УДК 681.5.015.3

ОПРЕДЕЛЕНИЕ ОПЕРАЦИОННЫХ ХАРАКТЕРИСТИК АЛГОРИТМОВ

А.М. Ольховой
(представил д.ф-м.н., проф. С.В. Смеляков)

Рассматривается аналитический метод определения операционных характеристик алгоритмов для предварительных оценок потребной производительности АСУ на основе анализа конечных марковских цепей.

Постановка проблемы. Одной из главных задач при разработке автоматизированной системы управления (АСУ) является оценка потребной производительности АСУ, на которую существенное влияние оказывает ее рабочая нагрузка. В качестве рабочей нагрузки выступают задачи по обработке информации. На начальных этапах проектирования информация о решаемых задачах бывает весьма незначительной и может быть только в форме математических моделей. Тем не менее, необходимость в предварительной оценке потребной производительности не пропадает. Для этих целей необходимо представить процесс выполнения задачи в вычислительной системе как последовательность шагов, связанную с выполнением операторов. Применительно к задачам определения трудоемкости операторы алгоритма будем подразделять на функциональные (F), перехода (U) и ввода-вывода (B). Функциональные операторы и операторы перехода задают совокупности операций над данными и относятся к одному классу операторов, называемых основными операторами.

Совокупность операторов и связей между ними наиболее наглядно представляется граф-схемой алгоритма, которая строится как композиция вершин, соответствующих операторам алгоритма, и дуг, отображающих связи между операторами. Выделяют вершины: начальные, конечные, операторные. Введенные операторы позволяют формализовать анализ алгоритмического обеспечения и получить искомые операционные характеристики на уровне описания алгоритмов. Задача определения операционных характеристик алгоритмов сводится к двум классам методов: аналитическим и имитационным. Аналитические методы оценки привлекают своей простотой, законченностью, но обладают одним существенным ограничением. Это ограничение связано с той степенью допущений, которая приемлема при анализе того или иного алгоритма и является неизбежной при постановке задачи оценки алгоритмов аналитическими методами.

Наиболее мощным аппаратом для решения указанной задачи являются методы имитационного моделирования. Имитационные модели позволяют получать более точные оценки и учитывать наиболее полно особенности, присущие исследуемым процессам. Однако разработка имитационных моделей требует значительных трудозатрат, что является основным недостатком этих методов. Поэтому на начальных этапах проектирования целесообразно использовать аналитические методы, так как для предварительных оценок точность их вполне удовлетворительна.

Анализ литературы [1 – 3] показал, что для определения операционных характеристик алгоритма целесообразно использовать аналитический метод, основанный на анализе конечных марковских цепей.

Целью данной статьи является разработка методики определения операционных характеристик алгоритмов аналитическим методом, основанным на анализе конечных марковских цепей.

Основной материал. Для применения аналитического метода, основанного на анализе конечных марковских цепей [1], представим алгоритм в виде граф-схемы (рис. 1).

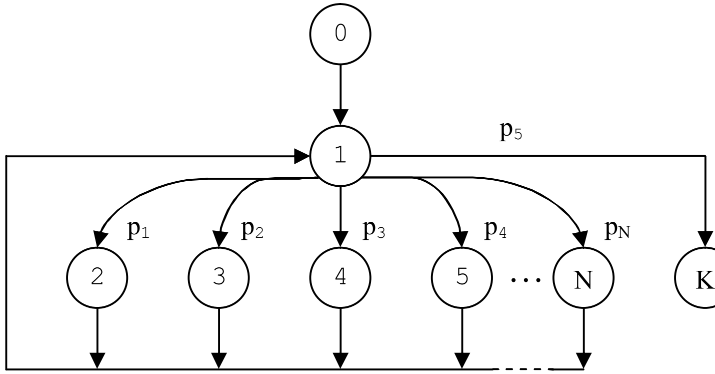

Рис. 1. Граф-схема алгоритма

Граф-схема выявляет структуру алгоритма, определяет множества операторов и дуг: $V=\{V_1,\ \dots\ ,V_{k-1}\}$; $D=\{(i,j)\}, i=\overline{0,k-1}, j=\overline{1,k}.$

Множество V, по существу, является объединением подмножеств операторов F, U, B. Подмножество функциональных операторов задает преобразование на множестве данных. Подмножество операторов перехода задает порядок, вычисления значений предикатов и правило выбора одного из возможных путей развития вычислительного процесса. Подмножество операторов ввода-вывода B задает обращение к определенному файлу с целью передачи некоторого количества информации.

