

Статистическое сравнение времени работы алгоритмов глобального апостериорного вывода в алгебраических байесовских сетях

А. А. Вяткин¹, А. Л. Тулупьев²

¹Санкт-Петербургский Федеральный исследовательский центр Российской академии наук

²Северо-Западный институт управления Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации

¹aav@dscs.pro, ²alt@dscs.pro

Аннотация. Среди вероятностных графических моделей выделяется класс алгебраических байесовских сетей, которые представляют собой модели баз фрагментов знаний. Важной задачей в теории алгебраических байесовских сетей является глобальный апостериорный вывод, моделирующий поступление новой информации и перестраивающий сеть на этой основе. Ранее было представлено несколько алгоритмов, решающих задачу проведения глобального апостериорного вывода. Данные алгоритмы используют различные глобальные структуры алгебраической байесовской сети – вторичную и третичную. Соответственно определяется задача сравнения этих алгоритмов, в частности необходим сравнительный статистический анализ времени их работы, на что и направлена данная статья. В работе также определяется алгоритм стохастической генерации ациклических алгебраических байесовских сетей, над которыми проводится сравнение алгоритмов глобального апостериорного вывода. В ходе исследований было установлено, что алгоритм, использующий третичную структуру, работает быстрее в 1.15–2 раза на алгебраических байесовских сетях, в которых присутствует 5–150 моделей фрагментов знаний. При этом ускорение тем выше, чем больше атомов в моделях фрагментов знаний и их пересечениях.

Ключевые слова: алгебраические байесовские сети; фрагмент знаний; логико-вероятностный вывод; третичная структура; вероятностные графические модели; машинное обучение; статистическое исследование сложности алгоритмов

I. ВВЕДЕНИЕ

Алгебраические байесовские сети (АБС) являются подклассом вероятностных графических моделей, находящихся свое применение в различных сферах [1–4]. В основе определения алгебраической байесовской сети лежит декомпозиция знаний на небольшие наборы – фрагменты знаний. В качестве моделей последних могут выступать, например, идеалы конъюнктов или наборы квантов, каждому из которых назначается интервальная или скалярная оценка вероятности истинности. Сама же алгебраическая байесовская сеть может рассматриваться с точки зрения различных глобальных структур, которыми являются, в частности, вторичная и третичная структуры.

Одним из основных вопросов, изучающихся в теории алгебраических байесовских сетей является глобальный апостериорный вывод, моделирующий поступление

новой информации о предметной области и учитывающий ее в модели. Ранее глобальный апостериорный вывод производился с применением вторичных структур и механизма распространения виртуального свидетельства [5]. В качестве альтернативы этому алгоритму может выступать способ, использующий третичную структуру [6, 7].

Таким образом возникает вопрос о сравнении двух данных алгоритмов, в частности, времени их работы. Для обоих алгоритмов приведены оценки вычислительной сложности, но сравнение их затруднено в силу возникающих скрытых численных характеристик, проявляющихся только после формирования искомого объекта. В связи с этим видится перспективным эмпирический подход к сравнению времени работы алгоритмов, на что и направлена данная статья. Следовательно, целью статьи является статистическое сравнение времен двух алгоритмов глобального апостериорного вывода, использующих вторичную и третичную структуры. В процессе достижения поставленной цели была также решена задача построения алгоритма, формирующего ациклические алгебраические сети, на наборе которых и производилось сравнение двух алгоритмов.

Теоретическая значимость работы состоит в определении нового алгоритма стохастического построения ациклических алгебраических байесовских сетей, что даст возможность сравнивать времена работы алгоритмов, использующихся с такого рода моделями. Практическая значимость заключается в выявлении оптимального по времени работы алгоритма глобального апостериорного вывода, что позволит расширить применение алгебраических байесовских сетей в практических задачах.

II. ПОСТАНОВКА ЗАДАЧИ

Опишем в обзорном формате систему терминов, рассматриваемых в данной работе и указанных в [5, 7–9].

Как было сказано ранее, АБС рассматривается как структура над более малыми объектами – *моделями фрагментов знаний* (ФЗ). В работе ФЗ представляется в виде набора *квантов* над некоторым *алфавитом* (*атомами*), где каждому кванту сопоставлена скалярная или интервальная оценка вероятности истинности. Под *квантом* понимается конъюнкция элементов заданного алфавита и их отрицаний. Например, для алфавита $\{x_0, x_1\}$ набором квантов будет $\{x_1 x_0, x_1 \bar{x}_0, x_1 \bar{x}_0 \bar{x}_1 x_0\}$.

