УДК 519.6 DOI:10.17587/mau.17.834-846

А. Г. Ченцов[1, 2], чл.-корр. РАН, гл. науч. сотр., проф., chentsov@imm.uran.ru,
А. М. Григорьев[1], зав. отделом, ag@uran.ru,
[1] Институт математики и механики имени Н. Н. Красовского УрО РАН, г. Екатеринбург,
[2] Уральский Федеральный Университет, г. Екатеринбург

Динамическое программирование в задаче маршрутизации: схема независимых вычислений

Рассматривается реализация схемы независимых вычислений для решения маршрутной задачи с условиями предшествования и (в теоретической части) с функциями стоимости, зависящими от списка заданий. Используется метод динамического программирования. Отдельно рассматривается параллельный алгоритм определения значения задачи (глобальный экстремум) и алгоритм "полного" решения, включающего построение оптимального маршрута. Последний алгоритм реализован на супервычислителе "Уран" при использовании (конечной) системы узлов, каждый из которых является совокупностью нескольких процессоров. В свою очередь, вся совокупность узлов образует вычислительный кластер. Допускается возможность вычисления некоторых значений функции Беллмана разными процессорами.

Ключевые слова: *динамическое программирование, маршрут, условия предшествования, существенные списки (заданий), слои пространства позиций, независимые вычисления, функция Беллмана*

Введение

Настоящая работа является продолжением работы [1]. В ней исследуются вопросы построения параллельных алгоритмов на основе схемы независимых вычислений [2, 3]. Рассматривается задача курьера, т.е. задача коммивояжера (ЗК), осложненная условиями предшествования; в теоретической части допускается также зависимость матрицы стоимостей от списка заданий. Подобная зависимость может возникать при рассмотрении некоторых задач, возникающих в атомной энергетике в связи с соображениями минимизации дозовой нагрузки работников при выполнении ими того или иного комплекса работ; см. в связи с этим работы [4—6]. Отметим, что зависимость стоимостей перемещений от списка заданий может возникать также в задачах, связанных с машиностроением; имеются в виду вопросы управления инструментом при листовой резке деталей на машинах с ЧПУ [7—9]. Введение в этих задачах упомянутой зависимости может быть связано с учетом технологических ограничений (жесткости листа, тепловых допусков) посредством соответствующих штрафов.

Как уже отмечалось, рассматриваемая задача имеет своим прототипом известную NP-полную ЗК (в англоязычной литературе — TSP). В связи с исследованием ЗК отметим работы [10—15]; в частности, в работах [14, 15] исследовалось применение для решения ЗК метода динамического программирования (ДП).

В настоящей работе используется вариант ДП, восходящий к работе [16] и не предусматривающий построение полного массива значений функции Беллмана. Используется схема независимых вычислений слоев данной функции, предложенная в работах [2, 3], а также вариант данной схемы, предложенный в работе [1]. В настоящей статье построен параллельный алгоритм вычисления значений функции Беллмана с применением суперкомпьютера. Данный алгоритм реализован на вычислительном кластере супервычислителя "Уран", проведен вычислительный эксперимент, показывающий работоспособность предлагаемого алгоритма.

Обозначения общего характера

Ниже используется стандартная теоретико-множественная символика (кванторы, связки; $\varnothing$ — пустое множество). Семейством называем множество, все элементы которого — множества (семейство — множество, "составленное" из множеств). Если x и y — объекты, то $\{x; y\}$ обозначаем (единственное) множество, содержащее в качестве своих элементов x, y и не содержащее никаких других элементов ($\{x; y\}$ — неупорядоченная пара объектов x, y). Тогда, в частности, для произвольного объекта z в виде $\{z\} \triangleq \{z; z\}$ имеем одноэлементное множество (синглетон), содержащее z. Каждое множество является объектом. С учетом этого, следуя работе [17, с. 67], для произвольных объектов a и b обозначаем (a, b) упорядоченную пару (УП) упомянутых объектов: $(a, b) \triangleq \{\{a\}; \{a, b\}\}$. Если же z — какая-либо УП, то $pr_1(z)$ и $pr_2(z)$ обозначаем соответственно первый и второй элементы УП z, однозначно определяемые условием $z = (pr_1(z), pr_2(z))$.

Если H — множество, то $\mathcal{P}(H)$ и $\mathcal{P}'(H)$ обозначаем соответственно семейства всех и всех непустых подмножеств (п/м) H; $Fin(H)$ обозначаем семейство всех конечных множеств из $\mathcal{P}'(H)$ ($Fin(H)$ есть, следовательно, семейство всех непустых конечных п/м H).

Как обычно [18, с. 17] определяем триплеты, "составленные" из произвольных объектов: если a, b и c — объекты, то $(a, b, c) \triangleq ((a, b), c)$. В связи с этим следуем также известному [18, с. 17] правилу: $A \times B \times C \triangleq (A \times B) \times C$ для произвольных непустых множеств A, B и C. Если, кроме того, D есть непустое множество, φ — отображение, действующие из $A \times B \times C$ в D (т.е. φ: $A \times B \times C \to D$), $x \in A \times B$ и $y \in C$, то определено значение $\varphi(x, y) \in D$ отображения φ в точке $(x, y) \in A \times B \times C$, для которого также используем обозначение $\varphi(pr_1(x), pr_2(x), y)$.

Для произвольных непустых множеств $\mathbb{U}$ и $\mathbb{V}$ обозначаем (Bi) $[\mathbb{U}; \mathbb{V}]$ множество всех биекций [19, с. 87] множества $\mathbb{U}$ на множество $\mathbb{V}$. Если Y и Z — непустые множества, а $\psi \in (Bi)[Y; Z]$, то обозначаем ψ^{-1} биекцию Z на Y, т.е. $\psi^{-1} \in (Bi)[Z; Y]$, обратную по отношению к ψ [19, с. 87]. Для каждого непустого множества L в виде $(Bi)[L; L]$ имеем множество всех перестановок L [19, с. 87].

Обозначаем $\mathbb{R}$ вещественную прямую с положительной полуосью

$$[0, \infty[\triangleq \{\xi \in \mathbb{R} \mid 0 \leqslant \xi\}, \mathbb{N} \triangleq \{1; 2; ...\} \in \mathcal{P}'(\mathbb{R}), \mathbb{N}_0 \triangleq \{0\} \cup \mathbb{N} = \{0; 1; 2; ...\} \text{ и}$$

$$\overline{p, q} \triangleq \{i \in \mathbb{N}_0 \mid (p \leqslant i) \ \& \ (i \leqslant q)\} \ \forall p \in \mathbb{N}_0 \ \forall q \in \mathbb{N}_0$$

(заметим, что $\overline{1, 0} = \varnothing$ и $\overline{1, m} = \{i \in \mathbb{N} \mid i \leqslant m\} \neq \varnothing$ при $m \in \mathbb{N}$). Непустому конечному множеству K сопоставляем его мощность $|K| \in \mathbb{N}$; тогда определено множество $\overline{1, |K|} = \{i \in \mathbb{N} \mid i \leqslant |K|\}$ и непустое множество $(bi)|K| \triangleq (Bi)[\overline{1, |K|}; K]$ всех биекций $\overline{1, |K|}$ на K. Кроме того, полагаем $|\varnothing| \triangleq 0$. Всюду в дальнейшем для каждого непустого множества S обозначаем $\mathcal{R}_+[S]$ множество всех функций из S в $[0, \infty[$.

1. Некоторые специальные обозначения; постановка задачи

Фиксируем число $N \in \mathbb{N}$, $N \geqslant 2$. Нашей целью является исследование перемещений

$$0 \to \alpha(1) \to ... \to \alpha(N), \tag{1.1}$$

где α — перестановка индексов из $\overline{1, N}$. В дальнейшем упомянутые перестановки именуем маршрутами. Полагаем, что выбор α может быть стеснен дополнительными ограничениями (условиями предшествования). В связи с этим фиксируем отношение

$$\mathbb{K} \in \mathcal{P}(\overline{1, N} \times \overline{1, N}), \tag{1.2}$$

определяющее условия предшествования. Итак, $\mathbb{K}$ есть п/м $\overline{1, N} \times \overline{1, N}$ (случай $\mathbb{K} = \varnothing$ из рассмотрения не исключается и соответствует отсутствию условий предшествования). Элементами $\mathbb{K}$ (1.2) являются УП; если $z \in \mathbb{K}$, то будем условно именовать $pr_1(z) \in \overline{1, N}$ отправителем, а $pr_2(z) \in \overline{1, N}$ — получателем z; сами же элементы $\mathbb{K}$ называем адресными парами. Всюду в дальнейшем полагаем, что

$$\forall \mathbb{K}_0 \in \mathcal{P}'(\mathbb{K}) \ \exists z_0 \in \mathbb{K}_0: pr_1(z_0) \neq pr_2(z) \ \forall z \in \mathbb{K}_0. \tag{1.3}$$

Тогда [16, часть 2] в виде

$$\mathbb{A} \triangleq \{\alpha \in \mathbb{P} | \forall z \in \mathbb{K} \forall t_1 \in \overline{1, N} \ \forall t_2 \in \overline{1, N} \ (z = (\alpha(t_1), \alpha(t_1))) \Rightarrow (t_1 < t_2)\} = \{\alpha \in \mathbb{P} \mid \alpha^{-1}(pr_1(z)) < \alpha^{-1}(pr_2(z)) \ \forall z \in \mathbb{K}\} \in \mathcal{P}'(\mathbb{P}), \tag{1.4}$$

где (здесь и ниже) $\mathbb{P} \triangleq (bi)[\overline{1, N}]$, имеем множество всех допустимых маршрутов (ДМ) в смысле условий предшествования, определяемых посредством $\mathbb{K}$ (1.2). Итак, $\mathbb{A}$, как и в работе [1], определяет множество допустимых решений формулируемой ниже задачи.

Пусть $\mathfrak{N} \triangleq \mathcal{P}'(\overline{1, N})$; элементы $\mathfrak{N}$ называем ниже списками (заданий). Полагаем заданными отображения

$$\mathbf{c} \in \mathcal{R}_+[\overline{0, N} \times \overline{1, N} \times \mathfrak{N}], f \in \mathcal{R}_+[\overline{1, N}], \tag{1.5}$$

участвующие в формировании аддитивного критерия. Следуя работе [1], введем аддитивный критерий качества маршрутов. Итак, если $\alpha \in \mathbb{P}$, то [1, (1)]

$$\mathfrak{C}_\alpha \triangleq \mathbf{c}(0, \alpha(1), \overline{1, N}) + \sum_{t=1}^{N-1} \mathbf{c}(\alpha(t), \alpha(t+1), \{\alpha(j): j \in \overline{t+1, N}\}) + f(\alpha(N)). \tag{1.6}$$

В качестве основной рассматриваем задачу

$$\mathfrak{C}_\alpha \to \min, \ \alpha \in \mathbb{A}. \tag{1.7}$$

Поскольку $\mathbb{A}$ — непустое конечное множество (см. (1.4)), то задаче (1.7) сопоставляется значение (экстремум)

$$V \triangleq \min_{\alpha \in \mathbb{A}} \mathfrak{C}_\alpha, \tag{1.8}$$

и непустое множество оптимальных ДМ:

$$\mathbb{A}_{\text{opt}} \triangleq \{\alpha_0 \in \mathbb{A} | \mathfrak{C}_{\alpha_0} = V\} \in \mathcal{P}'(\mathbb{A}). \tag{1.9}$$

Нашей целью является определение значения V (1.8) и какого-либо элемента множества (1.9). Для решения задачи (1.7) будем использовать аппарат ДП, действуя в духе работ [1, 16, 20, 21]. В следующем разделе конструкция решения на основе ДП излагается в краткой форме.

