

К ВОПРОСУ О МАРШРУТИЗАЦИИ ПЕРЕМЕЩЕНИЙ ПРИ ЛИСТОВОЙ РЕЗКЕ ДЕТАЛЕЙ

А.А. Петунин¹, А.Г. Ченцов^{1,2}, П.А. Ченцов^{1,2}

¹Уральский федеральный университет, г. Екатеринбург

²Институт математики и механики им. Н.Н. Красовского УрО РАН, г. Екатеринбург

Рассматривается решение задачи управления инструментом при листовой резке на машинах с ЧПУ. Предполагается, что исходная постановка осложнена различными ограничениями. Требуется построить решение возникающей задачи маршрутизации, соблюдающее ограничения и минимизирующее аддитивный критерий, включающий стоимости (внешних) перемещений и «внутренних» работ, связанных с резкой деталей по замкнутому контуру. Соблюдение ограничений предполагается обеспечивать за счет специального задания функций стоимости, т.е. (по сути) за счет формирования штрафов за нарушение требуемых условий. Главную роль играет при этом процедура на базе широко понимаемого динамического программирования. Конструируемый на данной основе алгоритм реализован в виде стандартной программы на многоядерной ПЭВМ. Изложение этого алгоритма составляет основную цель настоящей работы.

Ключевые слова: маршрутные задачи; условия предшествования; инженерные ограничения.

Введение

Задача управления движением инструмента при листовой резке деталей на машинах с ЧПУ является весьма актуальной в инженерных приложениях, связанных с раскроем, и, вместе с тем, содержит существенные трудности как качественного характера, так и в части вычислительной реализации. Дело в том, что она имеет достаточно большую размерность и включает большое число разнообразных ограничений (условия предшествования, жесткости листа и деталей, тепловые допуски и др.); см. [1–4] и др. Речь идет о построении траектории, посещающей (в режиме резки по замкнутому контуру) эквидистанты контуров вырезаемых деталей с осуществлением всякий раз выбора конкретного направления реза при соблюдении большой серии условий технологического характера. Эквидистанты формируются с тем, чтобы в процессе резки можно было бы избежать повреждений соответствующих деталей; поэтому и осуществляется резка с некоторым запасом. Сами подходы к эквидистантам и близкие к ним точки покидания эквидистант (по завершению реза) могут в принципе выбираться произвольно из соответствующих континуальных множеств. Однако трудности качественного характера (мы получаем для случая «непрерывных» эквидистант многомерную задачу нелинейного программирования со сложными ограничениями) вынуждают к использованию дискретизаций, которые, впрочем, аппроксимируют [5, 6] по результату исходную дискретно-непрерывную задачу в достаточно широком диапазоне условий.

1. Обсуждение задачи

Рассмотрим содержательную постановку задачи, фиксируя достаточно большой прямоугольник на плоскости, в пределах которого намечены детали $\Lambda_1, \dots, \Lambda_n$, подле-

жащие резке. Каждая из деталей Λ_j включает один или несколько замкнутых контуров, объединение которых составляет границу Λ_j . В свою очередь, возле каждого из контуров указана эквидистанта, определяющая безопасный режим резки для данного контура. Поскольку в процессе резки мы имеем дело только с данными (попарно дизъюнктными) эквидистантами, занумеруем их подряд, получая замкнутые кривые $\mathcal{M}_1, \dots, \mathcal{M}_N$, где N – натуральное число, $N \geq 2$. При этом каждое из множеств \mathcal{M}_j – компакт на плоскости.

В начальный момент инструмент находится в точке x^0 объемлющего прямоугольника, не принадлежащей ни одной из эквидистант. Реальный процесс резки состоит в следующем: выбирается способ α нумерации эквидистант, при котором их набор упорядочивается в виде $\mathcal{M}_{\alpha(1)}, \dots, \mathcal{M}_{\alpha(N)}$, после чего инструмент последовательно посещает множества набора в очередности α . Итак, первой посещается эквидистанта $\mathcal{M}_{\alpha(1)}$, затем $\mathcal{M}_{\alpha(2)}$ и т.д. При каждом посещении инструмент прибывает в точку, близкую к соответствующей эквидистанте; осуществляется врезка, после чего инструмент выходит на эквидистанту. При этом, исходя из технологических условий, выбирается направление реза (по часовой или против часовой стрелки), после выполнения которого инструмент вновь попадает в точку начала реза, из которой перемещается к точке выключения инструмента. Последняя близка к точке врезки. Итак, посещение эквидистанты связывается с выбором трех соседних точек, которые в первом приближении можно отождествить, получая упрощенную задачу. С учетом положений [5, 6] для практического решения возникающей дискретно-непрерывной задачи маршрутизации можно использовать модели дискретной оптимизации (ДО), позволяющие достигать экстремум исходной (упрощенной) задачи с любой степенью точности. В основе их построения находится идея дискретизации эквидистант, т.е. обращение к модели своеобразных мегаполисов (ниже используется данная терминология). Более того, в рамках данной модели ДО каждую из точек, образующих в совокупности сетку на соответствующей эквидистанте, можно «разукрупнить», вновь заменив ее триплетом, включающим точки врезки и выключения инструмента, а также точку на эквидистанте, определяющую начало реза. Собственно говоря, последняя точка реализуется при дискретизации эквидистанты, а две первые (близкие к ней точки) добавляются с учетом соображений технологического характера. Время, затрачиваемое непосредственно на рез всех контуров, одно и то же для всех решений исходной задачи, а потому его можно «исключить» из совокупного критерия.

Направление реза по каждой эквидистанте должно быть указано в качестве компоненты совокупного решения. Представляется, однако, что это можно сделать «во вторую очередь», располагая уже некоторым «каркасным» решением в виде пары маршрут-трасса. Данное решение связывается всего лишь с посещением мегаполисов; процессы, связанные с выполнением реза, из постановки на этом уровне исключаются. После осуществления оптимизации, т.е. после нахождения оптимального допустимого решения (ДР), направление реза по каждой отдельно взятой эквидистанте может быть реконструировано.

В настоящей работе исследуется решение задачи последовательного обхода мегаполисов; этап дискретизации «непрерывных» эквидистант, реализуемый по схеме [6], предполагается выполненным. На каждой из эквидистант намечены конечные множества (сетки) возможных точек начала реза, каждой из которых сопоставлены точка врезки и точка выключения инструмента. Работа, связываемая с посещением каж-

дого мегаполиса (локализованного возле эквидистанты), определяется в виде затрат (обычно временных) на перемещение от точки врезки к точке начала реза и от этой последней к точке выключения инструмента. Итак, мы получаем модель мегаполисов с внутренними работами, для которой рассматриваем задачу ДО. Заметим, что традиционные условия предшествования в этой модели относятся только к вариантам нумерации мегаполисов; эти варианты, являющиеся перестановками индексов, ниже именуются маршрутами, в то время как траектории посещения мегаполисов рассматриваются в качестве трасс.

Ограничения динамического характера, связанные, в частности, с тепловыми допусками, касаются в первую очередь трасс, что отражается затем и на выборе маршрутов. В [7] указан общий подход к построению оптимальных решений при таких ограничениях; последние явным образом присутствуют в постановке [7] наряду с условиями предшествования и функциями стоимости, зависящими от списка заданий. Другой вариант (см. [8]), используемый в настоящей работе, состоит в таком преобразовании функций стоимости, при котором нарушение ограничений (формально разрешаемое) приводит к большим потерям в смысле достигаемого качества. По существу речь идет об использовании штрафов за нарушение ограничений. Серьезной проблемой при данном подходе является само построение усложненным упомянутым образом функций стоимости. Один из вариантов такого построения рассматривается в статье в связи с последующей реализацией широко понимаемого динамического программирования (ДП); работа является естественным продолжением [8].