Переходы между вершинами графа рассматриваются как случайные события и характеризуются матрицей вероятностей перехода

$$P^* = \begin{vmatrix} P_{0,1} & \cdots & P_{0,k} \\ \vdots & & \vdots \\ P_{k-1,1} & \cdots & P_{k-1,k} \end{vmatrix}, \quad \sum_{j-1}^{k} P_{ij} = 1, \quad i = \overline{0, k-1}.$$

Значения P_{ij} определяются вероятностями значений предикатов, зависящими от распределения значений данных, отношения между которыми задаются предикатами. Допустим, что вероятности переходов P_{ij} постоянны и после выполнения оператора $V_i (i = \overline{1, k-1})$ переход к следующему оператору определяется только распределением вероятностей P_{ij}, т.е. не зависит от хода вычислительного процесса в прошлом. При указанных допущениях процесс выполнения алгоритма является марковским с состояниями S_0, … , S_k. Состояния с точностью до нумерации соответствуют вершинам графа и называются: S_0 – начальное (стартовое); S_1, … , S_{k-1} – невозвратные (процесс после какого-то числа шагов покидает их); S_k – поглощающее (достигнув его, процесс прекращается).

Для вычисления математического ожидания времени реализации алгоритма необходимо: во-первых, определить среднее число n_1, … , n_{k-1} пребываний марковского процесса в невозвратных состояниях; во-вторых, для каждого оператора подмножества F и U необходимо определить среднее количество операций k_0, составляющих оператор и для каждого оператора подмножества B среднее количество информации I_{BB}, передаваемой при выполнении оператора. Среднее число пребываний марковского процесса в невозвратных состояниях определяется корнями системы линейных алгебраических уравнений

$$n_i = \sum_{j=0}^{k-1} P_{ij} \cdot n_j, \quad i = \overline{1, k-1}. \tag{1}$$

Учитывая, что процесс находится в начальном состоянии один раз и вероятность $P_{01} = 1$, систему линейных уравнений (1) можно переписать в виде

$$n_i = \sum_{j=1}^{k-1} P_{ij} (n_j + \delta_{1i}), \quad i = \overline{1, k-1}, \tag{2}$$

где δ_{1i} – символ Кронекера.

Начальное состояние марковской цепи определяется дугой $(0, i)$ $i = \overline{1, k-1}$. Система (2) может иметь K–1 разных решений в зависимости от i. Учитывая K–1 начальное условие, запишем систему линейных уравнений (2) в матричном виде

$$\{n_{ij}\} = I + \{n_{ij}\}P. \tag{3}$$

Из (3) следует, что

$$\{n_{ij}\} = (I - P)^{-1} = N,$$

где N – фундаментальная матрица системы; P – матрица переходов, получается из P* путем отбрасывания первой строки и первого столбца.

Среднее время нахождения процесса в j-м состоянии с учетом того, что процесс начал выполняться с i-го оператора определяется уравнением

$$T = NW_{dq},$$

где W_{dq} – диагональная матрица, элементами которой является значение интервала времени, необходимого для выполнения i-го оператора. Время для выполнения i-го оператора определяется по формуле

$$W_{ii} = \begin{cases} k_{0i} / C_{пр} & \text{при} \quad V_i \in F \bigcup U, \\ l_{BB_i} / C_{BB} & \text{при} \quad V_i \in B, \quad i = \overline{1, k-1}. \end{cases}$$

где $C_{пр}$ – быстродействие процессора; C_{BB} – скорость ввода-вывода.

