УДК 519.6

А.Г. Ченцов, А.А. Ченцов

К ВОПРОСУ О НАХОЖДЕНИИ ЗНАЧЕНИЯ МАРШРУТНОЙ ЗАДАЧИ С ОГРАНИЧЕНИЯМИ*

Прототипом задач маршрутизации перемещений с ограничениями, возникающих в различных областях практической деятельности, является известная труднорешаемая задача коммивояжера (ЗК). В связи с решением ЗК и задач такого типа отметим метод ветвей и границ, а также динамическое программирование (ДП), однако из-за трудностей вычислительного характера (ЗК — одна из классических NP-полных задач) активно используются эвристические алгоритмы, особенно в тех вариантах ЗК, которые не являются «метрическими» (когда матрица затрат не определяется функцией расстояния). В частности, широко применяются варианты «жадного» алгоритма, реализующие в нужной форме идею оптимизации текущих затрат.

Задачи маршрутизации, возникающие в приложениях, обычно сопровождаются большим числом разнообразных ограничений, что приводит к применению эвристических алгоритмов с безусловным соблюдением упомянутых ограничений. Среди последних особо отметим так называемые условия предшествования (условия типа «одно после другого»), которые удается [1, § 4.9] использовать для снижения сложности вычислений при решении задачи по методу ДП. С другой стороны, данные условия часто встречаются в прикладных задачах, таких, в частности, как задачи, ориентированные на снижение облучения персонала АЭС (атомная энергетика), и задачи, связанные с управлением при листовой резке на станках с числовым программным управлением (ЧПУ) (см. [2, 3]). Так, например, в последнем случае для деталей с несколькими внутренними контурами вначале требуется выполнить резку именно внутренних контуров, а не внешних. На этапе раскроя возможно также размещение одних деталей внутри других, и в этом случае (схема «матрешки») внутренние детали следует вырезать раньше внешнего контура объемлющей детали. Упомянутые обстоятельства определяют особую роль условий предшествования: связанные с ними ограничения, с одной стороны, достаточно широко распространены, а с другой, могут быть использованы в своеобразном «положительном» направлении (в смысле снижения сложности вычислений). Другие ограничения, связанные с потребностями прикладных задач, являются на сегодняшний день серьезными осложняющими факторами, что делает эвристические алгоритмы еще более значимыми, хотя качество последних не всегда удовлетворительно.

В связи с этим представляет особый интерес разработка методов, позволяющих оценивать качество эвристик, что, собственно, и нашло свое отражение в многочисленных исследованиях по построению приближенных алгоритмов [4]. Еще одна возможность связана с попыткой найти значение маршрутной задачи, т.е. оптимальный результат, с которым можно сравнивать результаты, доставляемые эвристиками, и таким образом оценить качество последних. Именно такая цель ставится в настоящей работе; для ее достижения предполагается задействовать (в условиях ограничений различных типов) аппарат широко понимаемого ДП. Заметим, что основное затруднение, касающееся непосредственного применения ДП для поиска оптимальных решений задач маршрутизации, связано с дефицитом

---

* Работа выполнена при финансовой поддержке РФФИ (грант 15-01-07909).

памяти ЭВМ: требуется хранить массив значений функции Беллмана (в [1, § 4.9] показано, что упомянутый массив может насчитываться не полностью, если рассматривается задача с условиями предшествования; однако и в данной, «усеченной», версии требования к памяти вычислителя остаются жесткими). В настоящей работе показано, что если ограничиться поиском значения (экстремума) задачи, то нагрузку на память вычислителя можно существенно ослабить, введя «послойную» процедуру с обновлениями: слой функции Беллмана, не актуальный с точки зрения поиска значения задачи, удаляется и размещается новый слой функции. Целью данного исследования является нахождение значения (экстремума) по методу ДП для, возможно, большей размерности исходной задачи (заметим, что если строить оптимальное решение посредством ДП, то вышеупомянутое удаление слоев функции Беллмана недопустимо). Это значение предполагается использовать для оценивания качества эвристик.

В связи с применением ДП для решения ЗК отметим работы [5, 6]. Метод ветвей и границ [7] получил широкое распространение в задачах дискретной оптимизации и, в частности, при решении ЗК. Отметим обстоятельные обзоры имеющихся методов решения ЗК и задач такого типа в [4, 8–10]. В то же время имеется ряд работ инженерной направленности (см., в частности, [2, 3]), в которых развиваются подходы и эвристические методы решения прикладных задач со специфическими ограничениями различных типов. Рассматривая последнее направление, в настоящей работе опишем общий вариант ДП: речь идет о задаче последовательного обхода мегаполисов при условиях предшествования, причем и функции стоимости, и сами «текущие» ограничения зависят от списка заданий. В настоящей работе данный список отвечает заданиям, которые еще не выполнены на данный момент времени. Однако к этой постановке легко сводится случай, когда (при формировании функций стоимости и развивающихся в процессе реализации конкретных «маршрутных» решений динамических ограничений) существенна аналогичная зависимость от списка уже выполненных заданий (работ).

1. Некоторые обозначения общего характера

Если S — множество, то $\mathrm{P}(S)$ и $\mathrm{P}'(S)$ обозначим соответственно семейства всех и всех непустых подмножеств множества S; $\mathrm{Fin}(S)$ — семейство всех конечных множеств из $\mathrm{P}'(S)$ (элементы $\mathrm{Fin}(S)$ — непустые конечные подмножества S и только они). Произвольным объектам x и y сопоставляем единственное множество $\{x; y\}$, содержащее x, y и не содержащее никаких других элементов ($\{x; y\}$ — неупорядоченная пара объектов x, y). Объекту z сопоставляем синглетон $\{z\} \stackrel{\Delta}{=} \{z; z\}$ (здесь и ниже $\stackrel{\Delta}{=}$ — равенство по определению), содержащий z. Как обычно в теории множеств [11, с. 67], любым двум объектам p и q сопоставляем упорядоченную пару (УП) $(p, q) \stackrel{\Delta}{=} \{\{p\}; \{p; q\}\}$ с первым элементом p и вторым q. Если z — какая-либо УП, то $\mathrm{pr}_1(z)$ и $\mathrm{pr}_2(z)$ обозначим соответственно первый и второй элементы z, однозначно определяемые условием $z = (\mathrm{pr}_1(z), \mathrm{pr}_2(z))$; при $z \in A \times B$, где A и B — множества, имеем $\mathrm{pr}_1(z) \in A$ и $\mathrm{pr}_2(z) \in B$. Триплет (a, b, c) произвольных объектов a, b и c определяется [12, с. 17] в виде специальной УП: $(a, b, c) \stackrel{\Delta}{=} ((a, b), c)$. В связи с этим для любых непустых множеств A, B и C, используя определение [12, с. 17], имеем $A \times B \times C \stackrel{\Delta}{=} (A \times B) \times C$; если при этом $g : A \times B \times C \to D$, где D — непустое множество, то при $x \in A \times B$ и $y \in C$ определено значение $g(x, y) \in D$, отвечающее аргументу $(x, y) = (\mathrm{pr}_1(x), \mathrm{pr}_2(x), y)$ и обозначаемое также $g(\mathrm{pr}_1(x),\ \mathrm{pr}_2(x), y)$.