2. Динамическое программирование, 1

Для использования аппарата ДП в задаче с возможными ограничениями в виде условий предшествования, определяемых множеством $\mathbb{K}$, мы прежде всего осуществляем редукцию соответствующих ограничений [16, § 4.9]: допустимость по пред-

шествованию заменяется допустимостью по вычеркиванию. С этой целью введем в рассмотрение оператор вычеркивания $\mathbf{I}$, действующий в множестве (а, точнее, в семействе) $\mathfrak{N}$ по правилу

$$\mathbf{I}(K) \triangleq K\backslash\{pr_2(z)\colon z \in \Xi[K]\}, \qquad (2.1)$$

где $\Xi[K] \triangleq \{z \in \mathbb{K} \mid (pr_1(z) \in K)\ \&\ (pr_2(z) \in K)\}$ при $K \in \mathfrak{N}$. В терминах (2.1) вводятся частичные (вообще говоря) маршруты, допустимые по вычеркиванию: полагаем, что (см. [1, раздел 2], [16, § 4.9])

$$(\mathbf{I} - bi)[K] \triangleq \{\alpha \in (bi)[K] \mid \alpha(s) \in$$
$$\in \mathbf{I}(\{\alpha(t)\colon t \in \overline{s, |K|}\})\forall s \in \overline{1, |K|}\}\forall K \in \mathfrak{N}. \qquad (2.2)$$

В соотношении (2.2) имеем множество всех ДМ посещения "городов" с индексами из K. Согласно работе [16, предложения 2.2.2, 2.2.3] (при условии (1.3)) $(\mathbf{I} - bi)[K] \neq \varnothing\ \forall K \in \mathfrak{N}$. Стало быть, соотношение (2.2) определяет всякий раз непустое конечное множество, причем

$$\mathbb{A} = (\mathbf{I} - bi)[\overline{1, N}] = \{\alpha \in \mathbb{P} \mid (\alpha(1) \in \mathbf{I}(\overline{1, N}))\ \&$$
$$\&\ (\alpha(k) \in \mathbf{I}(\overline{1, N}\backslash\{\alpha(t)\colon t \in \overline{1, k-1}\})\ \forall k \in \overline{2, N}\}(2.3)$$

(имеется в виду случай $K = \overline{1, N}$). Посредством (2.2) вводим соответствующие частичные задачи: при $s \in \overline{0, N}$ и $K \in \mathfrak{N}$ со свойством $|K| \geqslant 2$ рассматриваем задачу

$$\mathbf{c}(s, \alpha(1), K) + \sum_{t=1}^{|K|-1} \mathbf{c}(\alpha(t), \alpha(t+1), \{\alpha(j)\colon j \in$$
$$\in \overline{t+1, |K|}\}) + f(\alpha(|K|)) \to \min,\ \alpha \in (\mathbf{I} - bi)[K]. (2.4)$$

Каждой задаче (2.4) сопоставляется значение $v(s, K)$ и непустое множество оптимальных ДМ. Итак, если $k \in \overline{0, N}$ и $K \in \mathfrak{N}$, $|K| \geqslant 2$, то

$$v(k, K) \triangleq \min_{\alpha \in (\mathbf{I} - bi)[K]} [\mathbf{c}(k, \alpha(1), K) + \sum_{t=1}^{|K|-1} \mathbf{c}(\alpha(t),$$
$$\alpha(t+1), \{\alpha(j)\colon j \in \overline{t+1, |K|}\}) + f(\alpha(|K|))] \in [0, \infty[. (2.5)$$

Если же $K \in \mathfrak{N}$ и $|K| = 1$, то $\exists! j \in \overline{1, N}\colon K = \{j\}$. Поэтому при $K \in \mathfrak{N}$ со свойством $|K| = 1$ мы для каждого $k \in \overline{0, N}$ полагаем, что

$$v(k, K) \triangleq \mathbf{c}(k, r, \{r\}) + f(r), \qquad (2.6)$$

где $r \in \overline{1, N}$ таково, что $K \in \{r\}$ (индекс r определяется по K единственным образом). Таким образом, посредством соотношения (2.6) определены все значения $v(s, \{t\})$, $t \in \overline{1, N}$. Отметим, однако, что в задаче, осложненной условиями предшествования, в последнем случае можно будет исключать из рассмотрения ситуации, когда $t = pr_1(z)$ при некотором выборе УП $z \in \mathbb{K}$.

Как обычно [1, 16, 20, 21], полагаем также, что

$$v(s, \varnothing) \triangleq f(s)\ \forall s \in \overline{0, N}. \qquad (2.7)$$

При $s \in \overline{0, N}$ введем в рассмотрение семейство $\mathfrak{N}_s \triangleq \{K \in \mathfrak{N} | s = |K|\}$. Тогда есть объединение всех множеств $\mathfrak{N}_s$, $s \in \overline{1, N}$.

При $k \in \overline{0, N}$, $l \in \overline{2, N}$ и $K \in \mathfrak{N}_l$ значение $v(k, K)$ определяется посредством (2.5). Если же $l = 1$ и $K \in \mathfrak{N}_l$, то при $k \in \overline{0, N}$ значение $v(k, K)$ определяется соотношением (2.6), где r соответствует K в смысле равенства $K = \{r\}$.

Полагаем здесь и ниже, что $\boldsymbol{N} \triangleq \mathfrak{N} \cup \{\varnothing\}$, получая, конечно, равенство

$$\boldsymbol{N} = \left(\bigcup_{s=1}^{N} \mathfrak{N}_s\right) \cup \{\varnothing\}. \qquad (2.8)$$

Теперь значения $v(s, K)$ определены (см. (2.7), (2.8)) при $s \in \overline{0, N}$ и $K \in \boldsymbol{N}$.

Иными словами, определена функция

$$v \in \mathcal{R}_+[\overline{0, N} \times \boldsymbol{N}], \qquad (2.9)$$

именуемая ниже функцией Беллмана.

Из общих результатов [20, 21] вытекает [1, с. 234] уравнение Беллмана

$$v(s, K) = \min_{j \in \mathbf{I}(K)} [\mathbf{c}(s, j, K) + v(j, K\backslash\{j\})]$$
$$\forall s \in \overline{0, N},\ \forall K \in \mathfrak{N}. \qquad (2.10)$$

Из соотношений (1.6), (1.8), (2.3) и (2.5) вытекает важное равенство

$$V = v(0, \overline{1, N}) =$$
$$= \min_{j \in \mathbf{I}(\overline{1, N})} [\mathbf{c}(0, j, \overline{1, N}) + v(j, \overline{1, N}\backslash\{j\})]. \qquad (2.11)$$

Поэтому построенную систему частичных задач можно рассматривать как расширение основной задачи (1.7).

3. Динамическое программирование, 2

В настоящем разделе излагается модификация схемы, предложенной в работе [16, § 4.9] (см. также более поздние варианты в работах [1, 20, 21]). Речь идет о том, чтобы (при условиях предшествования) использовать только "часть" массива значений функции Беллмана, а потому при реализации ДП оказывается возможным ограничиться этой "частью". Сначала введем в рассмотрение так называемые существенные списки заданий в виде элементов семейства

$$\mathbb{G} \triangleq \{K \in \mathfrak{N} | \forall z \in \mathbb{K}(pr_1(z) \in K) \Rightarrow (pr_2(z) \in K)\}. (3.1)$$

Кроме того, полагаем, что $\mathbb{G}_s \triangleq \{K \in \mathbb{G} \mid s = |K|\}$ $\forall\ s \in \overline{1, N}$. Тогда в виде $\{\mathbb{G}_1; \ldots; \mathbb{G}_N\}$ имеем разбиение $\mathbb{G}$ (3.1) в конечную сумму семейств. Легко

видеть, что $\mathbb{G}_N = \{\overline{1, N}\}$ (синглетон, содержащий множество $\overline{1, N}$).

Кроме того,

$$\mathbb{G}_1 = \{\{t\}: t \in \overline{1, N}\backslash\mathbb{K}_1\}, \quad (3.2)$$

где $\mathbb{K}_1 \triangleq \{pr_1(z): z \in \mathbb{K}\}$. Наконец [20, 21],

$$\mathbb{G}_{s-1} = \{K\backslash\{t\}: K \in \mathbb{G}_s, t \in \mathbf{I}(K)\} \ \forall s \in \overline{2, N}. \quad (3.3)$$

Свойства (3.2), (3.3) естественно связать с рекуррентной процедурой

$$\mathbb{G}_N \to \mathbb{G}_{N-1} \to \ldots \to \mathbb{G}_1, \quad (3.4)$$

у которой семейства $\mathbb{G}_1$ и $\mathbb{G}_N$ были указаны выше. В терминах семейств $\mathbb{G}_1, \ldots, \mathbb{G}_N$ определяются слои пространства позиций, обозначенные далее $D_0, D_1, \ldots, D_N$. При этом

$$D_0 \triangleq \{(s, \varnothing): s \in \overline{1, N}\backslash\mathbb{K}_1\}, \ D_N \triangleq \{(0, \overline{1, N})\}; \quad (3.5)$$

D_N есть синглетон, содержащий УП $(0, \overline{1, N})$.

Если же $s \in \overline{1, N-1}$, то для построения D_s нам потребуются некоторые вспомогательные понятия, а именно, при $K \in \mathbb{G}_s$ последовательно определяем множества

$$J_s(K) \triangleq \{j \in \overline{1, N}\backslash K \mid K \cup \{j\} \in \mathbb{G}_{s+1}\},$$
$$\mathbb{D}_s[K] \triangleq \{(j, K): j \in J_s(K)\}; \quad (3.6)$$

в терминах множеств (3.6) слой D_s определяется следующим образом:

$$D_s \triangleq \bigcup_{K \in \mathbb{G}_s} \mathbb{D}_s[K] \in \mathcal{P}'(\overline{1, N} \times \mathbb{G}_s). \quad (3.7)$$

Каждое множество (3.7) не пусто [16, § 4.9]. Итак, определены непустые множества $D_0, D_1, \ldots, D_N$, для которых $D_s \subset \overline{0, N} \times \boldsymbol{N} \ \forall s \in \overline{0, N}$.

