



УДК 519.6

А.А. Петунин, А.Г. Ченцов, П.А. Ченцов
Екатеринбург, Россия

К ВОПРОСУ О МАРШРУТИЗАЦИИ ДВИЖЕНИЯ ИНСТРУМЕНТА В МАШИНАХ ЛИСТОВОЙ РЕЗКИ С ЧИСЛОВЫМ ПРОГРАММНЫМ УПРАВЛЕНИЕМ

A.A. Petunin, A.G. Chentsov, P.A. Chentsov
Ekaterinburg, Russia

TO THE QUESTION ABOUT INSTRUMENT ROUTING IN THE AUTOMATED MACHINES OF THE SHEET CUTTING

Рассмотрена задача маршрутизации, связанная с процессом листовой резки на станках с числовым программным управлением. Построена математическая модель, а также точный и приближенный алгоритмы, реализованные на ПЭВМ.

ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ. ЭВРИСТИЧЕСКИЙ АЛГОРИТМ. ЗАДАЧА МАРШРУТИЗАЦИИ. МАРШРУТ. УСЛОВИЯ ПРЕДШЕСТВОВАНИЯ.

The routing problem in the sheet cutting machine is considered. The mathematical model, optimal and heuristic algorithms (with software implementation) are constructed.

DYNAMIC PROGRAMMING. HEURISTIC ALGORITHM. ROUTING PROBLEM. ROUTE. PRECEDING CONDITIONS.

При исследовании многих инженерных задач, возникающих в процессе автоматизированного проектирования управляющих программ (УП) для станков и машин с числовым программным управлением (ЧПУ), могут использоваться методы дискретной оптимизации. Некоторые из таких постановок, связанные с оптимизацией параметров времени и стоимости листовой резки на машинах с ЧПУ для термической и гидроабразивной резки материала, приведены, в частности, в [1, 2]. Термическая и гидроабразивная технологии резки предполагают, что для сохранения геометрии заготовки резка должна осуществляться не непосредственно по граничному контуру заготовки, с некоторым запасом, а именно по некоторой эквидистанте, определяемой необходимым припуском на рез. Еще одна особенность такого рода резки — необходимость осуществления предварительной врезки (пробивки) материала перед процессом резки по контуру заготовки. Пробивка материала сопровождается дополнительными деформациями материала в точке врезки, поэтому производится на расстоянии (дистанции) от

контура реза, большем, чем величина припуска на рез. Если используется стандартная техника резки «по замкнутому контуру», то в этом случае каждый замкнутый контур вырезается целиком, и после резки одного контура переход к следующей точке врезки происходит с выключенным инструментом на холостом ходу (рис. 1).

На практике кроме упомянутой стандартной техники резки применяются специальные приемы, которые (в отличие от стандартной техники)

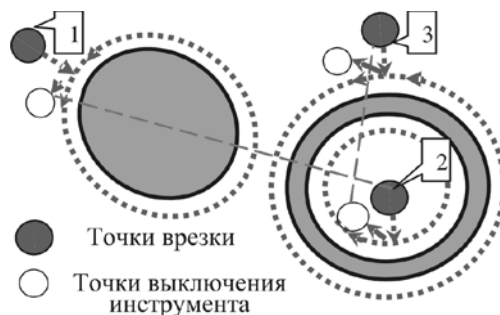


Рис. 1. Пример схемы резки двух заготовок «по замкнутому контуру»

используют либо резку нескольких контуров без выключения инструмента, либо частичную резку замкнутого контура с переходом на резку других контуров и последующим возвратом к первоначальному контуру.

В рамках настоящей статьи, целью которой является разработка эффективных алгоритмов оптимизации времени резки на машинах листовой резки с ЧПУ, будем полагать, что используется стандартная техника резки. Уже на уровне содержательной постановки весьма очевидна потребность в осуществлении рациональной маршрутизации перемещений. В этой связи имеет смысл упомянуть хорошо известную задачу коммивояжера [3–5]. Разумеется, возникают и некоторые особенности. Так, в частности, в рассматриваемой сейчас задаче, связанной с листовой резкой, возникают ограничения в виде условий предшествования (см. задачу курьера в [3]). Другая особенность, связанная с соображениями оптимизации, касается возможной многовариантности перемещений (хотя бы на гипотетическом уровне). Последнее приводит (в отсутствии условий предшествования) к обобщенной задаче коммивояжера или задаче последовательного обхода мегаполисов. Конструкция, реализующая комбинацию двух упомянутых выше особенностей, подробно исследовалась в [6]. В настоящей статье рассматривается применение конструкций [6] к решению задачи листовой резки. Предлагается специализированный вариант динамического программирования и эффективный эвристический алгоритм. В связи с другими подходами к решению проблемы маршрутизации листовой резки отметим [7–12].

Динамическое программирование (специализированная схема)

Рассмотрим некоторые подходы к решению задачи минимизации естественной целевой функции, учитывающие условия вложенности контуров заготовок. Для решения упомянутой выше задачи представляется логичным использовать весьма общую математическую модель, в основе которой находится идея, связанная с оптимизацией последовательного обхода мегаполисов в постановке [6, ч. 4]. Упомянутые мегаполисы имеет смысл разместить вблизи эквидистант контуров деталей, подлежащих резке. Разумеется, упомянутые мегаполисы являются на самом деле объектами виртуальными и используются для

оптимизации как расположения точек врезки, так и самой очередности выполнения операций, связанных с резкой контуров. Сохраняем естественные условия предшествования, придавая им форму, удобную для математического исследования. Будем связывать (в данной постановке) с каждым из контуров конечный набор точек на плоскости, расположенных всякий раз вблизи соответствующего контура. Данные точки – «города мегаполисов» – связываем с гипотетической возможностью врезки. После осуществления оптимизации (глобальной или локальной) можно будет осуществить выбор конкретных точек врезки.