ISSN 0572-2691

В дальнейшем $\mathbf{R}$ — вещественная прямая, $\mathbf{N} \overset{\Delta}{=} \{1; 2;...\} \in \mathcal{P}'(\mathbf{R})$ и $\mathbf{N}_0 \overset{\Delta}{=} \{0\} \cup \cup \mathbf{N} = \{0; 1; 2;...\} \in \mathcal{P}'(\mathbf{R})$; полагаем

$$\overline{p, q} \overset{\Delta}{=} \{i \in \mathbf{N}_0 \mid (p \le i) \,\&\, (i \le q)\} \quad \forall p \in \mathbf{N}_0 \quad \forall q \in \mathbf{N}_0.$$

Ясно, что $\overline{1, 0} = \varnothing$ (пустое множество) и $\overline{1, m} = \{k \in \mathbf{N} \mid k \le m\} \quad \forall m \in \mathbf{N}$. Каждому непустому конечному множеству K сопоставляем его мощность (количество элементов) $|K| \in \mathbf{N}$; полагаем $|\varnothing| \overset{\Delta}{=} 0$. Если K — непустое конечное множество, то $(\mathrm{bi})[K]$ обозначим (непустое) множество всех биекций множества $\overline{1, |K|}$ на K.

В дальнейшем $\mathcal{R}_+[T]$ обозначим множество всех функций, действующих из непустого множества T в полуось $[0, \infty[\overset{\Delta}{=} \{\xi \in \mathbf{R} \mid 0 \le \xi\}$; в качестве T используем далее конечное множество.

2. Постановка маршрутной задачи

Фиксируем непустое множество X, точку $x^0 \in X$, (натуральное) число $N \in \mathbf{N}$, $N \ge 2$, и множества

$$M_1 \in \mathrm{Fin}(X), ..., M_N \in \mathrm{Fin}(X), \tag{1}$$

называемые далее мегаполисами. В отношении (1) полагаем, что

$$(x^0 \notin M_j \quad \forall j \in \overline{1, N}) \,\&\, (M_p \cap M_q = \varnothing \quad \forall p \in \overline{1, N} \;\; \forall q \in \overline{1, N} \setminus \{p\}).$$

Кроме того, полагаем заданными отношения

$$\mathbb{M}_1 \in \mathcal{P}'(M_1 \times M_1), ..., \mathbb{M}_N \in \mathcal{P}'(M_N \times M_N), \tag{2}$$

элементы которых (а это — УП) определяют возможные варианты выполнения (внутренних) работ при посещении мегаполисов (1). Рассмотрим процессы вида

$$x^0 \to (x_1^{(1)} \in M_{\alpha(1)} \mapsto x_2^{(1)} \in M_{\alpha(1)}) \to ... \to (x_1^{(N)} \in M_{\alpha(N)} \mapsto x_2^{(N)} \in M_{\alpha(N)}), \tag{3}$$

где α — перестановка множества $\overline{1, N}$, $(x_1^{(1)}, x_2^{(1)}) \in \mathbb{M}_{\alpha(1)}, ..., (x_1^{(N)}, x_2^{(N)}) \in \mathbb{M}_{\alpha(N)}$. При естественных переобозначениях (3) сводится к виду

$$z^0 \to z^{(1)} \in \mathbb{M}_{\alpha(1)} \to ... \to z^{(N)} \in \mathbb{M}_{\alpha(N)}, \tag{4}$$

где (здесь и ниже) $z^0 \overset{\Delta}{=} (x^0, x^0)$. Будем считать, что и выбор α, и выбор $z^{(1)}, ..., z^{(N)}$ стеснены дополнительными ограничениями, для описания которых введем некоторые новые обозначения. Если $j \in \overline{1, N}$, то полагаем

$$\overline{\mathrm{M}}_j \overset{\Delta}{=} \{\mathrm{pr}_1(z) : z \in \mathbb{M}_j\}, \;\; \mathbf{M}_j \overset{\Delta}{=} \{\mathrm{pr}_2(z) : z \in \mathbb{M}_j\}$$

(у каждого мегаполиса выделены множества всех возможных пунктов прибытия и отправления соответственно (3)). Тогда

$$\mathbb{X} \overset{\Delta}{=} \{x^0\} \cup \left(\bigcup_{i=1}^{N} M_i\right) \in \mathrm{Fin}(X), \;\; \mathbf{X} \overset{\Delta}{=} \{x^0\} \cup \left(\bigcup_{i=1}^{N} \mathbf{M}_i\right) \in \mathrm{Fin}(\mathbb{X}). \tag{5}$$

С учетом (5) фиксируем мультифункции

$$A_1 : \mathbf{X} \times \mathfrak{N} \to \mathrm{P}'(M_1), \ldots, A_N : \mathbf{X} \times \mathfrak{N} \to \mathrm{P}'(M_N), \tag{6}$$

где $\mathfrak{N} \overset{\Delta}{=} \mathrm{P}'(\overline{1,\,N})$. Если $j \in \overline{1,\,N}$, $x \in \mathbf{X}$ и $K \in \mathfrak{N}$, то $A_j(x, K)$ определяет множество всех точек M_j, в которые возможно перемещение из x при условии, что K определяет список еще невыполненных заданий. Всюду в дальнейшем предполагается, что

$$A_j(x, K) \cap \overline{\mathrm{M}}_j \neq \varnothing \quad \forall x \in \mathbf{X} \ \forall K \in \mathfrak{N} \ \forall j \in K \tag{7}$$

(условие (7) позволяет стыковать внешние перемещения и внутренние работы).

Пусть $\mathbf{P} \overset{\Delta}{=} (\mathrm{bi})[\overline{1,\,N}]$ (множество всех перестановок индексного множества $\overline{1,\,N}$). Напомним, что в (3), (4) $\alpha \in \mathbf{P}$. Полагаем, однако, что выбор α может быть стеснен условиями предшествования. В связи с этим условимся, что при $\alpha \in \mathbf{P}$ обратная по отношению к α перестановка обозначается $\alpha^{-1} : \alpha^{-1} \in \mathbf{P}$ и

$$\alpha(\alpha^{-1}(k)) = \alpha^{-1}(\alpha(k)) = k \quad \forall k \in \overline{1,\,N}.$$

Фиксируем множество $\mathbf{K} \in \mathrm{P}(\overline{1,\,N} \times \overline{1,\,N})$; элементы $\mathbf{K}$, являющиеся УП, именуем адресными парами. Если $z \in \mathbf{K}$, то $\mathrm{pr}_1(z)$ имеет смысл отправителя, а $\mathrm{pr}_2(z)$ — получателя адресной пары z. Полагаем далее, что

$$\forall \mathbf{K}_0 \in \mathrm{P}'(\mathbf{K}) \ \exists z_0 \in \mathbf{K}_0 : \mathrm{pr}_1(z_0) \neq \mathrm{pr}_2(z) \ \ \forall z \in \mathbf{K}_0.$$

Тогда [1, ч. 2] имеем следующее свойство:

$$\mathbf{A} \overset{\Delta}{=} \{\alpha \in \mathbf{P} \,|\, \forall z \in \mathbf{K} \ \ \forall t_1 \in \overline{1,\,N} \ \ \forall t_2 \in \overline{1,N}$$

$$((\mathrm{pr}_1(z) = \alpha(t_1))\,\&\,(\mathrm{pr}_2(z) = \alpha(t_2))) \Rightarrow (t_1 < t_2)\} =$$

$$= \{\alpha \in \mathbf{P} \mid \alpha^{-1}(\mathrm{pr}_1(z)) < \alpha^{-1}(\mathrm{pr}_2(z)) \ \ \forall z \in \mathbf{K}\} \in \mathrm{P}'(\mathbf{P}), \tag{8}$$

т.е. $\mathbf{K}$-допустимые по предшествованию маршруты (перестановки индексов) существуют. Как видно из (3), (4), маршрут (перестановка индексов) еще не определяет течение процесса, поэтому введем понятие траектории, согласованной с тем или иным маршрутом. Сначала условимся, что $\mathbf{Z}$ — def-множество всех кортежей

$$(z_i)_{i \in \overline{0,N}} : \overline{0,\,N} \to \mathfrak{X} \times \mathbf{X}.$$