Исследуемая задача маршрутизации имеет своим прототипом известную труднорешаемую [9] задачу коммивояжера (TSP в англоязычной литературе); см. [10–17] и др. Отметим работы [15, 16], связанные с применением ДП для решения задачи коммивояжера, а также метод ветвей и границ (см. [17]). В связи с листовой резкой на машинах с ЧПУ отметим (наряду с [1]) работы [18–21], где основное внимание уделяется эвристическим алгоритмам. Подход, развиваемый в настоящей работе, восходит к [22], где рассматривались методы решения задач маршрутизации, основанные на идеях широко понимаемого ДП.

2. Обозначения и определения общего характера

Используется стандартная теоретико-множественная символика (кванторы, связки, \emptyset – пустое множество). Семейством называем множество, все элементы которого – множества. Произвольным объектам x, y сопоставляем множество $\{x; y\}$, содержащее x, y и не содержащее никаких других элементов. Если z – объект, то $\{z\} \triangleq \{z; z\}$ есть синглетон, содержащий z (одноэлементное множество, для которого $z \in \{z\}$). Кроме того, для любых двух объектов u и v в виде $(u, v) \triangleq \{\{u\}; \{u; v\}\}$ имеем упорядоченную пару (УП) с первым элементом u и вторым элементом v ; см. [23, с. 67]. Если z есть УП, то через $\text{pr}_1(z)$ и $\text{pr}_2(z)$ обозначаем соответственно первый и второй элементы z , однозначно определяемые равенством $z = (\text{pr}_1(z), \text{pr}_2(z))$.

Если a, b и c – объекты, то принимаем $(a, b, c) \triangleq ((a, b), c)$ (триплет определяем подобно [24, с. 17]). Соответственно, имеем [24, с. 17] для всяких непустых множеств A, B и C , что $A \times B \times C \triangleq (A \times B) \times C$; если ϕ есть отображение из $A \times B \times C$ в непустое множество D , $x \in A \times B$ и $y \in C$, то определено значение $\phi(x, y) \in D$ и при этом $\phi(x, y) = \phi(\text{pr}_1(x), \text{pr}_2(x), y)$.

Как обычно, \mathbb{R} – вещественная прямая с обычной упорядоченностью \leq , $\mathbb{R}_+ \triangleq \{\xi \in \mathbb{R} \mid 0 \leq \xi\}$, $\mathbb{N} \triangleq \{1; 2; \dots\}$, $\mathbb{N}_0 \triangleq \{0\} \cup \mathbb{N} = \{0; 1; 2; \dots\}$ и $\overline{p, q} = \{t \in \mathbb{N}_0 \mid (p \leq t) \& (t \leq q)\} \forall p \in \mathbb{N}_0 \forall q \in \mathbb{N}_0$. Если K – непустое конечное множество, то через $|K|, |K| \in \mathbb{N}$, обозначается его мощность; через $(\text{bi})[K]$ обозначаем множество всех биекций [25, с. 87] «промежутка» $\overline{1, |K|}$ на K . Кроме того, полагаем, что $|\emptyset| \triangleq 0$. Перестановка непустого множества есть [25, с. 87] биекция этого множества на себя.

Для всякого множества H через $\mathcal{P}(H)$ обозначаем семейство всех подмножеств (п/м) H ; $\mathcal{P}'(H) \triangleq \mathcal{P}(H) \setminus \{\emptyset\}$ – семейство всех непустых п/м H , а $\text{Fin}(H)$ есть по определению семейство всех конечных множеств из $\mathcal{P}'(H)$, т.е. семейство всех непустых конечных п/м H .

Через $\mathcal{R}_+[T]$ обозначаем множество всех неотрицательных вещественнозначных функций, определенных на непустом множестве T ; если $\mathbf{f} \in \mathcal{R}_+[T]$, то $\mathbf{f} : T \rightarrow \mathbb{R}_+$.

Если X и Y – непустые множества, $g : X \rightarrow Y$ и $Z \in \mathcal{P}(X)$, то $g^1(Z) \triangleq \{g(z) : z \in Z\}$ есть образ множества Z ; $g^1(Z) \neq \emptyset$ при $Z \neq \emptyset$.

3. Постановка задачи

В настоящем разделе совсем кратко излагается общая постановка задачи, приведенная в [8]. Фиксируем непустое множество X (в конкретной задаче, связанной с листовой резкой, X – прямоугольник на плоскости), $x^0 \in X$, $N \in \mathbb{N}$, $N \geq 2$, множества

$$M_1 \in \text{Fin}(X), \dots, M_N \in \text{Fin}(X), \quad (1)$$

а также отношения (п/м декартовых произведений) $\mathbb{M}_1 \in \mathcal{P}'(M_1 \times M_1), \dots, \mathbb{M}_N \in \mathcal{P}'(M_N \times M_N)$. Итак,

$$\emptyset \neq \mathbb{M}_1 \subset M_1 \times M_1, \dots, \emptyset \neq \mathbb{M}_N \subset M_N \times M_N. \quad (2)$$

Полагаем далее, что $x^0 \notin M_j \forall j \in \overline{1, N}$. Пусть $M_p \cap M_q = \emptyset \forall p \in \overline{1, N} \forall q \in \overline{1, N} \setminus \{p\}$.

Замечание 1. В конкретной задаче, связанной с листовой резкой на машинах с ЧПУ, (1) и (2) имеют следующий смысл: при $j \in \overline{1, N}$ множество M_j есть непустое плоское конечное множество, составленное из возможных точек врезки (ВТВ) и отвечающих им точек выключения инструмента (ТВИ), расположенных вблизи эквидистанты \mathcal{M}_j (см. раздел 1) (напомним, что каждая пара ВТВ и ТВИ дополняется единственным образом точкой начала реза, лежащей на \mathcal{M}_j и полученной при дискретизации эквидистант). В (2) мы (при $j \in \overline{1, N}$) группируем ВТВ и ТВИ, составляя из них УП (каждая ТВИ сопоставляется своей точке врезки); совокупность всех УП упомянутого типа составляет отношение \mathbb{M}_j .

Множества (1) именуем далее мегаполисами. Тогда (см. [8, раздел 3]) предметом нашего исследования будут процессы вида

$$(x^{(0)} \triangleq x^0) \longrightarrow (x_{1,1} \in M_{\alpha(1)} \rightsquigarrow x_{1,2} \in M_{\alpha(1)}) \longrightarrow \dots \longrightarrow (x_{N,1} \in M_{\alpha(N)} \rightsquigarrow x_{N,2} \in M_{\alpha(N)}), \quad (3)$$

где $(x_{1,1}, x_{1,2}) \in \mathbb{M}_{\alpha(1)}, \dots, (x_{N,1}, x_{N,2}) \in \mathbb{M}_{\alpha(N)}$; здесь $\alpha \in \mathbb{P}$, где (здесь и ниже) $\mathbb{P} \triangleq (\text{bi})[\overline{1, N}]$ (на выбор α накладываются также условия предшествования). Схема (3) очевидным образом приводится к виду

$$x^0 \rightarrow ((x_{1,1}, x_{1,2}) \in \mathbb{M}_{\alpha(1)}) \rightarrow \dots \rightarrow ((x_{N,1}, x_{N,2}) \in \mathbb{M}_{\alpha(N)}); \quad (4)$$

полагая $z_1 \triangleq (x_{1,1}, x_{1,2}), \dots, z_N \triangleq (x_{N,1}, x_{N,2})$, мы получаем в (4) реализацию процесса в виде системы УП (z_1, \dots, z_N) , которую удобно дополнить (вместо x^0) стартовой УП $z_0 = (x^0, x^0)$. Кортж (z_0, z_1, \dots, z_N) рассматриваем в качестве траектории или трассы.