Если алгоритм начинает выполняться с оператора с номером i = 1, то среднее время реализации алгоритма определяется суммой элементов первой строки матрицы T: $T_{cp} = \sum_{j=1}^{k-1} t_{1j}$. Для обеспечения необходимой точности решения некоторых задач необходимы сведения о дисперсии трудоемкости алгоритма. Дисперсия характеризует степень разброса значений случайной величины относительно своего среднего значения, и чем больше дисперсия, тем больше разброс значений. Для вычисления дисперсии времени выполнения алгоритма необходимо определить дисперсию числа пребываний марковского процесса б невозвратных состояниях $S_1, \ldots, S_{k-1}$.

Обозначим дисперсию числа пребываний марковского процесса в невозвратных состояниях $D[n_{ij}]$. Тогда

$$\{D[n_{ij}]\} = \{M[n_{ij}^{\ 2}]\} - \{M^2[n_{ij}]\},$$

где
$$\{M^2[n_{ij}]\} = \begin{vmatrix} n^2{}_{1,1} & \cdots & n^2{}_{1,k-1} \\ \vdots & & \vdots \\ n^2{}_{k-1,1} & \cdots & n^2{}_{k-1,k-1} \end{vmatrix} = N_{sq}.$$

Для нахождения матрицы $\{M[n^2{}_{ij}]\}$ найдем сначала $M[n^2{}_i]$, $i = \overline{1, k-1}$

$$M[n^2{}_i] = \sum_{j=1}^{k-1} P_{ij} M[(n_j + \delta_{1i})^2] = \sum_{j=1}^{k-1} P_{ij} M[n^2{}_j] + 2\sum_{j=1}^{k-1} P_{ij} M[n_j]\delta_{1i} + \delta_{1i}. \tag{4}$$

Далее так же, как в случае нахождения среднего числа пребываний марковского процесса в невозвратных состояниях $S_1, \dots, S_{k-1}$, учитывая, что начальное состояние определяется дугой (0,i), запишем (4) в матричном виде для K–1 решения

$$\{M[n^2{}_{ij}]\} = \{M[n^2{}_{ij}]\}P + 2(NP)_{dq} + I. \tag{5}$$

Появление диагональной матрицы $(NP)_{dq}$ объясняется тем, что множитель δ_{ij} обращает в нуль все элементы вне главной диагонали.

В [1] доказывается, что NP = PN = N – I. Подставим в (5), получим

$$\{M[n^2{}_{ij}]\} = N(2N_{dq} - I).$$

Таким образом,

$$\{D[n_{ij}]\} = N(2N_{dq} - I) - N_{sq} = N_2.$$

Учитывая, что $D[CX] = C^2 D[X]$ получим

$$\{D[t_{ij}]\} = N_2 W^2{}_{dq} = D.$$

Если алгоритм начинает выполняться от оператора с номером i = 1, то дисперсия времени прогонки определяется суммой элементов первой строки матрицы

$$D[t] = \sum_{j=1}^{k-1} D[t_{ij}], \qquad \sigma[t] = \sqrt{D[t]}.$$

Рассмотренный способ определения операционных характеристик алгоритма является универсальным. Он позволяет получать оценки для алгоритмов с любой структурой, прост с точки зрения программирования.

Выводы. Использованный метод оценки производительности алгоритмов позволяет получить сравнительные характеристики времени выполнения алгоритмов при использовании тех или иных вычислительных методов, рациональных параметров алгоритмического модуля, а также для выбора оптимальных, с точки зрения быстродействия, характеристик вычислительных устройств.

ЛИТЕРАТУРА

1. *Кемени Дж., Снелл Дж. Конечные цепи Маркова.– М.: Наука, 1970. – 244 с.*
2. *Гаврилов В.П., Русян С.Р. Методы определения операционных характеристик алгоритмов. – М.: ВВИА имени Н.Е. Жуковского, 1983. – 140 с.*
3. *Основы теории вычислительных систем / Под ред. С.А. Майорова. – М.: Высшая школа, 1978. – 346 с.*

Поступила 17.10.2003

ОЛЬХОВОЙ Алексей Михайлович, *адъюнкт ХВУ. В 1997 году окончил ХВУ. Область научных интересов – САПР и системы поддержки принятия решений.*