Итак, мы полагаем заданными «виртуальные мегаполисы» M_1, \dots, M_N , определяемые каждый в виде непустого конечного множества на плоскости; каждый из этих мегаполисов располагается возле контура соответствующей детали; здесь N – натуральное число, определяющее количество (замкнутых) контуров, подлежащих резке, $N \geq 2$. Полагаем, что в рамках выбранной модели фиксировано начальное состояние (база) $x^0 \in R \times R$ (здесь R – вещественная прямая), являющееся плоским вектором. Исполнитель (условный термин) покидает x^0 , посещает все мегаполисы M_1, \dots, M_N , после чего перемещается в заданную конечную точку $y^0 \in R \times R$. В процессе перемещения должны выполняться все условия предшествования. В данной постановке не обсуждается сам процесс резки – движения инструмента по контурам, поскольку при любом выборе маршрута время, затрачиваемое на саму резку, одно и то же. Однако это обстоятельство учитывается далее при постановке задачи о последовательном обходе мегаполисов, где (на макроуровне) отождествляются пункты прибытия и отправления для каждого мегаполиса, а особенности, связанные с врезкой и перемещением к эквидистанте, учитываются на уровне определения функций стоимости.

Рассмотрим множество \mathbf{P} всех перестановок множества $1, \bar{N} = \{i \in \mathbf{N} | i \leq N\}$, где $\mathbf{N} = \{1; 2; \dots\}$. Введем, кроме того, множество \mathbf{K} , $\mathbf{K} \subset 1, \bar{N} \times 1, \bar{N}$, элементами которого являются упорядоченные пары (У/П) индексов из $1, \bar{N}$. В терминах \mathbf{K} определяются условия предшествования. В простейшем варианте (ограничимся сейчас этим случаем), когда у каждой детали имеется самое большее один внутренний контур (речь, например, может идти о шайбе) \mathbf{K} составляется из У/П, для каждой из которых можно указать деталь, имею-

щую внутренней контур, причем первый элемент такой У/П соответствует индексу мегаполиса, связанного с обслуживанием внутреннего контура, а второй – индексу мегаполиса, связанного с обслуживанием внешнего контура. Такие У/П условимся называть *адресными*. Итак, \mathbf{K} есть совокупность всех адресных У/П. Мы не исключаем случая $\mathbf{K} = \emptyset$ (резка деталей без внутреннего контура; круги, квадраты и т. п.).

Будем полагать выполненными условия (см. [6, гл. 4]): 1) $x^0 \notin M_i \quad \forall i \in \overline{1, N}$; 2) $M_p \cap M_q = \emptyset$ при $p \neq q$; 3) для каждого непустого множества $\mathbf{K}_0, \mathbf{K}_0 \subset \mathbf{K}$, существует $z_0 \in \mathbf{K}_0 : pr_1(z_0) \neq pr_2(z) \quad \forall z \in \mathbf{K}_0$ (здесь $pr_1(\bullet)$ и $pr_2(\bullet)$ определяют проектирование У/П; см. [6, с. 16]). Впрочем, в рассматриваемой задаче третье условие, как правило, выполняется.

Нам требуется специальное понятие трассы, согласованной с маршрутом: если $\alpha \in \mathbf{P}$, то через \mathbf{X}_α обозначаем (непустое конечное) множество всех кортежей

$$(x_i)_{i \in \overline{0, N}} : \overline{0, N} \rightarrow R \times R \quad (1)$$

(здесь $\overline{0, N} := \{0\} \cup \overline{1, N}$), для каждого из которых $x_0 = x^0$ и $x_j \in M_{\alpha(j)} \quad \forall j \in \overline{1, N}$. Будем использовать только маршруты, допустимые по предшествованию (маршрут $\alpha \in \mathbf{P}$ допустим, если $\forall z \in \mathbf{K} \quad \forall t_1 \in \overline{1, N} \quad \forall t_2 \in \overline{1, N}$

$$((\alpha(t_1) = pr_1(z)) \& (\alpha(t_2) = pr_2(z))) \Rightarrow (t_1 < t_2);$$

итак, постулируется более ранняя обработка внутренних контуров): множество всех таких маршрутов есть \mathbf{A} [6, (2.1.5)], причем $\mathbf{A} \neq \emptyset$. Тогда У/П $(\alpha, (x_i)_{i \in \overline{0, N}})$, $\alpha \in \mathbf{A}$, $(x_i)_{i \in \overline{0, N}} \in \mathbf{X}_\alpha$, и только они – суть допустимые решения задачи, определяемой далее.

Введем $\mathbf{c} : (R \times R) \times (R \times R) \rightarrow [0, \infty[$ и $f : R \times R \rightarrow [0, \infty[$; \mathbf{c} оценивает внешние перемещения, а f – терминальное состояние. В основе конкретного определения \mathbf{c} и f будут позднее закладываться зависимости, связанные с задачей резки, но сейчас их конкретизировать не будем. Сопоставляем каждому кортежу (1) значение

$$\gamma((x_i)_{i \in \overline{0, N}}) := \sum_{i=0}^{N-1} \mathbf{c}(x_i, x_{i+1}) + f(x_N) \in [0, \infty[. \quad (2)$$

Тогда основная задача маршрутизации (ОЗМ) имеет вид

$$\gamma((x_i)_{i \in \overline{0, N}}) \rightarrow \min, \quad \alpha \in \mathbf{A}, \quad (x_i)_{i \in \overline{0, N}} \in \mathbf{X}_\alpha. \quad (3)$$

Заметим, что в задаче (3) критерий (см. (2)) явным образом от $\alpha \in \mathbf{A}$ не зависит. Поэтому на