Условия предшествования. Фиксируем отношение $\mathbf{K} \in \mathcal{P}(\overline{1, N} \times \overline{1, N})$ (итак, $\mathbf{K} \subset \overline{1, N} \times \overline{1, N}$; случай $\mathbf{K} = \emptyset$ не исключается), полагая далее, что $\forall \mathbf{K}_0 \in \mathcal{P}'(\mathbf{K}) \exists z_0 \in \mathbf{K}_0 : \text{pr}_1(z_0) \neq \text{pr}_2(z) \forall z \in \mathbf{K}_0$. Тогда [22, часть 2] имеем следующее свойство: $\mathbf{A} \triangleq \{\alpha \in \mathbb{P} \mid \forall z \in \mathbf{K} \forall t_1 \in \overline{1, N} \forall t_2 \in \overline{1, N} (z = (\alpha(t_1), \alpha(t_2))) \Rightarrow (t_1 < t_2)\} = \{\alpha \in \mathbb{P} \mid \alpha^{-1}(\text{pr}_1(z)) < \alpha^{-1}(\text{pr}_2(z)) \forall z \in \mathbf{K}\} \in \mathcal{P}'(\mathbb{P})$ (итак, $\mathbf{A} \in \text{Fin}(\mathbb{P})$); элементы \mathbf{A} назовем допустимыми по предшествованию (\mathbf{K} -допустимыми) маршрутами. С целью введения трасс и для некоторых других построений рассмотрим следующие п/м $X: \forall j \in \overline{1, N} (\mathfrak{M}_j \triangleq \{\text{pr}_1(z) : z \in \mathbf{M}_j\}) \& (\mathbf{M}_j \triangleq \{\text{pr}_2(z) : z \in \mathbf{M}_j\})$. Тогда, как легко видеть, имеем, что $\mathfrak{X} \triangleq (\bigcup_{i=1}^N \mathfrak{M}_i) \cup \{x^0\} \in \text{Fin}(X)$, $\mathbf{X} \triangleq (\bigcup_{i=1}^N \mathbf{M}_i) \cup \{x^0\} \in \text{Fin}(X)$, $\tilde{\mathfrak{X}} \triangleq \mathfrak{X} \cup \mathbf{X} \in \text{Fin}(X)$.

С этих пор при определении трассы мы вместо X используем множество $\tilde{\mathfrak{X}}$. В этой связи полагаем, что \mathbb{Z} есть множество всех кортежей $\mathbf{z} : \overline{0, N} \rightarrow \tilde{\mathfrak{X}} \times \mathbf{X}$. Тогда при $\alpha \in \mathbb{P}$ в виде $\mathcal{Z}_\alpha \triangleq \{\mathbf{z} \in \mathbb{Z} \mid (z_0 = (x^0, x^0)) \& (z_t \in \mathbf{M}_{\alpha(t)} \forall t \in \overline{1, N})\} \in \text{Fin}(\mathbb{Z})$ имеем (непустое конечное) множество всех трасс посещения мегаполисов, согласованных с маршрутом α . Элементы (непустого конечного) множества $\mathbf{D} \triangleq \{(\alpha, \mathbf{z}) \in \mathbf{A} \times \mathbb{Z} \mid \mathbf{z} \in \mathcal{Z}_\alpha\} \in \text{Fin}(\mathbf{A} \times \mathbb{Z})$ называем ДР (формулируемой ниже задачи). Пусть $\mathfrak{N} \triangleq \mathcal{P}'(\overline{1, N})$.

Функции стоимости. Для определения аддитивного критерия качества введем и зафиксируем функции

$$\mathbf{c} \in \mathcal{R}_+[\mathbf{X} \times \mathfrak{X} \times \mathfrak{N}], c_1 \in \mathcal{R}_+[\mathfrak{X} \times \mathbf{X} \times \mathfrak{N}], \dots, c_N \in \mathcal{R}_+[\mathfrak{X} \times \mathbf{X} \times \mathfrak{N}], f \in \mathcal{R}_+[\mathbf{X}]. \quad (5)$$

Функция \mathbf{c} в (5) используется для оценки внешних перемещений, функции c_1, \dots, c_N – для оценки работ, связанных с посещением мегаполисов, а f – для оценки терминального состояния. Если $\alpha \in \mathbb{P}$ и $\mathbf{z} \in \mathcal{Z}_\alpha$, то $\mathfrak{C}_\alpha[\mathbf{z}] \triangleq \sum_{t=1}^N [\mathbf{c}(\text{pr}_2(\mathbf{z}(t-1)), \text{pr}_1(\mathbf{z}(t)), \alpha^1(t, N))] + c_{\alpha(t)}(\mathbf{z}(t), \alpha^1(t, N))] + f(\text{pr}_2(\mathbf{z}(N)))$. В качестве основной рассматриваем следующую задачу

$$\mathfrak{C}_\alpha[\mathbf{z}] \rightarrow \min, (\alpha, \mathbf{z}) \in \mathbf{D}. \quad (6)$$

С задачей (6) связывается непустое множество оптимальных ДР и значение (данной задачи) или экстремум

$$V \triangleq \min_{(\alpha, \mathbf{z}) \in \mathbf{D}} \mathfrak{C}_\alpha[\mathbf{z}] = \min_{\alpha \in \mathbf{A}} \min_{\mathbf{z} \in \mathcal{Z}_\alpha} \mathfrak{C}_\alpha[\mathbf{z}] \in \mathbb{R}_+. \quad (7)$$

Нашей целью является нахождение значения (7) и какого-либо оптимального ДР.

4. Динамическое программирование

Для решения задачи (6) предлагается использовать вариант [7,8] широко понимаемого ДП, восходящий к [22, § 4.9]. Речь идет о построении слоев функции Беллмана, отвечающих естественному расширению задачи (6), связанному с построением частичных (или укороченных) задач; см., например, [8, раздел 4]. В данном кратком изложении мы стремимся к построению алгоритма, а потому опускаем теоретические

конструкции, подробно изложенные в [7, 8] и в ряде других работ (идейной основой является схема [22, § 4.9]).

Полагаем, что $\mathcal{G} \triangleq \{K \in \mathfrak{N} \mid \forall z \in \mathbf{K} \ (\text{pr}_1(z) \in K) \Rightarrow (\text{pr}_2(z) \in K)\}$, получая семейство существенных списков (заданий); пусть также

$$\mathcal{G}_s \triangleq \{K \in \mathcal{G} \mid s = |K| \ \forall s \in \overline{1, N}\}. \quad (8)$$

Легко видеть, что $\mathcal{G}_N = \{\overline{1, N}\}$, а $\mathcal{G}_1 = \{\{t\} : t \in \overline{1, N} \setminus \mathbf{K}_1\}$, где $\mathbf{K}_1 \triangleq \{\text{pr}_1(z) : z \in \mathbf{K}\}$. Введем оператор вычеркивания \mathbf{I} , действующий в \mathfrak{N} по следующему правилу: если $K \in \mathfrak{N}$, то

$$\mathbf{I}(K) \triangleq K \setminus \{\text{pr}_2(z) : z \in \Xi[K]\}, \quad (9)$$

где $\Xi[K] \triangleq \{z \in \mathbf{K} \mid (\text{pr}_1(z) \in K) \& (\text{pr}_2(z) \in K)\}$. Известно, что

$$\mathcal{G}_{s-1} = \{K \setminus \{t\} : K \in \mathcal{G}_s, t \in \mathbf{I}(K)\} \ \forall s \in \overline{2, N}. \quad (10)$$