С учетом этого (см. также (2.9)) для каждого из слоев определено сужение функции Беллмана на данный слой. Итак, полагаем при $s \in \overline{0, N}$, что

$$v_s \in \mathcal{R}_+[D_s] \quad (3.8)$$

есть такая функция, что

$$v_s(x, K) = v(x, K) \ \forall \ (x, K) \in D_s. \quad (3.9)$$

Функции (3.8), (3.9) называем слоями функции Беллмана, т.е. слоями функции (2.9). Для построения рекуррентной процедуры, доставляющей кортеж $(v_0, v_1, \ldots, v_N)$, напомним известное [1, (6); 16, предложение 4.9.4] свойство:

$$(t, K\backslash\{t\}) \in D_{s-1}$$
$$\forall s \in \overline{1, N} \ \forall (l, K) \in D_s \ \forall t \in \mathbf{I}(K). \quad (3.10)$$

С учетом (3.8), (3.10) получаем теперь, что при $s \in \overline{1, N}$, $(l, K) \in D_s$, $t \in \mathbf{I}(K)$ определено значение $v_{s-1}(t, K\backslash\{t\}) \in [0, \infty[$ функции v_{s-1} в "точке" $(t, K\backslash\{t\})$. Более того (см. [1, предложение 2]) при $s \in \overline{1, N}$ преобразование функции v_{s-1} в v_s определяется соотношениями

$$v_s(l, K) = \min_{t \in \mathbf{I}(K)} [\mathbf{c}(l, t, K) +$$
$$+ v_{s-1}(t, K\backslash\{t\})] \ \forall (l, K) \in D_s. \quad (3.11)$$

Отметим конкретизацию (3.11) для случая $s = 1$. Иными словами, рассмотрим преобразование $v_0 \in \mathcal{R}_+[D_0]$ в $v_1 \in \mathcal{R}_+[D_1]$. Для этого прежде всего отметим, что согласно (3.5) и (3.8) v_0 полностью определяется значениями $f(s) = v_0(s, \varnothing)$, где $s \in \overline{1, N}\backslash\mathbb{K}_1$. Иными словами, v_0 есть функция

$$(s, \varnothing) \to f(s): D_0 \to [0, \infty[. \quad (3.12)$$

Далее, согласно (3.6) мы при $K \in \mathbb{G}_1$ имеем равенство

$$J_1(K) = \{j \in \overline{1, N}\backslash K \mid K \cup \{j\} \in \mathbb{G}_2\}. \quad (3.13)$$

Из соотношений (3.2), (3.13) следует теперь, что при $t \in \overline{1, N}\backslash\mathbb{K}_1$ определено множество $J_1(\{t\})$, причем

$$J_1(\{t\}) = \{j \in \overline{1, N}\backslash\{t\} \mid \{t; j\} \in \mathbb{G}_2\}. \quad (3.14)$$

С учетом (3.2) и (3.6) имеем также, что при $t \in \overline{1, N}\backslash\mathbb{K}_1$ определено множество $\mathbb{D}_1[\{t\}]$, причем

$$\mathbb{D}_1[\{t\}] = \{(j, \{t\}): j \in J_1(\{t\})\}. \quad (3.15)$$

Из соотношений (3.2), (3.7), (3.14) и (3.15) вытекает, что

$$D_1 = \bigcup_{K \in \mathbb{G}_1} \mathbb{D}_1[K] = \bigcup_{t \in \overline{1, N}\backslash\mathbb{K}_1} \mathbb{D}_1[\{t\}] =$$
$$= \bigcup_{t \in \overline{1, N}\backslash\mathbb{K}_1} \{(j, \{t\}): j \in J_1(\{t\})\} =$$
$$= \{(j, \{t\}): t \in \overline{1, N}\backslash\mathbb{K}_1, j \in J_1(\{t\})\}. \quad (3.16)$$

Вместе с тем, согласно (3.10) $(j, K\backslash\{j\}) \in D_0 \ \forall (l, K) \in D_1 \ \forall j \in \mathbf{I}(K)$. При этом, однако, согласно (3.7) $D_1 \subset \overline{1, N} \times \mathbb{G}_1$, а потому при $(l, K) \in D_1$ непременно $K \in \mathbb{G}_1$, т.е. (см. (3.2)) $K = \{t\}$ при $t \in \overline{1, N}\backslash\mathbb{K}_1$, а тогда $\mathbf{I}(K) = \mathbf{I}(\{t\}) = \{t\}$ (см. [22, замечание 3.2]). Возвращаясь к (3.11), заметим, что

$$v_1(l, K) = \min_{t \in \mathbf{I}(K)} [\mathbf{c}(l, t, K) +$$
$$+ v_0(t, K\backslash\{t\})] \ \forall (l, K) \in D_1. \quad (3.17)$$

С учетом (3.16) функция v_1 полностью определяется значениями

$$v_1(j, \{s\}), \ s \in \overline{1, N}\backslash\mathbb{K}_1, j \in J_1(\{s\}).$$

Как следствие из соотношения (3.17) имеем при $s \in \overline{1, N} \backslash \mathbb{K}_1$ и $j \in J_1(\{s\})$, что

$$v_1(j, \{s\}) = \min_{t \in \mathbf{I}(\{s\})} [\mathbf{c}(j, t, \{s\}) + v_0(t, \{s\} \backslash \{t\})] =$$
$$= \min_{t \in \{s\}} [\mathbf{c}(j, t, \{s\}) + v_0(t, \{s\} \backslash \{t\})] =$$
$$= \mathbf{c}(j, s, \{s\}) + v_0(s, \varnothing) = \mathbf{c}(j, s, \{s\}) + f(s). \quad (3.18)$$

В силу (3.16) имеем, что посредством (3.18) полностью определена функция v_1. Итак, у нас полностью и достаточно просто определены функции v_0 и v_1. Дальнейшее построение, т.е. определение функций $v_2, \ldots, v_N$, осуществляется по правилу (3.11). Заметим, что согласно (2.11), (3.5), (3.9)

$$V = v_N(0, \overline{1, N}), \quad (3.19)$$

где v_N определяется на финальном этапе рекуррентной процедуры

$$v_0 \to v_1 \to \ldots \to v_N,$$

в основе которой находится преобразование (3.11). После этого, следуя работе [1, раздел 4], мы конструируем оптимальный маршрут. В целях полноты изложения приведем данную процедуру построения оптимального маршрута.

Сначала заметим, что в силу (3.11) и (3.19)

$$V = \min_{t \in \mathbf{I}(\overline{1, N})} [\mathbf{c}(0, t, \overline{1, N}) + v_{N-1}(t, \overline{1, N} \backslash \{t\})] \quad (3.20)$$

(мы используем (3.11) при $s = N$, $l = 0$, $K = \overline{1, N}$). С учетом (3.20) выбираем $\mathbb{J}_1 \in \mathbf{I}(\overline{1, N})$ так, что при этом

$$V = \mathbf{c}(0, \mathbb{J}_1, \overline{1, N}) + v_{N-1}(\mathbb{J}_1, \overline{1, N} \backslash \{\mathbb{J}_1\}). \quad (3.21)$$

Отметим при этом, что согласно (3.10)

$$(\mathbb{J}_1, \overline{1, N} \backslash \{\mathbb{J}_1\}) \in D_{N-1}, \quad (3.22)$$

а потому $v_{N-1}(\mathbb{J}_1, \overline{1, N} \backslash \{\mathbb{J}_1\})$ определено корректно. Согласно (3.10) имеем

$$(t, (\overline{1, N} \backslash \{\mathbb{J}_1\}) \backslash \{t\}) \in D_{N-2} \quad (3.23)$$

при $t \in \mathbf{I}(\overline{1, N} \backslash \{\mathbb{J}_1\})$, а потому для данного случая определено

$$v_{N-2}(t, \overline{1, N} \backslash \{\mathbb{J}_1; t\}) \in [0, \infty[.$$

С учетом (3.11) и (3.22) имеем теперь, что

$$v_{N-1}(\mathbb{J}_1, \overline{1, N} \backslash \{\mathbb{J}_1\}) =$$
$$= \min_{t \in \mathbf{I}(\overline{1, N} \backslash \{\mathbb{J}_1\})} [\mathbf{c}(\mathbb{J}_1, t, \overline{1, N} \backslash \{\mathbb{J}_1\}) +$$
$$+ v_{N-2}(t, \overline{1, N} \backslash \{\mathbb{J}_1, t\})]. \quad (3.24)$$

Используя соотношение (3.24), выберем $\mathbb{J}_2 \in \mathbf{I}(\overline{1, N} \backslash \{\mathbb{J}_1\})$ так, что

$$v_{N-1}(\mathbb{J}_1, \overline{1, N} \backslash \{\mathbb{J}_1\}) =$$
$$= \mathbf{c}(\mathbb{J}_1, \mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1\}) + v_{N-2}(\mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1, \mathbb{J}_2\}). \quad (3.25)$$

Вместе с тем, из (3.23) следует, что

$$(\mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1, \mathbb{J}_2\}) \in D_{N-2}, \quad (3.26)$$

и тогда согласно (3.9) определено значение $v_{N-2}(\mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1; \mathbb{J}_2\}) \in [0, \infty[$. По выбору $\mathbb{J}_1$ и $\mathbb{J}_2$ имеем, что $\mathbb{J}_1 \in \overline{1, N}$, $\mathbb{J}_2 \in \overline{1, N}$, $\mathbb{J}_1 \neq \mathbb{J}_2$.

Теперь подставим выражение (3.25) в соотношение (3.21).

Тогда

$$V = \mathbf{c}(0, \mathbb{J}_1, \overline{1, N}) + \mathbf{c}(\mathbb{J}_1, \mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1\}) +$$
$$+ v_{N-2}(\mathbb{J}_2, \overline{1, N} \backslash \{\mathbb{J}_1, \mathbb{J}_2\}). \quad (3.27)$$

Замечание 3.1. Пусть $N = 2$. Тогда из (3.27) вытекает, что

$$V = \mathbf{c}(0, \mathbb{J}_1, \overline{1, 2}) +$$
$$+ \mathbf{c}(\mathbb{J}_1, \mathbb{J}_2, \overline{1, 2} \backslash \{\mathbb{J}_1\}) + v_0(\mathbb{J}_2, \varnothing). \quad (3.28)$$

Согласно (3.26) $(\mathbb{J}_2, \varnothing) \in D_0$, а поэтому в силу (3.5) $\mathbb{J}_2 \in \overline{1, 2} \backslash \mathbb{K}_1$.

Тогда согласно (3.12) $v_0(\mathbb{J}_2, \varnothing) = f(\mathbb{J}_2)$, и в силу (3.28) имеем равенство

$$V = \mathbf{c}(0, \mathbb{J}_1, \overline{1, 2}) + \mathbf{c}(\mathbb{J}_1, \mathbb{J}_2, \{\mathbb{J}_2\}) + f(\mathbb{J}_2). \quad (3.29)$$

Отметим, что в нашем случае ($N = 2$)

$$J \triangleq (\mathbb{J}_t)_{t \in \overline{1, 2}} \in \mathbb{P},$$

а поэтому согласно (1.6) и (3.29)

$$\mathfrak{C}_J = \mathbf{c}(0, \mathbb{J}_1, \overline{1, 2}) + \mathbf{c}(\mathbb{J}_1, \mathbb{J}_2, \{\mathbb{J}_2\}) + f(\mathbb{J}_2) = V. \quad (3.30)$$

Согласно (2.3) имеем в нашем случае, что

$$\mathbb{A} = \{\alpha \in \mathbb{P} \mid (\alpha(1) \in \mathbf{I}(\overline{1, 2})) \,\&$$
$$\& \, (\alpha(2) \in \mathbf{I}(\overline{1, 2} \backslash \{\alpha(1)\})\}. \quad (3.31)$$

Заметим, что по выбору $\mathbb{J}_1$ (в нашем случае $N = 2$) $\mathbb{J}_1 \in \mathbf{I}(\overline{1, 2})$.