Теперь, ориентируясь на вариант (4), при $\alpha \in \mathbf{P}$ введем в рассмотрение (непустое) множество

$$\mathrm{Z}_\alpha \overset{\Delta}{=} \{(z_i)_{i \in \overline{0,N}} \in \mathbf{Z} \,|\, (z_0 = z^0)\,\&\,(z_\tau \in \mathfrak{M}_{\alpha(\tau)} \ \ \forall \tau \in \overline{1,\,N})\,\&$$

$$\&\,(\mathrm{pr}_1(z_s) \in A_{\alpha(s)}(\mathrm{pr}_2(z_{s-1}), \{\alpha(j) : j \in \overline{s,N}\}) \ \forall s \in \overline{1,\,N})\} \in \mathrm{P}'(\mathbf{Z}). \tag{9}$$

Тогда, в частности, в виде $\mathbf{D} \overset{\Delta}{=} \{(\alpha, (z_i)_{i \in \overline{0,N}}) \in \mathbf{A} \times \mathbf{Z} \,|\, (z_i)_{i \in \overline{0,N}} \in \mathrm{Z}_\alpha\}$ имеем непустое конечное множество допустимых решений (ДР) исследуемой ниже экстремальной задачи. Для ее точной формулировки введем функции стоимости, фиксируя

$$\mathbf{c} \in \mathrm{R}_+[\mathbf{X} \times \mathfrak{X} \times \mathfrak{N}],\ c_1 \in \mathrm{R}_+[\mathfrak{X} \times \mathbf{X} \times \mathfrak{N}], \ldots, c_N \in \mathrm{R}_+[\mathfrak{X} \times \mathbf{X} \times \mathfrak{N}],\ f \in \mathrm{R}_+[\mathbf{X}] \tag{10}$$

 ISSN 0572-2691

(реально функции стоимости задаются на подмножествах $\mathbf{X} \times \mathfrak{X} \times \mathfrak{N}$, $\mathfrak{X} \times \mathbf{X} \times \mathfrak{N}$ и $\mathbf{X}$, но продолжим их произвольным образом на упомянутые три множества, что не составляет труда, но несколько упрощает обозначения). Заметим, что при $\alpha \in \mathbf{A}$, $(z_i)_{i \in \overline{0,N}} \in \mathrm{Z}_\alpha$ и $s \in \overline{1, N}$ определены значения

$$\mathbf{c}(\mathrm{pr}_2(z_{s-1}), \mathrm{pr}_1(z_s), \{\alpha(t) : t \in \overline{s, N}\}) \in [0, \infty[,$$

$$c_{\alpha(s)}(z_s, \{\alpha(t) : t \in \overline{s, N}\}) \in [0, \infty[,$$

имеем также $f(\mathrm{pr}_2(z_N)) \in [0, \infty[$. С учетом этого введем аддитивный критерий, полагая, что

$$\mathbf{C}_\alpha[(z_i)_{i \in \overline{0,N}}] \overset{\Delta}{=} \sum_{s=1}^{N} [\mathbf{c}(\mathrm{pr}_2(z_{s-1}), \mathrm{pr}_1(z_s), \{\alpha(t) : t \in \overline{s, N}\}) + c_{\alpha(s)}(z_s, \{\alpha(t) : t \in \overline{s, N}\})] +$$

$$+ f(\mathrm{pr}_2(z_N)) \quad \forall \alpha \in \mathbf{A} \quad \forall (z_i)_{i \in \overline{0,N}} \in \mathrm{Z}_\alpha. \tag{11}$$

Итак, посредством (11) оценивается каждое ДР из $\mathbf{D}$. Задаче

$$\mathbf{C}_\alpha[(z_i)_{i \in \overline{0,N}}] \to \min, \ (\alpha, (z_i)_{i \in \overline{0,N}}) \in \mathbf{D}, \tag{12}$$

сопоставляется значение $V \in [0, \infty[$ (экстремум критерия):

$$V \overset{\Delta}{=} \min_{\alpha \in \mathbf{A}} \min_{(z_i)_{i \in \overline{0,N}} \in \mathrm{Z}_\alpha} \mathbf{C}_\alpha[(z_i)_{i \in \overline{0,N}}]. \tag{13}$$

В настоящей работе ставим своей целью определение V (13). Представляется, что данная цель достижима при меньших ресурсах памяти в сравнении с полным решением задачи (12) по методу ДП; найденное значение V предполагается использовать для оценивания качества эвристических решений данной (весьма сложной) маршрутной задачи. Схему построений, направленную на определение V, изложим ниже в виде соответствующего алгоритма на функциональном уровне.

3. Некоторые построения общего характера

В настоящем разделе кратко коснемся вопроса о представлении функции Беллмана, используя вариант расширения задачи (12), соответствующий идейно [13–15].

Введем сначала при $K \in \mathfrak{N}$ множество $\mathbf{Z}_K$ всех кортежей

$$(z_i)_{i \in \overline{0,|K|}} : \overline{0, |K|} \to \mathfrak{X} \times \mathbf{X}.$$

Тогда полагаем при $x \in \mathbf{X}$, $K \in \mathfrak{N}$ и $\alpha \in (\mathrm{bi})[K]$, что

$$\mathrm{Z}(x, K, \alpha) \overset{\Delta}{=} \{(z_i)_{i \in \overline{0,|K|}} \in \mathbf{Z}_K \mid (z_0 = (x, x)) \,\&\, (z_t \in \mathfrak{M}_{\alpha(t)} \ \forall t \in \overline{1, |K|}) \,\&$$

$$\& \, (\mathrm{pr}_1(z_s) \in A_{\alpha(s)}(\mathrm{pr}_2(z_{s-1}), \{\alpha(j) : j \in \overline{s, |K|}\}) \ \forall s \in \overline{1, |K|})\}. \tag{14}$$

Учитывая определение $\mathbf{P}$, получаем из (9), (14), что $\mathrm{Z}_\alpha = \mathrm{Z}(x^0, \overline{1, N}, \alpha)$ при $\alpha \in \mathbf{P}$. Для построения нужного расширения исходной задачи потребуется редукция ограничений [1, ч. 2]. С этой целью введем отображение $\mathbf{I}$, действующее в $\mathfrak{N}$ по правилу

$$\mathbf{I}(K) \overset{\Delta}{=} K \setminus \{\mathrm{pr}_2(z) : z \in \Xi[K]\},$$

где $\Xi[K] \overset{\Delta}{=} \{z \in \mathbf{K} \,|\, (\mathrm{pr}_1(z) \in K) \,\&\, (\mathrm{pr}_2(z) \in K)\}$ при $K \in \mathfrak{N}$. В этих терминах вводится [1, ч. 2] другой тип допустимости маршрутов: допустимость по вычеркиванию. Итак, при $K \in \mathfrak{N}$ получаем, что [1, предложения 2.2.2, 2.2.3]

$$(\mathbf{I} - \mathrm{bi})[K] \overset{\Delta}{=} \{\alpha \in (\mathrm{bi})[K] \,|\, \alpha(k) \in \mathbf{I}(\{\alpha(t) : t \in \overline{k, |K|}\}) \ \ \forall k \in \overline{1, |K|}\} \in \mathrm{P}'((\mathrm{bi})[K]), \quad (15)$$

т.е. (15) есть множество (непустое) допустимых по вычеркиванию частичных маршрутов посещения мегаполисов $M_k,\ k \in K.$ Напомним, что [1, ч. 2]

$$\mathbf{A} = (\mathbf{I} - \mathrm{bi})[\overline{1, N}] = \{\alpha \in \mathbf{P} \,|\, (\alpha(1) \in \mathbf{I}(\overline{1, N})) \,\&$$

$$\& \, (\alpha(k) \in \mathbf{I}(\overline{1, N} \setminus \{\alpha(t) : t \in \overline{1, k-1}\}) \ \ \forall k \in \overline{2, N})\} \in \mathrm{P}'(\mathbf{P}). \quad (16)$$

