ИЗВЕСТИЯ
ТОМСКОГО ОРДЕНА ОКТЯБРЬСКОЙ РЕВОЛЮЦИИ И ОРДЕНА ТРУДОВОГО КРАСНОГО ЗНАМЕНИ ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА им. С. М. КИРОВА

Том 223 1972

АЛГОРИТМ УПОРЯДОЧЕННОЙ ПЕРЕКОДИРОВКИ ВЕРШИН ГРАФА

Ю. Н. ЕФИМОВ, П. А. СЕДЕЛЬНИКОВ

(Представлена научным семинаром УВЛ)

В настоящее время широкий класс задач, решаемых на ЦВМ, формулируется в терминах теории графов. При этом возникают противоречивые требования к кодированию вершин графа (сети). Составителям сетей удобно пользоваться многоразрядными десятичными шифрами, имеющими содержательный смысл, однако с точки зрения удобства машинной обработки желательна сквозная нумерация вершин. Она позволяет записать топологию графа и другие информационные массивы более компактно и упрощает процессы сортировки и поиска. Однако известные алгоритмы перекодировки очень трудоемки, так как требуют многократного просмотра топологии и большого числа сравнений неперекодированных шифров с все возрастающим по величине массивом перекодированных шифров вершин [2].

С другой стороны, часто бывает удобным, а иногда и необходимым представление графа в виде слоев, расположенных в порядке возрастания порядковой функции φ его вершин, где $\varphi = 0, 1, 2, \ldots, k, \ldots, n$ [1]. Процесс нахождения порядковой функции без предварительной перекодировки вершин трудоемок. Если же такая перекодировка произведена, то упорядочивание топологии по возрастанию порядковой функции делает необходимой повторную перекодировку [2].

В данной работе предлагается алгоритм упорядоченной перекодировки вершин графа, являющийся модификацией алгоритма Кана [3]. Он позволяет одновременно производить упорядочивание топологии графа по возрастанию функции φ и сквозную нумерацию его вершин в прямом или обратном порядке при минимальном числе сравнений неперекодированных шифров с перекодированными шифрами вершин слоя, выделенного при предшествующем просмотре топологии.

Пусть граф G задан множеством вершин I и их прямым или обратным отображением Γ_i [4], т. е.

$$G = (I, \Gamma_i).$$

По значению порядковой функции он разбивается на слои I_{φ_k}, являющиеся подмножествами множества I и состоящие из вершин i с одинаковыми порядковыми функциями.

$$I_{\varphi_k} = \{i / \Gamma_i \subseteq (\bigcup_{\varphi=0}^{k-1} I_\varphi)\}.$$

Множеству вершин графа I, разнесенных по слоям, можно поставить во взаимнооднозначное соответствие равномощное ему множество натуральных чисел, т. е.

$$|I| \sim |N|,$$

что дает упорядоченную перекодировку вершин.

Упорядоченная перекодировка осуществляется за $n+2$ просмотра исходной топологии графа, во время которых составляется таблица соответствия (ТС) шифров вершин и чисел натурального ряда.

При каждом просмотре исходной топологии выделяются в очередной слой те вершины, у которых все элементы отображения на предшествующие слои, т. е. все связи, перекодированы.

При первом просмотре выделяется слой с $\varphi = 0$ (нулевой слой), вершины которого не имеют связей, т. е. $\Gamma_i = \varnothing$. Шифры вершин нулевого слоя записываются в ТС. В исходной топологии они заменяются соответствующими им числами натурального ряда и отмечаются признаками. Кроме того, вершины нулевого слоя записываются в массив топологии с упорядоченной перекодировкой.

При втором просмотре топологии перекодируются и отмечаются признаками те элементы отображения j вершин графа i, для которых $j \in (\Gamma_i \cap I_{\varphi_0})$, т. е. перекодируются все имеющиеся в графе связи вершин последующих слоев с вершинами нулевого слоя. Если для некоторой вершины $\Gamma_i \subseteq I_{\varphi_0}$, т. е. перекодированы все связи, то ее шифр записывается в ТС, в исходной топологии производится перекодировка и ставится признак. Вершина и ее отображения записываются в массив перекодированной топологии. После полного просмотра топологии в ТС будут выбраны шифры всех вершин первого слоя, в исходной топологии перекодированы все элементы отображения вершин последующих слоев на нулевой. Поэтому шифры вершин нулевого слоя ТС при последующих просмотрах топологии в перекодировке не участвуют. Вершины и элементы их отображения, отмеченные в топологии признаками, в дальнейшем процессе перекордировки также не участвуют.

При k-ом просмотре перекодируются и отмечаются признаками те элементы отображения вершин графа, для которых

$$j \in (\Gamma i \cap I_{\varphi_{k-1}}.$$

В k-й слой выбираются, заносятся в ТС, перекодируются, отмечаются признаками в исходной и записываются в перекодированную топологию те вершины, для которых

$$\Gamma i \subseteq \left(\bigcup_{\varphi=0}^{k-1} I_{\varphi}\right).$$

Если при очередном просмотре топологии не было перекодировано ни одной вершины, то процесс упорядоченной перекодировки заканчивается. Присутствие в исходной топологии неперекодированных шифров вершин и элементов их отображения говорит о наличии в графе контуров и обрывов. В этом случае такие вершины и элементы их отображения не имеют признаков, что позволяет быстро локализовать их.

Как видно из описания алгоритма, при прямом задании графа в нулевом слое упорядоченной топологии будут находиться стоки, а при обратном задании — источники. Порядок нумерации вершин в упорядоченной топологии (прямой или обратный) определяется порядком нумерации шифров в ТС. Произведя дополнительно сортировку упорядоченной топологии по убыванию порядковой функции, можно получить в начале записи топологии для прямого задания графа источники, а для обратного — стоки. Следовательно, имеется 8 режимов работы алгоритма.

К достоинствам рассмотренного алгоритма можно отнести следующее.

1. Совмещение в одном алгоритме двух трудоемких процессов: упорядочивания графа по слоям и перекордировки его вершин.

2. Минимизация числа сравнений шифров элементов отображения вершин графа с шифрами перекодированных вершин.

3. Возможность более рационального использования оперативной памяти ЭЦВМ, так как нет необходимости хранить в МОЗУ всю ТС.

4. Инвариантность алгоритма относительно способа задания графа (прямого или обратного).

Предлагаемый алгоритм был реализован на ЭЦВМ «Урал-11Б». Программа содержит около 400 команд. Время работы программы для графа, имеющего 1000 вершин и коэффициент сложности около 2, оказалось равным 55 сек. Топология графа полностью размещалась в оперативной памяти машины. Для графа с 400 вершин и коэффициентом сложности около 5 время работы программы было равно 29 мин. Массив топологии занимал около 90 зон по 256 слов. Основная доля времени работы программы падает на поиск в ТС шифров элементов отображения вершин путем последовательных сравнений.

ЛИТЕРАТУРА

1. Ю. Н. Ефимов. Алгоритм нахождения порядковой функции сетевого графика. Изв. ТПИ, т. 168. Изд. ТГУ, 1968.

2. Ю. Н. Ефимов. Применение малых ЭЦВМ в автоматизированных системах планирования и управления. Кандидатская диссертация, Томск, 1968.

3. A. B. Kahn. Topological Sorting of large Networks. Communication of the ACM, № 1, 1962.

4. К. Берж. Теория графов и ее применения. М., Изд. ИЛ, 1962.