Работа выполнена в рамках проекта по государственному заданию СПб ФИЦ РАН № FFZF-2022-0003.

На оценки вероятности истинности накладываются ограничения, исходящие из аксиоматики вероятностной логики и из предметной области. В частности, ограничениями, накладываемыми на скалярные оценки вероятности истинности для квантов q_1, \dots, q_n , будут:

$$\begin{cases} \sum_{i=1}^n p(q_i) = 1, \\ q_i \geq 0, \forall i \in \{1, \dots, n\}, \end{cases} \quad (1)$$

где $p(q)$ – оценка вероятности истинности кванта q . Стоит сказать, что над ФЗ определены операции пересечения и включения как над алфавитами этих ФЗ. При этом пересечением двух ФЗ будет также являться ФЗ.

Набор самих ФЗ определяет более обширную структуру – *первичную структуру* АБС. *Вторичная* же и *третичная структура* рассматриваются как графы с вершинами, нагруженными ФЗ. Более формально, *вторичная структура* – *граф смежности* с нагруженными ФЗ вершинами. *Граф смежности* – граф, в котором, во-первых, нагрузка одной вершины не входит полностью в нагрузку никакой другой вершины и, во-вторых, между каждой парой вершин, содержащих общие элементы, существует путь. При этом в нагрузки каждой из вершин этого пути входят элементы, общие для начальной и конечной вершины.

Каждому ребру в таком графе удобно также приписать нагрузку – пересечение нагрузок инцидентных этому ребру вершин (*сепаратор*). При этом рассматриваются только ребра, имеющие непустую нагрузку.

Для конкретной первичной структуры может существовать множество различных вторичных структур. В случае, если для первичной структуры существует вторичная, представляемая в виде дерева смежности, то первичная структура называется *ациклической*. Если рассмотреть диаграмму Хассе [10] для множества непустых сепараторов, объединенных с множеством нагрузок ФЗ и пустым множеством, то получим *третичную структуру (родительский граф)*.

Важно упомянуть понятие *непротиворечивости* АБС. В отличие от непротиворечивости ФЗ, задаваемой однозначно, непротиворечивость АБС бывает четырех видов, но в данной работе достаточно рассмотреть два из них:

- АБС *экстернально непротиворечива*, если каждый ФЗ в ней непротиворечив и оценки вероятности истинности для элементов, входящих в разные ФЗ, совпадают.
- АБС *глобально непротиворечива*, если ее можно погрузить в объемлющий непротиворечивый ФЗ так, что оценки на элементах АБС не изменятся.

Наконец, перейдем к понятию *апостериорного вывода*, который заключается в пересчете оценок на основе поступившего *свидетельства*. Такой вывод (*пропагация свидетельства*) может проводиться как в отдельном ФЗ (*локально*), так на всей АБС (*глобально*). Само *свидетельство* может быть разных видов, нам же важны *стохастические* свидетельства – свидетельства,

являющиеся непротиворечивыми ФЗ со скалярными оценками вероятности истинности. Важно упомянуть, что для пропагации свидетельства в ФЗ необходимо включение алфавита, над которым построено свидетельство, в алфавит, над которым построен ФЗ.

Сравнение будет проходить между двумя алгоритмами, реализующими глобальный апостериорный вывод. Обоим алгоритмам предшествует построение третичной структуры, время чего не учитывалось при сравнении. Первый алгоритм использует вторичную структуру и механизм распространения виртуальных свидетельств [5], перед которым проводилось построение вторичной структуры (алгоритм SinewsStars [9]). Алгоритм, использующий третичную структуру, описан в [6, 7]. Корректность обоих алгоритмов доказана только для ациклических АБС [5].

III. ФОРМИРОВАНИЕ НАБОРА ДАННЫХ

Перед тем, как перейти к статистическому сравнению времени, рассмотрим построение данных, на основе которых будет проводиться это сравнение – построение ациклических АБС.