Из (14), (15) получаем, что при $x \in \mathbf{X}$ и $K \in \mathfrak{N}$

$$\mathbf{D}_K(x) \overset{\Delta}{=} \{(\alpha, (z_i)_{i \in \overline{0,|K|}}) \in (\mathbf{I} - \mathrm{bi})[K] \times \mathbf{Z}_K \,|\, (z_i)_{i \in \overline{0,|K|}} \in \mathrm{Z}(x, K, \alpha)\} \in \mathrm{Fin}((\mathrm{bi})[K] \times \mathbf{Z}_K)$$

есть непустое множество ДР частичной задачи, отвечающей позиции (x, K). Для формулировки последней введем соответствующий аддитивный критерий: при $x \in \mathbf{X}$, $K \in \mathfrak{N}$, $\alpha \in (\mathbf{I} - \mathrm{bi})[K]$ и $(z_i)_{i \in \overline{0,|K|}} \in \mathrm{Z}(x, K, \alpha)$ полагаем, что

$$\mathfrak{C}_\alpha((z_i)_{i \in \overline{0,|K|}} \,|\, K) \overset{\Delta}{=} \sum_{s=1}^{|K|} [\mathbf{c}(\mathrm{pr}_2(z_{s-1}), \mathrm{pr}_1(z_s), \{\alpha(t) : t \in \overline{s, |K|}\}) +$$

$$+ c_{\alpha(s)}(z_s, \{\alpha(t) : t \in \overline{s, |K|}\})] + f(\mathrm{pr}_2(z_{|K|})). \quad (17)$$

С учетом (17) сопоставляем каждой позиции (x, K), $x \in \mathbf{X}$, $K \in \mathfrak{N}$, следующую частичную задачу:

$$\mathfrak{C}_\alpha((z_i)_{i \in \overline{0,|K|}} \,|\, K) \to \min, \ \ (\alpha, (z_i)_{i \in \overline{0,|K|}}) \in \mathbf{D}_K(x), \quad (18)$$

ограничения которой совместны, а значение (экстремум)

$$v(x, K) \overset{\Delta}{=} \min_{\alpha \in (\mathbf{I} - \mathrm{bi})[K]} \ \min_{(z_i)_{i \in \overline{0,|K|}} \in \mathrm{Z}(x, K, \alpha)} \mathfrak{C}_\alpha((z_i)_{i \in \overline{0,|K|}} \,|\, K) \in [0, \infty[. \quad (19)$$

Из (19), в частности, получаем (см. (16) и ранее упомянутые свойства пучков траекторий), что

$$V = v(x^0, \overline{1, N}), \quad (20)$$

а сама основная задача (12) может быть погружена в семейство частичных задач (18). Наконец, полагаем, что

$$v(x, \varnothing) = f(x) \ \ \forall x \in \mathbf{X}. \quad (21)$$

Таким образом, посредством (19), (21) на множестве $\mathbf{X} \times \mathrm{P}(\overline{1, N})$ определена функция Беллмана v, $v : \mathbf{X} \times \mathrm{P}(\overline{1, N}) \to [0, \infty[$, для которой имеет место важное равенство (20). Итак, в виде семейства задач (18), дополненного соглашением (21), получаем расширение исходной задачи (12). С учетом (7) получаем, что

$$\mathfrak{A}_j(x, K) \overset{\Delta}{=} \{z \in \mathfrak{M}_j \,|\, \mathrm{pr}_1(z) \in A_j(x, K)\} \neq \varnothing \ \ \forall x \in \mathbf{X} \ \ \forall K \in \mathfrak{N} \ \ \ \forall j \in K. \quad (22)$$

По аналогии с [13–15] устанавливается следующая теорема.

 ISSN 0572-2691

Теорема. Если $x \in \mathbf{X}$ и $K \in \mathfrak{N}$, то

$$v(x, K) = \min_{j \in \mathbf{I}(K)} \min_{z \in \mathcal{K}_j(x,K)} [\mathbf{c}(x, \mathrm{pr}_1(z), K) + c_j(z, K) + v(\mathrm{pr}_2(z), K \setminus \{j\})].$$

Из (20) и теоремы вытекает, в частности, что

$$V = \min_{j \in \mathbf{I}(\overline{1,N})} \min_{z \in \mathcal{K}_j(x^0, \overline{1,N})} [\mathbf{c}(x^0, \mathrm{pr}_1(z), \overline{1, N}) + c_j(z, \overline{1, N}) + v(\mathrm{pr}_2(z), \overline{1, N} \setminus \{j\})]. \quad (23)$$

Теорема определяет уравнение Беллмана, а формула (23) характеризует глобальный экстремум основной задачи.

4. Рекуррентная процедура построения слоев функции Беллмана

Ниже используется экономичная версия метода ДП, восходящая к [1, § 4.9] (более поздние варианты см. в [13–15]). Речь идет о том, чтобы отказаться (при наличии условий предшествования) от построения всего массива значений функции v. В связи с этим полагаем, что

$$\mathrm{G} \triangleq \{K \in \mathfrak{N} \mid \forall z \in \mathbf{K} \ \ (\mathrm{pr}_1(z) \in K) \Rightarrow (\mathrm{pr}_2(z) \in K)\}. \quad (24)$$

Множества, элементы семейства G, именуем существенными списками заданий. Упомянутые списки ранжируем по мощности; полагаем, что $\mathrm{G}_s \triangleq \{K \in \mathrm{G} \mid s = |K|\}$ $\forall s \in \overline{1, N}$. При этом, конечно, семейства G_s, $s \in \overline{1, N}$, образуют разбиение G. Ясно, что $\mathrm{G}_N = \{\overline{1, N}\}$ (синглетон, содержащий множество $\overline{1, N}$). С другой стороны, $\mathbf{K}_1 \triangleq \{\mathrm{pr}_1(z) : z \in \mathbf{K}\}$ определяет G_1 :

$$\mathrm{G}_1 = \{\{t\} : t \in \overline{1, N} \setminus \mathbf{K}_1\}. \quad (25)$$

Заметим также, что [13, (6.2)] имеет место свойство

$$\mathrm{G}_{s-1} = \{K \setminus \{t\} : K \in \mathrm{G}_s, t \in \mathbf{I}(K)\} \ \ \forall s \in \overline{2, N}. \quad (26)$$

При этом [1, предложение 4.2.9] $\mathrm{G}_s \neq \varnothing \ \forall s \in \overline{1, N}$. В результате получим (см. (25), (26)) рекуррентную процедуру $\mathrm{G}_N \to \mathrm{G}_{N-1} \to \ldots \to \mathrm{G}_1$ с известными крайними семействами G_N и G_1. При этом [13, с. 69] $\overline{1, N} \setminus \mathbf{K}_1 \neq \varnothing$, тогда

$$\widetilde{\mathbf{M}} \triangleq \bigcup_{i \in \overline{1,N} \setminus \mathbf{K}_1} \mathbf{M}_i \in \mathrm{P}'(\mathbf{X})$$

и, как следствие, определено (непустое) множество

$$D_0 \triangleq \{(x, \varnothing) : x \in \widetilde{\mathbf{M}}\} \in \mathrm{P}'(\mathbf{X} \times \mathrm{P}(\overline{1, N})). \quad (27)$$