самом деле (3) можно представлять как задачу

$$\gamma((x_i)_{i \in \overline{0, N}}) \rightarrow \min, (x_i)_{i \in \overline{0, N}} \in \mathbf{X}, \quad (4)$$

где \mathbf{X} есть объединение всех множеств \mathbf{X}_α , $\alpha \in \mathbf{A}$. Разумеется, задачи (3) и (4) эквивалентны. Имеются, однако, некоторые обстоятельства, которые делают постановку (3) предпочтительной (это, в частности, связано со структурой уравнения Беллмана; [см. 6, предложение 3.2.1] в частном случае постоянных мультифункций [6, (2.1.1)], здесь мы ограничиваемся этим случаем). Через V обозначаем значение (экстремум) (3), т. е. наименьшее из чисел $\gamma((x_i)_{i \in \overline{0, N}})$, $(x_i)_{i \in \overline{0, N}} \in \mathbf{X}$. Оптимальным решением называем У/П $(\alpha^0, (x_i^0)_{i \in \overline{0, N}})$, $\alpha^0 \in \mathbf{A}$, $(x_i^0)_{i \in \overline{0, N}} \in \mathbf{X}_{\alpha^0}$, для которой $\gamma((x_i^0)_{i \in \overline{0, N}}) = V$. Разумеется, справедливо равенство

$$V = \min_{\alpha \in \mathbf{A}} \min_{(x_i)_{i \in \overline{0, N}} \in \mathbf{X}_\alpha} \gamma((x_i)_{i \in \overline{0, N}}).$$

Надо найти V и какое-либо из оптимальных решений – в этом состоит решение ОЗМ. В [6, § 4.9] построена экономичная версия метода динамического программирования (МДП), доставляющая оптимальное решение ОЗМ. Сейчас кратко (на идейном уровне) обсудим решение ОЗМ (3) с использованием нестандартной версии МДП, подробно изложенной в монографии [6] и в целом ряде журнальных статей (см. [13, 14] и др.).

Итак, введем семейство \mathbf{F} всех непустых подмножеств $\overline{1, N}$ и $\mathbf{F}_0 := \mathbf{F} \cup \{\emptyset\}$ (семейство всех подмножеств $\overline{1, N}$). Введем также оператор \mathbf{I} [6, (2.2.27), (2.2.28)], действующий в \mathbf{F} (значения \mathbf{I} определены в [6, (2.2.26), (2.2.28)]). При $\mathbf{K} \in \mathbf{F}$ полагаем, что $|\mathbf{K}|$ есть мощность \mathbf{K} , $|\emptyset| = 0$. Введем частичные маршруты и трассы, имея в виду выполнение заданий с индексами из $\mathbf{K} \in \mathbf{F}$: при $J_{\mathbf{K}} := \{i \in \mathbf{N} \mid i \leq |\mathbf{K}|\}$ определяем $(bi)[\mathbf{K}]$ как множество всех биекций [15] $J_{\mathbf{K}}$ на \mathbf{K} . Следуя определению [6, с. 28] и учитывая [6, (2.2.27), (2.2.28), предложение 2.2.3], получаем при $\mathbf{K} \in \mathbf{F}$ непустое [6, (2.2.54)] множество $(\mathbf{I} - bi)[\mathbf{K}]$, для которого $(\mathbf{I} - bi)[\mathbf{K}] \subset (bi)[\mathbf{K}]$. Элементы $(\mathbf{I} - bi)[\mathbf{K}]$ – частичные маршруты посещения мегаполисов $(M_k, k \in \mathbf{K})$, обладающие допустимостью «по вычеркиванию» (напомним, что \mathbf{I} [6] есть оператор вычеркивания задания из списка). При этом, кстати, [6, с. 33], $\mathbf{A} = (\mathbf{I} - bi)[\overline{1, N}]$, чем объясняется тождественность двух типов допустимости в ОЗМ (допустимость по предшествованию и допустимость «по вычеркиванию»).

Полагаем теперь при $K \in F$, что $J_K^0 := \{0\} \cup J_K$, и, при $x \in R \times R$ и $\alpha \in (bi)[K]$, определяем $X(x, K, \alpha)$ как (непустое) множество всех кортежей

$$(x_i)_{i \in J_K^0} : J_K^0 \rightarrow R \times R, \quad (5)$$

таких, что $x_0 = x$ и $x_j \in M_{\alpha(j)} \quad \forall j \in J_K$. Тем самым введены частичные трассы (кстати, $X_\alpha = X(x^0, 1, N, \alpha) \quad \forall \alpha \in P$); для нас это определение существенно в случае $\alpha \in A$. При $K \in F$ каждому кортежу (5) сопоставляется значение

$$\gamma_K((x_i)_{i \in J_K^0}) := \sum_{j=0}^{|K|-1} c(x_j, x_{j+1}) + f(x_N). \quad (6)$$

В частности, (7) можно использовать для оценивания частичных трасс. С учетом этого рассматриваем частичные (укороченные, вообще говоря) задачи:

$$\gamma_K((x_i)_{i \in J_K^0}) \rightarrow \min, \quad \alpha \in (I - bi)[K], \quad (7)$$

$$(x_i)_{i \in J_K^0} \in X(x, K, \alpha),$$

где $x \in R \times R$ и $K \in F$. Для каждой такой задачи определен экстремум (значение)

$$v(x, K) := \min_{\alpha \in (I - bi)[K]} \min_{(x_i)_{i \in J_K^0} \in X(x, K, \alpha)} \gamma_K((x_i)_{i \in J_K^0}) \in [0, \infty[. \quad (8)$$

Полагаем также $v(x, \emptyset) \equiv f(x) \quad \forall x \in R \times R$. Тем самым мы определили функцию v , действующую из $R \times R \times F := (R \times R) \times F$ в $[0, \infty[$.