Вместе с тем, по выбору $\mathbb{J}_2$, $\mathbb{J}_2 \in \mathbf{I}(\overline{1, 2} \backslash \{\mathbb{J}_1\})$.

Поскольку $J \in \mathbb{P}$, $J(1) = \mathbb{J}_1$ и $J(2) = \mathbb{J}_2$, имеем из (3.31), что

$$J \in \mathbb{A}. \quad (3.32)$$

Тогда из соотношений (1.9), (3.30) и (3.32) получаем, что $J \in \mathbb{A}_{opt}$, ч.т.д. ■

Если $N > 2$, то процедуру последовательного решения локальных задач, подобных (3.21) и (3.25), следует продолжать вплоть до исчерпывания всего списка. В результате будет построен ДМ $\eta = (\mathbb{J}_t)_{t \in \overline{1, N}} \in \mathbb{A}$ со свойством $\mathfrak{C}_\eta = V$ (см. (1.8)), что означает оптимальность маршрута η: $\eta \in \mathbb{A}_{opt}$.

4. Элементы параллельной структуры

В настоящем разделе излагается теоретическая конструкция, восходящая к [2—3, 22] (см. также более частный вариант в работе [1]). В связи с этим напомним, что (см. (3.3))

$$\mathbb{G}_{N-1} = \{\overline{1, N}\backslash\{t\}: t \in \mathbf{I}(\overline{1, N})\}. \quad (4.1)$$

С учетом свойства $\mathbb{A} \neq \emptyset$ и [16, предложение 4.9.2] получаем, что $\mathbb{G}_{N-1} \neq \emptyset$; элементами (4.1) являются множества $K \in \mathbb{G}$ такие, что $|K| = N - 1$. Заметим, что в (3.20) используются значения v_{N-1} в позициях из D_{N-1}, у которых вторым элементом является всякий раз множество из семейства (4.1). С этой точки зрения наиболее важным представляется построение функции

$$v_{N-1}: D_{N-1} \to [0, \infty[. \quad (4.2)$$

(см. (3.8)). При этом, согласно (3.6) для $K \in \mathbb{G}_{N-1}$

$$J_{N-1}(K) = \{j \in \overline{1, N}\backslash K \mid K \cup \{j\} \in \mathbb{G}_N\} =$$
$$= \{j \in \overline{1, N}\backslash K \mid K \cup \{j\} = \overline{1, N}\} = \overline{1, N}\backslash K; \quad (4.3)$$

$$\mathbb{D}_{N-1}[K] =$$
$$= \{(j, K): j \in J_{N-1}(K)\} \in \mathcal{P}'(\overline{1, N} \times \mathbb{G}_{N-1}). \quad (4.4)$$

Наконец, из (3.7) вытекает, что

$$D_{N-1} = \bigcup_{K \in \mathbb{G}_{N-1}} \mathbb{D}_{N-1}[K]. \quad (4.5)$$

Учтем (4.1). Тогда из (4.3) и (4.4) получаем при $t \in \mathbf{I}(\overline{1, N})$, что $\overline{1, N}\backslash\{t\} \in \mathbb{G}_{N-1}$, и определены множества

$$J_{N-1}(\overline{1, N}\backslash\{t\}) = \overline{1, N}\backslash(\overline{1, N}\backslash\{t\}) = \{t\},$$
$$\mathbb{D}_{N-1}[\overline{1, N}\backslash\{t\}] =$$
$$= \{(j, \overline{1, N}\backslash\{t\}): j \in J_{N-1}(\overline{1, N}\backslash\{t\})\} =$$
$$= \{(t, \overline{1, N}\backslash\{t\})\}. \quad (4.6)$$

Наконец, из (4.1) и (4.5) получаем, что

$$D_{N-1} = \bigcup_{t \in \mathbf{I}(\overline{1, N})} \mathbb{D}_{N-1}[\overline{1, N}\backslash\{t\}] =$$
$$= \bigcup_{t \in \mathbf{I}(\overline{1, N})} \{(t, \overline{1, N}\backslash\{t\})\} =$$
$$= \{(t, \overline{1, N}\backslash\{t\}): t \in \mathbf{I}(\overline{1, N})\} \quad (4.7)$$

(см. [1, (17)]). Из (3.20) и (4.7) следует, что глобальный экстремум полностью определяется значениями $v_{N-1}(t, \overline{1, N}\backslash\{t\})$, $t \in \mathbf{I}(\overline{1, N})$, в совокупности реализующими функцию v_{N-1}. Выражение (3.20) будем рассматривать в качестве оконечного для схемы независимых вычислений, излагаемой ниже.

Вспомогательные области достижимости. Всюду в дальнейшем полагаем, что $N \geqslant 3$. При этом условии, следуя работам [2, 3, 22], рассмотрим вспомогательные процессы с дискретным временем, фазовое пространство которых составляют всевозможные списки заданий.

Если $K \in \mathbb{G}_{N-1}$ (т.е. $K = \overline{1, N}\backslash\{t\}$, где $t \in \mathbf{I}(\overline{1, N})$), то $\boldsymbol{T}[K]$ обозначим множество всех таких кортежей

$$(K_i)_{i \in \overline{0, N-2}}: \overline{0, N-2} \to \mathbb{G}, \quad (4.8)$$

что

$$(K_0 = K) \ \& \ (K_t = K_{t-1}\backslash\{s\} \ \forall t \in$$
$$\in \overline{1, N-2} \ \forall s \in \mathbf{I}(K_{t-1})). \quad (4.9)$$

Иными словами, начальному состоянию K мы сопоставим пучок траекторий (4.8), каждая из которых имеет динамику, определяемую в (4.9). Последнее можно связать с разностными включениями: полагая при $P \in \mathbb{G}$, что $\mathcal{K}[P] \triangleq \{P\backslash\{s\}: s \in \mathbf{I}(P)\}$, мы рассматриваем траектории (4.8), отвечающие начальному условию $K_0 = K$ и включениям

$$K_t \in \mathcal{K}[K_{t-1}], \ t \in \overline{1, N-2}. \quad (4.10)$$

Напомним, что $K \in \mathbb{G}_{N-1}$ у нас зафиксировано; $\boldsymbol{T}[K]$ есть множество всех решений включения (4.10), где $K_0 = K$.

С процессами (4.9), (4.10) естественно связываются области достижимости (ОД) в конечный и промежуточные "моменты времени": если $K \in \mathbb{G}_{N-1}$ и $t \in \overline{0, N-2}$, то

$$\widetilde{\mathbb{T}}[K; t] \triangleq \{K_t : (K_i)_{i \in \overline{0, N-2}} \in \boldsymbol{T}[K]\} \in \mathcal{P}(\mathbb{G}). \quad (4.11)$$

В работе [3, предложение 12] установлено, что

$$\mathbb{G}_{N-(t+1)} = \bigcup_{K \in \mathbb{G}_{N-1}} \widetilde{\mathbb{T}}[K; t] =$$
$$= \bigcup_{j \in \mathbf{I}(\overline{1, N})} \widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; t] \ \forall t \in \overline{0, N-2}. \quad (4.12)$$

Заметим в связи с (4.11) и (4.12), что $\widetilde{\mathbb{T}}[K; 0] = \{K\}$ и $\widetilde{\mathbb{T}}[K; N-2] \subset \mathbb{G}_1$ при $K \in \mathbb{G}_{N-1}$. Разумеется, в силу (4.1) последнее означает, что при $j \in \mathbf{I}(\overline{1, N})$

$$\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; 0] = \{\overline{1, N}\backslash\{j\}\},$$
$$\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; N-2] \subset \mathbb{G}_1. \quad (4.13)$$

При этом [1—3, 22] в условиях фиксированного списка $K \in \mathbb{G}_{N-1}$ система ОД подчиняется следующей рекуррентной процедуре: $\widetilde{\mathbb{T}}[K; 0] = K$ и при $t \in \overline{0, N-3}$

$$\widetilde{\mathbb{T}}[K; t+1] = \{P\backslash\{s\}: P \in \widetilde{\mathbb{T}}[K; t], s \in \mathbf{I}(P)\}. \quad (4.14)$$

Данная процедура позволяет нам построить при заданном начальном условии $K \in \mathbb{G}_{N-1}$ всю систему ОД для наших вспомогательных процессов:

$$(\widetilde{\mathbb{T}}[K; 0] = K) \to \widetilde{\mathbb{T}}[K; 1] \to \ldots \to \widetilde{\mathbb{T}}[K; N-2]. \quad (4.15)$$

С учетом (4.1) получаем, что (4.15) определяет при фиксации $j \in \mathbf{I}(\overline{1, N})$ систему ОД

$$(\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; t])_{t \in \overline{0, N-2}}.$$

С учетом (4.1) мы, следовательно, получаем, что при $j \in \mathbf{I}(\overline{1, N})$ имеет место $\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; 0] = \{\overline{1, N}\backslash\{j\}\}$ и, кроме того, при $t \in \overline{0, N-3}$

$$\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; t+1] = \\ = \{P\backslash\{s\}: P \in \widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; t]), s \in \mathbf{I}(P)\}. \quad (4.16)$$

Таким образом реализуется пошаговая процедура

$$(\widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; 0] = \{\overline{1, N}\backslash\{j\}\}) \to \widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; 1] \to \ldots \\ \ldots \to \widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\}; N-2]. \quad (4.17)$$

5. Представление слоев пространства позиций

Напомним, что при $s \in \overline{1, N}$ и $K \in \mathbb{G}_s$ определено множество $\mathbb{D}_s[K] \in \mathcal{P}'(\overline{1, N} \times \mathbb{G}_s)$. Данные множества условимся именовать клетками пространства позиций. С помощью клеток удается [2, 3] построить некоторые новые слои пространства позиций. В связи с этим напомним, что при $s \in \overline{1, N-1}$, $K \in \mathbb{G}_{N-1}$ и $H \in \widetilde{\mathbb{T}}[K; N-(s+1)]$ (отметим, что $N-(s+1) \in \overline{0, N-2}$) $\mathbb{D}_s[H] \subset D_s$. Как следствие при $s \in \overline{1, N-1}$ и $K \in \mathbb{G}_{N-1}$ определено (см. [1, (28)])

$$\mathcal{D}_s[K] \triangleq \bigcup_{H \in \widetilde{\mathbb{T}}[K;\, N-(s+1)]} \mathbb{D}_s[H] \in \mathcal{P}'(D_s). \quad (5.1)$$

С учетом (4.1) и (5.1) получаем, как следствие, при $s \in \overline{1, N-1}$ и $j \in \mathbf{I}(\overline{1, N})$, что

$$\mathcal{D}_s[\overline{1, N}\backslash\{j\}] = \\ = \bigcup_{H \in \widetilde{\mathbb{T}}[\overline{1, N}\backslash\{j\};\, N-(s+1)]} \mathbb{D}_s[H] \in \mathcal{P}'(D_s). \quad (5.2)$$

В силу [3, предложение 17] имеем теперь

$$D_s = \bigcup_{K \in \mathbb{G}_{N-1}} \mathcal{D}_s[K] = \\ = \bigcup_{j \in \mathbf{I}(\overline{1, N})} \mathcal{D}_s[\overline{1, N}\backslash\{j\}]\ \forall s \in \overline{1, N-1}. \quad (5.3)$$