Кроме того, как и в [13–15], полагаем $D_N \triangleq \{(x^0, \overline{1, N})\}$ (синглетон, содержащий УП $(x^0, \overline{1, N})$). Если $s \in \overline{1, N-1}$ и $K \in \mathrm{G}_s$, то последовательно конструируем множества

$$\mathrm{J}_s(K) \triangleq \{t \in \overline{1, N} \setminus K \mid \{t\} \cup K \in \mathrm{G}_{s+1}\} \in \mathrm{P}'(\overline{1, N} \setminus K),$$

$$\mathfrak{M}_s[K] \triangleq \bigcup_{j \in \mathrm{J}_s(K)} \mathbf{M}_j \in \mathrm{P}'(\mathbf{X}),$$

$$\mathfrak{D}_s[K] \triangleq \{(x, K) : x \in \mathfrak{M}_s[K]\} \in \mathrm{P}'(\mathbf{X} \times \mathrm{G}_s).$$

Следуя [13–15], в этих обозначениях получаем, что

$$D_s \overset{\Delta}{=} \bigcup_{K \in \mathrm{G}_s} \tilde{D}_s[K] \in \mathrm{P}'(\mathbf{X} \times \mathrm{G}_s) \quad \forall s \in \overline{1,\, N-1}. \tag{28}$$

Итак, введены слои $D_0, D_1, ..., D_N$ пространства позиций, являющиеся каждый непустым множеством. При этом [13, с. 69]

$$(y, K \setminus \{k\}) \in D_{s-1} \quad \forall s \in \overline{1,\, N} \quad \forall K \in \mathrm{G}_s \quad \forall k \in \mathbf{I}(K) \quad \forall y \in \mathbf{M}_k. \tag{29}$$

Из (29) следует, в частности, понятное свойство:

$$(\mathrm{pr}_2(z), K \setminus \{j\}) \in D_{s-1} \quad \forall s \in \overline{1,\, N} \quad \forall (x, K) \in D_s \quad \forall j \in \mathbf{I}(K) \quad \forall z \in \tilde{A}_j(x, K). \tag{30}$$

С учетом (27), (28) вводим такие определения: если $s \in \overline{0,\, N},$ то полагаем, что $v_s \in \mathrm{R}_+[D_s]$ — такое отображение, что

$$v_s(x, K) \overset{\Delta}{=} v(x, K) \quad \forall (x, K) \in D_s; \tag{31}$$

иными словами, v_s определяем как сужение функции v на множество D_s. Таким образом, введены функции

$$v_0 : D_0 \to [0, \infty[,\ v_1 : D_1 \to [0, \infty[, ..., v_N : D_N \to [0, \infty[; \tag{32}$$

при этом функция v_N в (32) определяется единственным значением (20):

$$V = v_N(x^0, \overline{1,\, N}). \tag{33}$$

Функция v_0 определяется в силу (21), (27) очень просто:

$$v_0(x, \varnothing) = f(x) \quad \forall x \in \tilde{\mathbf{M}}. \tag{34}$$

С учетом (30) имеем, что при $s \in \overline{1,\, N},$ $(x, K) \in D_s,$ $j \in \mathbf{I}(K)$ и $z \in \tilde{A}_j(x, K)$ определено значение $v_{s-1}(\mathrm{pr}_2(z), K \setminus \{j\}) \in [0, \infty[$. Легко видеть (с учетом теоремы), что справедливо следующее предложение.

Предложение. Если $s \in \overline{1,\, N},$ то функция v_{s-1} преобразуется в v_s по следующему правилу:

$$v_s(x, K) = \min_{j \in \mathbf{I}(K)} \min_{z \in \tilde{A}_j(x,K)} [\mathbf{c}(x, \mathrm{pr}_1(z), K) + c_j(z, K) + v_{s-1}(\mathrm{pr}_2(z), K \setminus \{j\})] \\ \forall (x, K) \in D_s. \tag{35}$$

Таким образом, получаем рекуррентную процедуру

$$v_0 \to v_1 \to ... \to v_N \tag{36}$$

построения всех функций (32), именуемых далее слоями функции Беллмана: функция v_0 в (36) известна (см. (34)), а преобразование v_{s-1} в v_s, где $s \in \overline{1,\, N},$ осуществляется по правилу (35). Тогда, в частности, согласно (33) определено значение $V,$ т.е. глобальный экстремум основной задачи.

Отметим теперь, что в силу (33), (35)

$$V = \min_{j \in \mathbf{I}(\overline{1,N})} \min_{z \in \tilde{A}_j(x^0, \overline{1,N})} [\mathbf{c}(x^0, \mathrm{pr}_1(z), \overline{1,N}) + c_j(z, \overline{1,\, N}) + v_{N-1}(\mathrm{pr}_2(z), \overline{1,\, N} \setminus \{j\})]. \tag{37}$$

Из (37) видно, что для определения V требуются не все функции (32), а только функция v_{N-1}. Аналогичным образом из (35) имеем следующее свойство: при $s \in \overline{1,\, N}$ для построения v_s достаточно функции v_{s-1}. Это существенно с точки зрения ресурсов памяти используемой ЭВМ. В связи с этим рассмотрим процедуру (36) с целью определения V (13).

 ISSN 0572-2691

Начальный (подготовительный) этап состоит в построении функции v_0 на основе формулы (34). Тем самым каждой позиции $(x, K) \in D_0$ сопоставляется значение $v_0(x, K) \in [0, \infty[$.

Последующие этапы процедуры реализуются однотипно: при $s \in \overline{1, N}$, располагая функцией v_{s-1}, значения которой находятся в памяти компьютера, осуществляется насчитывание массива значений функции v_s посредством (35), после чего новый массив значений, т.е. $(v_s(x, K) : (x, K) \in D_s)$, заносится в память компьютера вместо уже использованного массива значений $(v_{s-1}(x, K) :$ $: (x, K) \in D_{s-1})$; последний уничтожается.

Данная операция, включающая полную замену содержимого памяти, повторяется N раз. Результатом ее применения является значение V (33). Точнее, нахождение V осуществляется на основании v_{N-1}. Для этого, располагая массивом значений $(v_{N-1}(x, K) : (x, K) \in D_{N-1})$, искомое значение V рассчитываем по формуле (37).

Замечание. Упомянутая процедура с постоянным обновлением памяти не позволяет, в отличие от [1, 13–15], построить ДР, реализующее значение V, поскольку [13–15] для этого требуются все функции (32). Однако именно за счет данного обновления удается увеличить значение N, для которого V (33) может быть найдено с помощью процедуры на основе ДП.

5. Оценивание эвристического алгоритма

Располагая значением V, можем оценить качество эвристических алгоритмов, применяемых для построения ДР. Данные алгоритмы, не гарантируя близости к V по результату, позволяют быстро находить соответствующее ДР. Рассмотрим простейший вариант такого рода — модификацию «жадного» алгоритма [16].

Введем следующее правило. Итак, если $x \in \mathbf{X}$ и $K \in \mathfrak{N}$, то выбираем $\mathbf{j} \in \mathbf{I}(K)$ и $\mathbf{z} \in \mathfrak{A}_{\mathbf{j}}(x, K)$ так, что

$$\mathbf{c}(x, \mathrm{pr}_1(\mathbf{z}), K) + c_{\mathbf{j}}(\mathbf{z}, K) = \min_{j \in \mathbf{I}(K)} \min_{z \in \mathfrak{A}_j(x,K)} [\mathbf{c}(x, \mathrm{pr}_1(z), K) + c_j(z, K)]; \qquad (38)$$

при этом в качестве x может использоваться либо x^0, либо $\mathrm{pr}_2(z)$, где $z \in \mathfrak{M}_l$ и $l \in \overline{1, N} \setminus K$.

Алгоритм действует следующим образом. Находясь в начальной позиции $(x^0, \overline{1, N})$, определяем $\mathbf{z}^{(0)} \in \mathfrak{X} \times \mathbf{X}$ посредством z^0 (см. разд. 1), т.е. полагаем $\mathbf{z}^{(0)} \overset{\Delta}{=} z^0$. Далее используем (38) при $x = \mathrm{pr}_2(\mathbf{z}^{(0)})$ и $K = \overline{1, N}$; находим в этих условиях $\mathbf{j}$ и $\mathbf{z}$, после чего полагаем $\mathbf{j}_1 = \mathbf{j}$ и $\mathbf{z}^{(1)} = \mathbf{z}$.