Предложение 1. Если $x \in R \times R$ и $K \in F$, то справедливо равенство

$$v(x, K) = \min_{j \in I(K)} \min_{y \in M_j} [c(x, y) + v(y, K \setminus \{j\})]. \quad (9)$$

Доказательство легко извлекается из [6, предложение 3.2.1]. Предложение 1 определяет уравнение Беллмана, содержащее, как видно из (9), некоторые особенности (внешний минимум реализуется с учетом правила вычеркивания). Для практического применения предложения 1 следует использовать конструкцию [6, § 4.9], где установлена возможность построения (без потери качества) оптимального решения на основе усеченного массива значений функции Беллмана. Множества из F именуем списками. Следуя [6, § 4.9], введем семейство

$$G \equiv \{K \in F \mid \forall z \in K (pr_1(z) \in K) \Rightarrow (pr_2(z) \in K)\}$$

(при $K = \emptyset$ имеем равенство $G = F$) всех существенных (и неупорядоченных) списков, которые ранжируем по мощности,

полагая $G_s := \{K \in G \mid |s| = |K|\} \quad \forall s \in \overline{1, N}$. При $K_1 \equiv \{pr_1(z) : z \in K\}$ имеем равенство $G_1 = \{\{t\} : t \in \overline{1, N} \setminus K_1\}$ (семейство всех одноэлементных существенных списков). Понятно, что $G_N = \{\overline{1, N}\}$ (семейство, содержащее единственное множество $\overline{1, N}$); наконец,

$$G_{s-1} = \{K \setminus \{t\} : K \in G_s, t \in I(K)\} \quad \forall s \in \overline{2, N}. \quad (10)$$

В (10) определена рекуррентная процедура построения семейств G_1, \dots, G_N . На их основе конструируются слои пространства позиций, обозначаемые D_0, D_1, \dots, D_N . Для определения D_0 введем сначала множество M в виде объединения всех множеств $M_i, i \in \overline{1, N} \setminus K_1$, после чего полагаем $D_0 := \{(x, \emptyset) : x \in M\}$. Пусть, кроме того, $D_N := \{(x^0, \overline{1, N})\}$ (одноэлементное множество, содержащее $\forall \Pi (x^0, \overline{1, N})$ – «настоящую» начальную позицию). Если же $s \in \overline{1, N-1}$, то полагаем $J_s(K) := \{i \in \overline{1, N} \setminus K \mid \{i\} \cup K \in G_{s+1}\} \quad \forall K \in G_s$; после этого определяем D_s в виде

$$D_s := \bigcup_{K \in G_s} \{(x, K) : x \in \bigcup_{i \in J_s(K)} M_i\}.$$

Тогда каждое из множеств D_0, D_1, \dots, D_N непусто и состоит из позиций – элементов множества-произведения $R \times R \times F$. Основное свойство этих множеств имеет вид

$$(y, K \setminus \{k\}) \in D_{s-1} \quad \forall s \in \overline{1, N} \quad \forall (x, K) \in D_s \quad \forall k \in I(K) \quad \forall y \in M_k. \quad (11)$$

Теперь определяем слои функции Беллмана: если $s \in \overline{0, N}$, то

$$v_s : D_s \rightarrow [0, \infty[\quad (12)$$

определяем правилом $v_s(x, K) := v(x, K) \quad \forall (x, K) \in D_s$. Итак, слои (12) определены в виде сужений функции Беллмана на слои пространства позиций. Из предложения 1 и (11) вытекает

Предложение 2. Если $s \in \overline{1, N}$, то v_s определяется системой равенств:

$$v_s(x, K) = \min_{k \in I(K)} \min_{y \in M_k} [c(x, y) + v_{s-1}(y, K \setminus \{k\})] \quad \forall (x, K) \in D_s. \quad (13)$$

Итак, функция v_0 известна ($v_0(x, \emptyset) = f(x) \quad \forall x \in M$), а предложение 2 определяет преобразование $v_{s-1} \rightarrow v_s$ при $s \in \overline{1, N}$.

Рассмотрим нужную конкретизацию функций c и f ; мы полагали эти функции максимально продолженными. Однако с точки зрения ОЗМ реально нас интересуют, в вопросах конкретного задания значений $c(x^{(1)}, x^{(2)})$, лишь следующие

два случая: 1) $x^{(1)} = x^0$ и $x^{(2)} \in M_j$ при некотором $j \in \overline{1, N}$; 2) $x^{(1)} \in M_p$ и $x^{(2)} \in M_q$, где $p \in \overline{1, N}$, $q \in \overline{1, N}$ и $p \neq q$. Для всех прочих векторов $x^{(1)} \in R \times R$ и $x^{(2)} \in R \times R$ можно, например, полагать $c(x^{(1)}, x^{(2)}) := 0$ (имеется в виду У/П $(x^{(1)}, x^{(2)})$), не соответствующие ни первому, ни второму случаям. Воздержимся сейчас от точного описания всех таких несущественных для дальнейшего У/П с элементами в $R \times R$. Заметим далее, что не существует У/П $(x^{(1)}, x^{(2)})$, $x^{(1)} \in R \times R$, $x^{(2)} \in R \times R$, одновременно удовлетворяющих обоим случаям. Итак, значения $c(x^{(1)}, x^{(2)})$ можно задавать отдельно для первого и второго случаев; соответствующее определение c будет корректным. Первый случай является наиболее простым: для $x \in M_j$, где $j \in \overline{1, N}$, полагаем, что $c(x^0, x) := \rho(x^0, x)$, где (здесь и ниже) ρ есть евклидово расстояние в $R \times R$.