Случай s = 1. Отдельно рассмотрим вариант, отвечающий случаю $s = 1$. В связи с этим напомним, что согласно (3.16)

$$D_1 = \{(k, \{t\}): t \in \overline{1, N}\backslash\mathbb{K}_1,\ k \in J_1(\{t\})\}. \quad (5.4)$$

Вместе с тем из (4.12) вытекает, что при $K \in \mathbb{G}_{N-1}$

$$\widetilde{\mathbb{T}}[K; N-2] \subset \mathbb{G}_1, \quad (5.4')$$

и при этом $\widetilde{\mathbb{T}}[K; N-2] \neq \varnothing$. Из (5.1) вытекает (как частный случай), что при $K \in \mathbb{G}_{N-1}$

$$\mathcal{D}_1[K] = \bigcup_{H \in \widetilde{\mathbb{T}}[K;\, N-2]} \mathbb{D}_1[H] \in \mathcal{P}'(D_1), \quad (5.5)$$

где согласно (5.4') при всяком выборе $\mathbb{H} \in \widetilde{\mathbb{T}}[K; N-2]$ для некоторого $t \in \overline{1, N}\backslash\mathbb{K}_1$

$$\mathbb{H} = \{t\}. \quad (5.6)$$

В интересах конкретизации индекса t в (5.6) введем [1, (36)] в рассмотрение при $K \in \mathbb{G}_{N-1}$

$$\mathbb{M}_0[K] \triangleq \{h \in \overline{1, N}\backslash\mathbb{K}_1 | \{h\} \in \widetilde{\mathbb{T}}[K; N-2]\}. \quad (5.7)$$

Из (5.5)—(5.7) вытекает, что

$$\mathcal{D}_1[K] = \bigcup_{h \in \mathbb{M}_0[K]} \mathbb{D}_1[\{h\}], \quad (5.8)$$

где [1, предложение 6.1]

$$\widetilde{\mathbb{T}}[K; N-2] = \{\{h\}: h \in \mathbb{M}_0[K]\}. \quad (5.9)$$

Заметим, что множества $\mathbb{M}_0[K]$, $K \in \mathbb{G}_{N-1}$, определяются всякий раз после исполнения рекуррентной процедуры (4.15).

Заметим с учетом (3.2) и (5.4'), что

$$\forall K \in \mathbb{G}_{N-1}\ \forall P \in \widetilde{\mathbb{T}}[K; N-2] \\ \exists\, h \in \overline{1, N}\backslash\mathbb{K}_1: P = \{h\}. \quad (5.10)$$

С учетом (5.7) и (5.10) получаем, что

$$\forall K \in \mathbb{G}_{N-1}\ \forall P \in \widetilde{\mathbb{T}}[K; N-2] \\ \exists\, h \in \mathbb{M}_0[K]: P = \{h\}. \quad (5.11)$$

Замечание 5.1. В самом деле, пусть $K \in \mathbb{G}_{N-1}$ и $P \in \widetilde{\mathbb{T}}[K; N-2]$. Используя (5.10), подберем $p \in \overline{1, N}\backslash\mathbb{K}_1$ так, что $P = \{p\}$. Тогда по выбору P имеем, что $\{p\} \in \widetilde{\mathbb{T}}[K; N-2]$. Итак,

$$p \in \overline{1, N}\backslash\mathbb{K}_1: \{p\} \in \widetilde{\mathbb{T}}[K; N-2]. \quad (5.12)$$

Из (5.7) и (5.12) вытекает, что $p \in \mathbb{M}_0[K]$, чем и завершается проверка (5.11). ■

С учетом [1, предложение 6.1] имеем, что (см. (5.9))

$$\widetilde{\mathbb{T}}[K; N-2] = \{\{h\}: h \in \mathbb{M}_0[K]\}\ \forall K \in \mathbb{G}_{N-1}. \quad (5.13)$$

В свою очередь, из (5.5) и (5.13) вытекает свойство

$$\mathcal{D}_1[K] = \bigcup_{h \in \mathbb{M}_0[K]} \mathbb{D}_1[\{h\}],\ \forall K \in \mathbb{G}_{N-1}. \quad (5.14)$$

Из (3.14) и (5.7) получаем, однако, что

$$J_1(\{t\}) = \\ = \{j \in \overline{1, N}\backslash\{t\} | \{t; j\} \in \mathbb{G}_2\}\ \forall K \in \mathbb{G}_{N-1}\ \forall t \in \mathbb{M}_0[K]. \quad (5.15)$$

В свою очередь, из (3.15) вытекает, что (см. (5.7))

$$\mathbb{D}_1[\{t\}] = \{(j, \{t\}): j \in J_1(\{t\})\}$$
$$\forall\, K \in \mathbb{G}_{N-1}\ \forall t \in \mathbb{M}_0[K]. \qquad (5.16)$$

Из (5.14) и (5.16) вытекает следующее положение (см. (5.15)):

$$\mathcal{D}_1[K] = \bigcup_{h \in \mathbb{M}_0[K]} \{(j, \{h\}): j \in J_1(\{h\})\} =$$
$$= \{(j, \{h\}): h \in \mathbb{M}_0[K], j \in J_1(\{h\})\}\ \forall K \in \mathbb{G}_{N-1}. \qquad (5.17)$$

Вернемся к соотношению (5.1). С учетом (3.8) и (5.1) корректно определяется при $s \in \overline{1, N-1}$ и $K \in \mathbb{G}_{N-1}$ функция

$$\mathcal{W}_s[K] \in \mathcal{R}_+[\mathcal{D}_s[K]] \qquad (5.18)$$

посредством следующего правила:

$$\mathcal{W}_s[K](j, P) \triangleq v_s(j, P)\ \forall\ (j, P) \in \mathcal{D}_s[K]. \qquad (5.19)$$

Тогда, в частности, из (3.9) и (5.19) следует, что

$$\mathcal{W}_s[K](j, P) =$$
$$= v(j, P) \forall s \in \overline{1, N-1}\ \forall K \in \mathbb{G}_{N-1}\ \forall (j, P) \in \mathcal{D}_s[K]. \qquad (5.20)$$

В частности, посредством (5.19), (5.20) при $K \in \mathbb{G}_{N-1}$ определена функция $\mathcal{W}_1[K]$.

Согласно (5.19) и (5.20) имеем

$$\mathcal{W}_1[K](j, P) = v_1(j, P) = v(j, P)\ \forall (j, P) \in \mathcal{D}_1[K]. \qquad (5.21)$$

С учетом (5.17) и (5.21) получаем при $K \in \mathbb{G}_{N-1}$, что функция

$$\mathcal{W}_1[K] \in \mathfrak{R}_+[\mathcal{D}_1[K]] \qquad (5.21')$$

полностью определяется значениями

$$\mathcal{W}_1[K](j, \{h\}) \in [0, \infty[,\ h \in \mathbb{M}_0[K], j \in J_1(\{h\}).$$

При определении данных значений имеет смысл учитывать (3.18). Для этого заметим, что согласно (5.7)

$$\mathbb{M}_0[K] \subset \overline{1, N} \backslash \mathbb{K}_1\ \forall K \in \mathbb{G}_{N-1}. \qquad (5.22)$$

Тогда из (3.18), (5.21) и (5.22) вытекает, что

$$\mathcal{W}_1[K](j, \{h\}) = v_1(j, \{h\}) =$$
$$\mathbf{c}(j, h, \{h\}) + f(h)\ \forall K \in \mathbb{G}_{N-1}$$
$$\forall h \in \mathbb{M}_0[K]\ \forall j \in J_1(\{h\}). \qquad (5.23)$$

Из (5.17), (5.21') и (5.23) вытекает, что функции $W_1[K]$ при $K \in \mathbb{G}_{N-1}$ построены (учитываем (5.22), (5.7), (3.14)).

Рассмотрим теперь вопросы, связанные с преобразованиями

$$\mathcal{W}_t[K] \to \mathcal{W}_{t+1}[K],\ K \in \mathbb{G}_{N-1},\ t \in \overline{1, N-2}. \qquad (5.23')$$

Для этого заметим, прежде всего, что согласно работе [22, предложение 9.1]

$$(s, Q \backslash \{s\}) \in \mathcal{D}_t[K]\ \forall K \in \mathbb{G}_{N-1}\ \forall t \in$$
$$\in \overline{1, N-2}\ \forall\ (x, Q) \in \mathcal{D}_{t+1}[K]$$
$$\forall s \in \mathbf{I}(Q). \qquad (5.24)$$

Из (5.18) и (5.24) вытекает, в частности, что при $K \in \mathbb{G}_{N-1}$, $t \in \overline{1, N-2}$, $(v, Q) \in \mathcal{D}_{t+1}[K]$ и $s \in \mathbf{I}(Q)$ определено

$$\mathcal{W}_t[K](s, Q \backslash \{s\}) \in [0, \infty[.$$

Как следствие, при $K \in \mathbb{G}_{N-1}$, $t \in \overline{1, N-2}$ и $(v, Q) \in \mathcal{D}_{t+1}[K]$ определено значение

$$\min_{s \in \mathbf{I}(Q)} [\mathbf{c}(v, s, Q) + \mathcal{W}_t[K](s, Q \backslash \{s\})] \in [0, \infty[.$$

Согласно работе [22, (10.17)] имеем (см. также конкретизированный для рассматриваемого частного случая вариант [1, предложение 6.3]), что

$$\mathcal{W}_{t+1}[K](v, Q) =$$
$$= \min_{s \in \mathbf{I}(Q)} (\mathbf{c}(v, s, Q) + \mathcal{W}_t[K](s, Q \backslash \{s\})]$$
$$\forall K \in \mathbb{G}_{N-1}\ \forall t \in \overline{1, N-2}\ \forall\ (v, Q) \in \mathcal{D}_{t+1}[K]. \qquad (5.25)$$

Посредством (5.23), (5.25) при каждом $K \in \mathbb{G}_{N-1}$ реализуется следующая рекуррентная процедура:

$$\mathcal{W}_1[K] \to \mathcal{W}_2[K] \to \ldots \to \mathcal{W}_{N-1}[K]. \qquad (5.26)$$

Действительно, $W_1[K]$ определяется посредством (5.23), а преобразование (5.23') определяется посредством (5.25). Отметим, что (5.26) можно с учетом (4.1) рассматривать как процедуру

$$\mathcal{W}_1[\overline{1, N} \backslash \{j\}] \to \mathcal{W}_2[\overline{1, N} \backslash \{j\}] \to \ldots$$
$$\ldots \to \mathcal{W}_{N-1}[\overline{1, N} \backslash \{j\}], \qquad (5.27)$$

где $j \in \mathbf{I}(\overline{1, N})$. В виде (5.26), (5.27) имеем всякий раз процедуру, локализованную для использования того или иного процессора.

