мат. наук, доцент А.Л. Тулупьев.

**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 грантом правительства Санкт-Петербурга для победителей конкурса грантов Санкт-Петербурга для студентов, аспирантов, молодых ученых, молодых кандидатов наук 2010 г., диплом ПСП№ 10697.

# РЕФЕРАТ

*Фильченков А.А.* **Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников.**

Цель данной работы — предложить улучшение существующего алгоритма построения множества минимальных графов смежности на основе построения основного множества вершин для стереоклик вместо построения для них множества вершин и реализовать данное улучшение, разработав новый алгоритм построения этого множества.

Приведена теория глобальной структуры алгебраических байесовских сетей, в частности, классификация владений клик и классификация клик графов смежности. Приведена базовая схема алгоритма построения множества минимальных графов смежности, четыре ее улучшения и известный улучшенный алгоритм построения множества минимальных графов смежности (при помощи самоуправляемых клик).

Над множеством вершин строится максимальный граф смежности. По этому графу строится граф клик. Каждой клике сопоставляется множество вершин, которые в нее попали. На графе клик задается порядок. По заданному порядку перебираются все клики из упорядоченного дерева клик. Для каждой клики при помощи алгоритма Прюфера перебираются все оммажи, поскольку оммаж является деревом — и для каждого оммажа перебираются все соответствующие ему жилы. Таким образом, для каждой клики перебираются все жилы, соответствующие ее весу. Все такие жилы записываются в массив жил для данной клики. Согласно теореме о множестве минимальных графов смежности, нам достаточно построить все возможные пучки, чтобы получить все минимальные графы. Это можно сделать, перебрав всевозможные комбинации жил для каждой клики, и затем объединяя такие жилы в единый граф.

Первое улучшение состояло в том, чтобы рассматривать при построении графа клик только клики, т.е. только те подграфы максимального графа смежности, вес которых совпадает с весом какого-либо из ребер максимального графа смежности.

Второе улучшение состояло в том, чтобы рассматривать при переборе клик из графа клик только те клики, которые имеют более одной компоненты связности при удалении из них ребер веса, равного весу клики (т.е. имеющих ровно одно владение).

Третье улучшение состояло в том, чтобы клики, состоящие ровно из двух вершин, не использовать для построения особых множеств ребер — жил, а выделять из них единственное ребро, которое добавлять к специальному графу — графу обязательных ребер.

Четвертое улучшение состояло в том, чтобы строить владения путем поиска компонент связности не на множестве вершин клики, а на множестве вассалов, к которым далее добавляются доменные вершины.

Предложено новое улучшение, которое строит множество вершин стереоклик не прямым перебором, а путем построения основного множества вершин — множества концов ребер с весом, совпадающим с весом клики.

Предложен алгоритм построения множества минимальных графов смежности, реализующий описанные улучшения, доказана его корректность.

Дальнейший анализ глобальной структуры алгебраической байесовской сети может быть полезен в целях создания новых эффективных алгоритмов построения семейства минимальных графов смежности. Открытым также остается вопрос об оценки времени работы алгоритма.

## ABSTRACT

*Filchenkov A.A.* **Minimal join graph set synthesis self-managed possession cliques algorithm.**

The goal of this work is to suggest an improvement of the existing minimal join graph set synthesis algorithm based on designing main vertices set of a stereoclique instead of designing vertex set of that and to actualize the improvement by designing a new algorithm for the minimal join graph set synthesis.

Algebraic Bayesian network global structure theory, classification of clique possessions and classification of join graph cliques in particular, were overviewed. Minimal join graph set synthesis algorithm basic scheme, four its improvement and the known minimal join graph set synthesis (self-managed cliques) algorithm are represented.

Maximal join graph should be built over a given vertex set. A clique graph should be built by that graph. Every clique should be collated to the set of vertexes that belong to the clique. Cliques order should be specified. With the order all the cliques should be sorted out. All homages should be sorted out for every clique by the means of the Pruefer algorithm, because an homage is a tree. Than all sinews should be sorted out for every homage. So that, all the sinews would be sorted out for each clique. All such sinews should be reordered to sinew array for the clique. According to the minimal join graph set theorem, it's enough to design all branches in the purpose to design all minimal join graphs. It can be done by sorting out all sinew combinations chosen one for each clique and uniting them into the one graph.

On the base of observed classification three basic algorithm improvements that base some cliques specifies, are suggested.

The first improvement is to use only the cliques (e.g. only the maximal join graph subgraphs), the weight of which equals to weigh of any maximal join graph edge.

The second improvement is to use only the cliques that have more than one connection components having lost all the edges of the same weight. (e.g. the cliques that has more than one possession).

The third improvement is to use cliques that have only two vertices not in designing sinews, but to add their only edge to a special graph — necessary edges graph.

The fourth improvement is to design possessions by searching connection components not in clique vertex set, but in vassal set to which domain vertex are added aftermath.

A new improvement to design stereoclique vertex set not by entire enumeration but by designing main vertices set — such a set of vertices that belong to edges with the weight that the clique has, is suggested.

A new minimal join graph set synthesis algorithm that actualize the improvement is designed and its robustness is proven.

The further algebraic Bayesian network global structure analysis might be useful in the purpose to create new efficient minimal join graph set designing algorithms. Also the question of the algorithm running time is still open.