Начнем с того, что наивный подход к случайному построению АБС (в чем может помочь, например, работа [11]) и последующей проверке ациклическости [6, 12] на практике не оправдал себя – ациклические АБС встречаются достаточно редко и построение необходимого набора моделей занимает значительное время. На основе этого было решено формировать ациклические АБС непосредственно. Помимо этого, было принято решение о построении АБС только со скалярными оценками. Данный выбор позволит сэкономить время на анализе работы алгоритмов, но не даст потерь в корректности сравнения – пропагация свидетельств в сами ФЗ будет производиться в равном количестве в обоих алгоритмах, разницу же в скорости работы может дать размер свидетельств, но она будет заметна и при использовании скалярных оценок. Также стоит отметить о необходимости наличия параметров для настройки генерирующего алгоритма, что позволит формировать структуры с теми или иными свойствами (количество атомов в ФЗ, количество соседей у вершин и т.д.) и анализировать работу алгоритмов пропагации на наборах моделей различного характера.

A. Алгоритм стохастической генерации ациклических АБС

Таким образом, был определен следующий алгоритм генерации ациклической АБС, состоящей из n ФЗ:

1. Формируем начальный ФЗ. Добавляем этот ФЗ в множество S .
 - a. Случайным образом выбираем k_1 атомов, где $1 \leq k_1 \leq \text{Max}_{\text{atom}}$. Количество атомов ограничивается Max_{atom} , так как в теории АБС предполагается, что ФЗ рассматриваются над относительно небольшим набором атомов. При проведении экспериментов Max_{atom} был принят равным 9.
 - b. Формируем оценки вероятности истинности. Для k_1 атомов генерируется $2^{k_1} - 1$ случайное

число в отрезке $[0,1]$. Данные числа разбивают отрезок на 2^k частей (часть может иметь нулевую длину), что дает необходимые оценки вероятности над квантами.

2. Выбираем случайный элемент K из множества S и, далее, формируем для него ФЗ-соседи и кладем их также в S . Формирование ФЗ-соседей:
 - а. Случайным образом генерируем s соседних ФЗ, где $s \geq 1$. При этом учитываем, что общее число ФЗ не должно превосходить n .
 - б. Для каждого ответвления
 - i. Случайным образом ищем k_\cap атомов, которые будут лежать на пересечении с K , где $1 \leq k_\cap \leq k_K - 1$ и k_K – количество атомов в K .
 - ii. Ищем k_{new} – количество новых атомов для ФЗ-соседа, где $1 \leq k_{new} \leq Max_{atom} - k_\cap$.
 - iii. Генерируем оценки вероятности для ФЗ-соседа – по известным оценкам пересечения формируем новые оценки (более подробно будет оговорено далее).
3. Если число ФЗ в АБС меньше n , то переходим к шагу 2, иначе завершаем работу алгоритма.

Теорема 1. Вышеописанный алгоритм генерации ациклической АБС, во-первых, всегда сформирует ациклическую глобально непротиворечивую АБС и, во-вторых, любая ациклическая АБС размера n со скалярными оценками и количеством атомов в ФЗ, не превышающих Max_{atom} , может с некоторой вероятностью быть порождена этим алгоритмом (с точностью до переименования атомов).

Доказательство. Покажем, что алгоритм формирует ациклическую глобально непротиворечивую АБС. Во-первых, будет сгенерировано дерево смежности, так как:

- Будет деревом по построению.
- Если у двух ФЗ есть непустое пересечение, то на единственном пути между ними все промежуточные ФЗ будут содержать это пересечение. Для доказательства этого факта рассмотрим два некоторых ФЗ K_1 и K_2 и какой-либо атом x из их непустого пересечения. Заметим, что по построению существует единственный ФЗ K (может быть один из K_1 или K_2), который является общим предком для K_1 и K_2 . Атом x обязан лежать в K , в противном случае K_1 и K_2 не смогли бы получить его одновременно, так как атомы для нового ФЗ берутся либо из пересечения с ФЗ-соседем, либо совершенно новые. По этой же

причине ФЗ, встречающиеся на путях от K до K_1 и K_2 , будут содержать x . Соответственно все атомы из пересечения будут лежать в ФЗ на пути между K_1 и K_2 .

- Ни один ФЗ не может лежать в другом. Докажем методом от противного. Пусть ФЗ K_1 лежит в ФЗ K_2 . Тогда K_1 лежит в любом ФЗ на пути между K_1 и K_2 согласно предыдущему пункту. В этом случае K_1 лежит в своем ФЗ-соседе K_h . Если формирование сети шло от K_h к K_1 , то возникает противоречие, так как количество новых атомов при формировании ФЗ-соседа не меньше 1 (пункт 2.b.ii). Невозможно также формирование сети от K_1 к K_h , так как алгоритм не позволит взять все атомы из K_1 для пересечения (пункт 2.b.i).