Вернемся к построению слоев $v_1, \ldots, v_{N-1}$. Используя процедуру (5.26), мы при каждом $s \in \overline{1, N-1}$ получаем множества $\mathcal{D}_s[K]$, $K \in \mathbb{G}_{N-1}$, которые в объединении реализуют D_s (см. (5.3)); на каждом из этих множеств определена (после реализации процедуры (5.26)) соответствующая функция $W_s[K]$. Теперь (при $s \in \overline{1, N-1}$) построение слоя v_s функции Беллмана осуществляется по следующему правилу: если $(j, Q) \in D_s$, то используя (5.3), подбираем любое множество $\tilde{K} \in \mathbb{G}_{N-1}$, для которого

$$(j, Q) \in \mathcal{D}_s[\tilde{K}], \qquad (5.28)$$

(такое множество $\tilde{K}$ существует в силу (5.3)), после чего, следуя (5.19), полагаем

$$v_s(j, Q) = \mathcal{W}_s[\tilde{K}](j, Q). \qquad (5.29)$$

Итак, v_s конструируется посредством склеивания функций

$$\mathcal{W}_s[K],\ K \in \mathbb{G}_{N-1}.$$

Данная схема построения $v_1, ..., v_{N-1}$ может быть реализована и в терминах рекуррентной процедуры (5.27). Здесь этап (5.28), (5.29) следует осуществлять в терминах подходящего индекса из $\mathbf{I}(\overline{1, N})$.

Итак, пусть $s \in \overline{1, N-1}$. Тогда для построения v_s применяем следующее рассуждение. Если $(l, Q) \in D_s$, то с учетом (5.3) подбираем любой индекс $j \in \mathbf{I}(\overline{1, N})$, для которого

$$(l, Q) \in \mathcal{D}_s[\overline{1, N} \backslash \{j\}],$$

(такой индекс непременно существует; см. (5.3)), после чего с учетом (5.3) и (5.19) имеем

$$v_s(l, Q) = \mathcal{W}_s[\overline{1, N} \backslash \{j\}](l, Q). \qquad (5.30)$$

В данной версии v_s получается склеиванием

$$\mathcal{W}_s[\overline{1, N} \backslash \{j\}], \; j \in \mathbf{I}(\overline{1, N}).$$

Таким образом, в терминах (5.29), (5.30) будут построены все нужные функции v_s, $s \in \overline{1, N-1}$ (напомним, что функция v_0 известна; см. (3.12)). Теперь для определения v_N (см. (3.12)) следует использовать (3.20); напомним, что функция v_N полностью определяется (см. (3.19)) значением V в силу (3.5). Располагая набором $(v_0, v_1, ..., v_N)$, мы можем теперь построить оптимальный маршрут, действуя в духе (3.21), (3.25); см. также заключение раздела 3.

Параллельный алгоритм определения глобального экстремума. Рассмотрим вариант процедуры (5.26), (5.27), ориентированный на определения V (2.11). Для этого модифицируем процедуру (5.26), имея своей целью экономию памяти. Будем следовать при этом идее работы [23, с. 48, 49] (в [23], однако, не рассматривались вопросы, связанные с распараллеливанием).

1^0. Зафиксируем множество $\widetilde{\mathbb{K}} \in \mathbb{G}_{N-1}$ и обсудим модификацию (5.26) при условии $K = \widetilde{\mathbb{K}}$. Сначала определяем $\mathcal{W}_1[\widetilde{\mathbb{K}}]$, используя нужный вариант (5.23), т.е. вычисляя при $h \in \mathbb{M}_0[\widetilde{K}]$ и $j \in J_1(\{h\})$

$$\mathcal{W}_1[\widetilde{\mathbb{K}}](j, \{h\}) = \mathbf{c}(j, h, \{h\}) + f(h). \qquad (5.31)$$

После выполнения всех вычислений вида (5.31) при переборе $h \in \mathbb{M}_0[\widetilde{\mathbb{K}}]$ и $j \in J_1(\{h\})$ мы получаем функцию $\mathcal{W}_1[\widetilde{\mathbb{K}}]$.

Пусть вообще $\tau \in \overline{1, N-2}$, и функция $\mathcal{W}_\tau[\widetilde{\mathbb{K}}]$ уже найдена. Тогда согласно (5.25) определяем

$$\mathcal{W}_{\tau+1}[\widetilde{\mathbb{K}}] \in \mathfrak{R}_+[\mathcal{D}_{\tau+1}[\widetilde{\mathbb{K}}]]:$$

при $(v, Q) \in \mathcal{D}_{\tau+1}[\widetilde{\mathbb{K}}]$ вычисляем

$$\mathcal{W}_{\tau+1}[\widetilde{\mathbb{K}}](v, Q) =$$

$$= \min_{s \in \mathbb{I}(Q)} [\mathbf{c}(v, s, Q) + \mathcal{W}_\tau[\widetilde{\mathbb{K}}](s, Q \backslash \{s\})]. \qquad (5.32)$$

Система значений (5.32) полностью определяет функцию $\mathcal{W}_{\tau+1}[\widetilde{\mathbb{K}}]$.

Если $\tau = N-2$, то в виде $\mathcal{W}_{\tau+1}[\widetilde{\mathbb{K}}]$ получаем $\mathcal{W}_{N-1}[\widetilde{\mathbb{K}}]$ и останавливаем процедуру, являющуюся модификацией (5.26). Если же $\tau < N-2$, то заменяем в памяти вычислителя массив значений $W_\tau[\widetilde{\mathbb{K}}]$ массивом значений $\mathcal{W}_{\tau+1}[\widetilde{\mathbb{K}}]$, который будет использоваться при построении $\mathcal{W}_{\tau+2}[\widetilde{\mathbb{K}}]$ (действуем в духе (5.25) при очевидной смене обозначений).

В итоге реализации данной процедуры

$$\mathcal{W}_1[\widetilde{\mathbb{K}}] \to ... \to \mathcal{W}_{N-1}[\widetilde{\mathbb{K}}]$$

мы получаем только "финальную" функцию $\mathcal{W}_{N-1}[\widetilde{\mathbb{K}}]$.

2^0. Полагаем теперь, что по правилу пункта 1^0 при каждом множестве (списке) $K \in \mathbb{G}_{N-1}$ выполнена процедура

$$\mathcal{W}_1[K] \to ... \to \mathcal{W}_{N-1}[K] \qquad (5.32')$$

с перезаписью слоев вплоть до построения $\mathcal{W}_{N-1}[K]$. В результате мы получаем набор функций

$$\mathcal{W}_{N-1}[K], \; K \in \mathbb{G}_{N-1}. \qquad (5.33)$$

Теперь в терминах (5.33) конструируем v_{N-1} посредством склеивания, подобного (5.29). Итак, поскольку согласно (5.3)

$$D_{N-1} = \bigcup_{K \in \mathbb{G}_{N-1}} \mathcal{D}_{N-1}[K], \qquad (5.34)$$

мы поступаем следующим образом при подсчете значений функции v_{N-1}. Если $(l, Q) \in D_{N-1}$, то с учетом (5.34) подбираем $\hat{K} \in \mathbb{G}_{N-1}$, для которого

$$(l, Q) \in \mathcal{D}_{N-1}[\hat{K}]$$

(такое множество $\hat{K}$ непременно существует в силу (5.34)), после чего полагаем (см. (5.29)), что

$$v_{N-1}(j, Q) = \mathcal{W}_{N-1}[\hat{K}](j, Q). \qquad (5.35)$$

Подчеркнем, что функция $\mathcal{W}_{N-1}[\hat{K}]$ нам известна (см. (5.33)). Таким образом, удается построить всю функцию v_{N-1}, после чего по формуле (3.20) определяется искомое значение V.

Заметим, что в случае, когда нам необходимо наряду с глобальным экстремумом V найти оптимальный маршрут, мы должны для каждого $K \in \mathbb{G}_{N-1}$ построить все функции $\mathcal{W}_1[K], ..., \mathcal{W}_{N-1}[K]$, после чего склеиванием в духе (5.29) найти все функции $v_1, ..., v_{N-1}$ (и разместить их в памяти вычислителя), после чего определить (3.20) и выполнить процедуру построения оптимального маршрута, используя соотношения, подобные (3.21), (3.25).

6. Реализация процедуры построения слоев функции Беллмана независимыми процессорами

В этом разделе описана практическая реализация процедуры построения слоев функции Бел-

лмана независимыми друг от друга процессорами. Итак, рассмотрим схему реализации процедуры с использованием схемы независимых вычислений. Кроме того, мы изменяем здесь некоторые элементы нашей теоретической конструкции. А именно, в дальнейшем мы предполагаем, что каждый слой, связанный с $K \in \mathbb{G}_{N-1}$, обрабатывается несколькими процессорами с общей оперативной памятью (ОЗУ). Называем данную совокупность процессоров узлом кластера. Совокупность узлов образует вычислительный кластер. Такая конструкция (т. е. узел) используется вместо каждого отдельного процессора из предыдущего раздела.

Хранение данных

Рассмотрим вариант хранения данных для одного узла кластера. Для любого множества целевых точек K в хранилище может храниться набор кратчайших путей (далее НКП) (имеются в виду слои функции Беллмана $\mathcal{W}_s[K]$, $K \in \mathbb{G}_{N-1}$), проходящих через это множество. Каждый кратчайший путь в НКП отличается от остальных путей этого НКП своей начальной точкой и является кратчайшим среди всех возможных путей с той же начальной точкой. В НКП может быть максимум столько путей, какова мощность соответствующего множества K, но, как правило, их меньше, так как какие-то точки данного множества не могут быть начальными из-за ограничений в виде условий предшествования. Ввиду этого в качестве контейнера для хранилища НКП используется хэш-таблица в целях экономии памяти по сравнению с массивом. Ключом в хэш-таблице является битовая маска множества, в которой каждый единичный бит в позиции i означает присутствие в данном множестве точки с номером i.

Основной алгоритм

1. Главный процессор формирует семейство $\mathbb{G}_{N-1}$ ($\mathbb{G}_{N-1} = \{\overline{1, N} \backslash \{t\}: t \in \mathbf{I}(\overline{1, N})\}$), для которого каждый элемент представляет собой множество мощности $N-1$. Множества $K \in \mathbb{G}_{N-1}$ распределяются между узлами с помощью протокола MPI. Каждый узел содержит k вычислительных ядер с общей оперативной памятью. На узле, связанном со списком $K \in \mathbb{G}_{N-1}$, слои функции Беллмана $\mathcal{W}_s[K]$, $s \in \overline{1, N-1}$, равномерно распределены между процессорами. Поскольку оперативная память на узле общая, то обмена данными между процессорами не проводится. Распределение фрагментов слоев пространства позиций и слоев функции Беллмана