Итак (см. (9), (10)), определена рекуррентная процедура $\mathcal{G}_N \rightarrow \mathcal{G}_{N-1} \rightarrow \dots \rightarrow \mathcal{G}_1$, где \mathcal{G}_N известно, а преобразование списков соответствует (10). С помощью списков (8) конструируются слои пространства позиций, обозначаемые через D_0, D_1, \dots, D_N . Полагаем, что

$$D_0 \triangleq \{(x, \emptyset) : x \in \tilde{\mathcal{M}}\}, \quad (11)$$

где $\tilde{\mathcal{M}}$ есть объединение всех множеств $\mathbf{M}_i, i \in \overline{1, N} \setminus \mathbf{K}_1$. Кроме того, полагаем, что $D_N \triangleq \{(x^0, \overline{1, N})\}$, получая синглетон, содержащий УП $(x^0, \overline{1, N})$. Если же $s \in \overline{1, N-1}$ и $K \in \mathcal{G}_s$, то полагаем, что $J_s(K) \triangleq \{j \in \overline{1, N} \setminus K \mid \{j\} \cup K \in \mathcal{G}_{s+1}\}$, $\mathcal{M}_s[K] \triangleq \bigcup_{j \in J_s(K)} \mathbf{M}_j$, $\mathbb{D}_s[K] \triangleq \{(x, K) : x \in \mathcal{M}_s[K]\}$. После этих предваряющих построений

определяем при $s \in \overline{1, N-1}$ слой D_s в следующем виде: $D_s \triangleq \bigcup_{K \in \mathcal{G}_s} \mathbb{D}_s[K] \in \mathcal{P}'(\mathbf{X} \times \mathcal{G}_s)$. Тем самым завершается [8, 22] построение непустых множеств D_0, D_1, \dots, D_N в пространстве $\mathbf{X} \times \mathcal{P}(\overline{1, N})$. Легко видеть, что

$$D_1 = \{(x, \{t\}) : t \in \overline{1, N} \setminus \mathbf{K}_1, x \in \mathcal{M}_1[\{t\}]\}. \quad (12)$$

При $s \in \overline{1, N}$, $K \in \mathcal{G}_s$, $k \in \mathbf{I}(K)$ и $y \in \mathbf{M}_k$ имеем $(y, K \setminus \{k\}) \in D_{s-1}$. Как следствие

$$(\text{pr}_2(z), K \setminus \{j\}) \in D_{s-1} \ \forall s \in \overline{1, N} \ \forall (x, K) \in D_s \ \forall j \in \mathbf{I}(K) \ \forall z \in \mathbf{M}_j. \quad (13)$$

Наконец, мы конструируем функции $v_0 \in \mathcal{R}_+[D_0], v_1 \in \mathcal{R}_+[D_1], \dots, v_N \in \mathcal{R}_+[D_N]$, полагая с учетом (11), что $v_0(x, \emptyset) \triangleq f(x) \ \forall x \in \tilde{\mathcal{M}}$, и определяя v_1, \dots, v_N посредством рекуррентной процедуры, использующей (13). Итак, пусть $s \in \overline{1, N}$; тогда [8, предложение 4.1] $v_s(x, K) = \min_{j \in \mathbf{I}(K)} \min_{z \in \mathbf{M}_j} [c(x, \text{pr}_1(z), K) + c_j(z, K) + v_{s-1}(\text{pr}_2(z), K \setminus \{j\})] \ \forall (x, K) \in D_s$. Итак, имеем рекуррентную процедуру $v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_N$, для которой

$$V = v_N(x^0, \overline{1, N}). \quad (14)$$

Свойство (14) вытекает из общих результатов [6–8] и связано с тем, что функции v_0, v_1, \dots, v_N являются сужениями (единой) функции Беллмана, отвечающей задаче (6). Заметим, что [8, (4.11)] преобразование $v_0 \rightarrow v_1$ приводит (см. (12)) к следующему представлению v_1 :

$$v_1(x, \{t\}) = \min_{z \in \mathbf{M}_t} [c(x, \text{pr}_1(z), \{t\}) + c_t(z, \{t\}) + f(\text{pr}_2(z))] \ \forall t \in \overline{1, N} \setminus \mathbf{K}_1 \ \forall x \in \mathcal{M}_1[\{t\}].$$

В [8, с. 106] изложен естественный алгоритм определения значения задачи, т.е. V , с «перезаписью» слоев функции Беллмана, а также алгоритм построения оптимального ДР.

5. Построение функций стоимости, ответственных за соблюдение динамических ограничений

В настоящем разделе мы возвращаемся к задаче, связанной с листовой резкой, следуя разделу 1. Мы сохраняем традиционную для таких задач структуру условий предшествования: у каждой детали внутренние контура должны вырезаться раньше внешнего; для вложенных деталей более ранней должна быть резка внутренних деталей. Впрочем, не исключаются и другие варианты условий предшествования, но мы их сейчас обсуждать не будем.

Режущий инструмент может перемещаться с двумя скоростями. Именно, со скоростью холостого хода и скоростью реза. В режиме холостого хода инструмент перемещается между контурами. Скорость холостого хода далее будем обозначать через v_i , ее значение обычно равно сотням миллиметров в секунду. Скорость реза используется в режиме реза контура и в режиме подхода к контуру после врезки и отхода от него для выключения инструмента. Далее скорость реза будем обозначать через v_c , ее величина обычно составляет десятки миллиметров в секунду.

В части описания динамических ограничений будем стремиться к учету условий следующего типа, связанного с вопросом об отводе тепла: важно, чтобы завершающий участок резки замкнутого контура приходился на часть листа, в пределах которой было бы достаточно «много металла» (см. [8, раздел 5]). Это требование можно, конечно, отнести, в частности, к выбору направления реза. Важно, однако, чтобы такая возможность действительно существовала при выбранной локализации врезки.

Напомним, что в настоящей конкретной задаче X – невырожденный прямоугольник на плоскости, N соответствует разделу 3, мегаполисы M_1, \dots, M_N локализованы возле соответствующих эквидистант: при $j \in \overline{1, N}$ мегаполис M_j состоит из плоских векторов (элементов X), расположенных возле эквидистанты M_j ; отношение \mathbb{M}_j есть совокупность УП, для каждой из которых первым элементом является точка врезки, а вторым – точка выключения инструмента. Каждая такая пара точек «привязана» к точке начала реза, полученной при дискретизации эквидистанты (см. раздел 1). Это означает, что УП $z \in \mathbb{M}_j$ сопоставлена точка $u(z) \in M_j$, из которой резка может осуществляться в одном из двух направлений; заканчиваться же она будет снова в точке $u(z)$. С этой точкой связываются два возможных финишных участка на эквидистанте, отвечающие резке по часовой и против часовой стрелки. Каждому финишному участку сопоставляется его внешняя ε -окрестность, где $\varepsilon \in \mathbb{R}_+ \setminus \{0\}$ задано: рассматривается пересечение «обычной» евклидовой ε -окрестности множества, определяемого финишным участком, и внешности контура с номером j . Для получающегося плоского множества определена площадь $S(j, z, d)$, где $d = 1$ при резке по часовой стрелке и $d = 0$ в противном случае; значение $S(j, z, d)$ не равно нулю. Кроме того (см. [8, раздел 6]), мы располагаем дополнением множества X до всей плоскости и множеством пустот, образовавшихся при выполнении предыдущих заданий; данное множество может пересекаться с вышеупомянутой внешней ε -окрестностью. Для нас важна площадь получающегося множества-пересечения (мы полагаем при этом,

