УДК 519.682.1

**С.В. Листровой**, д-р техн. наук
Украинская государственная академия
железнодорожного транспорта
(Украина,61050, Харьков, пл. Фейрбаха, 7,
тел. 0509355042, e-mail: om1@yandex.ru)

# Метод перечисления максимальных независимых множеств в произвольных неориентированных графах

Предложена процедура перечисления только максимальных независимых множеств в неориентированных произвольных графах, позволяющая уменьшить временную сложность реализации алгоритма.

Запропоновано процедуру перелічування тільки максимальних незалежних множин у неорієнтованих довільних графах, яка дозволяє зменшити часову складність реалізації алгоритму.

*Ключевые слова: максимальные независимые множества, клики.*

Во многих прикладных задачах синтеза и анализа вычислительных систем и сетей, а также при разработке специального математического обеспечения для их функционирования требуется найти в конечном множестве объектов максимальную систему объектов, попарно не связанных друг с другом, или выбрать минимальную систему объектов, связанных со всеми другими. Формулировки подобных задач на языке теории графов приводят к понятиям независимости и покрытия.

Множество вершин $U$ графа $G$ ($V$, $E$) называется независимым (внутренне устойчивым), если никакие две вершины из этого множества не смежные, т.е. если $U \subseteq V$ и $U$ независимо в графе $G$, то порожденный подграф $G$ ($U$) является пустым. Очевидно, что если при этом $U^* \subseteq U$, то $U^*$ — также независимое множество.

Внутренне устойчивое множество называется максимальным, если оно не является собственным подмножеством некоторого другого независимого множества.

Наибольшее по мощности независимое множество называется наибольшим. Число вершин в наибольшем независимом множестве графа

$G(V, E)$ называется числом независимости, или числом внутренней устойчивости, и обозначается через $\alpha_0(G)$. Ясно, что наибольшее независимое множество является максимальным.

Нетрудно видеть, что задача о клике и задача о независимом множестве, по сути, эквивалентны: каждая из них вытекает из другой при построении дополнения графа $G(V, E)$, т.е. такого графа $\overline{G}(V, E^1)$, в котором есть все вершины исходного. При этом в дополнении графа вершины соединены ребром только в том случае, если они не были соединены в исходном графе. Число вершин в наибольшей клике называют также плотностью $\varphi(G)$ графа $G(V, E)$. Очевидно, что $\varepsilon(G) = \varphi(\overline{G})$ и $\varphi(G) = \varepsilon(\overline{G})$.

Обе задачи тесно связаны с задачей о минимальном вершинном покрытии. Число вершин в наименьшем покрытии графа $G(V, E)$ называется числом вершинного покрытия $\beta_0(G)$ и связано с $\alpha_0(G)$ следующим равенством: $\alpha_0(G) + \beta_0(G) = n$, где $n$ — число вершин в графе $G(V, E)$. Один из способов нахождения независимого множества вершин графа $G(V, E)$ состоит в построении всех максимальных независимых множеств (МНМ) и в выборе из них множества с наибольшей мощностью.

Одним из широко используемых алгоритмов для определения МНМ является алгоритм Брона—Кербоша, используемый для нахождения МНМ вершин, если построено дополнение к исходному графу [1, 2]. Этот алгоритм основан на том, что всякая клика в графе является его максимальным по включению полным подграфом. Начиная с одиночной вершины (образующей полный подграф) на каждом шаге алгоритма происходит увеличение уже построенного полного подграфа добавлением вершины из множества кандидатов. Отсечение неперспективных вариантов, заведомо не приводящих к построению клики, обеспечивается использованием дополнительного множества, в которое помещаются вершины, уже использованные для увеличения полного подграфа.

Алгоритм оперирует тремя множествами вершин графа: $Q$-множеством, содержащим на каждом шаге рекурсии полный подграф для данного шага (строится рекурсивно); множеством вершин $H$, которые могут увеличить множество $Q$, множеством вершин $N$, которые уже были использованы для расширения $Q$ на предыдущих шагах. Алгоритм является рекурсивной процедурой, применяемой к этим трем множествам, и его сложность линейна относительно числа клик в графе, где $n$ — число вершин, $m$ — число ребер, $\mu$ — число клик. В работе [1] показано, что в худшем случае временная сложность алгоритма не превышает $\boldsymbol{O}(3^{n/3})$. Все известные алгоритмы определения МНМ имеют экспоненциальную сложность. Разработанная процедура позволяет уменьшить временную сложность алгоритма перечисления МНМ в произвольных неориентированных графах.

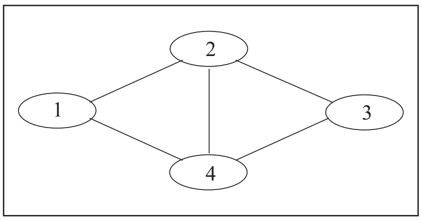


*Рис. 1.* Граф *G*

**Формализация и решение задачи.** Пусть задан исходный граф $G(V, E)$ с $n$ вершинами и требуется перечислить в нем все МНМ. Важное свойство графов, которое понадобится для решения поставленной задачи, состоит в том, что подмножество $X \subseteq V$ является МНМ графа $G(V, E)$ тогда и только тогда, когда его дополнение $Y$ является вершинным покрытием (т.е. $X = V - Y$). Ясно, что если в графе $G(V, E)$ есть вершина $i$ со степенью, равной $n-1$, то вершина $i$ является МНМ.

