

УДК 519.81

Профессор Ю.В. Бугаев, доцент И.Е. Медведкова,
доцент С.В. Чикунов, аспирант Ф.А. Музалевский
(Воронеж. гос. ун-т инж. технол.) кафедра информационных технологий,
моделирования и управления, тел. (473) 255-25-50

Механизм выбора вариантов при заданных отношениях для компонент объектов сравнения

Рассмотрен синтез механизма выбора составных объектов на основе сравнения их компонент.

The synthesis of composite object selection mechanism based on a comparison of their components.

Ключевые слова: выбор вариантов, бинарные отношения.

В работе [1] предложено обобщение схемы динамического программирования (ДП), описанное в терминах языка функций выбора (ФВ) – наиболее универсального и удобного для анализа описания концепции выбора.

Напомним, что в [1] рассматривалась следующая задача. Имеем некоторый многошаговый процесс, на каждой стадии которого необходимо выбрать набор управлений, позволяющий построить множество «хороших» траекторий. Пусть T – некоторое множество траекторий процесса. Рассматривая каждую траекторию $p \in T$ как допустимое альтернативное решение некоторой задачи выбора, а T – как некое предъявление, введем ФВ $C(T)$, позволяющую осуществлять выбор «хороших» траекторий.

Определение 1. Назовем C -оптимальными все траектории предъявления, попавшие в выбор.

Предположим, что ФВ определена не только на целых траекториях, но и на каждой их части. Пусть все траектории некоторого множества Z имеют общий фрагмент p , т.е. имеем соотношение:

$$Z = Y \& p, \quad (1)$$

где символ «&» означает операцию стыковки частей траекторий.

Определение 2. Будем говорить, что ФВ обладает свойством слабой аддитивности, если для любого множества Y неполных траекторий, удовлетворяющего соотношению (1)

при некоторой совокупности траекторий Z и фиксированном фрагменте p , справедливо равенство

$$C(Y \& p) = C(Y) \& p.$$

В [1] дана формулировка обобщенного принципа оптимальности Беллмана (ОПОБ), который определяет метод отбора C -оптимальных траекторий, предложена численная схема поиска таких траекторий и доказана следующая теорема.

Теорема 1. Если ФВ $C(\bullet)$, определенная на множестве траекторий и их отдельных фрагментов, обладает свойствами наследования, отбрасывания и слабой аддитивности, то численная схема, основанная на ОПОБ, позволяет найти все C -оптимальные траектории процесса.

Этот результат позволил с единых позиций провести анализ ряда алгоритмов, вписывающихся в данную концепцию, однако проблема синтеза новых численных схем остается открытой. Рассмотрим, например, следующую задачу.

Пример 1. Пусть имеется некоторый граф, на котором требуется найти набор в определенном смысле нехудших путей. Для осуществления выбора необходимо сравнивать альтернативные варианты путей, например, посредством какого-то бинарного отношения. Очевидно, что в постановке задачи невозможно описать это отношение в явном виде на множестве всех путей, т.к. их количество, скорее всего, будет слишком велико. Намного реальнее ввести правило сравнения отдельных ребер графа. В простейшем случае

это можно сделать посредством задания их критериальных весов на числовой шкале [2, 3]. В силу аддитивности критериев сложностей со сравнением путей здесь не возникает. Но при задании правила сравнения рёбер на более слабых шкалах, не допускающих применение обычных арифметических операций [4], возникает проблема корректного использования этого правила в вычислительном алгоритме, т.к. в методах, основанных на схеме ДП, придётся сравнивать как полные пути, так и их фрагменты, причём количество рёбер у сравниваемых фрагментов может различаться.

Подобные задачи возникают в сетевом планировании, при построении маршрутов в компьютерных сетях и в других приложениях.

Данная работа посвящена синтезу ФВ, основанной на механизме блокировки по бинарному отношению и построению на этой основе вычислительных алгоритмов.

1. Построение бинарного отношения для объектов. Пусть на множестве некоторых элементов $X = \{x_1, x_2, \dots, x_n\}$ задано бинарное отношение (БО) R^* , определяющее предпочтение одного элемента перед другим. На базе R^* построим отношение R , позволяющее сравнить два объекта A и B , представляющие собой произвольные наборы элементов множества X . При этом предполагается, что количество элементов в одном объекте может не совпадать с количеством элементов в другом, а один и тот же элемент может входить в оба объекта.