Подробнее рассмотрим второй случай, полагая в дальнейшем, что точка выключения инструмента находится всякий раз на эквидистантном контуре, по которому осуществляется резка, и отвечает завершению процесса резки для соответствующего контура. Полагаем, что задан коэффициент $\theta \in R$, $\theta > 1$, характеризующий замедление движения в металле по сравнению с движением холостого хода; можно считать, что реальное расстояние в металле умножается на θ , что позволяет конструировать функцию c в метрических терминах. Пусть при $p \in \overline{1, N}$ L_p есть эквидистантный контур, отвечающий контуру заготовки, имеющему номер p . Полагаем, что L_p есть всякий раз компактное множество. Если к тому же задана точка $x^{(1)} \in M_p$ и точка $x^{(2)} \in R \times R$, то определена некоторая точка $y_p^0[x^{(1)}; x^{(2)}] \in L_p$, с которой будет начата, после перемещения из $x^{(1)}$, резка и закончена; затем последует перемещение в точку $x^{(2)}$ на холостом ходу. Затраты на эту операцию полагаются равными величине

$$c_p(x^{(1)}, x^{(2)}) = \theta \rho(x^{(1)}, y_p^0[x^{(1)}; x^{(2)}]) + \rho(y_p^0[x^{(1)}; x^{(2)}], x^{(2)}) \in [0, \infty[. \quad (14)$$

Мы можем рассматривать различные варианты задания $y_p^0[x^{(1)}; x^{(2)}]$. Так, например, можно полагать, что $y_p^0[x^{(1)}; x^{(2)}] \in L_p$ минимизирует зависимость

$$y \mapsto \theta \rho(x^{(1)}, y) + \rho(y, x^{(2)}) : L_p \rightarrow [0, \infty[.$$

Можно допустить определенное огрубление:

считать, что $y_p^0[x^{(1)}, x^{(2)}]$ – одна из ближайших (в L_p) точек к $x^{(1)}$.

Мы будем считать, что \tilde{M} есть def объединение всех $M_i, i \in \overline{1, N}$. В этих условиях упомянутые выше точки $x^{(1)}$ и $x^{(2)}$ будем выбирать сейчас только в пределах \tilde{M} , фиксируя при $p \in \overline{1, N}$, $x^{(1)} \in M_p$ и $x^{(2)} \in \tilde{M} \setminus \{x^{(1)}\}$ точку $y_p^0[x^{(1)}, x^{(2)}] \in L_p$ и определяя при этом $c_p(x^{(1)}, x^{(2)})$ посредством (14).

Отметим далее, что в силу попарной дизъюнктивности мегаполисов для каждой точки из объединения всех множеств $M_i, i \in \overline{1, N}$, определен единственный индекс $\eta[x] \in \overline{1, N}$, для которого $x \in M_{\eta[x]}$. С учетом этого полагаем в условиях второго случая, что $c(x^{(1)}, x^{(2)}) := c_{\eta[x^{(1)}]}(x^{(1)}, x^{(2)})$. Таким образом, определена существенная «часть» функции стоимости c . Будем считать заданной точку $y^0 \in R \times R$ (точка финиша), для которой $f(x) := \rho(x, y^0) \quad \forall x \in R \times R$.

Эвристический алгоритм

Процедура строится на основе известного жадного алгоритма решения маршрутных задач. Особенность конструкции состоит в итерационном методе его использования. Именно после каждой итерации один из переходов от мегаполиса к мегаполису искусственно делается невыгодным (его стоимость становится очень большой).

Разделим алгоритм на две части: непосредственно сам алгоритм и правило его итерационного применения.

Описание алгоритма по шагам.

1. Находим контур с номером, не являющимся второй компонентой ни в одной из адресных пар, и содержащий точку врезки, стоимость перехода в которую из точки старта минимальна. Делаем данный контур первым на маршруте, а данную точку врезки первой на трассе.

2. Находим контур с номером, не являющимся второй компонентой ни в одной из адресных пар, в которых первая компонента еще не на маршруте, и содержащий точку врезки, стоимость перехода в которую с предыдущего контура на маршруте минимальна. Делаем данный контур и данную точку врезки следующими на маршруте и трассе.

3. Выполняем шаг 2 для всех оставшихся контуров, пока маршрут не будет построен.

4. Для первого контура маршрута среди всех его точек врезки находим такую, которая дает минимальное значение суммы стоимости перехода в нее из старта и стоимости перехода на следующий контур маршрута.

5. Для следующего контура маршрута среди всех его точек врезки находим такую, которая дает минимальное значение суммы стоимости перехода в нее с предыдущего контура маршрута и стоимости перехода на следующий контур маршрута.

6. Выполняем шаг 5 для всех последующих контуров маршрута, кроме последнего.

7. Для последнего контура маршрута среди всех его точек врезки находим такую, которая дает минимальное значение суммы стоимости перехода в нее с предыдущего контура маршрута и стоимости перехода в финишную точку.

8. Выполняем шаги 4–7 заданное количество раз – количество итераций упорядочения точек врезки (обозначим данное количество через $M1$).

Основная идея итерационного подхода состоит в том, чтобы всякий раз направлять алгоритм по новому пути, заставляя находить выходы из локальных экстремумов. Для этого вводятся специальные коррекции стоимостей переходов от контура к контуру – матрица A с коэффициентами $A_{i,j} \in \{0,1\}$, $i \in \overline{1,N}$, $j \in \overline{1,N}$. Если $A_{i,j} = 1$, то стоимость перехода на контур с индексом i в случае, если он находится в маршруте на позиции j , заменяется очень большой константой. В противном случае стоимость рассчитывается традиционным способом.