что при выполнении последовательности операций по резке контуров получающееся множество-пересечение обладает площадью и ее определение не доставляет затруднений), зависящая от списка $\tilde{K} \in \mathcal{P}(\overline{1, N})$ выполненных заданий (\tilde{K} – список уже вырезанных контуров). Разумеется, $\tilde{K} = \overline{1, N} \setminus K$, где K – список еще не выполненных работ, связанных с резкой. Интересующую нас площадь множества-пересечения обозначим через $\tilde{S}(j, z, d, \tilde{K})$, получая число из \mathbb{R}_+ . Обозначая через ρ евклидово расстояние на плоскости и при условии, что $\mathbf{v}_c \in \mathbb{R}_+ \setminus \{0\}$ есть скорость движения инструмента в металле, полагаем $\hat{c}_j(z, d, K) \triangleq \frac{\rho(\text{pr}_1(z), u(z))}{\mathbf{v}_c} + \frac{\rho(u(z), \text{pr}_2(z))}{\mathbf{v}_c} + \frac{\tilde{S}(j, z, d, \overline{1, N} \setminus K)}{S(j, z, d)} P$, где $P \in \mathbb{R}_+$ – заданное и достаточно большое число. Наконец, нужное нам значение стоимости внутренних работ определяем в виде $c_j(z, K) \triangleq \inf(\{\hat{c}_j(z, 0, K); \hat{c}_j(z, 1, K)\})$. Напомним, что $\mathbf{c}, c_1, \dots, c_N, f$ достаточны для постановки задачи (6) и для нахождения в ней оптимального «каркасного» решения (см. [8, раздел 4]).

Предполагается, что стоимости внешних перемещений для $x_1 \in \mathbf{X}, x_2 \in \mathbf{X}$ и $K \in \mathfrak{N}$ имеют вид: $\mathbf{c}(x_1, x_2, K) = \frac{\rho(x_1, x_2)}{\mathbf{v}_i}$, где \mathbf{v}_i – скорость ходостого хода (обычно превышает скорость реза в десятки раз). Итак, полагаем сейчас, что функция \mathbf{c} не зависит от списка заданий.

Пусть «каркасное» оптимальное решение $(\alpha^0, \mathbf{z}^0) \in \mathbf{D}, \mathfrak{C}_{\alpha^0}[\mathbf{z}^0] = V$, уже найдено. Теперь нам следует для каждого $j \in \overline{1, N}$ рассмотреть значения $\hat{c}_{\alpha^0(j)}(\mathbf{z}^0(j), 0, \{\alpha^0(t) : t \in \overline{j, N}\})$, $\hat{c}_{\alpha^0(j)}(\mathbf{z}^0(j), 1, \{\alpha^0(t) : t \in \overline{j, N}\})$, $c_{\alpha^0(j)}(\mathbf{z}^0(j), \{\alpha^0(t) : t \in \overline{j, N}\})$, после чего положить $d^0(j) \triangleq 0$ при $\hat{c}_{\alpha^0(j)}(\mathbf{z}^0(j), 0, \{\alpha^0(t) : t \in \overline{j, N}\}) \leq \hat{c}_{\alpha^0(j)}(\mathbf{z}^0(j), 1, \{\alpha^0(t) : t \in \overline{j, N}\})$ и $d^0(j) \triangleq 1$ в противном случае. В результате будет построен кортеж $\mathbf{d} \triangleq (d^0(j))_{j \in \overline{1, N}}$ со значениями в $\{0; 1\}$. Если $s \in \overline{1, N}$ и $d^0(s) = 1$, то рез контура с номером $\alpha^0(s) \in \overline{1, N}$ следует осуществлять по часовой стрелке; в случае же $d^0(s) = 0$ упомянутую резку надо осуществлять против часовой стрелки. Стоимость внутренних работ, связанных с посещением мегаполиса $\mathbb{M}_{\alpha^0(s)}$, для «каркасного» решения будет при этом совпадать с $c_{\alpha^0(s)}(\mathbf{z}^0(s), \{\alpha^0(t) : t \in \overline{s, N}\})$.

Область завершения реза (уточнение). Пусть на контуре выбрана некоторая точка начала реза. В противоположном резу направлении на контуре откладываем участок длиной $L \in \mathbb{R}_+ \setminus \{0\}$, который далее будем называть участком завершения реза. Вокруг данного участка строим внешнюю ε -окрестность шириной $\varepsilon = H \in \mathbb{R}_+ \setminus \{0\}$, как показано на рис. 1. При этом внешность понимается следующим образом: контур рассматривается как простая замкнутая кривая; в качестве внешней рассматривается неограниченная компонента связности. Далее эту окрестность будем называть областью завершения реза. На рис. 1 в первом случае (слева) данная область всегда полностью покрыта металлом. Во втором случае (справа) верхняя часть области всегда пересекается с пространством за пределами листа. Кроме того, второй участок (прямоугольник с двойной штриховкой) также может представлять из себя участок пересечения с пустотой, если правая деталь уже вырезана, как показано на рис. 1.

На рис. 1 $S(j, z, d)$ соответствует площади всей заштрихованной фигуры (области завершения реза), а $\tilde{S}(j, z, d, \tilde{K})$ – площади фигур с двойной штриховкой (площадь пересечения области завершения реза с пустотами в металле и пространством за пределами листа). Вычисление значений функций $S(j, z, d)$ и $\tilde{S}(j, z, d, \tilde{K})$ представляет из себя достаточно сложную задачу. Так как набор уже вырезанных контуров \tilde{K} все

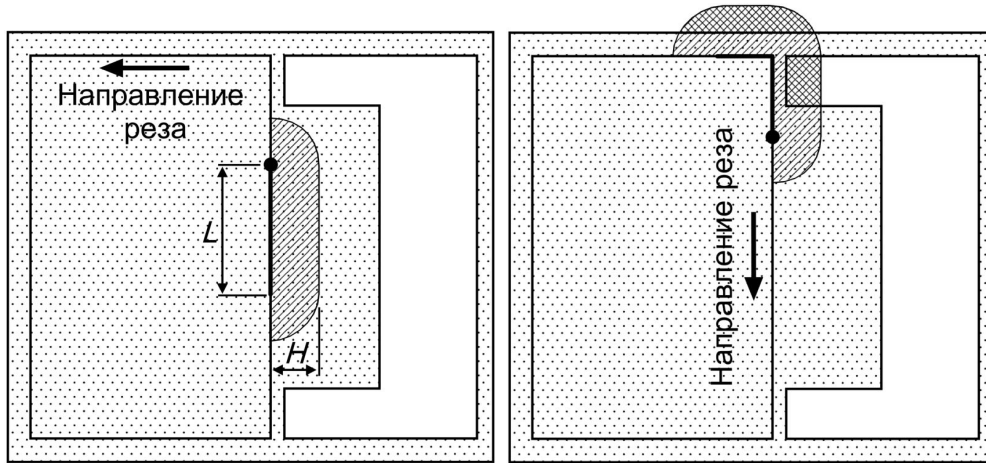


Рис. 1. Область завершения реза

время меняется в процессе решения задачи, то и значение $\tilde{S}(j, z, d, \tilde{K})$ по идее нужно вычислять всякий раз, когда требуется оценить стоимость перехода к следующему контуру. Но при анализе задачи была обнаружена важная особенность: пустоты от разных контуров никогда не пересекаются между собой и с пространством вне листа. Площадь пересечения области завершения реза и фигур, ограниченных выбранным набором контуров, есть сумма площадей пересечений области завершения реза с внутренними частями плоскости, ограниченными каждым из этих контуров.