В работе [3] показано, что если $f$-булева функция, построена по графу $G(V, E)$ в виде произведения дизъюнктов $(V_i \vee V_j)$, где $\{V_i\} \in \{0, 1\}$, $i = \overline{1, n}$, $j = \overline{1, n}$, $i \neq j$, и при этом каждый дизъюнкт $(V_i \vee V_j)$ соответствует ребру $(V_i, V_j)$, то все наборы переменных $\{V_i, V_j\}$, на которых она принимает значение истинно, соответствуют вершинным покрытиям в графе $G(V, E)$. Следовательно, для перечисления всех вершинных покрытий графа $G(V, E)$ необходимо определить те системы значений переменных $\{V_i, V_j\}$, при которых высказывание

$$f(V_1, V_2, ..., V_n) = 1 \qquad (1)$$

есть истинно. Для того чтобы найти эти системы значений переменных $\{v_i, v_j\}$, необходимо привести левую часть (1) к минимальной дизъюнктивной нормальной форме, раскрывая скобки и пользуясь законом поглощения. Такая форма единственна ввиду отсутствия в (1) логических отрицаний. Покажем это на примере графа $G$ ( рис. 1), булева функция которого имеет вид

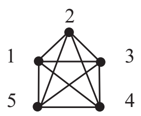
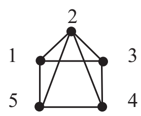
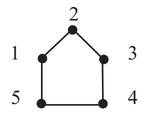
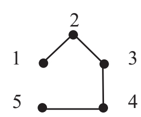
$$f = (V_1 \vee V_2)(V_2 \vee V_3)(V_3 \vee V_4)(V_2 \vee V_4)(V_1 \vee V_4) =$$
$$= (V_2 \vee V_1V_3)(V_4 \vee V_1V_2V_3) = V_2V_4 \vee V_1V_2V_3 \vee V_1V_3V_4\,. \qquad (2)$$

Как видно из (2), в результате раскрытия скобок и приведения подобных получен полный перечень вершинных покрытий графа $G$. Ими являются подмножества вершин $\{2, 4\}$, $\{1, 2, 3\}$, $\{1, 3, 4\}$. Соответственно их дополнения образуют МНМ графа. В данном случае такими являются подмножества $\{1, 3\}$, $\{4\}$, $\{2\}$. Множество покрытий, полученных после приведения подобных на основе операции поглощения, будем называть множеством непоглощаемых вершинных покрытий, т.е. покрытий, не содержащихся ни в каких других покрытиях. Обозначив число непоглощаемых вершинных покрытий в графе через $\gamma$, а число МНМ через $\lambda$, убедимся, что справедливо равенство $\gamma = \lambda$.

Для графа $G(V, E)$ и его дополнения $G^1(V, E^1)$ справедливо равенство $2E + 2E^1 = n(n-1)$, из которого видно, что число пар независимых вершин в графе $G(V, E)$ не может превысить числа ребер в графе $G^1$. Если независимые пары вершин объединять в независимые множества и далее пытаться наращивать мощность независимых множеств, то число таких объединений будет ограничено наличием связей между вершинами в графе $G(V, E)$, т.е. множеством ребер $E$ в графе $G(V, E)$, которое в произвольном графе не может превысить $E_{\max} = \dfrac{n(n-1)}{2}$. При этом следует учесть, что число МНМ $\lambda$ определяется числом $\gamma$ непоглощаемых вершинных покрытий в графе. Ясно, что число $\gamma$ в графе пропорционально удвоенному числу ребер $2E$ в графе $G(V, E)$, поскольку каждое ребро может быть покрыто двумя вершинами. Следовательно, число $\lambda^* = Cn(n-1)$ является оценкой сверху числа МНМ в графе $G(V, E)$, где $C$ — некоторая константа, возможно, достаточно большая.

В полном графе $\gamma = n$, а если граф представляет дерево, то число $\gamma$ может приближаться к $Cn(n-1)$. Данная оценка является весьма грубой. В табл. 1 представлены графы, полученные из первого графа посредством удаления ребер. При этом число непоглощаемых вершинных покрытий $\gamma$ и число МНМ определены согласно соотношению (1).

*Таблица 1*

| Граф | Вершинное покрытие | МНМ | $\gamma$ | $\lambda$ | $m$ |
|---|---|---|---|---|---|
|  | 2345 v 1345 v 1245 v 1235 v 1234 | 1 v 2 v 3 v 4 v 5 | –5 | 5 | 10 |
|  | 235 v 124 v 1345 | 14 v 35 v 2 | 3 | 3 | 8 |
|  | 124 v 134 v 135 v 235 v 245 | 35 v 25 v 24 v 14 v 13 | –5 | 5 | 5 |
|  | 24 v 235 v 134 v 135 | 135 v 14 v 25 v 24 | –4 | 4 | 4 |

Как видно из табл. 1, удаление ребер может приводить к уменьшению числа $\gamma$ и, следовательно, числа $\lambda$, но оно может и возрастать с увеличением числа вершин в графе.