Пусть отношение R^* позволяет отсортировать элементы, составляющие объект, внутри самого набора, т.е. при $A = \{x_1, x_2, \dots, x_k\}$ положим $A^S = \{x_{i_1}, x_{i_2}, \dots, x_{i_k}\}$ – отсортированный объект A , где $\forall u, 1 \leq u \leq k-1, (x_{i_u}, x_{i_{u+1}}) \in R^*$. Введём бинарное отношение R для отсортированных объектов. Пусть два сравниваемых объекта A^S и B^S имеют одинаковое число элементов – m , т.е. $A^S = (a_1, a_2, \dots, a_m), B^S = (b_1, b_2, \dots, b_m)$. Тогда

$$A^S R B^S \Leftrightarrow \forall i = 1, \dots, m \quad (a_i, b_i) \in R^* \quad (2)$$

Если два сравниваемых объекта A^S и B^S имеют разное число элементов, то более «короткий» из них дополним сначала «пустыми» объектами '0', для которых по определению будем считать

$$\forall x_i \in X \quad ('0', x_i) \in R^*. \quad (3)$$

Тогда для произвольных объектов A^S и B^S соотношение (2) будем строить с учётом (3).

2. Применимость механизма к динамическому программированию. Определим условия, которые необходимо наложить на бинарное отношение R^* для возможности применения схемы (2) в ДП. Перечень ограничений представляется логичным разбить на две части. Первая – связана с необходимостью сортировки элементов внутри объекта, вторая – с непосредственным применением предложенной схемы.

Для осуществления сортировки, очевидно, необходимо, чтобы R^* было антисимметрично, полно и транзитивно, т.е. являлось линейным порядком [5]. Далее будем полагать, что элементы внутри объектов сравнения отсортированы должным образом.

Определим вторую часть ограничений, которые следует наложить на R^* . Известно [5], что структуры, представленные асимметричными и транзитивными бинарными отношениями, порождают непустые функции выбора, обладающие свойствами наследования, согласия и отбрасывания (что необходимо для применения метода динамического программирования при поэтапном выборе).

Докажем транзитивность R . Пусть имеем три отсортированных объекта A^S, B^S и C^S , причём $A^S R B^S$ и $B^S R C^S$. Тогда согласно (2)

$$\forall i = 1, \dots, m \quad (a_i, b_i) \in R^* \wedge (b_i, c_i) \in R^*,$$

где m – максимум из длин наборов A, B и C . Поскольку ранее на R^* было наложено условие транзитивности, можно утверждать, что $\forall i = 1, \dots, m \quad (a_i, c_i) \in R^*$, а это означает $A^S R C^S$, что и требовалось доказать.

Определим условия, при которых R асимметрично.

Имеем $A^S R B^S$. Из формулы (2) видно, что одновременное выполнение $B^S R A^S$ возможно лишь при полном поэлементном равенстве оценок в A^S и B^S . В случае, когда оценки A^S и B^S полностью поэлементно совпадают, следует использовать некоторый дополнительный критерий выбора, например, выбирать альтернативу с наименьшим порядковым номером. Таким образом достигается асимметричность.

Покажем, при каком условии функция выбора, порождаемая БО R , будет обладать свойством слабой аддитивности. Для этого дадим следующее определение.

Определение 3. Пусть на множестве траекторий и их фрагментов некоторого многостадийного процесса определено БО P . Будем говорить, что P обладает свойством инвариантности относительно сдвига, если выполнено условие:

$$\forall x, y \in A, \forall b \mid x \& b, y \& b \in A' \quad (x, y) \in P \Leftrightarrow (x \& b, y \& b) \in P \quad (4)$$

Теорема 2. Для того чтобы ФВ, основанная на механизме блокировки по некоторому бинарному отношению P , обладала свойством слабой аддитивности, необходимо и достаточно, чтобы P было инвариантно относительно сдвига.