Во-вторых, сеть по построению будет экстернально непротиворечива. Затем, в силу ее ациклическости и наличия скалярных оценок, следует необходимая глобальная непротиворечивость [5].

Далее покажем, что любая ациклическая АБС может быть получена с некоторой долей вероятности. Возьмем некоторую ациклическую АБС N , представленную в виде дерева смежности, и некоторый ее ФЗ K . С точностью до переименования атомов, этот ФЗ может быть сгенерирован на шаге 1. Далее каждый соседний ФЗ K_h для K мог быть сформирован на шаге 2, так как, во-первых, между ним и K будет непустое пересечение (они связаны ребром в дереве смежности), и, во-вторых, атомы в K_h , не лежащие на пересечении, не могли встретиться ранее. Покажем это также методом от противного. Допустим, такие атомы встретились. Тогда существовал бы проходящий через K путь до ФЗ, содержащий эти атомы, поэтому, так как N представлена в виде дерева смежности, ФЗ K также содержал бы эти атомы, что невозможно. Проводя эти же рассуждения для новых формируемых ФЗ получим, что вся сеть может быть построена представленным алгоритмом. *Конец доказательства.*

В. Дополнения к алгоритму

В процессе описания алгоритма возникнет два вопроса, связанных со случайной генерацией, которые необходимо оговорить.

Первый вопрос – формирование случайного натурального числа (количества атомов или соседей). В ходе реализации данный вопрос был решен с применением геометрического распределения, где «неудачей» является формирование нового элемента (атома или нового соседа) с вероятностью q . Данное распределение использовалось с единственной оговоркой – рассматривалась возможность ограничить формируемое число значением сверху (например, Max_{atom}). В случае превышения ограничения проводилась повторная генерация.

Второй вопрос – формирование оценок вероятности истинности для ФЗ на основе известных оценок для части ФЗ (пункт 2.b.iii). Будем рассматривать оценки над квантами. Обозначим кванты итогового ФЗ за q_1, \dots, q_{2^N} , кванты ФЗ, образованного пересечением – $q_1^{\wedge}, \dots, q_{2^n}^{\wedge}$. Кванты $q_1^{\wedge}, \dots, q_{2^n}^{\wedge}$ имеют известную оценку вероятности истинности. Для корректного формирования скалярных оценок необходимо выполнение условий (1).

С учетом введенных ограничений вытекает следующий способ генерации оценок для q_1, \dots, q_{2^N} :

1. Для каждого q_i^{\wedge} формируем 2^{N-n} оценок вероятности $p_1^i, \dots, p_{2^{N-n}}^i$, дающих в сумме 1 (как в пункте 1.b алгоритма генерации ациклической АБС).
2. Каждую оценку p_j^i , полученную в пункте 1 умножаем на $p(q_i^{\wedge})$, что и будет соответствующей оценкой вероятности для q_j т.е. $p(q_j) = p_j^i \cdot p(q_i^{\wedge})$.

С учетом такого построения все необходимые ограничения для оценок q_j будут выполнены.

Отдельно стоит отметить формирование свидетельства. Будем рассматривать стохастические свидетельства, которые полностью лежат в каком-либо ФЗ. Данные свидетельства формировались случайным выбором ФЗ, атомов, из которых будет состоять свидетельство (как в пункте 2.b.i) и оценок вероятности истинности (как в пункте 1.b).

IV. ОПИСАНИЕ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

После определения алгоритма случайного формирования набора данных опишем процесс проведения эксперимента, который выполнялся с использованием языка программирования Python. Прежде всего, необходимо определить формат представления результатов, на основе чего будет сформирован дизайн эксперимента. С предположением о том, что разница времени работы алгоритмов будет увеличиваться с ростом количества ФЗ в АБС, результаты будут представляться в виде графика зависимости статистик от N – количества ФЗ в АБС. В результате рассмотрены N от 5 до 150 с шагом 5. Далее для каждого N формировалась выборка из $M = 100$ пар ациклических АБС и стохастических свидетельств, после чего рассчитывалось отношение времен работы алгоритма, применяющего вторичную структуру к алгоритму, применяющему третичную структуру. С целью уменьшить влияние случайных факторов, которые связаны с работой операционной системы, пропаганда свидетельства для каждого конкретного свидетельства и ациклической алгебраической байесовской сети проводилась $K = 50$ раз, и далее время работы алгоритмов усреднялось. В качестве отображаемых на графиках статистик, после удаления явных выбросов, были выбраны средние по отношениям времен,