Пусть $s \in \overline{1, N}$ и уже построены $\mathbf{j}_t \in \overline{1, N}$, $t \in \overline{1, s}$, и $\mathbf{z}^{(t)} \in \mathfrak{X} \times \mathbf{X}$, $t \in \overline{0, s}$. При этом $\mathbf{z}^{(0)} = z^0$, УП $(\mathbf{j}_1, \mathbf{z}^{(1)})$ определена выше, а при $t \in \overline{2, s}$ УП $(\mathbf{j}_t, \mathbf{z}^{(t)})$ определяется посредством (38) в следующей конкретизации:

$$x = \mathrm{pr}_2(\mathbf{z}^{(t-1)}), \ K = \overline{1, N} \setminus \{\mathbf{j}_l : l \in \overline{1, t-1}\}, \ \mathbf{j}_t = \mathbf{j}, \ \mathbf{z}^{(t)} = \mathbf{z}.$$

Если $s = N$, то процедура завершена. Если же $s < N$, то для построения $\mathbf{j}_{s+1}$ и $\mathbf{z}^{(s+1)}$ используем (38) при $x = \mathrm{pr}_2(\mathbf{z}^{(s)})$ и $K = \overline{1, N} \setminus \{\mathbf{j}_l : l \in \overline{1, s}\}$, полагая в данной конкретизации, что $\mathbf{j}_{s+1} = \mathbf{j}$ и $\mathbf{z}^{(s+1)} = \mathbf{z}$.

Из упомянутой выше схемы видно, что $\mathbf{j}_t \in \overline{1, N}$ при $t \in \overline{1, N}$. Кроме того, из определения множеств (22) имеем, что $\mathbf{z}^{(t)} \in \mathfrak{X} \times \mathbf{X} \ \forall t \in \overline{0, N}$. При этом

$$(\mathbf{j}_1 \in \mathbf{I}(\overline{1, N})) \ \& \ (\mathbf{j}_s \in \mathbf{I}(\overline{1, N} \setminus \{\mathbf{j}_l : l \in \overline{1, s-1}\}) \ \ \forall s \in \overline{2, N}). \qquad (39)$$

Из (39) следует, в частности, что $\mathbf{j}_1 \in \overline{1, N}$ и $\mathbf{j}_s \in \overline{1, N} \setminus \{\mathbf{j}_l : l \in \overline{1, s-1}\} \quad \forall s \in \overline{2, N}$. Тогда, как легко видеть,

$$(\mathbf{j}_s)_{s \in \overline{1,N}} \in \mathbf{P} \tag{40}$$

(если $\mathbf{j}_k = \mathbf{j}_l$ при $k \neq l$, то среди k и l выбираем наименьшее p и наибольшее q; тогда $p < q$, а потому

$$\mathbf{j}_q \in \overline{1, N} \setminus \{\mathbf{j}_l : l \in \overline{1, q-1}\}$$

и, как следствие, $\mathbf{j}_p \neq \mathbf{j}_q$). Из (16), (39) и (40) получаем, что

$$(\mathbf{j}_s)_{s \in \overline{1,N}} \in \mathbf{A}. \tag{41}$$

Далее в силу (22) и правила выбора на основе (38) имеем, что

$$\mathbf{z}^{(j)} \in \tilde{\mathbb{M}}_j \quad \forall j \in \overline{1, N}. \tag{42}$$

Наконец, из (22) имеем по правилу (38), что

$$\mathrm{pr}_1(\mathbf{z}^{(s)}) \in A_{\mathbf{j}_s}(\mathrm{pr}_2(\mathbf{z}^{(s-1)}), \{\mathbf{j}_t : t \in \overline{s, N}\}) \quad \forall s \in \overline{1, N}. \tag{43}$$

В самом деле, $\mathbf{j}_1 \in \mathbf{I}(\overline{1, N})$, а $\overline{1, N} = \{\mathbf{j}_s : s \in \overline{1, N}\}$ и, стало быть,

$$\mathbf{j}_1 \in \mathbf{I}(\{\mathbf{j}_s : s \in \overline{1, N}\}). \tag{44}$$

Поэтому (см. (38) при $x = \mathrm{pr}_2(\mathbf{z}^{(0)})$ и $K = \overline{1, N}$) имеем, что

$$\mathbf{z}^{(1)} \in \tilde{A}_{\mathbf{j}_1}(\mathrm{pr}_2(\mathbf{z}^{(0)}), \{\mathbf{j}_t : t \in \overline{1, N}\}), \tag{45}$$

а в этом случае из (22) следует, что

$$\mathrm{pr}_1(\mathbf{z}^{(1)}) \in A_{\mathbf{j}_1}(\mathrm{pr}_2(\mathbf{z}^{(0)}), \{\mathbf{j}_t : t \in \overline{1, N}\}). \tag{46}$$

Пусть $s \in \overline{2, N}$. Тогда $s - 1 \in \overline{1, N-1}$ и $\mathbf{z}^{(s-1)} = (\mathrm{pr}_1(\mathbf{z}^{(s-1)}), \mathrm{pr}_2(\mathbf{z}^{(s-1)}))$, где согласно правилу выбора в (38)

$$\mathbf{z}^{(s)} \in \tilde{A}_{\mathbf{j}_s}(\mathrm{pr}_2(\mathbf{z}^{(s-1)}), \overline{1, N} \setminus \{\mathbf{j}_t : t \in \overline{1, s-1}\}). \tag{47}$$

С учетом (22) имеем из (47), что

$$\mathrm{pr}_1(\mathbf{z}^{(s)}) \in A_{\mathbf{j}_s}(\mathrm{pr}_2(\mathbf{z}^{(s-1)}), \overline{1, N} \setminus \{\mathbf{j}_t : t \in \overline{1, s-1}\}). \tag{48}$$

Выбор s был произвольным, и в силу (40) получили, что

$$\mathrm{pr}_1(\mathbf{z}^{(s)}) \in A_{\mathbf{j}_s}(\mathrm{pr}_2(\mathbf{z}^{(s-1)}), \{\mathbf{j}_t : t \in \overline{s, N}\}) \quad \forall s \in \overline{2, N}. \tag{49}$$

Из (45), (49) вытекает, что

$$\mathrm{pr}_1(\mathbf{z}^{(s)}) \in A_{\mathbf{j}_s}(\mathrm{pr}_2(\mathbf{z}^{(s-1)}), \{\mathbf{j}_t : t \in \overline{s, N}\}) \quad \forall s \in \overline{1, N}. \tag{50}$$

Теперь из (9), (43) и (50) имеем по выбору $\mathbf{z}^{(0)}$, что

$$(\mathbf{z}^{(s)})_{s \in \overline{0,N}} \in \mathrm{Z}_{(\mathbf{j}_s)_{s \in \overline{1,N}}}. \tag{51}$$

В свою очередь, из (41), (51) вытекает, в частности, что

$$((\mathbf{j}_s)_{s \in \overline{1,N}}, (\mathbf{z}^{(s)})_{s \in \overline{0,N}}) \in \mathbf{D}. \tag{52}$$

Итак, согласно (52) «жадный» алгоритм, определяемый правилом (38), порождает ДР основной задачи и можно оценить его близость по результату к V.