Следует заметить, что число различных независимых множеств и клик в произвольном графе может быть экспоненциально большим. Так, в работах [1, 2] показано, что наибольшее число возможных клик в графе с $n$ вершинами может достигать значения $3^{n/3}$, т.е следует различать МНМ, число которых не превышает $\lambda^*$, и произвольные независимые множества, число которых существенно больше.

Общим недостатком всех известных процедур поиска МНМ и максимальных клик, аналогичных процедуре, используемой в алгоритме Брона—Кербоша, является попытка построить наибольшее МНМ посредством перечисления всех независимых множеств и затем выделить в них максимальное. При этом организация поиска происходит в ширину или в глубину в дереве вариантов с перебиранием всех независимых множеств. При таком поиске число шагов, выполняемых процедурой, является экспоненциально большим, и только после окончания неявного полного перебора вариантов поиска может быть выделено наибольшее МНМ.

Рассмотрим подход, позволяющий преодолеть данный недостаток. Такой подход основан на построении процедуры, перечисляющей только все МНМ, число которых не превышает $\lambda^*$. Построение такой процедуры возможно на основе составления последовательности графов $G^1, G^2, \ldots, G^p$, где граф $G^1$ построен из исходного графа $G(V, E)$ посредством соединения ребрами вершин, которые в графе $G(V, E)$ не связаны между собой. Затем объединяем вершины в подмножества, которые в графе $G^1$ соединены ребрами. Вершины $i$ в графах $G^2, ..., G^p$ будем характеризовать двумя подмножествами: подмножеством $X_i^r$ из $r$ независимых вершин и подмножеством $Y_i^k$, состоящим из $k$ вершин, с которыми связаны вершины из $X_i^r$.

Формирование графов $G^i$ осуществляется до тех пор, пока для всех сформированных множеств, соответствующих вершинам графа $G^i$, не будет выполнено условие $X_i^r \cup Y_i^k = V$, которое означает, что сформированное множество является максимально независимым, поскольку в этом случае подмножество $Y_i^k$ будет представлять вершинное покрытие в графе $G(V, E)$, а подмножество $X_i^r$ — МНМ вершин графа $G(V, E)$, дополняющее это подмножество до $n$.

Если $X_i^r \cup Y_i^k \neq V$, то, очевидно, существуют вершины, которые можно добавить в подмножество $X_i^r$, и оно останется при этом независимым. Число таких вершин не превосходит $n - r$. Если $X_i^r \cup Y_i^k = V$, то таких вершин в графе $G(V, E)$ не существует.

В текущем графе $G^{\,i}$, если вершины ($i, j$ ... $p$) описываются подмножествами $(X_i^r, Y_i^k), (X_j^r, Y_j^k) \ldots (X_p^r, Y_p^k)$, у которых $Y_i^k = Y_j^k = \ldots = Y_p^k = Y$ могут быть объединены в одну вершину, описываемую подмножествами $(X_{\mu i}^r, Y)$, где $X_{\mu i}^r = X_i^r \cup X_j^r \cup \ldots \cup X_p^r$, и при этом если $X_{\mu i}^r \cup Y = V$,то $X_{\mu i}^r$ — МНМ. Вершины в графе $G^{\,i}$ соединяются ребром ($i, j$), если объединения $(X_i^r \cup X_j^r)$ являются независимыми множествами. Процесс формирования графа $G^{\,i}$ прекращается, когда все его вершины удовлетворяют условию $X_i^r \cup Y_i^k = V$ или дальнейшее объединение невозможно.

При завершении работы процедуры может возникать зацикливание, заключающееся в том, что все МНМ уже построены и процедура добавляет дублирующее МНМ. Поэтому необходимо проверять, изменился ли список сформированных МНМ по сравнению с предыдущим шагом. Если список не изменился, то процедура заканчивает работу. Следует заметить, что в процессе формирования графов $G^i$, могут возникать одинаковые вершины. При этом дублирующие вершины необходимо удалять.

Введем процедуру $\boldsymbol{A}$ формирования подмножеств независимых вершин, позволяющую их перечислить.

**Процедура *A*.**

Ш а г 1. Строим граф $G^{\,i=1}$, являющийся дополнением исходного графа до полного. Формируем вершины графа $G^{\,i=2}$ (объединяя вершины, соединенные ребрами в графе $G^{\,i=1}$ в подмножества $(X_i^r)$ мощности 2, число которых равно числу ребер в графе $G^{\,i=1}$) и для каждой сформированной вершины определяем подмножества вершин $(Y_i^k)$, с которыми связаны вершины из подмножеств $(X_i^r)$, в графе $G(V, E)$.

Ш а г 2. Проверяем, есть ли вершины с одинаковыми подмножествами $Y_i^k$. Если есть, то объединяем соответствующие $X_i^r$ в одну вершину и проверяем, есть ли вершины, для которых выполняется условие $X_i^r \cup Y_i^k = V$. Если такие вершины есть, то подмножества $X_i^r$, соответствующие данным вершинам, являются МНМ. Запоминаем их и исключаем эти вершины из дальнейшего анализа, иначе — переходим к выполнению следующего шага.