полученным ранее, а также доверительный интервал для этого среднего, полученный с помощью метода bootstrap [13]. Помимо этих оценок рассматривалась также медиана, что помогло оценить выбросы, 1 и 3-й квартили, но в целях удобочитаемости графика эти статистики не отображались. Также стоит отметить, что гистограммы распределений отношений времен имели куполообразную форму, а также между векторами времен, полученными при проведении пропагации с использованием вторичной структуры и третичной структуры, был проведен парный t-критерий Стьюдента, по результатам которого отверглась гипотеза о равенстве средних (p-value имел порядок $10^{-15} - 10^{-5}$).

В алгоритме, формирующем ациклические АБС, присутствуют гиперпараметры, которыми будут являться вероятности «неудач» в геометрических распределениях. Экспериментальным путем было установлено, что на результаты существенно влияют лишь комбинация двух параметров – вероятность «неудачи» при формировании количества новых атомов (пункты 1.a и 2.b.ii), а также вероятность «неудачи» при определении количества атомов, лежащих на пересечении (пункт 2.b.i). В итоге, для отображения наиболее различных графиков, были выделены два набора параметров, отличающихся двумя вышеописанными параметрами. В первом наборе значения двух параметров низки (по 0.1), во втором – высоки (по 0.9). В результате во втором наборе данных ФЗ, а также их пересечения, состоят в среднем из большего числа атомов.

Итогом проведения эксперимента являются два графика, изображенных на рис. 1 и рис. 2. График средних по отношениям времен обозначается как “M”, графики доверительных интервалов – как “R-” и “R+”.

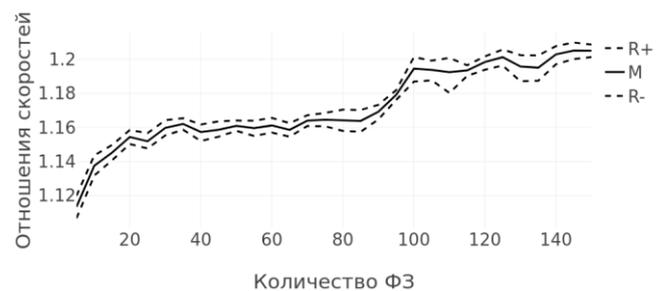


Рис. 1. Первый набор данных – вероятность появления ФЗ из большого числа атомов низка

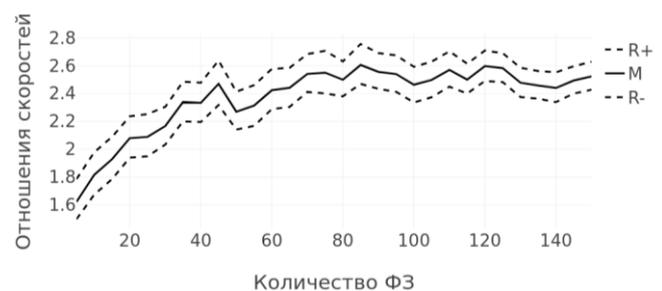


Рис. 2. Второй набор данных – вероятность появления ФЗ из большого числа атомов высока

Из графиков видно, что алгоритм, использующий третичную структуру, работает в целом быстрее, при этом на первом наборе данных ускорение есть, но

незначительное (приблизительно в 1.15 раз), во втором наборе данных ускорение более ощутимо (приблизительно в 2 раза). Более детальный разбор показал, что разница во времени работы в основном обусловлена тем, что при использовании третичной структуры формируются свидетельства меньшего размера, чем при использовании вторичной (где виртуальными свидетельствами будут являться пересечения между ФЗ). При этом на втором наборе данных такая разница заметна сильнее, так как ФЗ, а значит их пересечения, в целом состоят из большего числа атомов.

V. ЗАКЛЮЧЕНИЕ

В данной работе проведено статистическое сравнение времени работы двух алгоритмов глобального апостериорного вывода, один из них использует вторичную структуру, другой – третичную. Помимо этого, был предложен алгоритм стохастической генерации ациклических АБС, на основе чего были сформированы данные для эксперимента. В результате было установлено, что использование алгоритма с третичной структурой более целесообразно с точки зрения времени выполнения (работает быстрее в 1.15–2 раза на АБС, где количество моделей фрагментов знаний равно 5–150). Ускорение тем выше, чем больше атомов в ФЗ и их пересечениях.