Доказательство. Напомним, что механизм блокировки определяется следующим правилом:

$$C^P(X) = \{x \in X \mid \forall y \in X (y, x) \notin P\}.$$

Достаточность. Пусть для P выполнено соотношение (4), и пусть Y – некоторое множество неполных траекторий. Выберем произвольную пару альтернатив x и y , что $x \in C^P(Y)$. Тогда $(y, x) \notin P$, а следовательно, $(y \& b, x \& b) \notin P$. Значит, $(x \& b) \in C^P(Y \& b)$. Отсюда $C^P(Y) \& b \subseteq C^P(Y \& b)$. Аналогично доказывается и обратное включение. Значит $C^P(Y \& b) = C^P(Y) \& b$.

Необходимость. Пусть $(x \& b) \in C^P(Y \& b)$, а значит и $x \in C^P(Y)$. Тогда для любой траектории $(y \& b)$ будет $(y \& b, x \& b) \notin P$. Предположим, что отношение $(y, x) \notin P$ не выполняется. Но тогда $x \notin C^P(Y)$ – противоречие. Значит $(y \& b, x \& b) \notin P \Rightarrow (y, x) \notin P$. Аналогично доказывается и обратное следование. В итоге получим условие (4). Теорема доказана.

Следовательно, надо показать, что БО R инвариантно относительно сдвига. Перепишем определение (4):

$$\forall A, B \in X, \forall x \mid A \& x, B \& x \in X' \\ (A, B) \in R \Leftrightarrow (A \& x, B \& x) \in R.$$

Пусть при сортировке вектора $A \& x$ элемент x попал на порядковое место $i_a \in \{1, \dots, m+1\}$, а при сортировке $B \& x$ – на i_b . Рассмотрим следующие случаи.

1. $i_a = i_b = k$. Тогда для элементов A^S и B^S с индексами j как меньше, так и больше k , сохранится условие $(a_j, b_j) \in R^*$, а в силу полноты и $(a_k, b_k) = (x, x) \in R^*$. Следовательно, $(A \& x)^S R (B \& x)^S$.

2. $i_a > i_b$. В таком случае для $i < i_b$ и $i > i_a$ сохранится отношение $(a_i, b_i) \in R^*$. Для остальных элементов A^S и B^S в силу транзитивности

и полноты R^* выполняется следующая цепочка отношений:

$$a_{i_b} R^* a_{i_b+1} R^* \dots R^* a_{i_a-1} R^* x R^* x R^* b_{i_b} R^* b_{i_b+1} \dots R^* b_{i_a} \quad (5)$$

Обозначим $\bar{a}_i$ и $\bar{b}_i$ – элементы, стоящие на i -м месте в наборах $(A \& x)^S$ и $(B \& x)^S$. Очевидно, справедливы следующие соотношения:

$$\bar{a}_i = \begin{cases} a_i & \text{при } i < i_a \\ x & \text{при } i = i_a \\ a_{i-1} & \text{при } i > i_a \end{cases}; \quad \bar{b}_i = \begin{cases} b_i & \text{при } i < i_b \\ x & \text{при } i = i_b \\ b_{i-1} & \text{при } i > i_b \end{cases}. \quad (6)$$

Тогда из (5) и (6) имеем следующий набор отношений:

$$\begin{aligned} a_{i_b} &= \bar{a}_{i_b} R^* x = \bar{b}_{i_b}; \\ a_{i_b+1} &= \bar{a}_{i_b+1} R^* b_{i_b} = \bar{b}_{i_b+1}; \bar{a}_{i_b+2} R^* \bar{b}_{i_b+2}; \\ x &= \bar{a}_{i_a} R^* b_{i_a-1} = \bar{b}_{i_a}. \end{aligned} \quad (7)$$

Иными словами, в этом случае $(A \& x) R (B \& x)$.

3. $i_a < i_b$. Тогда для элементов A^S и B^S должны выполняться следующие отношения:

$$\begin{aligned} x R^* a_{i_a} R^* \dots R^* a_{i_b-1} R^* b_{i_b-1} R^* x \\ x R^* a_{i_a} R^* b_{i_a} R^* \dots R^* b_{i_b-1} R^* x. \end{aligned}$$

Очевидно, в силу антисимметричности R^* , это может иметь место только при

$$x = a_{i_a} = b_{i_a} = \dots = a_{i_b-1} = b_{i_b-1} = x.$$

Отсюда получим

$$\begin{aligned} \bar{a}_{i_a} &= x R^* b_{i_a} = \bar{b}_{i_b}; \\ \bar{a}_{i_a+1} &= a_{i_a} R^* b_{i_a} R^* b_{i_a+1} = \bar{b}_{i_b+1}; \dots \\ \bar{a}_{i_b} &= a_{i_b-1} R^* b_{i_b-1} = \bar{b}_{i_b} = x; \end{aligned}$$

Следовательно, $(A \& x) R (B \& x)$. Теорема доказана

Таким образом, бинарное отношение R может быть положено в основу функции выбора, применяемой при динамическом программировании.

