

8. *Biere A., Heule V., van Maaren H, and Walsh T.* Handbook of Satisfiability. Amsterdam: IOS Press, 2009.
9. *Богачкова И. А., Заикин О. С., Кочемазов С. Е., Отпущенников И. В., Семёнов А. А.* Задачи поиска коллизий для криптографических хеш-функций семейства MD как варианты задачи о булевой выполнимости // Вычислительные методы и программирование. 2015. Т. 16. С. 61–77.
10. *Отпущенников И. В., Семёнов А. А.* Технология трансляции комбинаторных проблем в булевы уравнения // Прикладная дискретная математика. 2011. № 1. С. 96–115.
11. *De D., Kumarasubramanian A., and Venkatesan R.* Inversion attacks on secure hash functions using SAT solvers // LNCS. 2007. V. 4501. P. 377–382.
12. *Dobbertin H.* The first two rounds of MD4 are not One-Way // Proc. 5th Intern. Workshop Fast Software Encryption. London, UK: Springer Verlag, 1998. P. 284–292.
13. *Nadel A. and Ryvchin V.* Efficient SAT solving under assumptions // LNCS. 2012. V. 7317. P. 242–255.

УДК 519.178

DOI 10.17223/2226308X/8/55

ВЫЧИСЛЕНИЕ ВЕРХНЕЙ ОЦЕНКИ ВЕРШИННОЙ ЦЕЛОСТНОСТИ ГРАФА НА ОСНОВЕ МИНИМАЛЬНЫХ СЕПАРАТОРОВ

В. В. Быкова, Ю. И. Кириллов

Рассматривается трудно вычисляемый числовой параметр графа, называемый вершинной целостностью и используемый в анализе и синтезе отказоустойчивых сложных технических систем. Для нахождения данного параметра необходимо знание всех сепараторов исходного графа. Предлагается алгоритм, который ограничивается построением и анализом только всех минимальных сепараторов. Поэтому алгоритм даёт верхнюю оценку вершинной целостности графа. Вычислительная сложность предлагаемого алгоритма полиномиально зависит от числа вершин и числа минимальных сепараторов графа. Результаты экспериментов показали, что вычисленные оценки являются хорошими и часто достижимыми.

Ключевые слова: *алгоритмы на графах, вершинная целостность графа, минимальные сепараторы.*

Все последние десятилетия проблеме исследования мер целостности графов уделяется особое внимание, поскольку она тесно связана с вопросами анализа и синтеза сложных технических систем, для которых отказоустойчивость является важнейшим показателем качества функционирования. Вершинная целостность — одна из детерминированных мер целостности графа [1].

Пусть $G = (V, E)$ — простой связный граф с множеством вершин V и множеством рёбер E , при этом $n = |V| \geq 1$ — порядок графа G . Под вершинной целостностью (*Vertex Integrity*) графа G понимается числовой параметр $I(G)$, вычисляемый по формуле

$$I(G) = \min_{S \subseteq V} \{|S| + w(G - S)\}, \quad (1)$$

где $w(G - S)$ — порядок наибольшей компоненты связности графа $G - S$, который получается из G удалением всех вершин, входящих в $S \subseteq V$. Множество S , для которого достигается равенство в (1), принято называть I -множеством. В неполном связном графе G всякое I -множество является сепаратором этого графа [2]. Доказано [3], что задача вычисления вершинной целостности графа NP-трудная. Ввиду высокой вы-

числительной сложности нахождения точных значений вершинной целостности произвольного графа актуальны нижние и верхние оценки для $I(G)$.

Множество всех вершин графа $G = (V, E)$, смежных с некоторой вершиной $v \in V$, образует в G окрестность вершины v , которая обозначается через $N(v)$. Под замкнутой окрестностью вершины v понимается множество $N[v] = N(v) \cup \{v\}$. Множество вершин $N(S) = \left(\bigcup_{v \in S} N(v) \right) \setminus S$ называется окрестностью для $S \subseteq V$ в G . Множество вершин $S \subseteq V$ разделяет несмежные вершины u и v графа G , если в графе $G - S$ вершины u и v принадлежат различным компонентам связности. Множество S при этом называют (u, v) -сепаратором; (u, v) -сепаратор называют минимальным, если в нём нет собственного подмножества, являющегося (u, v) -сепаратором. Сепаратор S минимальный, если в G существует такая пара вершин u и v , что S является минимальным (u, v) -сепаратором, то есть минимальным относительно, по крайней мере, двух вершин этого графа. Известно, что S — минимальный сепаратор графа G , если и только если существуют две различные области связности C_1 и C_2 графа $G - S$, такие, что $N(C_1) = N(C_2) = S$ [4]. Если S является минимальным (u, v) -сепаратором и $S \subseteq N(v)$, то S называется минимальным сепаратором, близким к v . Обозначим через M множество всех минимальных сепараторов графа $G = (V, E)$. В общем случае множество M может содержать экспоненциальное число сепараторов.