В итерационном подходе будут использованы следующие параметры. Общее количество итераций $N2$ и размерность цикла итераций $N3$. На каждой из $N2$ итераций, за исключением первой, для одного из контуров, выбранного случайно, делается коррекция. По завершении каждого цикла из $N3$ итераций все коррекции сбрасываются. Далее идет детальное описание алгоритма.

1. Выполняем алгоритм без коррекций. Сохраняем результат счета.

2. Выбираем случайно позицию $i, i \in \overline{1,N}$ на маршруте. В этой позиции находим контур с индексом $j, j \in \overline{1,N}$. Принимаем $A_{j,i} = 1$.

3. Выполняем алгоритм с учетом коррекции. Если результат счета улучшился по сравнению с сохраненным, запоминаем результат и маршрут с трассой.

4. Выполняем шаги 2 и 3 еще $N3-1$ раз ($N3-2$ для первого цикла итераций – первая итерация была на шаге 1 без коррекции).

5. Обнуляем матрицу итераций.

6. Выполняем шаги 2–5 до тех пор, пока количество итераций не превысит число $N2$.

7. Используем сохраненный наилучший результат счета и соответствующие ему маршрут и трассу как результат работы итерационного алгоритма.

Вычислительный эксперимент

Вычисления производились на ПЭВМ с процессором Intel i7-2630QM с 8 Гб оперативной памяти, работающей под управлением Windows 7 (64-bit).

В ходе вычислительного эксперимента рассматривались два примера. Первый пример – небольшой размерности. Данное обстоятельство позволило применить к нему динамическое программирование, чтобы оценить, насколько результат эвристического алгоритма близок к оптимальному. Во втором примере требуется обойти сто пятьдесят контуров, что вполне соответствует средней задаче раскроя. Во всех случаях числовые данные не приводятся в целях экономии места. Точки x^0 и y^0 совпадают с началом координат.

Пример 1. Требуется обойти двадцать четыре контура, при этом восемь из них являются вложенными. Использовались следующие значения параметров: $N1 = 3$, $N2 = 10\,000$, $N3 = 5$. Для учета движения резака в металле $\theta = 50$.

Эвристический алгоритм. Результаты счета показаны на рис. 2. Время счета составило 1 с. Результат счета 724,7. Тот же пример, но без использования итерационного подхода: время счета меньше секунды, результат счета 821,58.

На рисунке квадратиками обозначены точки врезки, маленькими крестиками – точки завершения реза.

Результаты для динамического программирования: время счета 2 мин 10 с, экстремум 718,49. Видно, что результат эвристического алгоритма получился достаточно близким к экстремуму.

Пример 2. Требуется произвести обход ста пятидесяти контуров (пятьдесят контуров были вложенными). Использовались следующие значения параметров: $N1 = 3$, $N2 = 25\,000$, $N3 = 5$. Для учета движения резака в металле $\theta = 50$.

Результаты счета показаны на рис. 3. Время счета составило 90 с. Результат счета 4402,04. Тот же пример, но без использования итерационного подхода: время счета меньше секунды, результат счета 4811,65.

Математическая модель, использующая элементы маршрутизации и условия предшествова-