3. Пример практического применения.

При построении оптимальных маршрутов в вычислительных сетях на уровне протоколов маршрутизации (RIP, OSPF) применяется некоторый скалярный критерий (метрика). Использование различных вариантов метрики невозможно в силу сложности интегрирования разнородных критериев [6]. Предлагаемый метод позволяет сравнивать маршруты, элементы которых оценены различными метриками (пропускной способностью канала, дли-

ной очереди на маршрутизаторе, процентом потерь при доставке пакетов и т.п.) при выполнении одного единственного условия – необходимо задать отношения предпочтения для всех неделимых элементов графа-сети.

Отдельно следует рассмотреть случай возникновения нескольких элементов в множестве недоминируемых альтернатив. В таком случае следует учесть схему построения сети. Если предполагается (хотя бы теоретически) возможность отправки пакета сразу по нескольким адресам, то сложностей при применении метода возникнуть не должно. Если же есть ограничение на количество отправляемых копий пакета, то необходимо ввести дополнительное ограничение.

Простейшим примером такого ограничения может служить выбор n первых элементов множества (где n – максимально допустимое число копий пакетов). Данное ограничение имеет существенный недостаток: при различных вариантах представления одних и тех же входных данных (например, различном порядке следования альтернатив), будут получены различные решения. Этот недостаток можно обойти, проводя предварительную сортировку предъявляемого множества альтернатив, на котором необходимо осуществить выбор. Данная сортировка может замедлить поиск решения, но она не является обязательной. Ведь в реально функционирующих информационных системах сбор данных осуществляется автоматизировано, следовательно, предоставляемый формат данных является константным, то есть искажения порядка следования входных данных не будет происходить.

На основании сказанного сформулируем задачу. Необходимо составить список нехудших маршрутов из вершины s в вершину t , по которым могут быть отправлены пакеты данных. Лучшим будем считать маршрут, у которого число звеньев минимально и из них максимальное количество звеньев обладает «хорошим качеством». Качество ребра определим некоторым бинарным отношением R^* , свойства которого были оговорены выше.

Отметим, что от порядка следования участков маршрута пакета качество связи не зависит, следовательно, сортировка элементов не внесёт искажения в полученный результат.

При обратном ходе от вершины t к вершине s воспользуемся ОПОБ для построения оптимальных траекторий.

Пусть оптимизируемый процесс содержит N стадий. Состояние процесса после завершения k -й стадии определяется значением некоторого параметра v_k , $k = 1, \dots, N$, а v_0 определяет начальное состояние. Обозначим $D_k(v_{k-1})$ – множество C -оптимальных траекторий вида $v_{k-1} \rightarrow v_N$. В этих обозначениях $D_1(v_0)$ – множество решений задачи. Согласно [1] для $D_k(v_{k-1})$ существует рекуррентное соотношение

$$D_k(v_{k-1}) = C \left[\bigcup_{u \in U_k} [(v_{k-1} \rightarrow \chi_k(v_{k-1}, u)) \& D_{k+1}(\chi(v_{k-1}, u))] \right], \quad k = N, \dots, 1 \quad (8)$$

при начальном условии $D_{N+1}(v_N) = \emptyset$. Здесь $v_k = \chi_k(v_{k-1}, u_k)$, а χ_k – некоторая функция, реализующая переход из состояния v_{k-1} в v_k под действием управления u_k .

Задачу о поиске маршрутов будем решать в предположении, что граф, моделирующий компьютерную сеть, не содержит контуров. В противном случае необходимо будет ввести цикл по возможным модификациям наборов $D(x)$, как это сделано в скалярном [7] или векторном [3] вариантах алгоритма Форда-Беллмана поиска оптимальных путей на графе произвольного вида.