В связи с последними построениями полезно рассмотреть подробнее некоторые понятия, связанные с контурами. Итак, пусть Q – произвольный контур, отвечающий какой-то детали. Ограничиваясь сейчас вариантом резки по замкнутому контуру, можно считать, что Q есть простая замкнутая кривая, т.е. гомеоморф окружности. Согласно теореме Жордана Q разбивает плоскость на две компоненты связности, одна из которых ограничена, а вторая неограничена. Мы выбираем ограниченную компоненту связности $(\text{всс})[Q]$ (внутреннюю «часть» плоскости, ограниченную контуром Q), после чего рассматриваем семейство всех контуров, содержащихся в $(\text{всс})[Q]$. Нам потребуется отображение, сопоставляющее по смыслу контуру упомянутое семейство.

Полагаем теперь, что все N контуров занумерованы (напомним, что они попарно дизъюнкты): Q_1, \dots, Q_N . В этом случае введем при $k \in \overline{1, N}$ множество $(\text{Всс})[k] \triangleq \{j \in \overline{1, N} \mid Q_j \subset (\text{всс})[Q_k]\}$; полагаем теперь, что $\mathbf{J} : \overline{1, N} \rightarrow \mathcal{P}'(\overline{1, N})$ имеет вид $\mathbf{J}(s) \triangleq (\text{Всс})[s] \forall s \in \overline{1, N}$. Подчеркнем, что $k \in \mathbf{J}(k) \forall k \in \overline{1, N}$.

Для каждого контура $j, j \in \overline{1, N}$, для каждой УП $z, z \in \mathbb{M}_j$, и для каждого направления реза $d, d \in \{0, 1\}$ можно заранее, еще до работы алгоритма маршрутизации и построения матрицы стоимости, вычислить площадь области завершения реза $S(j, z, d)$ и площади пересечения этой области с фигурами, ограниченными каждым из контуров, имеющим индекс $k, k \in \overline{1, N} \setminus \mathbf{J}(j)$, а также с пространством за пределами листа. Если нет пересечения с каким-либо контуром $k, k \in \overline{1, N} \setminus \mathbf{J}(j)$, то площадь пересечения считается нулевой. Кроме того, нулевой считается и площадь пересечения с фигурой, ограниченной любым контуром $k, k \in \mathbf{J}(j)$ – это в дальнейшем облегчает вычисления (избавляет от надобности проверять, является ли данный контур вложенным или нет). Через $\mathbf{S}(j, z, d, k)$ обозначим функцию, которая каждому контуру с индексом j , каждой его УП z и направлению реза d , а также индексу

любого другого контура k , $k \in \overline{1, N} \setminus \{j\}$, ставит в соответствие площадь пересечения области завершения реза контура с индексом j через УП z в направлении d с фигурой, ограниченной контуром, имеющим индекс k , за вычетом фигур, ограниченных контурами с индексами $\mathbf{J}(k) \setminus \{k\}$. Через $\mathbb{S}(j, z, d)$ обозначим функцию, которая каждому контуру с индексом j , каждой его УП z и направлению реза d ставит в соответствие площадь пересечения области завершения реза контура с индексом j через УП z в направлении d с пространством за пределами листа.

Пусть $\tilde{K}, \tilde{K} \subset \overline{1, N}$, есть множество индексов уже вырезанных к данному моменту контуров. Тогда значение $\tilde{S}(j, z, d, \tilde{K})$ может быть вычислено следующим способом: $\tilde{S}(j, z, d, \tilde{K}) = \mathbb{S}(j, z, d) - \sum_{k \in \tilde{K}} \mathbf{S}(j, z, d, k)$. Таким образом, следует вычислить только лишь значения $S(j, z, d)$, $\mathbf{S}(j, z, d, k)$ и $\mathbb{S}(j, z, d)$ для каждого контура j , для каждой УП z и для каждого направления реза d . Следует отметить, что абсолютная точность здесь не нужна. Следовательно, можно воспользоваться эвристическими алгоритмами.

Именно, для вычисления указанных площадей предлагается использовать специальные матрицы. Лист металла и прилегающая область шириной H делится на равные прямоугольники (ячейки). Каждому такому прямоугольнику приписывается числовое значение.

Значение -1 говорит о том, что координаты ячейки (всех четырех ее угловых точек) находятся за пределами листа и отрезки, составляющие прямоугольник ячейки, не пересекаются с краями листа. Ячейке присваивается значение 0 , если хотя бы одна угловая точка находится в пределах листа. Целое число больше нуля соответствует индексу контура, с которым ячейка пересекается, или лежит во внутренней его области. На рис. 2 показан пример такой матрицы. Данный пример приводится исключительно для облегчения восприятия – на практике размерность матрицы составляет тысячи ячеек по каждой оси, что обеспечивает достаточную точность расчетов.

Итак, фиксируем некоторый контур j , УП z , через которую будет производиться рез, и одно из двух направлений реза d . Откладываем в противоположную сторону направлению реза участок длиной L на контуре. Располагаем на нем с равным шагом центры n^* окружностей радиуса H . Для каждой окружности проверяются попавшие в нее ячейки. Если ячейке соответствует индекс текущего контура или одного из его вложенных контуров, то такая ячейка игнорируется, иначе инкрементируется счетчик ячеек области завершения реза. Разумеется, перекрывающиеся ячейки должны учитываться только один раз. После обработки всех окружностей значение этого счетчика умножается на площадь ячейки – так получается площадь области завершения реза, именно, значение $S(j, z, d)$. Кроме того, производится подсчет всех ячеек со значением -1 в пределах указанных окружностей. Приблизительным значением функции $\mathbb{S}(j, z, d)$ будет это количество, умноженное на площадь ячейки.

Точно так же подсчитывается количество ячеек в пределах окружностей, которым соответствует индекс контура. Для $k \in \overline{1, N} \setminus \mathbf{J}(k)$ значение $\mathbf{S}(j, z, d, k)$ получается умножением посчитанного количества ячеек с индексом k в пределах окружностей на площадь ячейки. Для $k \in \mathbf{J}(k)$ значение $\mathbf{S}(j, z, d, k)$ равно 0 .

Для исключения повторного использования одних и тех же ячеек используется булевская матрица той же размерности, что и основная. В ней отмечаются уже обработанные ячейки.

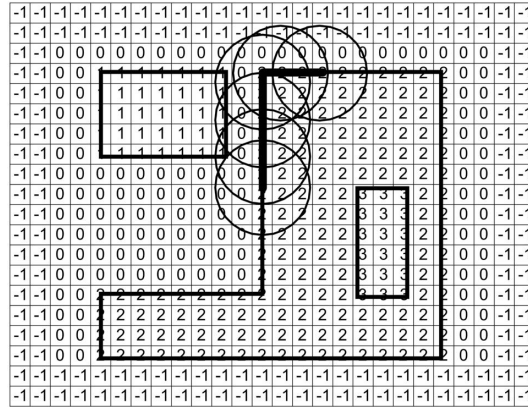


Рис. 2. Подсчет площадей

6. Вычислительный эксперимент

Для проведения вычислительного эксперимента был использован компьютер с процессором Intel i5-2400 и 8Гб оперативной памяти, работающий под управлением Windows 7 (64-bit). Программа разработана на языке C++, в среде Microsoft Visual C++ 2013.

В примерах скорость холостого хода $v_i = 500$ мм/с, скорость реза $v_c = 10$ мм/с. Длина финишного участка реза $L = 150$ мм, ширина области завершения реза $H = 50$ мм. Коэффициент штрафа для внутренних работ $P = 1000000$.

Пример 1. Количество контуров $N = 21$, количество адресных пар $|\mathbf{K}| = 0$. Данное обстоятельство указывает на то, что ни один контур не вложен в другой. Непосредственно координаты контуров и точек не приводятся из соображений объема. Результат счета 70,52 – штрафы от тепловых ограничений отсутствуют (отсутствуют пересечения области завершения реза с пустотами в металле и пространством за пределами листа). Время счета 4 мин 37 с.