В настоящей работе предлагается алгоритм, позволяющий с помощью минимальных сепараторов вычислять верхнюю оценку меры вершинной целостности и находить соответствующие приближения к I -множествам заданного графа за время $O(n^3|M|)$. Поиск всех минимальных сепараторов осуществляется по следующему алгоритму.

Алгоритм. Поиск минимальных сепараторов

Вход: Граф $G(V, E)$

Выход: M — множество всех минимальных сепараторов графа G

1. $M := \emptyset$
 2. Для каждого $v \in V$
 3. $H := G - N[v]$
 4. Для каждой области связности C графа H
 5. $S := N(v) \cap N(C)$
 6. Если $S \neq \emptyset$, то $M := M \cup \{S\}$
 7. Для каждого $S \in M$
 8. Для каждого $x \in S$
 9. $X := S \cup N(x)$
 10. $H := G - X$
 11. Для каждой области связности C графа H
 12. $S := N(C) \cap X$
 13. Если $S \neq \emptyset$, то $M := M \cup \{S\}$
-

Каждый из найденных минимальных сепараторов подставляется в выражение $|S| + w(G - S)$. Минимальное полученное значение I^* является верхней оценкой $I(G)$.

Программный код алгоритма написан на языке C++ в среде разработки Code::Blocks. Вычисленное значение верхней оценки I^* и время работы t_a предложенного алгоритма сравнивались с точным значением вершинной целостности графа, полученным методом полного перебора за время t . Вычислительный эксперимент был проведён на случайных связных графах различных порядков; результаты эксперимента для $n = 20$ представлены в таблице.

№ графа	t	I	t_a	I^*	$I^* - I$	t/t_a
1	31,922	15	0,066	16	1	483,6667
2	27,653	16	0,638	16	0	43,34326
3	27,634	15	3,636	15	0	7,60011
4	25,34	16	16,881	16	0	1,501096
5	25,95	15	0,05	16	1	519
6	26,487	15	0,624	15	0	42,44712
7	26,717	15	5,793	15	0	4,611945
8	26,04	15	0,05	15	0	520,8
9	24,56	15	0,03	16	1	818,6667
10	23,096	14	0,587	14	0	39,34583

Алгоритм был также опробован на графах с известным типом структур (на двумерных решётках, на графах Гретша, Хватала, Хивуда и др.). Результаты экспериментов позволяют говорить о том, что вычисленные значения оценок являются хорошими и часто достижимыми. При этом время нахождения оценки существенно меньше, чем при использовании метода полного перебора. Однако было замечено, что алгоритм даёт большую ошибку на графах с регулярной структурой (например, на двумерной решётке).

ЛИТЕРАТУРА

1. *Быкова В. В.* О мерах целостности графа // Прикладная дискретная математика. 2014. № 4(26). С. 96–111.
2. *Barefoot C. A., Entringer R., and Swart H. C.* Vulnerability in graphs — a comparative survey // J. Combin. Math. Combin. Comput. 1987. No. 1. P. 13–22.
3. *Clark L. H., Entringer R. C., and Fellows M. R.* Computational complexity of integrity // J. Combin. Math. Combin. Comput. 1987. No. 2. P. 179–191.
4. *Berry A. and Bordat J.-P.* Structuring the minimal separators of an undirected graphs. Technical Report 152, LIM, Marseille, 1996.

УДК 681.5.015

DOI 10.17223/2226308X/8/56

ПОСТРОЕНИЕ ФУНКЦИИ ОШИБКИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ИДЕНТИФИКАЦИИ АЛГОРИТМА РАНЖИРОВАНИЯ

О. А. Кожушко

Предлагается функция ошибки для постановки задачи идентификации алгоритма ранжирования результатов текстового поиска. Приводится обоснование некорректности применения функций ошибки, используемых в задачах, где выходные значения принимают действительные значения. В качестве альтернативы предлагается функция ошибки, которая учитывает не абсолютное, а относительное изменение результатов ранжирования. Приводится частный случай постановки задачи идентификации, в которой результат ранжирования рассматривается как класс релевантности.

Ключевые слова: алгоритм ранжирования, идентификация системы, функция ошибки.

Задача идентификации системы подразумевает построение модели, устанавливающей взаимосвязь между входными и выходными значениями данной системы, и в общем виде ставится следующим образом [1].

Пусть исходная система реализует неизвестное отображение $F : DX \rightarrow DY$. Необходимо построить отображение $M_F : DX \rightarrow DY$ таким образом, что $M_F(x) = F(x)$