Ш а г 3. Объединяем сформированные вершины ребрами ($i, j$), если объединения $(X_i^r \cup X_j^r)$ являются независимыми множествами, и проверяем, есть ли вершины, в которых подмножества $X_i^r$ не являются максимальными множествами. Если нет, то среди сформированных множеств удаляем дублирующие и процедура заканчивает работу, поскольку все МНМ построены. Иначе — проверяем, произошло добавление новых множеств в сформированный список МНМ или нет. Если нет, то процедура заканчивает работу, если добавление произошло, то выполняем следующий шаг.

Ш а г 4. Формируем вершины графа $G^{\,i:=i+1}$, объединяя вершины, соединенные ребрами на предыдущем шаге, и для каждой сформирован-

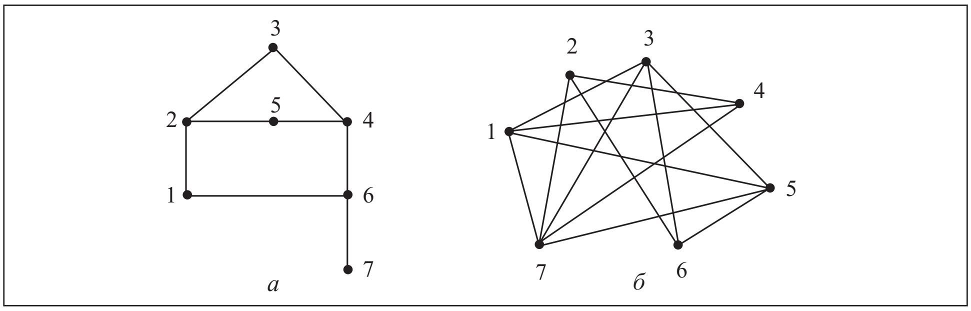



*Рис. 2.* Исходный граф $G\ (V,\ E\ )$ (*а*) и граф $G^1(V,\ E^1\ )$ — дополнение графа $G(V,\ E\ )$ до полного графа (*б*)

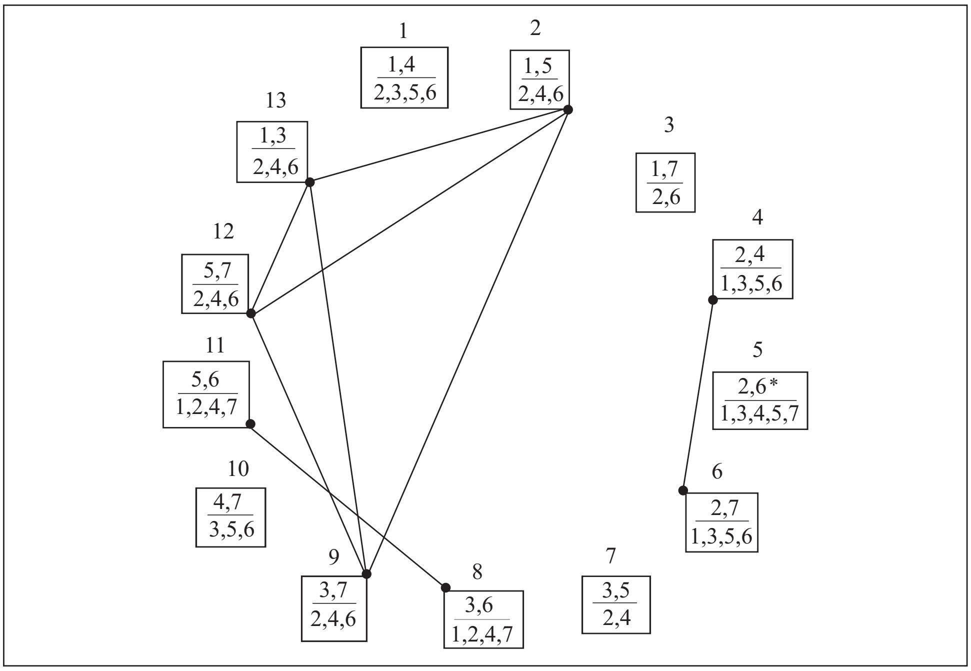



*Рис. 3.* Процесс формирования графа $G^2(V^2,\ E^2\ )$

ной вершины определяем подмножества вершин ($Y_i^k$), с которыми связаны вершины из подмножеств ($X_i^r$), в графе $G(V,\ E)$. После этого переходим к выполнению шага 2.

Рассмотрим пример работы процедуры $\boldsymbol{A}$ для графа $G(V,E)$ приведенного на рис. 2, *а*. В графе $G^1(V,\ E')$ вершины $u$ и $v$ соединены ребрами, если в графе $G(V,\ E)$ эти вершины не связаны. Вершины в графе $G^2(V^2,\ E^2)$ обозначены квадратами (рис. 3). В верхней части квадрата записаны объе-

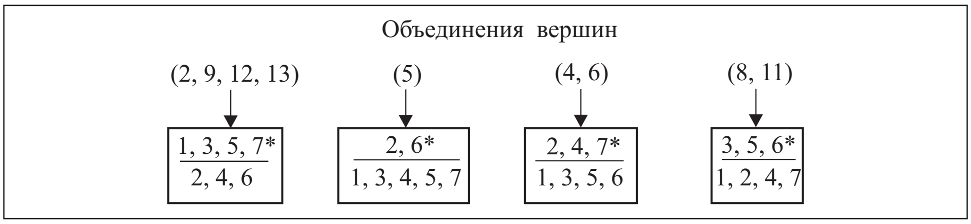



*Рис. 4.* Результат объединения вершин с одинаковыми значениями $Y_i^2$

динения вершин, соединенные ребрами в графе $G^{i=1}$ в подмножества $(X_i^r)$ мощности 2, а в нижней части — подмножества вершин $(Y_i^k)$, с которыми связаны вершины из подмножеств $(X_i^r)$ в графе $G(V, E)$.