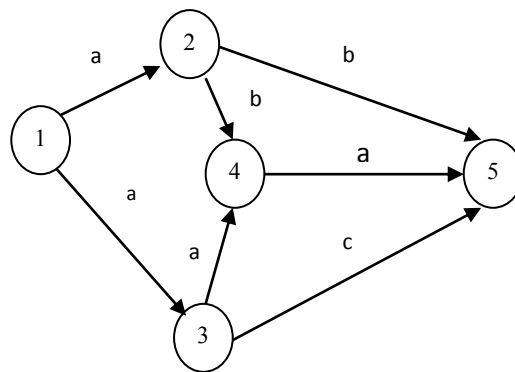


Рисунок. Граф, моделирующий компьютерную сеть

Перепишем соотношение (8) в более кратком виде, предпочтительном для данной задачи:

$$D(x) = C \left[\bigcup_{y \in \Gamma^-(x)} [(x, y) \& D(y)] \right], \quad (9)$$

где $\Gamma^-(x)$ – множество конечных вершин дуг вида (x, y) .

ЛИТЕРАТУРА

Рассмотрим задачу на графе, представленном на рисунке. В нём отношения между ребрами графа заданы в виде отношения «меньше» между указанными символьными оценками (в алфавитном порядке).

Построим множества нехудших путей от вершины $s = 1$ до $t = 5$, основываясь на рекуррентном соотношении (9) и начальном условии $D_5 = \emptyset$.

1) $x = 4$. $D_4 = \{4 - 5\}$ с оценкой (a) – единственный путь из 4 в 5.

2) $x = 3$. Так как $\Gamma^-(3) = \{4, 5\}$, при нахождении D_3 надо рассмотреть два варианта: путь $(3, 4) \& D_4$ с оценкой (a, a) и путь $(3, 5)$ с оценкой $(0', c)$. Поскольку эти варианты конфликтуют, то в выбор помещаем обе альтернативы:

$$D_3 = \{3 - 4 - 5, 3 - 5\} = \{D_3^1, D_3^2\}.$$

3) $x = 2$. $D_2 = C[2 \& D_4 \mid (a, b), 2 - 5 \mid (0', b)] = (2 - 5)$, т.к. по первой оценке доминирует альтернатива $2 - 5$, её и включим в выбор.

4) $x = 1$. $D_1 = C[1 \& D_3^1 \mid (a, a, a), 1 \& D_3^2 \mid (0', a, c), 1 \& D_2 \mid (0', a, b)] = \{1 - 3 - 4 - 5, 1 - 2 - 5\}$, т.к. третья альтернатива доминирует над второй.

Работа выполнена в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технического комплекса России на 2007-2013 гг.» по гос. контракту № 16.516.11.6040 от 21.04.11 г.

1. Бугаев, Ю. В. Обобщение схемы динамического программирования [Текст] / Ю. В. Бугаев, С. В. Чикунов // Автоматика и телемеханика. -2009. -№ 2. -С. 90 – 100.

2. Сысоев, В. В. Структурные и алгоритмические модели автоматизированного проектирования производства изделий электронной техники [Текст] / В. В. Сысоев. - Воронеж: ВТИ, 1993. - 207 с.

3. Бугаев, Ю. В. Применение прямого обобщения скалярных алгоритмов в векторной оптимизации на графах [Текст] / Ю. В. Бугаев // Дискр. матем. -2001. Т. 13. Вып. 3. - С. 110 – 124.

4. Орлов, А. И. Теория принятия решений [Текст]: учебное пособие / А. И. Орлов. - М.: Изд-во «Экзамен», 2005. - 656 с.

5. Юдин, Д. Б. Вычислительные методы теории принятия решений [Текст] / Д. Б. Юдин. - М.: Наука, 1989. - 316 с.

6. Семенов, Ю.А. Алгоритмы телекоммуникационных сетей. Часть 2. Протоколы и алгоритмы маршрутизации в Internet [Текст] / Ю.А. Семенов // Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2007. - С. 390-432.

7. Липский, В. Комбинаторика для программистов [Текст] / В. Липский., пер. с польск. - М.: Мир, 1988. - 213 с.