ISSN 0572-2691

6. Вычислительный эксперимент

Рассматриваемые в настоящей статье алгоритмы решения маршрутной задачи обхода мегаполисов (точный алгоритм на основе ДП и «жадный» алгоритм) реализованы в виде программ для ПЭВМ. В качестве мегаполисов в модельном примере использованы равномерные «сетки» на плоскости, получаемые при размещении 12 точек на окружностях на равных угловых расстояниях друг от друга, включая точку, имеющую нулевую угловую координату. Таким образом, каждый мегаполис M_i (1) однозначно представляется парой (O_i, R_i), где O_i — точка, являющаяся центром окружности, а R_i — радиус окружности; $i \in \overline{1, N}$. Пусть начальная точка x^0 совпадает с началом координат, т.е. $x^0 = (0, 0)$, а терминальное состояние процесса перемещений (3) связано с возвратом в точку x^0.

Пусть функции затрат $\mathbf{c}$ и f в (10) пропорциональны евклидову расстоянию, а именно: f определяется евклидовым расстоянием, а функция $\mathbf{c}$ — утроенное евклидово расстояние между соответствующими точками. Значения функций $c_1, ..., c_N, A_1, ..., A_n$ определяются в виде экстремумов «внутренних» незамкнутых задач коммивояжера, связанных с посещением всех городов мегаполиса при условии, что матрица затрат всякий раз определяется посредством евклидовых расстояний; аргументами упомянутых функций являются при этом точка прибытия в мегаполис и точка отправления из мегаполиса. Предполагается, что в данном варианте $\mathbf{c}$, $c_1, ..., c_N, A_1, ..., A_N$ зависимость от списка заданий отсутствует.

Программа написана на языке программирования C++ (использован 64-разрядный компилятор), работает под управлением 64-разрядной операционной системы семейства Windows, начиная с Windows 7, в многопотоковом режиме (вычислительная часть выполняется в отдельном от интерфейса пользователя потоке); при решении задачи на плоскости имеется возможность представления траектории движения по мегаполисам в графическом виде, увеличения отдельных участков графика и сохранения изображения в файл графического формата bmp. Исходные данные и результаты работы программы сохраняются в текстовом файле специального формата. Вычислительный эксперимент проводился на компьютере с центральным процессором Intel Core i7 объемом ОЗУ 64 Гбайт с установленной операционной системой Windows 7×64 (64 разряда) и максимальной Sp1.

Из-за ограничения объема статьи в явном виде перечислим точки мегаполисов и адресные пары, составляющие условия предшествования, а также элементы маршрутов и узлы трассы, ограничившись только графическим изображением траекторий движения.

Пусть $N = 34$ и заданы 39 адресных пар (условия предшествования). В результате применения точного алгоритма на основе ДП в его полной версии [17] (нахождение глобального экстремума (13) и его доставляющего ДР) получим следующие результаты: величина совокупных затрат $V = 4305{,}78$; время счета 9 ч. 19 мин. 43 с. График маршрута и трассы приведен на рис. 1.

В случае применения точного алгоритма на основе ДП в усеченной версии (без построения маршрута и трассы) получаем сокращение времени счета в 1,7 раза (время поиска глобального экстремума составило 5 ч. 33 мин. 45 с.).

Рассмотрим теперь результаты работы (для упомянутого выше модельного примера) «жадного» алгоритма, о котором шла речь в предыдущем разделе: величина совокупных затрат 6088,12; время счета 1 с. График маршрута и трассы приведен на рис. 2.

Как видно из приведенной информации, проигрыш «жадного» алгоритма по отношению к точному методу составил примерно 30 % при несоизмеримом выигрыше во времени счета.

Рассмотрим теперь пример использования усеченной версии точного метода для решения задачи обхода $N = 37$ мегаполисов при условии, что заданы 66 адресных пар (элементы множества $\mathbf{K}$), ограничивающих порядок посещения. Пусть функции c_i, $i \in \overline{1, 37}$, аналогичны рассмотренным в предыдущем примере, а функция $\mathbf{c}$ — суть евклидово расстояние между соответствующими плоскими векторами. В итоге получим величину совокупных затрат (глобальный экстремум) $V = 3237{,}03$. В случае применения «жадного» алгоритма затраты на перемещения по мегаполисам равны $4053{,}34$. Графическое изображение траектории движения, полученной в результате работы «жадного» алгоритма, приведено на рис. 3.