Например, вершина 1 в графе $G(V, E)$ соответствует объединению вершин 1 и 4 в графе $G^{i=1}$, которые соединены ребром. В графе $G(V, E)$ вершина 1 связана с вершинами 2 и 6, а вершина 4 с вершинами $\{3, 5, 6\}$, т.е. множество вершин 1, 4 связано с множеством вершин $\{2, 3, 5, 6\}$. Звездочками на рис. 4 помечены те вершины, в которых сформированные подмножества вершин являются МНМ. Так, вершина 5 соответствует МНМ, поскольку объединение подмножеств имеет вид $(X_5^2 \cup Y_2^5) = \{2, 6\} \cup \{1, 3, 4, 5, 7\} = \{1, 2, 3, 4, 5, 6, 7\} = V$, т.е. образует множе`ство вершин $V$ графа $G(V, E)$.

Вершины $\{2, 9, 12, 13\}$ графа $G^2(V^2, E^2)$ связаны с одним и тем же множеством вершин в графе $G(V, E)$, т.е. $Y_2^2 = Y_9^2 = Y_{12}^2 = Y_{13}^2 = \{2, 4, 6\}$. Эти вершины соединены между собой ребрами, аналогично соединены ребром вершины 4 и 6, так как $Y_4^2 = Y_6^2 = \{1, 3, 5, 6\}$ и вершины 8 и 11 у них следующие: $Y_8^2 = Y_{11}^2 = \{1, 2, 4, 7\}$.

После объединения вершин с одинаковыми подмножествами $Y_i^2$ получим вершины, соответствующие МНМ, так как объединения соответствующих подмножеств $X_i^r \cup Y_i^r$ в этих вершинах образуют множество вершин $V$ в графе $G(V, E)$. Запоминаем эти множества, а соответствующие им вершины исключаем из дальнейшего анализа. Оставшиеся вершины графа соединяем, если объединения $(X_i^r \cup X_j^r)$ являются независимыми множествами, и получаем граф $G^2(V^2, E^2)$ (рис. 5).

Объединения подмножества в соответствии с графом $G^2(V^2, E^2)$ начинаем формировать с вершины графа $G^3(V^3, E^3)$ (рис. 6). Поскольку все сформированные вершины соответствуют МНМ, процедура заканчивает работу и удаляются дублирующие вершины. Так, из объединений вершин (1, 3), (1, 10), (3, 10) оставляем только одно.

Таким образом, с помощью процедуры $\boldsymbol{A}$ построено пять МНМ: $\{1, 3, 5, 7\}$; $\{1, 4, 7\}$; $\{3, 5, 6\}$; $\{2, 4, 7\}$; $\{2, 6\}$. При этом суммарное число вершин, которое пришлось построить при формировании графов $G^2(V^2, E^2)$,

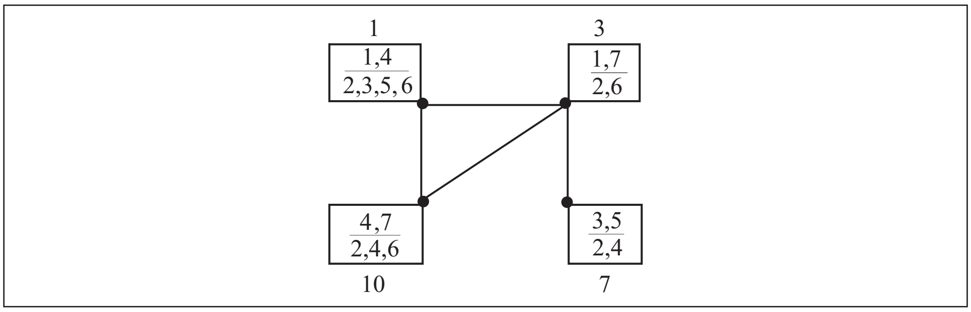



*Рис. 5.* Граф $G^2(V^2,\ E^2)$

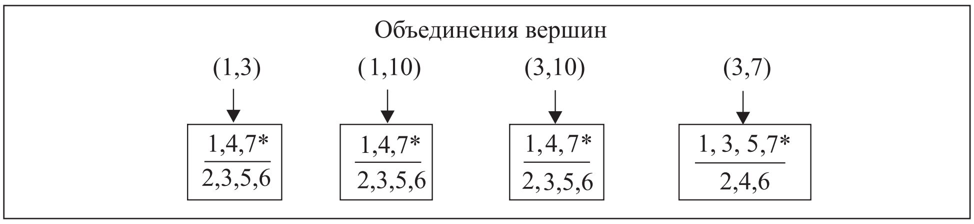



*Рис. 6.* Процесс формирования графа $G^3(V^3,\ E^3)$

$G^3(V^3,\ E^3)$, не превысило числа ребер в графе $G^1(V, E^1)$. Графы $G^1$ и $G(V,\ E)$ являются исходными данными для работы процедуры ***A***.