Пример 2. Количество контуров $N = 25$, количество адресных пар $|\mathbf{K}| = 11$. Результат счета 83,56 – штрафы от тепловых ограничений отсутствуют. Время счета 4 мин 3 с.

Пример 3. Количество контуров $N = 28$, количество адресных пар $|\mathbf{K}| = 21$. Результат счета 94,85 – штрафы от тепловых ограничений отсутствуют. Время счета 4 мин 26 с.

Выводы по вычислительному эксперименту. Использование адресных пар позволяет существенно сузить множество возможных решений. Теоретически, чем больше адресных пар, тем большую размерность могут иметь примеры, решаемые за сходное время на одном и том же компьютере. Данное предположение полностью подтверждается вычислительным экспериментом. В эксперименте рассматривались три примера. В первом количество контуров 21. Вложений контуров не было вообще, следовательно, адресные пары отсутствовали. Во втором примере количество контуров 25, адресных пар было среднее количество – 11. В третьем случае количество контуров было 28, адресных пар было достаточно много (а именно, 21 адресная пара для 28 контуров при максимально возможном количестве адресных пар 27 (если все

контура вложены один в другой)). Время счета и объем требуемой памяти при этом различаются незначительно.

При расчетах использовались сложные функции стоимости, учитывающие тепловые ограничения на основе описанной в разделе 6 методики. Данный подход позволяет вычислять функции стоимости с быстротой, сопоставимой с вычислением евклидова расстояния. Такое быстродействие достигается за счет предварительного начисывания площадей областей, описанных в разделе 6 и используемых в функциях стоимости.

Во всех случаях прибавка к стоимости внутренних работ от штрафа, связанного с тепловыми ограничениями, равнялась нулю. Это означает, что всякий раз было найдено такое решение, которое позволяет исключить пересечение области завершения реза с пустотами от вырезанных контуров и пространством за пределами листа. И это несмотря на то, что область имела не самые маленькие размеры – 150 мм в длину и 50 мм в ширину.

Данный подход представляется вполне приемлемым для раскроев с небольшим количеством контуров (которое (количество) в некоторых пределах меняется в зависимости от количества вложенных контуров, что несложно учесть). Время счета в несколько минут является вполне приемлемым для производственных целей.

Работа выполнена при финансовой поддержке постановления № 211 Правительства Российской Федерации, контракт № 02.А03.21.0006, и РФФИ (проект 17-08-01385).

Литература

1. Петунин, А.А. О некоторых стратегиях формирования маршрута инструмента при разработке управляющих программ для машин термической резки материала / А.А. Петунин // Вестник УГАТУ. – 2009. – Т. 13, № 2 (35). – С. 280–286.
2. Фроловский, В.Д. Автоматизация проектирования управляющих программ тепловой резки металла на оборудовании с ЧПУ / В.Д. Фроловский // Информационные технологии в проектировании и производстве. – 2005. – № 4. – С. 63–66.
3. Верхотуров, М.А. Математическое обеспечение задачи оптимизации пути режущего инструмента при плоском фигурном раскрое на основе цепной резки / М.А. Верхотуров, П.Ю. Тарасенко // Вестник УГАТУ. – 2008. – Т. 10, № 2 (27). – С. 123–130.
4. Петунин, А.А. К вопросу о маршрутизации движения инструмента в машинах листовой резки с числовым программным управлением / А.А. Петунин, А.Г. Ченцов, П.А. Ченцов // Научно-технические ведомости СПбГПУ. Серия: Информатика. Телекоммуникации. Управление. – 2013. – № 2 (169). – С. 103–111.
5. Chentsov, A.A. Dynamic Programming Method in the Generalized Traveling Salesman Problem: the Influence of Inexact Calculation / A.A. Chentsov, A.G. Chentsov // Mathematical and Computer Modelling. – 2001. – V. 33 – P. 801–819.
6. Ченцов, А.Г. Дискретно-непрерывная задача маршрутизации с условиями предшествования / А.Г. Ченцов, А.А. Ченцов // Труды Института математики и механики УрО РАН. – 2017. – Т. 23, № 1. – С. 275–292.
7. Ченцов, А.Г. Задача маршрутизации с ограничениями, зависящими от списка заданий / А.Г. Ченцов, А.А. Ченцов // Доклады Академии наук. – 2015. – Т. 465, № 2. – С. 154–158.
8. Ченцов, А.Г. Маршрутизация в условиях ограничений: задача о посещении мегаполисов / А.Г. Ченцов, П.А. Ченцов // Автоматика и телемеханика. – 2016. – № 11. – С. 96–117.

9. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М.: Мир, 1982.
10. Меламед, И.И. Задача коммивояжера. Вопросы теории / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 9. – С. 3–34.
11. Меламед, И.И. Задача коммивояжера. Точные методы / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 10. – С. 3–29.
12. Меламед, И.И. Задача коммивояжера. Приближенные алгоритмы / И.И. Меламед, С.И. Сергеев, И.Х. Сигал // Автоматика и телемеханика. – 1989. – № 11. – С. 3–26.
13. Gutin, G. The Traveling Salesman Problem and Its Variations / G. Gutin, A. Punnen. – Berlin: Springer, 2002.
14. Cook, W.J. In Pursuit of the Traveling Salesman. Mathematics at the Limits of Computation / W.J. Cook. – New Jersey: Princeton University Press, 2012.
15. Беллман, Р. Применение динамического программирования к задаче о коммивояжере / Р. Беллман // Кибернетический сборник. – 1964. – Т. 9. – С. 219–228.
16. Хелд, М. Применение динамического программирования к задачам упорядочения / М. Хелд, Р.М. Карп // Кибернетический сборник. – 1964. – Т. 9. – С. 202–218.
17. Литл, Дж. Алгоритм для решения задачи о коммивояжере / Дж. Литл, К. Мурти, Д. Суини, К. Кэрел // Экономика и математические методы. – 1965. – Т. 1, № 1. – С. 94–107.
18. Wang, G.G. Optimal Process Planning for a Combined Punch-and-Laser Cutting Machine Using and Colony Optimization / G.G. Wang, S.Q. Xie // International Journal of Production Research. – 2005. – V. 43, № 11. – P. 2195–2216.
19. Lee, M.-K. Cutting Path Optimization in CNC Cutting Processes Using a Two-Step Genetic Algorithm / M.-K. Lee, K.-B. Kwon // International Journal of Production Research. – 2006. – V. 44, № 24. – P. 5307–5326.
20. Jing, Y. An Optimized Algorithm of Numerical Cutting-Path Control in Garment Manufacturing / Y. Jing, C. Zhige // Advanced Materials Research. – 2013. – V. 796. – P. 454–457.
21. Ганелина, Н.Д. Исследование методов построения кратчайшего пути обхода отрезков на плоскости / Н.Д. Ганелина, В.Д. Фроловский // Сибирский журнал вычислительной математики. – 2006. – Т. 9, № 3. – С. 241–252.
22. Ченцов, А.Г. Экстремальные задачи маршрутизации и распределения заданий: вопросы теории / А.Г. Ченцов. – Ижевск: НИЦ «Регулярная и хаотическая динамика». Ижевский институт компьютерных исследований, 2008.
23. Куратовский, К. Теория множеств / К. Куратовский, А. Мостовский. – М.: Мир, 1970.
24. Дьедонне, Ж. Основы современного анализа / Ж. Дьедонне. – М.: Мир, 1964.
25. Кормен, Т. Алгоритмы: Построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М.: Московский центр непрерывного математического образования, 2002.