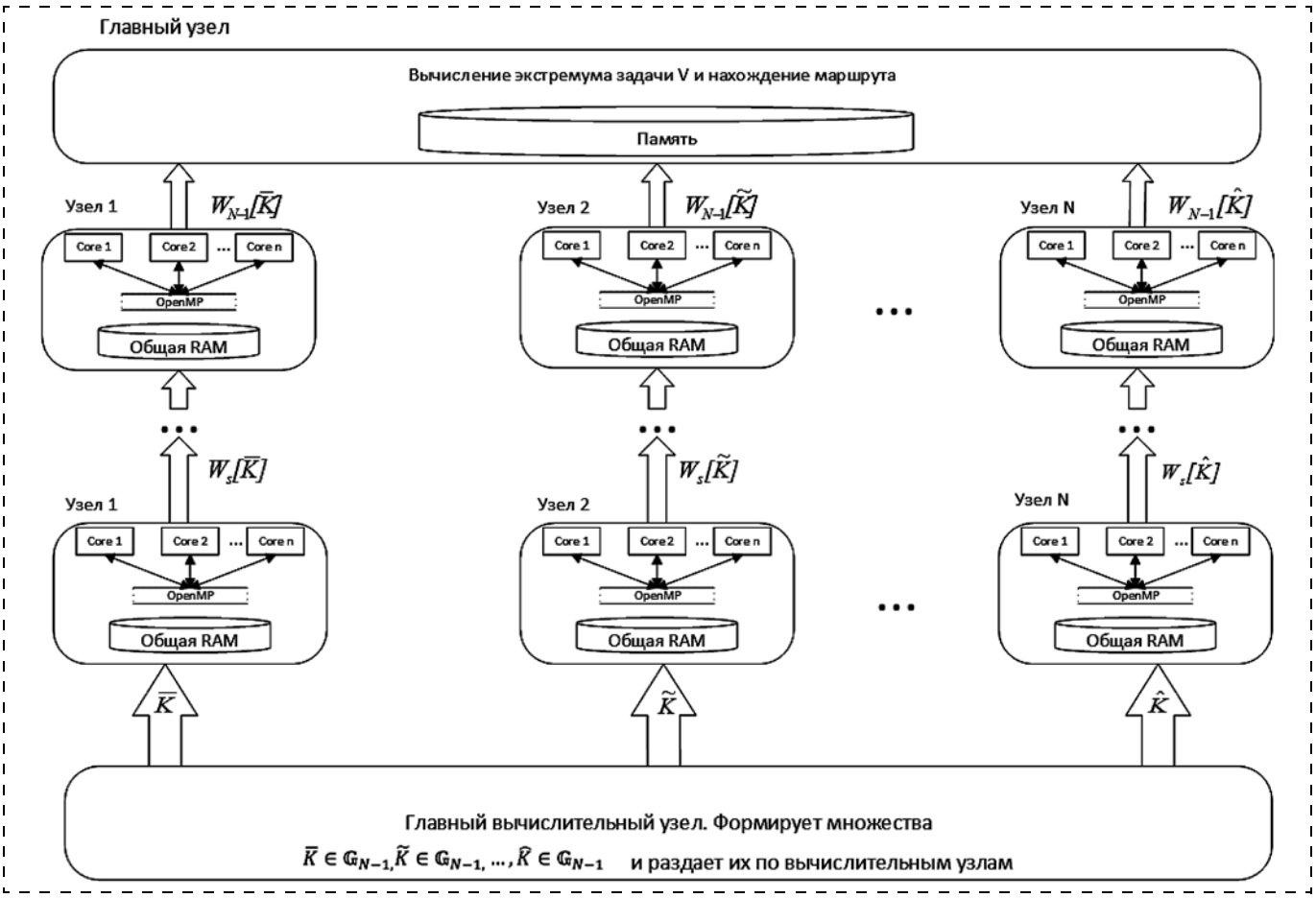

Рис. 1. Структура параллельного алгоритма

между процессорами внутри одного узла выполняется с помощью библиотеки OpenMP.

2. Слои значений функции Беллмана, полученные на первом шаге каждым вычислительным узлом, собираются на главный узел. После этого слой v_{N-1} вычисляется с помощью (5.34) и (5.35). И, наконец, рассчитывается значение экстремума задачи, т.е. V из (3.20).

3. Главный процессор строит оптимальный маршрут путем нахождения локальных экстремумов подобно (3.21) и (3.24).

Работу алгоритма иллюстрирует рис. 1.

7. Практическая реализация решения задачи маршрутизации, осложненной условиями предшествования

В данном разделе рассматривается решение задачи маршрутизации на плоскости с использованием супервычислителя "Уран". Функцию стоимости **c** полагаем сейчас для простоты не зависящей от списка заданий и определяем посредством евклидова расстояния на плоскости $\mathbf{R} \times \mathbf{R}$, функцию f полагаем тождественно равной нулю. Будем рассматривать в данном разделе случай, когда число городов равно 45, т.е. $N = 45$. Города определяются точками на плоскости, базе отвечает стартовая точка $x^0 = (95{,}492;\ 87{,}568)$. Мы рассматриваем здесь множество $\mathbb{K}$, содержащее 35 адресных пар, определяющих условия предшествования: множество $\mathbb{K}$, задается следующими адресными парами: (5,25) (2,25) (29,28) (3,18) (43,30) (20,40) (25,38) (20,0) (29,2) (30,24) (20,7) (44,26) (27,18) (5,3) (11,5) (5,9) (18,29) (36,28) (15,34) (1,32) (7,39) (26,42) (27,21) (8,41) (32,4) (41,28) (5,29) (35,12) (39,4) (1,10) (32,15) (31,15) (44,32) (37,25) (34,6), где в качестве первого аргумента задается отправитель, а второй аргумент указывает на получателя.

В целях практической проверки теоретической работы была написана программа для суперкомпьютера "Уран" на языке программирования C++. Программа работает в среде 64-разрядной операционной системы семейства Linux. Вычислительный эксперимент проводили на вычислительных узлах кластера "Уран" со следующими характеристиками:

- два шестиядерных процессора Intel® Xeon® X5675 (3.07 GHz);
- оперативная память 192 GB;
- кэш-память 2 × 12 MB Level 2 cache;
- 8 GPU Tesla M2090 (6 ГБ Global Memory);
- локальный жесткий диск 400 GB.

В эксперименте было использовано 19 узлов кластера, каждый узел содержит 12 вычислительных ядер. Таким образом, в нашей практической реализации алгоритма было задействовано 228 вычислительных ядра.

В результате вычислений был найден маршрут: ($i_1 = 8$, $i_2 = 37$, $i_3 = 1$, $i_4 = 13$, $i_5 = 44$, $i_6 = 41$, $i_7 = 35$, $i_8 = 11$, $i_9 = 43$, $i_{10} = 16$, $i_{11} = 27$, $i_{12} = 17$, $i_{13} = 22$, $i_{14} = 5$, $i_{15} = 31$, $i_{16} = 10$, $i_{17} = 14$, $i_{18} = 26$, $i_{19} = 3$, $i_{20} = 9$, $i_{21} = 32$, $i_{22} = 33$, $i_{23} = 30$, $i_{24} = 15$, $i_{25} = 18$, $i_{26} = 23$, $i_{27} = 21$, $i_{28} = 20$, $i_{29} = 42$, $i_{30} = 34$, $i_{31} = 6$, $i_{32} = 36$, $i_{33} = 29$, $i_{34} = 7$, $i_{35} = 39$, $i_{36} = 12$, $i_{37} = 4$, $i_{38} = 2$, $i_{39} = 25$, $i_{40} = 28$, $i_{41} = 0$, $i_{42} = 40$, $i_{43} = 38$, $i_{44} = 19$, $i_{45} = 24$) (рис. 2).

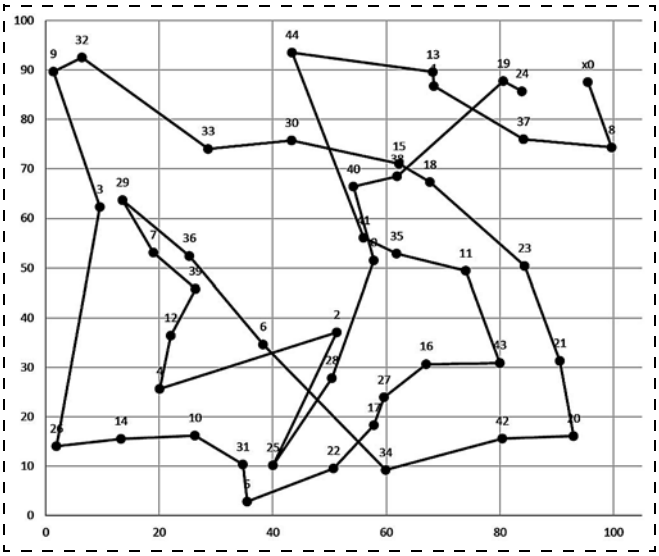

Рис. 2. Оптимальный маршрут в евклидовой задаче с условиями предшествования (стоимости перемещений не зависят от списка заданий)

Были получены следующие результаты: $V = 777{,}322449$ (экстремум задачи), время счета 24 662 с, максимальный объем оперативной памяти, используемый на одном вычислительном узле, составил 94 620 Мбайт.

Заключение

В статье построен параллельный алгоритм для решения задачи курьера, т.е. задачи коммивояжера с условиями предшествования и функциями стоимости, допускающими зависимость от списка заданий. Такого рода зависимость может возникать, в частности, в задачах атомной энергетики, связанных со снижением облучаемости персонала АЭС. В основе алгоритма находится параллельный вариант метода ДП, реализующий построение слоев функции Беллмана (построение всего массива данной функции не предполагается).

Список литературы

1. **Ченцов А. Г., Кошелева М. С.** Динамическое программирование в задаче курьера со стоимостями, зависящими от списка заданий // Мехатроника, автоматизация, управление. 2015. Т. 16, № 4. С. 232—244.

2. **Ченцов А. Г.** Одна параллельная процедура построения функции Беллмана в обобщенной задаче курьера с внутренними работами // АиТ. 2012. № 3. С. 134—149.

3. **Ченцов А. Г.** Одна параллельная процедура построения функции Беллмана в обобщенной задаче курьера с внутренними работами // Вестник ЮУрГУ. Сер. Мат. моделирование и программирование. 2012. № 18. Вып. 12. С. 53—76.

4. **Ташлыков О. Л.** Методы оценки и снижения дозовых нагрузок при ремонте АЭС. Екатеринбург: УГТУ-УПИ, 2009. 118 с.

5. **Ташлыков О. Л.** Дозовые затраты персонала в атомной энергетике. Анализ. Пути снижения. Оптимизация. Saarbrücken, Germany: LAP LAMBERT Academic Publishing GmbH & Co. RG, 2011. 232 с.

6. **Коробкин В. В., Сесекин А. Н., Ташлыков О. Л., Ченцов А. Г.** Методы маршрутизации и их приложения в задачах повышения безопасности и эффективности эксплуатации атомных станций. М.: Новые технологии, 2012. 234 с.

7. **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. Т. 13. № 2 (35). С. 280—286.

8. **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.

9. **Петунин А. А., Ченцов А. Г., Ченцов П. А.** К вопросу о маршрутизации движения инструмента в машинах листовой резки с числовым программным управлением // Науч.-техн. ведомости СПбГПУ. Сер. Информатика. Телекоммуникации. Управление. 2013. № 2 (169). С. 103—111.

10. **Меламед И. И., Сергеев С. И., Сигал И. Х.** Задача коммивояжера. I. Вопросы теории; II. Точные алгоритмы; III. Приближенные алгоритмы // АиТ. 1989. № 9. С. 3—34; № 10. С. 3—29; № 11. С. 3—26.

11. **Gutin G., Punnen A. P.** The Traveling Salesman Problem and Its Variations // Kluwer. 2002. 830 p.

12. **William J. Cook.** In pursuit of the traveling salesman. Mathematics at the limits of computation. Princeton University Press, NJ. 2012. P. 248.

13. **Литл Дж., Мурти К., Суини Д., Кэрел К.** Алгоритм для решения задачи о коммивояжере // Экономика и математические методы. 1965. Т. 1 (Вып. 1). С. 94—107.