Для того чтобы проверить правильность результата, представим исходный граф $G(V, E)$ в виде булевой функции:

$$f = (1\vee 2)(1\vee 6)(2\vee 3)(2\vee 5)(3\vee 4)(4\vee 5)(4\vee 6)(6\vee 7)\,.$$

Проведя объединение по 2, 4 и 6, получим $f = (2\vee 135)(4\vee 356)(6\vee 17)$. Далее, раскрывая скобки, приводя подобные и используя форму поглощения, получаем перечень непоглощаемых вершинных покрытий графа $G\,(V,\ E)$, приведенного на рис. 2:

$$f = 1247\vee 246\vee 2356\vee 1356\vee 13457. \qquad (3)$$

Поскольку МНМ являются дополнениями вершинных покрытий, на основании (3) получаем перечень МНМ $f^{\,1} = 356\vee 1357\vee 147\vee 247\vee 26$, т.е. результат идентичен результату процедуры ***A***. В процессе работы процедуры ***A*** исключаются из анализа только независимые подмножества вершин, дублирующие уже сформированные МНМ вершин, т.е. перечисляются все МНМ в исследуемом графе.

Оценим временную сложность работы процедуры ***A***. В графе $G^{\,i=1}$ число вершин равно $n$, а число ребер не может превышать $E^1 = \dfrac{n\,(n-1)}{2} - E$.

В графе $G^{i=2}$ число вершин не может превысить $E^1 = \frac{n(n-1)}{2} - E < \frac{n(n-1)}{2}$. Поскольку число МНМ пропорционально $n(n-1)$, общее число вершин, сформированных в процессе формирования одного из графов $G^i$, не может превысить $\frac{n^2(n-1)^2}{2}$ при плотностях ребер, превышающих 0,5. При формировании графов $G^i$ происходит процесс удвоения числа вершин в подмножествах, соответствующих вершинам графа, который можно описать последовательностью

$$2^1 + 2^2 + 2^4 + 2^8 + \ldots + 2^p. \qquad (4)$$

Поскольку число вершин в исходном графе равно $n$, процесс (4) может быть завершен, когда $2^p = n$, откуда $p = \log_2 n$. Ясно, что справедливо неравенство

$$2^1 + 2^2 + 2^4 + 2^8 + \ldots + 2^p < \log_2 n \, 2^{\log_2 n} = n \log_2 n.$$

Следовательно, число графов $G^i$, которое придется построить при реализации процедуры $\boldsymbol{A}$, не может превысить $n \log_2 n$. При этом общее число вершин, которое придется построить, не превысит $\frac{n^2(n-1)^2}{2} n \log_2 n$. Поскольку на формирование одной вершины графа требуется не более $2n^2$ операций сравнения, общее число операций, затрачиваемое процедурой $\boldsymbol{A}$ на формирование всех МНМ, не может превысить $n^5(n-1)^2 \log_2 n \approx n^7 \log_2 n$. Таким образом, временная сложность работы процедуры в худшем случае при плотностях ребер в графе $\rho > 0{,}5$ не превысит $\boldsymbol{O}(n^7 \log_2 n)$.

Данная оценка является оценкой сверху, поэтому представляет интерес получить оценку временной сложности данной процедуры в среднем при различных плотностях ребер в графе. Для этого было проведено экспериментальное исследование:

а) временной сложности работы процедуры $\boldsymbol{A}$ при различных плотностях ребер в графах $G(V, E)$;

б) зависимости числа МНМ от плотности ребер в графе при фиксированном значении числа вершин;

в) зависимости числа $\alpha_0(G)$ от плотности ребер в графе $G(V, E)$ при фиксированном числе вершин.

При оценке временной сложности подсчитано число операций сравнения и объединения, выполняемых процедурой $\boldsymbol{A}$ при формировании вершин графов $G^i$. Для получения каждого значения, приведенного в табл. 2

и на рис.7, 8, генерировалось не менее 40 графов размерности $n$ и 40 раз решалась задача перечисления всех МНМ, определялось среднее число элементарных операций (ЭО), выполняемых процедурой $\boldsymbol{A}$ для всех значений $n$. При этом ребра в графах генерировались по равномерному закону распределения.

Как видно из табл. 2 и рис. 8, *а*, число МНМ существенно меньше величины теоретической оценки $\lambda^* = Cn(n-1)$, а значение $\alpha_0(G)$ с увеличением плотности графа уменьшается и в среднем при малых плотностях