Александр Александрович Петунин, доктор технических наук, профессор, кафедра «Информационные технологии и автоматизация проектирования», Уральский федеральный университет (г. Екатеринбург, Российская Федерация), aaretunin@gmail.com.

Александр Георгиевич Ченцов, член-корреспондент РАН, профессор, отдел «Управляемые системы», Институт математики и механики им. Н.Н. Красовского

УрО РАН; кафедра «Прикладная математика и механика», Уральский федеральный университет (г. Екатеринбург, Российская Федерация), chentsov@imm.uran.ru.

Павел Александрович Ченцов, кандидат физико-математических наук, отдел «Вычислительные сети», Институт математики и механики им. Н.Н. Красовского УрО РАН; лаборатория «Оптимальный раскрой промышленных материалов и оптимальные маршрутные технологии», Уральский федеральный университет (г. Екатеринбург, Российская Федерация), chentsov.p@mail.ru.

Поступила в редакцию 6 июня 2017 г.

MSC 93CXX

DOI: 10.14529/mmp170303

ABOUT ROUTING IN THE SHEET CUTTING

A.A. Petunin¹, A.G. Chentsov^{1,2}, P.A. Chentsov^{1,2}

¹Ural Federal University, Ekaterinburg, Russian Federation

²N.N. Krasovskii Institute of Mathematics and Mechanics of the Ural Branch of the RAS, Ekaterinburg, Russian Federation

E-mail: aapetunin@gmail.com, chentsov@imm.uran.ru, chentsov.p@mail.ru

The solution of the problem of tool routing in CNC sheet cutting machines is considered. It is assumed that the initial problem formulation is complicated by different restrictions. It is required to construct a solution of this routing problem that respects the constraints and minimizes the additive criterion, including the costs of (external) displacements and "internal" related to cutting parts on a closed contour works. Compliance of the constraints is supposed to be provided through a special assignment of cost functions, i.e. (in fact) due to the formation of penalties for the restriction violation. The procedure based on widely understood dynamic programming is the main way of problem solving in this paper. The program of problem solving on a multi-core PC is constructed. The presentation of this algorithm is the main goal of this paper.

Keywords: routing problem; precedence conditions; engineering constraints.

References

1. Petunin A.A. [About Some Strategies of the Programming of Tool Route by Developing of Control Programs for Thermal Cutting Machines]. *Scientific Journal of Ufa State Aviation Technical University*, 2009, vol. 13, no. 2 (35), pp. 280–286. (in Russian)
2. Frolovskiy V.D. [Automation of Designing of Control Programs of Thermal Cutting of Metal on the Equipment with CNC]. *Information Technology of CAD/CAM/CAE*, 2005, no. 4, pp. 63–66. (in Russian)
3. Verhoturov M.A., Tarasenko P.Ju. [Mathematical Support of the Task of Optimizing the Path of the Cutting Tool for Flat Pattern Cutting Based on Chain Cutting]. *Scientific Journal of Ufa State Aviation Technical University*, 2008, vol. 10, no. 2 (27), pp. 123–130. (in Russian)
4. Petunin A.A., Chentsov A.G., Chentsov P.A. [To the Question about Instrument Routing in the Automated Machines of the Sheet Cutting]. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2013, no. 2 (169), pp. 103–111.
5. Chentsov A.A., Chentsov A.G. Dynamic Programming Method in the Generalized Traveling Salesman Problem: the Influence of Inexact Calculation. *Mathematical and Computer Modelling*, 2001, vol. 33, issues 8-9, pp. 801–819. DOI: 10.1016/S0895-7177(00)00282-X

6. Chentsov A.G., Chentsov A.A. [A Discrete-Continuous Routing Problem with Precedence Conditions]. *Proceedings of the Institute of Mathematics and Mechanics*, 2017, vol. 23, no. 1, pp. 275–292. (in Russian)
7. Chentsov A.G., Chentsov A.A. Route Problem with Constraints Depending on a List of Tasks. *Doklady Mathematics*, 2015, vol. 92, no. 3, pp. 685–688. DOI: 10.1134/S1064562415060083
8. Chentsov A.G., Chentsov P.A. Routing under Constraints: Problem of Visit to Megalopolises. *Automation and Remote Control*, 2016, vol. 77, no. 11, pp. 1957–1974. DOI: 10.1134/S0005117916110060
9. Garey M., Johnson D. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. N.Y., W.H. Freeman & Co., 1979, 338 p.
10. Melamed I.I., Sergeev S.I., Sigal I.Kh. The Traveling Salesman Problem. Issues in Theory. *Automation and Remote Control*, 1989, vol. 50, no. 9, pp. 1147–1173. (in Russian)
11. Melamed I.I., Sergeev S.I., Sigal I.Kh. The Traveling Salesman Problem. Exact Methods. *Automation and Remote Control*, 1989, vol. 50, no. 10, pp. 1303–1324.
12. Melamed I.I., Sergeev S.I., Sigal I.Kh. The Traveling Salesman Problem. Approximate Algorithms. *Automation and Remote Control*, 1989, vol. 50, no. 11, pp. 1459–1479.
13. Gutin G., Punnen A. *The Traveling Salesman Problem and Its Variations*. Berlin, Springer, 2002.
14. Cook W.J. *In Pursuit of the Traveling Salesman, Mathematics at the Limits of Computation*. New Jersey, Princeton University Press, 2012.
15. Bellman R. Dynamic Programming Treatment of the Travelling Salesman Problem. *Journal of the Association for Computing Machinery*, 1962, vol. 9, issue 1, pp. 61–63. DOI: 10.1145/321105.321111
16. Held M., Karp R.M. A Dynamic Programming Approach to Sequencing Problems. *Journal of the Society for Industrial and Applied Mathematics*, 1962, vol. 10, no. 1, pp. 196–210. DOI: 10.1137/0110015
17. Little J., Murty K., Sweeney D., Karel C. An Algorithm for the Traveling Salesman Problem. *Operations Research*, 1963, vol. 11, issue 6, pp. 972–989. DOI: 10.1287/opre.11.6.972
18. Wang G.G., Xie S.Q. Optimal Process Planning for a Combined Punch-and-Laser Cutting Machine Using and Colony Optimization. *International Journal of Production Research*, 2005, vol. 43, issue 11, pp. 2195–2216. DOI: 10.1080/00207540500070376
19. Lee M.-K., Kwon K.-B. Cutting Path Optimization in CNC Cutting Processes Using a Two-Step Genetic Algorithm. *International Journal of Production Research*, 2006, vol. 44, issue 24, pp. 5307–5326. DOI: 10.1080/00207540600579615
20. Jing Y., Zhige C. An Optimized Algorithm of Numerical Cutting-Path Control in Garment Manufacturing. *Advanced Materials Research*, 2013, vol. 796, pp. 454–457.
21. Ganelina N.D., Frolovsky V.D. [On Constructing the Shortest Circuits on a Set of Line Segments]. *Siberian Journal of Numerical Mathematics*, 2006, vol. 9, no. 3, pp. 241–252. (in Russian)
22. Chentsov A.G. *Ekstremal'nye zadachi marshrutizacii i raspredeleniya zadaniy: voprosy teorii* [Extreme Tasks of Routing and Distribution of Tasks: Theory Questions], Izhevsk, 2008.
23. Kuratowski K., Mostowski A. *Set Theory*. Amsterdam, North-Holland Publishing Company, 1967.
24. Dieudonne J. *Foundations of Modern Analysis*. N.Y., London, Academic Press, 1960.
25. Cormen T., Leiserson C., Rivest R. *Introduction to Algorithms*. MIT Press, McGraw-Hill, 1990.

Received June 6, 2017