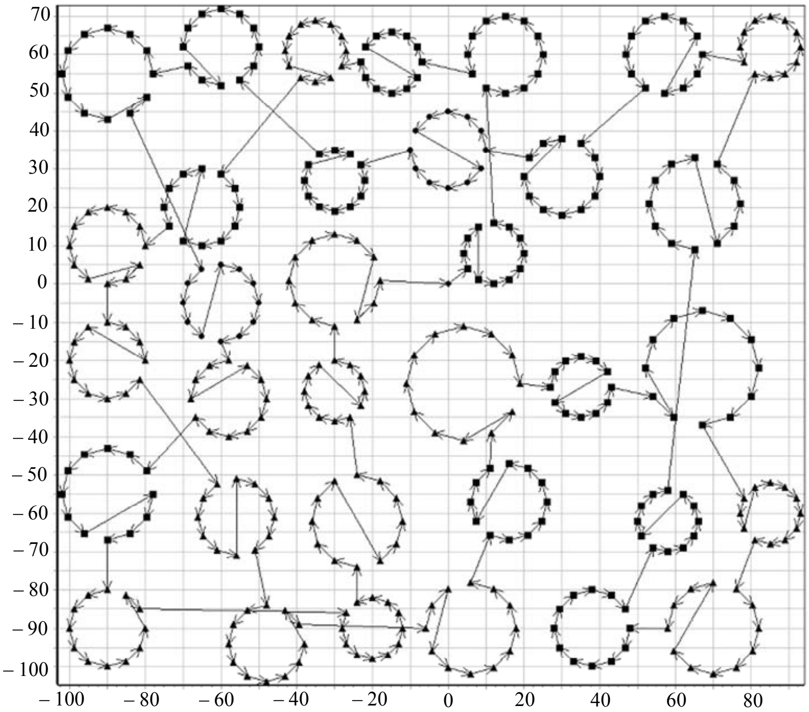

Рис. 1

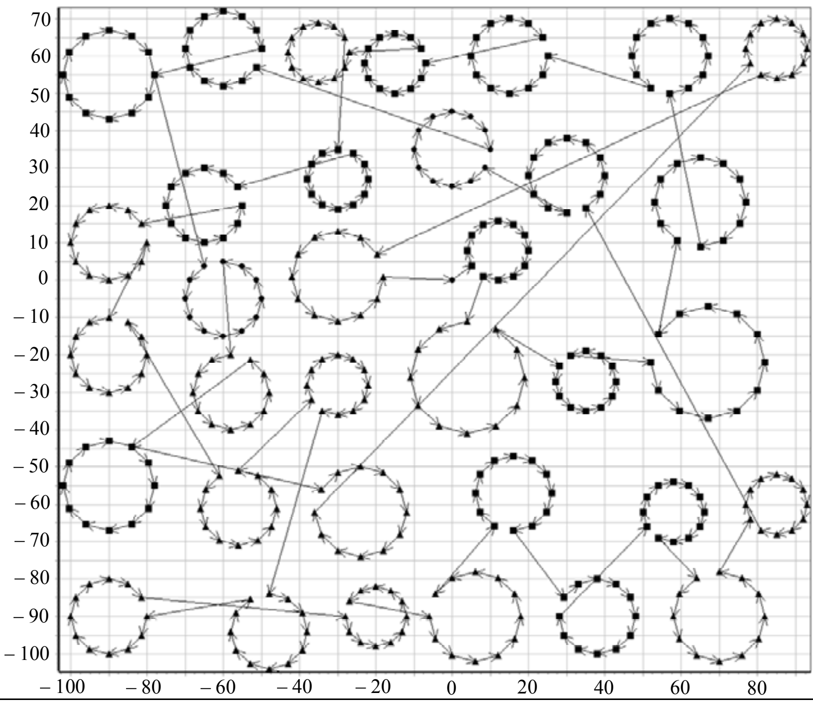

Рис. 2

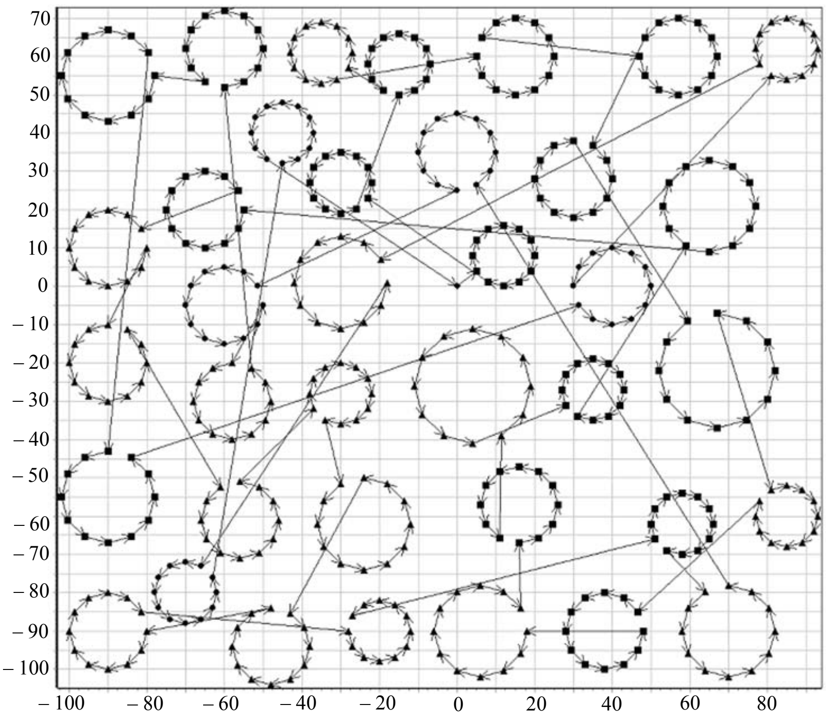

Рис. 3

В данном примере проигрыш «жадного» алгоритма по отношению к глобальному экстремуму составил примерно 20 %.

Итак, применение усеченной (без построения маршрута и трассы) версии точного метода на основе ДП позволяет экономить память и существенно сократить время счета (по сравнению с полной версией); полученный в итоге глобальный экстремум весьма полезен в качестве оценки результатов работы эвристических алгоритмов, которые, несмотря на проигрыш по результату, имеют несравненные преимущества в виде очень малого времени счета и низкие требования к вычислительным ресурсам компьютера (производительность процессора и объем оперативной памяти); в ряде прикладных задач отступление от оптимальности вполне допустимо в угоду скорости счета.

О.Г. Ченцов, О.О. Ченцов

ДО ПИТАННЯ ПРО ЗНАХОЖДЕННЯ ЗНАЧЕННЯ МАРШРУТНОЇ ЗАДАЧІ З ОБМЕЖЕННЯМИ

Розглянуто задачу послідовного обходу мегаполісів з обмеженнями різних типів. Вважається, що функції вартості, а також «поточні» обмеження можуть залежати від списку завдань (можлива залежність від списку виконаних або, навпаки, ще не виконаних завдань). Запропоновано підхід до визначення глобального екстремуму (значення задачі) на основі динамічного програмування в широкому сенсі. Завдяки такому підходу досягається економія пам'яті комп'ютера, що дозволяє визначати экстремум в задачі більшої розмірності та використовувати його для тестування евристичних алгоритмів. Для побудови шарів функції Беллмана використовується скорочена процедура, що дозволяє зменшити складність обчислювань (за умов передування не передбачається побудова всього масиву значень функції Беллмана).

A.G. Chentsov, A.A. Chentsov

ON THE PROBLEM OF OBTAINING THE VALUE OF ROUTING PROBLEM WITH CONSTRAINTS

The problem of sequential travelling of megapolises with constraints of different types is considered. It is supposed that cost functions and «current» constraints can be dependent on the tasks list (it is possible that dependence on the fulfilled or nonfulfilled tasks arises). Approach to determination of global extremum (the problem value) on the basis of widely interpreted dynamic programming is proposed. Under given approach, economy of computer memory is reached; this permits to determine extremum in problem with larger dimensionality and use it for testing of heuristic algorithms. Under construction of layers of Bellman function, truncated procedure which enables one to decrease computing complexity is used (under preceding conditions, construction of all array of the Bellman function values is not provided).

1. *Ченцов А.Г.* Экстремальные задачи маршрутизации и распределения заданий: вопросы теории. — Москва; Ижевск : РХД, 2008. — 238 с.
2. *Петунин А.А.* О некоторых стратегиях формирования маршрута инструмента при разработке управляющих программ для машин термической резки материала // Вестник УГАТУ. Серия: Управление, вычислительная техника и информатика. — 2009. — **13**, № 2(35). — С. 280–286.
3. *Фроловский В.Д.* Автоматизация проектирования управляющих программ тепловой резки металла на оборудовании с ЧПУ // Информационные технологии в проектировании и производстве. — 2005. — № 4. — С. 63–66.
4. *Gutin G., Punnen A.P.* The Traveling salesman problem and its variations — Boston; London; Dordrecht : Kluwer Academic Publishers Group, 2002. — 830 p.
5. *Беллман Р.* Применение динамического программирования к задаче о коммивояжере // Кибернетический сборник. — М. : Мир, 1964. — **9**. — С. 219–228.
6. *Хелд М., Карп Р.М.* Применение динамического программирования к задачам упорядочения // Там же. — М. : Мир, 1964. — **9**. — С. 202–218.
7. *Литл Дж., Мурти К., Суини Д., Кэрел К.* Алгоритм для решения задачи о коммивояжере // Экономика и математические методы. — 1965. — **1** (вып. 1). — С. 94–107.
8. *Меламед И.И., Сергеев С.И., Сигал И.Х.* Задача коммивояжера. Вопросы теории // Автоматика и телемеханика. — 1989. — № 9. — С. 3–34.
9. *Меламед И.И., Сергеев С.И., Сигал И.Х.* Задача коммивояжера. Точные алгоритмы // Там же. — 1989. — № 10. — С. 3–29.
10. *Меламед И.И., Сергеев С.И., Сигал И.Х.* Задача коммивояжера. Приближенные алгоритмы // Там же. — 1989. — № 11. — С. 3–26.
11. *Куратовский К., Мостовский А.* Теория множеств. — М. : Мир, 1970. — 416 с.
12. *Дьедонне Ж.* Основы современного анализа. — М. : Мир, 1964. — 430 с.
13. *Ченцов А.Г.* К вопросу о маршрутизации комплексов работ // Вестник Удмуртск. ун-та. Математика. Механика. Комп. науки. — 2013. — № 1. — С. 59–82.
14. *Ченцов А.А., Ченцов А.Г., Ченцов П.А.* Элементы динамического программирования в экстремальных задачах маршрутизации // Проблемы управления. — 2013. — № 5. — С.12–21.
15. *Ченцов А.Г.* Задача последовательного обхода мегаполисов с условиями предшествования // Автоматика и телемеханика. — 2014. — № 4. — С. 170–190.
16. *Ченцов А.А., Ченцов А.Г., Ченцов П.А.* Метод итераций в задаче маршрутизации с внутренними потерями // Труды Института математики и механики УрО РАН. — 2009. — **15**, № 4. — С. 268–287.
17. *Ченцов А.Г., Ченцов А.А.* Задача маршрутизации с ограничениями, зависящими от списка заданий // Доклады Академии наук. — 2015. — **465**, № 2. — С. 154–158.

Получено 02.12.2015