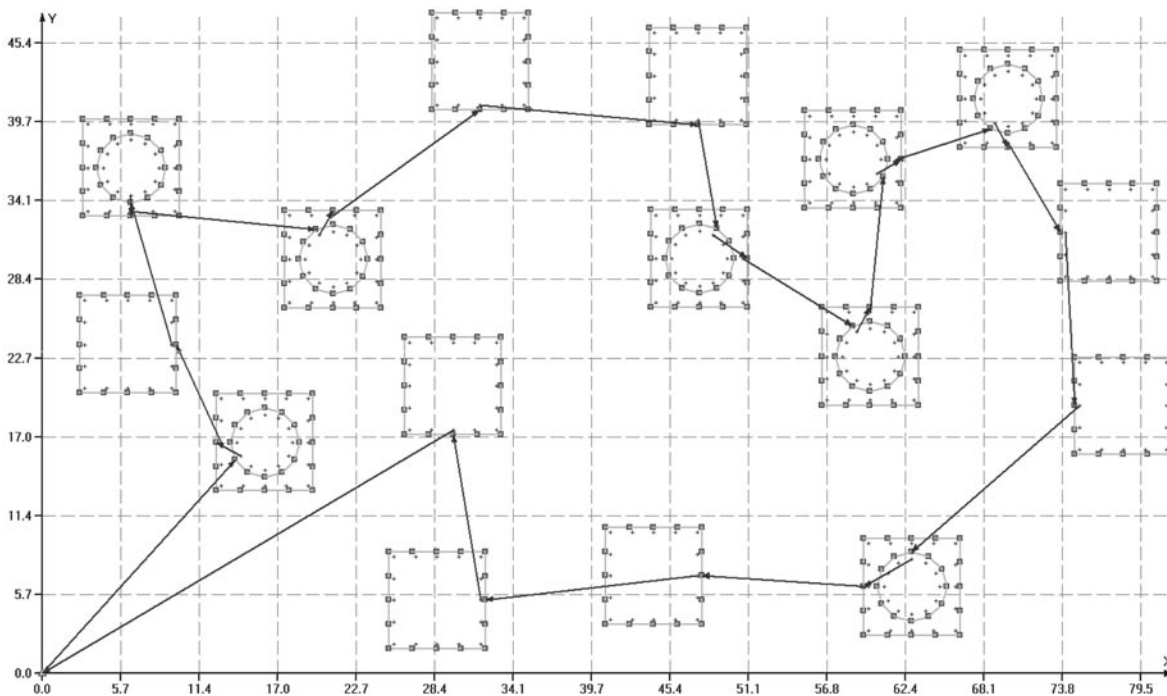


Рис. 2. Результаты счета эвристического алгоритма для примера 1

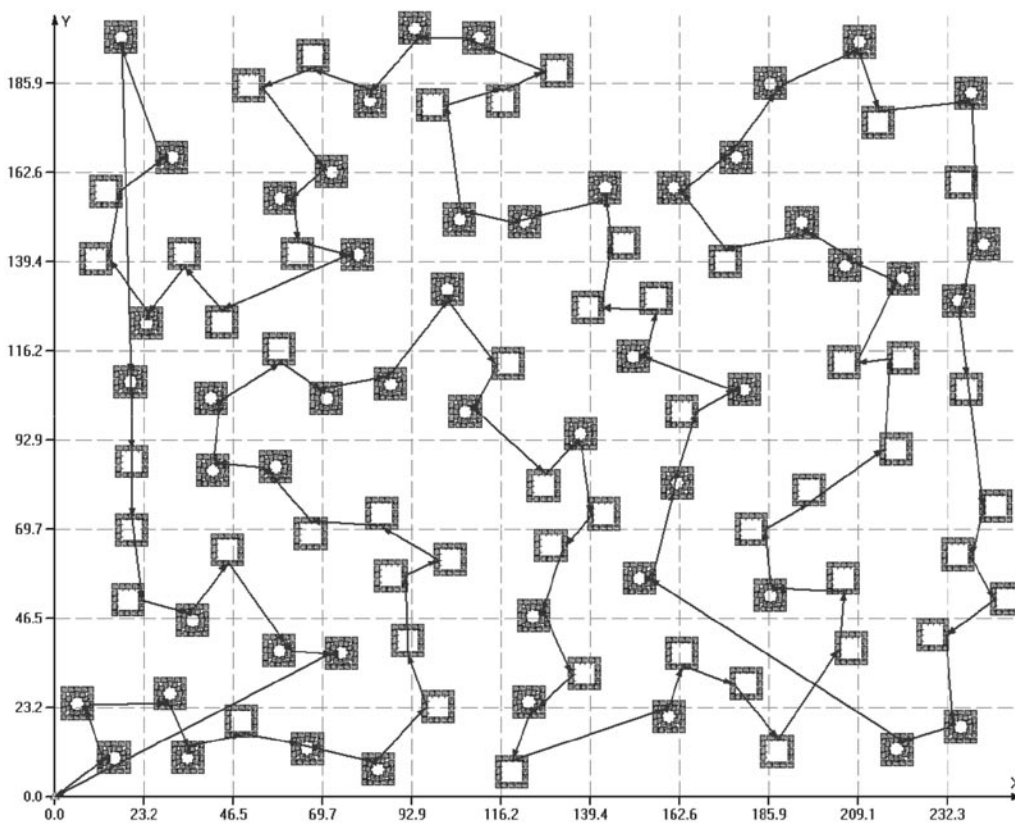


Рис. 3. Результаты счета эвристического алгоритма для примера 2

ния, может быть весьма полезной при решении задач оптимизации процесса резки деталей с применением станков с ЧПУ, включая те случаи, когда размерность задачи достаточно велика. Использование оптимальной процедуры Беллмана в таких случаях затруднено сложностью вычислительной реализации и прежде всего ограниченностью памяти, что существенно в вопросах насчитывания требуемого массива значений функции Беллмана. В случае, когда требуется решать задачи значительной размерности, могут использоваться специальные эвристические алгоритмы. Один из таких алгоритмов приведен в данной статье. Отметим, что при умеренной размерности оптимальная процедура, типа используемой

в [5, § 4.9], может успешно применяться. Один из способов применения оптимальной процедуры – тестирование эвристических алгоритмов.

Эвристический алгоритм, предложенный в статье, обеспечивает соблюдение условий предшествования и обладает хорошим быстродействием, сочетая в себе идею жадного алгоритма и логику итераций. Безусловно, для оценки границ его применимости требуется большой вычислительный эксперимент. В алгоритме присутствует ряд параметров, выбор которых сам по себе представляет серьезную задачу и поле для дальнейших исследований. Кроме того, алгоритм коррекции стоимостей также может быть изменен для достижения наилучших результатов.