*Таблица 2*

| $n$ | Оценка сверху числа ЭО | Число выполненных ЭО | $t$, с | Оценка сверху $\lambda^*$ | $\lambda$ | $\alpha_0(G)$ | Число одинаковых МНМ |
|---|---|---|---|---|---|---|---|
| $\rho = 0{,}02 \div 0{,}2$; $C = 1$ | | | | | | | |
| 10 | 1300000 | 16598 | 0,03 | 90 | 16 | 5 | 6 |
| 11 | 67620483 | 50195 | 0,06 | 110 | 21 | 6 | 1 |
| 12 | 128636191 | 215043 | 0,15 | 132 | 28 | 6 | 7 |
| 13 | 232796998 | 472669 | 0,32 | 156 | 37 | 7 | 1 |
| 14 | 402679585 | 1688780 | 0,89 | 182 | 49 | 7 | 8 |
| 15 | 669768750 | 4030206 | 2,1 | 210 | 65 | 7 | 8 |
| 16 | 1076426179 | 12280294 | 5,7 | 240 | 86 | 8 | 8 |
| 17 | 410338673 | 32790573 | 14,7 | 272 | 114 | 9 | 1 |
| 18 | 2559079734 | 88805250 | 38,1 | 306 | 151 | 9 | 9 |
| 19 | 3807893608 | 238947325 | 98,44 | 342 | 200 | 10 | 1 |
| 20 | 554240000 | 593443818 | 239 | 380 | 265 | 10 | 2 |
| $\rho = 0{,}5$; $C = 1$ | | | | | | | |
| 20 | 5504000000 | 239244 | 0,1 | 380 | 32 | 5 | — |
| 30 | 107163000000 | 6967613 | 1,7 | 870 | 149 | 6 | — |
| 40 | 874905600000 | 100974931 | 18,2 | 1560 | 466 | 7 | — |
| 50 | 4375000000000 | 693336285 | 102,9 | 2450 | 1012 | 8 | — |
| 60 | 16572211200000 | 1005307365 | 724,4 | 3540 | 1950 | 8 | — |
| $\rho = 0{,}9$; $C = 1$ | | | | | | | |
| 50 | 4375000000000 | 35955 | 0,04 | 2450 | 81 | 3 | — |
| 100 | 660000000000000 | 3541338 | 0,51 | 9900 | 226 | 4 | — |
| 150 | 12387304687500000 | 55974348 | 4,6 | 22350 | 340 | 4 | — |
| 200 | 98176000000000000 | 381577400 | 26,1 | 39800 | 389 | 5 | — |
| 250 | 487670898437500000 | 11500160701 | 99,3 | 62250 | 436 | 4 | — |

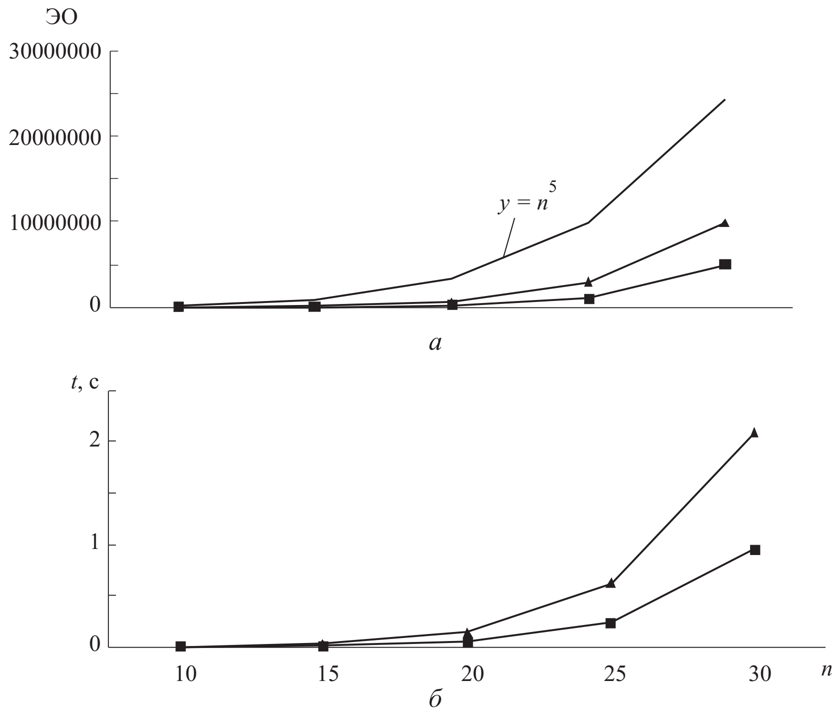



*Рис. 7.* Зависимость среднего числа ЭО (*а*) и среднего времени *t* (*б*) работы процедуры ***A*** от числа вершин в графе при ρ = 50 % : ▲ — алгоритм RM; ■ — алгоритм IND

ребер в графе не превышает значения $n/2$ (см. рис. 8, *б*). Среднее число операций, выполняемых процедурой ***A***, также не превышает полученной оценки сверху (см. табл. 2 и рис. 7).

Результаты эксперимента показали, что временная сложность процедуры ***A*** начинает приближаться к полученной верхней оценке при больших значениях *n* и малых ρ. Поскольку степень полинома, оценивающего временную сложность ее работы, очень большая, время решения задачи может возрастать до десятков сотен минут и часов. Поэтому для эффективной работы предложенного алгоритма при малых плотностях ребер и большом числе вершин необходим процесс распараллеливания данной процедуры, что возможно осуществить, применив для ее реализации CUDA технологии.