В дальнейшем предполагается провести сравнение алгоритмов проверки ациклическости, указанных в работах [6, 12], а также более детальный анализ выбросов, полученных при проведении эксперимента, указанного в статье. В данном случае анализ выбросов, возможно, поможет указать направления для последующей оптимизации алгоритмов. Также результаты статьи будут полезны в контексте прикладного использования АБС, в частности, в сфере оценки защищенности систем от социоинженерных атак [14, 15].

СПИСОК ЛИТЕРАТУРЫ

[1] Tavana M., Abtahi A.-R., Caprio D.D., Poortarigh M. An Artificial Neural Network and Bayesian Network model for liquidity risk assessment in banking // *Neurocomputing*. 2018. Vol. 275. P. 2525–2554. DOI: 10.1016/j.neucom.2017.11.034.

[2] Amin M.T., Khan F., Ahmed S., Imtiaz S. A data-driven Bayesian network learning method for process fault diagnosis // *Process Safety and Environmental Protection*. 2021. Vol. 150. P. 110–122. DOI: 10.1016/j.psep.2021.04.004.

[3] Wu Y., McLeod C., Blyth C., Bowen A., Martin A., Nicholson A., Mascaro S., Snelling T. Predicting the causative pathogen among children with osteomyelitis using Bayesian networks – improving antibiotic selection in clinical practice // *Artificial Intelligence in Medicine*. 2020. Vol. 107. p. 101895.

[4] De Iulius M., Kammouh O., Cimellaro G.P., Tesfamariam S. Quantifying restoration time of power and telecommunication lifelines after earthquakes using Bayesian belief network model // *Reliability Engineering & System Safety*. 2021. Vol. 208. p.107320.

[5] Тулупьев А.Л. Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности. СПб.: ООО Издательство «Анатолия», 2007. 40 с.

[6] Вяткин А.А., Абрамов М.В., Харитонов Н.А., Тулупьев А.Л. Применение третичной структуры алгебраической байесовской сети в задаче апостериорного вывода // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2023. (В печати).

[7] Фроленков К.В., Фильченков А.А., Тулупьев А.Л. Апостериорный вывод в третичной полиструктуре алгебраической байесовской сети // *Информатика и автоматизация*. 2012. № 23. С. 343–356. DOI: 10.15622/sp.23.17.

[8] Тулупьев А.Л. Алгебраические байесовские сети: локальный логико-вероятностный вывод. СПб.: ООО Издательство «Анатолия», 2007. 80 с.

[9] Фильченков А.А., Фроленков К.В., Сироткин А.В., Тулупьев А.Л. Система алгоритмов синтеза подмножеств минимальных графов смежности // *Информатика и автоматизация*. 2013. № 27. С. 200–244. DOI: 10.15622/sp.27.17.

[10] Aho A., Garey M., Ullman J. The Transitive Reduction of a Directed Graph // *SIAM Journal on Computing*. 1972. Vol. 1, № 2. P. 131–137. DOI: 10.1137/0201008.

[11] Зотов М.А., Левенец Д.Г., Тулупьев А.Л., Золотин А.А. Синтез вторичной структуры алгебраических байесовских сетей: инкрементальный алгоритм и статистическая оценка его сложности // *Научно-технический вестник информационных технологий, механики и оптики*. 2016. Т. 16. № 1. С. 122–132.

[12] Фильченков А.А., Тулупьев А.Л. Алгоритм выявления ациклическости первичной структуры алгебраической байесовской сети по ее четвертичной структуре // *Информатика и автоматизация*. 2011. № 19. С. 128–145. DOI: 10.15622/sp.19.7.

[13] Efron B. *Bootstrap methods: another look at the jackknife*. Springer New York, 1992. pp. 569–593.

[14] Khlobystova A.O., Abramov M.V. The models separation of access rights of users to critical documents of information system as factor of reduce impact of successful social engineering attacks // *Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 8th International Conference on “Fuzzy Systems, Soft Computing and Intelligent Technologies (FSSCIT 2020)”*. Smolensk, Russia, June 29–July 1, 2020. P. 264–268.

[15] Korepanova A.A., Oliseenko V.D., Abramov M. V. Applicability of Similarity Coefficients in Social Circle Matching // *2020 XXIII International Conference on Soft Computing and Measurements (SCM)*. St. Petersburg, Russia. 2020. P. 41–43. DOI: 10.1109/SCM50615.2020.9198782.