СПИСОК ЛИТЕРАТУРЫ

1. **Петунин, А.А.** Разработка САМ системы для машин резки листового материала как пример инновации [Текст] / А.А. Петунин // Инновационные информационные технологии: Теория и практика: Матер. Междунар. сем. (Карлсруе – Уфа – Дрезден, 8–13 апр. 2011). – Уфа, 2011. – С. 47–50.
2. **Петунин, А.А.** Об оптимизационных моделях и алгоритмах решения задач проектирования управляющих программ для машин листовой резки [Текст] / А.А. Петунин // Информационные Технологии и Системы: Матер. Междунар. научн. конф. ИТИС-2012. – Челябинск: ЧелГУ, 2012.
3. **Меламед, И.И.** Задача коммивояжера. Вопросы теории [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 9. – С. 3–34.
4. **Меламед, И.И.** Задача коммивояжера. Точные алгоритмы [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 10. – С. 3–29.
5. **Меламед, И.И.** Задача коммивояжера. Приближенные алгоритмы [Текст] / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 11. – С. 3–26.
6. **Ченцов, А.Г.** Экстремальные задачи маршрутизации и распределения заданий: вопросы теории [Текст] / А.Г. Ченцов. – Ижевск: НИЦ «Регулярная и хаотическая динамика», Ижевский институт компьютерных исследований, 2008. – 240 с.
7. **Петунин, А.А.** О некоторых стратегиях формирования маршрута инструмента при разработке управляющих программ для машин термической резки материала [Текст] / А.А. Петунин // Вестник УГАТУ. Сер. Управление, вычислительная техника и информатика. – 2009. – Т. 13. – № 2 (35). – С. 280–286.
8. **Fischetti, M.** A Branch-and-Cut algorithm for the symmetric generalized traveling salesman problem [Text] / M. Fischetti, J.J. Salazar-Gonzalez, P. Toth // Operations Research. – 1997. – Vol. 45 (3). – P. 378–394.
9. **Snyder, L.V.** A random-key genetic algorithm for the generalized traveling salesman problem [Text] / L.V. Snyder, M.S. Daskin // European J. of Operational Research. – 2006. – Vol. 174. – P. 38–53.
10. **Silberholz, J.** The Generalized Traveling Salesman Problem: a new Genetic Algorithm approach [Text] / J. Silberholz, B. Golden // Extending the Horizons: Advances in Computing, Optimization, and Decision Technologies. – 2007. – P. 165–181.
11. **Gutin, G.** A Memetic Algorithm for the Generalized Traveling Salesman Problem [Text] / G. Gutin, D. Karapetyan // Natural Computing. – Springer, 2010. – Vol. 9 (1). – P. 47–60.
12. **Кочетов, Ю.А.** Методы локального поиска для дискретных задач размещения. Модели и алгоритмы [Текст] / Ю.А. Кочетов. – Saarbrücken: Lambert Academic Publishing, 2011. – 259 с.
13. **Ченцов, А.А.** Экстремальная задача маршрутизации перемещений с ограничениями и внутренними потерями [Текст] / А.А. Ченцов, А.Г. Ченцов, П.А. Ченцов // Изв. вузов. Математика. – 2010. – № 6. – С. 64–81.
14. **Ченцов, А.А.** Об одной задаче маршрутизации с внутренними работами [Текст] / А.А. Ченцов, А.Г. Ченцов // Тр. Ин-та математики и механики УрО РАН. – 2012. – Т. 18. – № 1. – С. 298–317.
15. **Кормен Т.** Алгоритмы: построение и анализ [Текст] / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М.: МЦНМО, 1999. – 960 с.

REFERENCES

1. Petunin A.A. Razrabotka CAM sistemy dlia mashin rezki listovogo materiala kak primer innovatsii / Innovatsionnye informatsionnye tekhnologii: Teoriia i praktika: mater. Mezhdunar. sem. (Karlsruhe – Ufa – Dresden, 8–13 apr. 2011). –Ufa, 2011. –S. 47–50. (rus)
2. Petunin A.A. Ob optimizatsionnykh modeliakh i algoritmakh resheniia zadach proektirovaniia upravliaiushchikh programm dlia mashin listovoi rezki / Informat-sionnye Tekhnologii i Sistemy: Mezhdunar. nauchn. konf. ITIS-2012. –Cheliabinsk: ChelGU, 2012. (rus)
3. Melamed I.I., Sergeev S.I., Sigal I.Kh. Zadacha kommivoiazhera. Voprosy teorii / Avtomatika i telemekhanika. –1989. –№ 9. –S. 3–34. (rus)
4. Melamed I.I., Sergeev S.I., Sigal I.Kh. Zadacha kommivoiazhera. Tochnye algoritmy / Avtomatika i telemekhanika. –1989. –№ 10. –S. 3–29. (rus)
5. Melamed I.I., Sergeev S.I., Sigal I.Kh. Zadacha kommivoiazhera. Priblizhennye algoritmy / Avtomatika i telemekhanika. –1989. –№ 11. –S. 3–26. (rus)
6. Chentsov A.G. Ekstremal'nye zadachi marshrutizatsii i raspredeleniia zadani: voprosy teorii. –Izhevsk: NITs «Reguliarniia i khaoticheskaia dinamika», Izhevskii institut komp'yuternykh issledovani, 2008. –240 s. (rus)
7. Petunin A.A. O nekotorykh strategiiakh formirovaniia marshruta instrumenta pri razrabotke upravliaiushchikh programm dlia mashin termicheskoi rezki materiala / Vestnik UGATU. Ser. Upravlenie, vychislitel'naia tekhnika i informatika. –2009. –T. 13. –№ 2 (35). –S. 280–286. (rus)
8. Fischetti M., Salazar-Gonzalez J.J., Toth P. A Branch-and-Cut algorithm for the symmetric generalized traveling salesman problem. *Operations Research*. –1997. –№ 45 (3). –P. 378–394.
9. Snyder L.V., Daskin M.S. A random-key genetic algorithm for the generalized traveling salesman problem / *European J. of Operational Research*. –2006. –№ 174. –P. 38–53.
10. Silberholz J., Golden B. The Generalized Traveling Salesman Problem: a new Genetic Algorithm approach / *Extending the Horizons: Advances in Computing, Optimization, and Decision Technologies*, 2007. –P. 165–181.
11. Gutin G., Karapetyan D. A Memetic Algorithm for the Generalized Traveling Salesman Problem / *Natural Computing*. –Springer, 2010. –Vol. 9 (1). –P. 47–60.
12. Kochetov Iu.A. Metody lokal'nogo poiska dlia diskretnykh zadach razmeshcheniia. *Modeli i algoritmy*. –Saarbrücken: Lambert Academic Publishing, 2011. –259 s.
13. Chentsov A.A., Chentsov A.G., Chentsov P.A. Ekstremal'naia zadacha marshrutizatsii peremeshchenii s ogranicheniiami i vnutrennimi poteriami / *Izv. vuzov. Matematika*. –2010. –№ 6. –S. 64–81. (rus)
14. Chentsov A.A., Chentsov A.G. Ob odnoi zadache marshrutizatsii s vnutrennimi rabotami // *Tr. In-ta matematiki i mekhaniki UrO RAN*. –2012. –T. 18. – № 1. –S. 298–317. (rus)
15. Kormen T., Leizeron Ch., Rivest R. *Algoritmy: postroenie i analiz*. – Moscow: MTsNMO. –1999. –960 s. (rus)