Для проверки правильности работы процедуры ***A*** использован точный ранговый алгоритм RM [4], имеющий экспоненциальную сложность, и алгоритм IND, реализующий процедуру ***A***. Из рис. 7 видно, что число ЭО, выполняемых процедурой ***A***, при ρ = 50 % в графах не превышает зна-

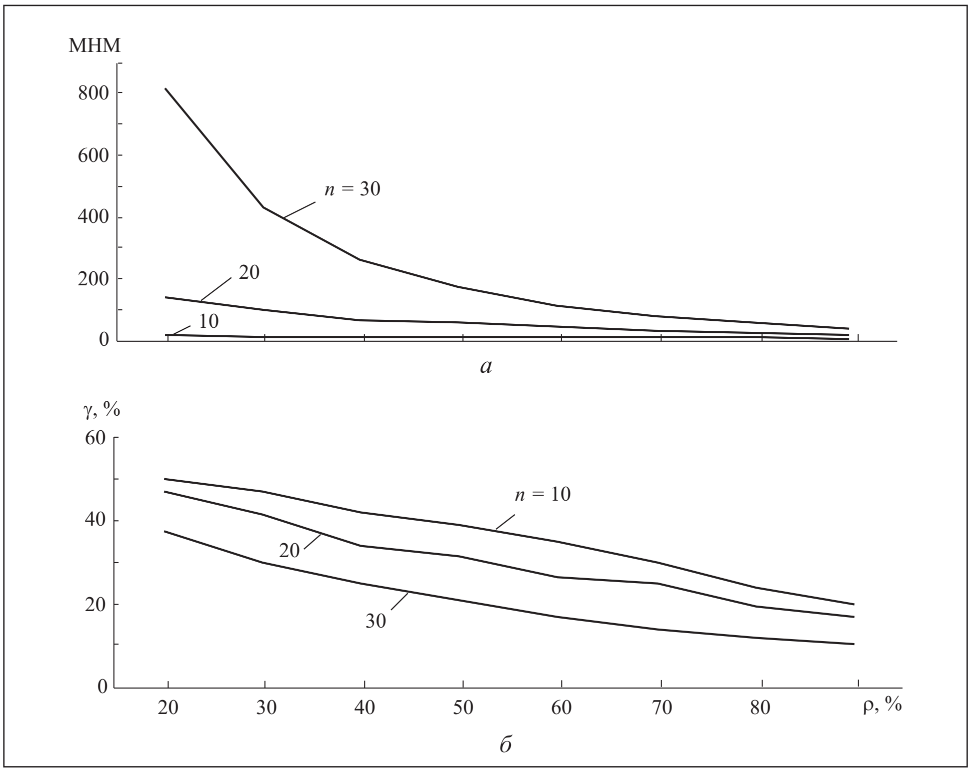



*Рис. 8.* Зависимость среднего числа МНМ (*а*) и средней мощности наибольшего МНМ (*б*) от плотности ребер при различном числе вершин в графе

чений данного полинома. В процессе экспериментальных исследований несовпадений при определении МНМ ранговым алгоритмом и исследуемой процедурой ***A*** не наблюдалось.

## Выводы

Предложенная процедура ***A*** при $\rho > 0{,}5$ в среднем за полиномиальное время позволяет решать такие задачи как определение МНМ, вершинных покрытий и клик в произвольных неориентированных графах, степень вершин которых больше или равна единице. Однако при малых плотностях ребер в графе процедура выполняет экспоненциальное число ЭО, поскольку число МНМ возрастает экспоненциально. В худшем случае при $\rho < 0{,}5$ процедура ***A*** выполняет не более, чем $\boldsymbol{O}\,(2^n)$ ЭО, а при $\rho \geq 0{,}5$ — $\boldsymbol{O}\,(n^7 \log_2 n)$ ЭО, что выгодно отличает ее от алгоритма Брона — Кербоша, имеющего временную сложность $\boldsymbol{O}\,(3^{n/3})$. Кроме того, в алгоритме Брона —

Кербоша перечисляются все независимые множества и среди них выбираются максимально независимые, а в предложенной процедуре ***A*** перечисляются только МНМ, что позволяет существенно сократить время на их перечисление.

A procedure of enumeration of only maximum independent sets in unoriented arbitrary graphs has been proposed; it allows reducing a temporary difficulty of the algorithm realization.

СПИСОК ЛИТЕРАТУРЫ

1. *Кристофидес Н.* Теория графов. Алгоритмический подход. — М. : Мир, 1978. — 309 с.
2. *Емеличев В.А., Мельников О.И., Сарванов В.И., Тышкевич Р.И.* Лекции по теории графов. — М. : Наука, 1990. — 383 с.
3. *Листровой С.В., Яблочков С.В.* Метод решения задачи определения минимальных вершинных покрытий и независимых максимальных множеств // Электрон. моделирование. — 2003. — **25**, № 2. — С. 31—40.
4. *Листровой С.В., Гуль А.Ю.* Метод решения задачи о минимальном покрытии на основе рангового подхода // Там же. — 1999. — **21**, № 1 — С. 58—70.

Поступила 06.08.13

*ЛИСТРОВОЙ Сергей Владимирович, д-р техн. наук, профессор, профессор кафедры специализированных компьютерных систем Украинской государственной академии железнодорожного транспорта. В 1972 г. окончил Харьковское высшее военное командно-инженерное училище. Область научных исследований — задачи дискретной оптимизации и теории графов и их приложение к анализу вычислительных систем и сетей.*