14. **Беллман Р.** Применение динамического программирования к задаче о коммивояжере // Кибернет. сборник. Т. 9. М.: Мир, 1964. С. 219—228.

15. **Хелд М., Карп Р. М.** Применение динамического программирования к задачам упорядочения // Кибернет. сборник. 1964. Т. 9. С. 202—218.

16. **Ченцов А. Г.** Экстремальные задачи маршрутизации и распределения заданий: вопросы теории. М., Ижевск: НИЦ "Регулярная и хаотическая динамика", 2008. 240 с.

17. **Куратовский К., Мостовский А.** Теория множеств. М.: Мир. 1970. 416 с.

18. **Дьедонне Ж.** Основы современного анализа. М.: Мир, 1964. 430 с.

19. **Кормен Т., Лейзерсон Ч., Ривест Р.** Алгоритмы: построение и анализ. М.: МЦНМО, 1990. 960 с.

20. **Ченцов А. А., Ченцов А. Г., Ченцов П. А.** Элементы динамического программирования в экстремальных задачах маршрутизации // Проблемы управления. 2013. № 5. С. 12—21.

21. **Ченцов А. Г.** Задача последовательного обхода мегаполисов с условиями предшествования // АиТ. 2014. № 4. С. 170—190.

22. **Ченцов А. Г.** К вопросу о маршрутизации комплексов работ // Вестник Удм. ун-та. Математика. Механика. Комп. науки. 2013. № 1. С. 59—82.

23. **Ченцов А. Г., Ченцов А. А.** К вопросу о нахождении значения маршрутной задачи с ограничениями // Проблемы управления и информатики. 2016. № 1. С. 41—54.

Dynamic Programming Method in a Routing Problem: a Scheme of Independent Computations

A. G. Chentsov[1, 2], chentsov@imm.uran.ru✉, **A. M. Grigoryev**[1], ag@uran.ru,
N. N. Krasovskii Institute of Mathematics and Mechanics, 620990, Ekaterinburg, Russian Federation
[2]Ural Federal University, 620002, Ekaterinburg, Russian Federation

Corresponding author: **Chentsov Aleksandr G.,** D. Sc., Professor,
N. N. Krasovskii Institute of Mathematics and Mechanics,
620990, Ekaterinburg, Russian Federation;
Ural Federal University, 620002, Ekaterinburg, Russian Federation
E-mail: chentsov@imm.uran.ru

Received on July 7, 2016
Accepted on July 14, 2016

The article is devoted to an implementation of the scheme of independent computations for solving the routing problem with precedence constraints and (in the theoretical part) with the cost functions depending on the list of tasks. The prototype of the considered statement is the well-known intractable traveling salesman problem, however, the considered statement is distinguished by the above-mentioned qualitative features. The problem is solved by the dynamic programming method. It is not necessary to construct the whole array of the values of the Bellman function, which is motivated by a desire to decrease the computational complexity. The parallel algorithm for determining the value of the problem value (the global extremum) and the "complete" solution algorithm, which includes the construction of an optimal route, are treated separately. The latter algorithm was realized on the Uran supercomputer; it made use of the (finite) nodes system. Every node had multiple processors, and the totality of the nodes formed the computational cluster. It was possible for certain values of the Bellman function to be (independently) computed by different processors. The developed methods may be applied to certain problems of the nuclear power generation, including the problem of minimization of the radiation exposure of the staff during the emergency situations, similar to Chernobyl and Fukushima. Another possible application may be connected with machine engineering problems, such as routing of a cutting tool of CNC cutting machines. And the last but not the least, this statement may be used to model many problems in the sea and air transportation, and also certain problems in the airborne wildfire detection.

Keywords: *dynamic programming, route, precedence constraints, feasible sets (of tasks), state space layers, independent computations, Bellman function*

For citation:

Chentsov A. G., Grigoryev A. M. Dynamic Programming Method in a Routing Problem: a Scheme of Independent Computations, *Mekhatronika, Avtomatizatsiya, Upravlenie,* 2016, vol. 17, no. 12, pp. 834—846.

DOI: 10.17587/mau.17.834-846

References

1. **Chentsov A. G., Kosheleva M. S.** *Dinamicheskoe programmirovanie v zadache kur'era so stoimostjami, zavisjashhimi ot spiska zadanij* (Dynamic Programming in the Precedence Constrained TSP with the Costs Depending on a List of Tasks), *Mehatronika, Avtomatizacija, Upravlenie*, 2015, vol. 16, no. 4, pp. 232—244 (in Russian).
2. **Chentsov A. G.** *Odna parallel'naja procedura postroenija funkcii Bellmana v obobshhennoj zadache kur'era s vnutrennimi rabotam*i (One parallel procedure for the construction of the Bellman function in the generalized problem of the courier with the inner workings), *AiT*, 2012, no. 3, pp. 134—149 (in Russian).
3. **Chentsov A. G.** *Odna parallel'naja procedura postroenija funkcii Bellmana v obobshhennoj zadache kur'era s vnutrennimi rabotami* (One parallel procedure for the construction of the Bellman function in the generalized problem of the courier with the inner workings), *Vestnik JuUrGU. Ser. Mat. Modelirovanie i Programmirovanie*, 2012, no. 18, vol. 12, pp. 53—76 (in Russian).
4. **Tashlykov O. L.** *Metody ocenki i snizhenija dozovyh nagruzok pri remonte AJeS* (Assessment methods and reduction radiation exposure at nuclear power plant repair), Ekaterinburg, UGTU-UPI, 2009, 118 p. (in Russian).
5. **Tashlykov O. L**. *Dozovye zatraty personala v atomnoj jenergetike. Analiz. Puti snizhenija. Optimizacija* (Personnel Dose Costs in the Nuclear Industry. Analysis. Ways to Decrease. Optimization.), Saarbrüsken, Germany, LAP LAMBERT Academic Publishing GmbH & Co, RG, 2011, 232 p. (in Russian).
6. **Korobkin V. V., Sesekin A. N., Tashlykov O. L., Chentsov A. G.** *Metody marshrutizacii i ih prilozhenija v zadachah povyshenija bezopasnosti i jeffektivnosti jekspluatacii atomnyh stancij* (Routing Methods and Their Applications to the Enhancement of Safety and Efficiency of Nuclear Plant Operation), Moscow, Novye tehnologii, 2012, 234 p. (in Russian).
7. **Petunin A. A.** *O nekotoryh strategijah formirovanija marshruta instrumenta pri razrabotke upravljajushhih programm dlja mashin termicheskoj rezki materiala* (About some strategy of formation of a route of the cutting tool by development of the controlling programs for the thermal sheet cutting machines), *Vestnik UGATU. Serija: Upravlenie, Vychislitel'naja Tehnika i Informatika*, 2009, vol. 13, no. 2 (35), pp. 280—286 (in Russian).
8. **Frolovskij V. D.** *Avtomatizacija proektirovanija upravljajushhih programm teplovoj rezki metalla na oborudovanii s ChPU* (Automation of designing control programs for thermal cutting of metal by CNC equipment), *Informacionnye Tehnologii v Proektirovanii i Proizvodstve*, 2005, no. 4. pp. 63—66 (in Russian).
9. **Petunin A. A., Chentsov A. G., Chentsov P. A.** *K voprosu o marshrutizacii dvizhenija instrumenta v mashinah listovoj rezki s chislovym programmnym upravleniem* (About routing of tool motion in the sheet cutting machines with numerical program control), *Nauch.-tehn. vedomosti SPbGPU. Serija: Informatika. Telekommunikacii. Upravlenie*, 2013, no. 2 (169), pp. 103—111 (in Russian).
10. **Melamed I. I., Sergeev S. I., Sigal I. H.** *Zadacha kommivojazhera. I. Voprosy teorii; II. Tochnye algoritmy; III. Priblizhennye algoritmy* (The traveling salesman problem. I. Issues in theory; II Exact methods; III Approximate algorithms), *AiT*, 1989, no. 9, pp. 3—34; no. 10, pp. 3—29; no. 11, pp. 3—26 (in Russian).
11. **Gutin G., Punnen A. P.** The Traveling Salesman Problem and Its Variations, Kluwer, 2002, 830 p.
12. **William J. Cook.** In pursuit of the traveling salesman. Mathematics at the limits of computation, Princeton University Press, NJ, 2012, pp. 248.
13. **Litl Dzh., Murti K., Suini D., Kjerel K.** *Algoritm dlja reshenija zadachi o kommivojazhere* (Algorithms for Solving the Traveling Salesman Problem), *Jekonomika i Matematicheskie Metody*, 1965, vol. 1 (no. 1), pp. 94—107 (in Russian).
14. **Bellman R.** *Primenenie dinamicheskogo programmirovanija k zadache o kommivojazhere* (Dynamic Programming Treatment of the Travelling Salesman Problem), *Kibernet. Sbornik*, vol. 9, 1964, pp. 219—228 (in Russian).
15. **Held M., Karp R. M.** *Primenenie dinamicheskogo programmirovanija k zadacham uporjadochenija* (A Dynamic Programming Approach to Sequencing Problems. Journal of the Society for Industrial and Applied Mathematics), *Kibernet. Sbornik*, 1964, vol. 9, pp. 202—218 (in Russian).
16. **Chentsov A. G.** *Jekstremal'nye zadachi marshrutizacii i raspredelenija zadanij: voprosy teorii* (Extreme Problems of Routing and Tasks Distribution), Moscow, Izhevsk, NIC "Reguljarnaja i haoticheskaja dinamika", 2008, 240 p. (in Russian).
17. **Kuratovskij K., Mostovskij A.** *Teorija mnozhestv* (*Set Theory*), Moscow, Mir. 1970. 416 p. (in Russian).
18. **D'edonne Zh.** *Osnovy sovremennogo analiza* (Foundations of modern analysis), Moscow: Mir. 1964. 430 p. (in Russian).
19. **Kormen T., Lejzerson Ch., Rivest R.** *Algoritmy: postroenie i analiz* (Introduction to Algorithms), Moscow, MCNMO, 1990, 960 p. (in Russian).
20. **Chentsov A. A., Chentsov A. G., Chentsov P. A.** *Jelementy dinamicheskogo programmirovanija v jekstremal'nyh zadachah marshrutizacii* (Elements of dynamic programming in extremal route problems), *Problemy Upravlenija*, 2013, no. 5, pp. 12—21 (in Russian).
21. **Chentsov A. G.** *Zadacha posledovatel'nogo obhoda megapolisov s uslovijami predshestvovanija* (Problem of successive megalopolis traversal with the precedence conditions), *AiT*, 2014, no. 4, pp. 170—190 (in Russian).
22. **Chentsov A. G.** *K voprosu o marshrutizacii kompleksov rabot* (To question of routing of works complexes), *Vestnik Udm. un-ta. Matematika. Mehanika. Komp. nauki*, 2013, no. 1, pp. 59—82 (in Russian).
23. **Chentsov A. G., Chentsov A. A.** *K voprosu o nahozhdenii znachenija marshrutnoj zadachi s ogranichenijami* (On the problem of obtaining the value of routing problem with constraints), *Problemy Upravlenija i Informatiki*. 2016, no. 1, pp. 41—54 (in Russian).