

ОЦЕНКА ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТИ АЛГОРИТМОВ ПОИСКА ПУТИ

Грикень В.Г., Хинкель Е.Р.

*Институт информационных технологий Белорусского государственного университета
информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Майсеня Л.И. – док. пед. наук, профессор

Аннотация. Представлена оценка вычислительной сложности алгоритмов поиска пути. Результаты оценки вычислительной сложности алгоритмов получены с использованием модели неветвящейся программы.

Задача нахождения эффективных алгоритмов поиска пути является актуальной во многих приложениях. Классической задачей поиска пути является задача о коммивояжере, когда расстояние, пройденное им, должно быть минимальным. Эффективный алгоритм решения этой задачи неизвестен, и она является NP-полной. Другой пример, имеющий важное значение в области современной электроники – разработка цифрового процессора на основе технологии проектирования на кристалле, когда топология проводников схемы непосредственно связана с таким важнейшим параметром любого процессора как быстродействие. Выбор того или иного алгоритма является нетривиальным процессом. Но, при некоторых условиях можно указать такие алгоритмы, при реализации которых полученное расстояние будет ненамного превышать минимально возможное. Для многих приложений, связанных с информационными системами, построенными на оптимальных алгоритмах обработки сигналов [1] (системы космического базирования военного назначения, цифровая широкополосная коммуникация, робототехнические промышленные комплексы, сложное медицинское оборудования и пр.) основным требованием к применяемым алгоритмам является минимально возможная вычислительная сложность вычислений. К настоящему времени не существует общего определения понятия сложности алгоритмов. Это понятие допускает различные трактовки, отражающие особенности задач, уровень развития процессорных технологий. Например, могут учитываться или не учитываться такие параметры, как пропускная способность канала, размер машинного слова, точность вычислений, аппаратное обеспечение и др.

В работе представлена оценка вычислительной сложности четырех алгоритмов поиска. Рассматривались алгоритмы: Дейкстры; модификация алгоритма Дейкстры; BFS (breadth-first search); DFS (depth-first search) [2]. Исследования проводились на квадратном лабиринте размером $M \times N = 25 \times 25$. Размер входа обработки $n = 148$. Алгоритмы реализовывались в виде компьютерных программ. В качестве технологии реализации алгоритмов была принята модель обобщенной однопроцессорной машины с памятью с произвольным доступом. В этой модели команды процессора выполняются последовательно; одновременно выполняемые операции отсутствуют. Наиболее употребительной моделью вычислительного процесса, позволяющей раскрыть вычислительную сложность алгоритма и сравнить различные алгоритмы, является модель неветвящейся программы [3]. Число шагов такой программы как функция от размера входа n определяет временную сложность, а число переменных, участвующих в вычислениях определяют емкостную сложность программы. При определении сложности неветвящейся программы учитывают: набор входных переменных c_0, c_1, \dots, c_n ; операции над множествами в кольце G или поле F ; множество базисных операций $P = \{+, \times, /, \cup, \cap\}$, элемент $\beta \in G$ или $\beta \in F$. Сложность базисной операции $f \in P$ описывается числом $\lambda(f)$. Результаты оценки вычислительной сложности алгоритмов по временному показателю представлены в таблице 1. Каждое алгоритмическое решение сводилось к определению значения нормированного времени, за которое обрабатывался вход. Данные получены путем экспериментальных исследований на основе компьютерного моделирования. Сложность каждого алгоритма (программы) определялась суммой всех чисел $\lambda(f) \in P$ по всем командам программы.

Таблица 1 – Вычислительная сложность алгоритмов

Алгоритм	Дейкстры	Модификация алгоритма Дейкстры	BFS	DFS
Временная сложность	2995	2999	2985	2998

Выводы:

1. Одна и та же задача разными алгоритмами решается практически одинаковым числом шагов неветвящейся программы.

58-я научная конференция аспирантов, магистрантов и студентов БГУИР, 2022 г.

2. При увеличении времени работы вычислителя выигрыш от применения алгоритма BFS возрастает в сравнении с другими исследуемыми алгоритмами.
3. С ростом быстродействия процессора эффективность алгоритма BFS увеличивается.
4. При увеличении времени работы вычислителя и быстродействия алгоритма BFS, выигрыш в обработке размера входа будет еще более существенным.

Список использованных источников:

1. Митюхин, А.И. *Прикладная теория кодирования: учеб. - метод пособие.* / А. И. Митюхин – Минск : БГУИР, 2018.
2. Кормен, Т.Х., Алгоритмы: построение и анализ. / Т.Х. Кормен [и др.] / 2-е издание. – М: Пер. с англ. Изд-во «Вильямс», 2009.
3. Лосев, В.В. Микропроцессорные устройства обработки информации. Алгоритмы цифровой обработки: учеб. пособие. – Мн.: Высшая школа, 1990.