Достаточное условие оптимальности решения задачи непрерывной резки

# Введение

Задача построения оптимального маршрута инструмента является важным этапом разработки управляющих программ для машин резки листового материала с ЧПУ.

Как и предшествующая ей задача раскроя, то есть оптимального размещения деталей, подлежащих резке, на листе материала, в общем случае задача резки является NP-полной. В силу этого, её точное решение не может быть получено для реально возникающих в производстве задач (для сотен деталей) за разумное время, поэтому для её решения (так же, как и для задачи раскроя) систематически применяются разнообразные эвристики, дающие решения приемлемого качества в осмысленное время. В то же время вопрос оценки качества полученных решений в сравнении с оптимальным решением как правило остаётся нерешённым и представляет значительный научный интерес.

## Классификация

GSCCP

GSCCP

TSP

GTSP

CCP

ECP

SCCP

ICP

GSCCP

Конечное множество точек врезки

Фиксированные точки врезки

Непрерывное множество точек врезки

Рис 1. Классификация задач резки

## Технологические ограничения

# Задача непрерывной резки

Рассмотрим эвклидову плоскость и на ней – область , ограниченную замкнутым контуром (как правило – прямоугольник), которая представляет собой модель листового материала, подлежащего резке. Пусть внутри задано попарно непересекающихся плоских контуров , ограничивающих деталей . Деталь может быть ограничена как одним контуром, так и несколькими (внешним и внутренними отверстиями), так что в общем случае .

Контура могут иметь произвольную форму, но мы будем рассматривать только случай, когда они состоят из (конечного числа) отрезков прямых и дуг окружностей, что определяется существующим технологическим оборудованием. В частном случае, когда контуры состоят только из отрезков прямых, задача непрерывной резки сводится к одному из вариантов задачи обхода многоугольников (Touring Polygon Problem, TPP).

Далее, в области (как правило, на её границе) задаются две точки, обозначим их (на практике как правило ), представляющие собой начало и конец маршрута резки.

Задача непрерывной резки заключается в поиске:

1. точек врезки
2. Порядка обхода контуров , то есть перестановки элементов

Целевая функция в данном случае сильно упрощается по сравнению с общей задачей резки и сводится к минимизации длины холостого хода

Где для простоты мы вводим обозначения .

Кроме того, мы будем решать задачу оптимизации с дополнительным ограничением, так называемым «ограничением предшествования» (precedence constraint). Хотя контуры не пересекаются, они могут быть вложены друг в друга, то есть , где обозначает 2-мерную фигуру, ограниченную контуром (в более привычных обозначениях ). В общей задаче резки это может быть вызвано двумя разными обстоятельствами (отверстия в деталях и размещение меньших деталей в отверстиях больших для экономии материала), но в данной задаче эти варианты обрабатываются одинаково.

Если один контур расположен внутри другого, то вложенный контур должен быть вырезан (посещён) ранее внешнего: в перестановке. Таким образом, не все перестановки контуров оказываются допустимы.

# Общий алгоритм решения

Предлагаемый алгоритм решения состоит из нескольких этапов, естественно связанных с природой решаемой задачи.

## Удаление внешних контуров

Для автоматического соблюдения ограничений предшествования на первом этапе удаляются все контуры, содержащие вложенные

Это в общем случае приводит к уменьшению размера задачи (в некоторых случаях – значительному), и тем самым сокращает время расчётов на втором и особенно третьем этапе.

## Непрерывная оптимизация

Mi-1

Mi+1

Ci-1

Ci+1

Ci

Mi

M’i

Mi-1

Mi+1

Ci-1

Ci+1

Ci

Mi

M’i

M\*i-1

Рис. 2. Выбор оптимальной точки врезки

На этом этапе мы предполагаем последовательность обхода заданной и ищем координаты точек врезки в каждый контур , минимизирующих суммарную длину холостого хода. Для этого выбираются начальные положения точек (например, случайным образом) и положение каждой точки изменяется в предположении, что все остальные неподвижны: . Большая часть слагаемых в целевой функции константны, поэтому она упрощается до

Простейший геометрический анализ показывает, что если точки расположены по разные стороны сегмента контура , то оптимальная позиция точки врезки – на пересечении с отрезком: . В случае же, если точки расположены по одну сторону сегмента, то решение также легко находится с применением принципа Ферма, порождающем знаменитое правило «угол падения равен углу отражения» (см. Рис. 2).

Общая схема оптимизации на данном этапе, таким образом выглядит так:

* Выбираем произвольные начальные положения точек врезки .
* находим оптимальную позицию как описано выше за константное время .
* Повторяем предыдущий шаг, пока позиции всех точек не сойдутся (с некоторой заранее заданной точностью .

На практике весь процесс хорошо сходится за время и поэтому многократно используется как подпрограмма на следующем шаге.

## Дискретная оптимизация

Самый вычислительно сложный этап заключается в поиске перестановки , минимизирующей длину холостого хода , то есть фактически решение задачи коммивояжера (Travelling Salesman Problem, TSP) с функцией расстояния, вычисляемой при помощи непрерывной оптимизации, как описано на предыдущем шаге.

Для поиска решения используется эвристический метод переменных окрестностей (Variable Neighborhood Search, VNS). Схема его применения:

1. Начальная перестановка выбирается случайным образом
2. Пока
   1. Из окрестности  выбирается перестановка , минимизирующая
   2. Если
   3. Иначе
3. Конец

На шаге 3.1 многократно применяется шаг непрерывной оптимизации:

Для построения окрестностей разного размера применяются различные приёмы, например:

* Все возможные парные перестановки (фактически это окрестность размера 2 в смысле расстояния Левенштейна)
* Циклические перестановки 3 контуров. Все такие перестановки потребовали бы времени , поэтому выбираются только такие варианты, в которых переставляемые контуры удалены в исходной перестановке не более, чем на заранее заданное расстояние (параметр алгоритма).
* Аналогичным образом применяются циклические перестановки 4 контуров, находящихся в пределах некоторого расстояния друг от друга в исходной перестановке .
* Выбирается последовательный блок контуров произвольной длины и осуществляется их циклическая перестановка.
* Перестановка контуров в последовательном блоке произвольной длины «задом наперёд»
* Обмен двух последовательных (но не соседних) блоков контуров
* Циклический сдвиг контуров между несколькими последовательными блоками одинаковой длины
* И ещё порядка десяти различных рецептов построения «близких» перестановок

Если размер окрестности, генерируемой некоторым приёмом, оказывается слишком велик, он может быть легко ограничен введением дополнительного параметра, аналогично тому, как это сделано для тройных и четверных перестановок.

Кроме того, метод переменных окрестностей имеет некоторые вариации, уменьшающие перебор на шаге 3.1, такие как «Первый подходящий» или метод Монте-Карло, однако их влияние на качество и скорость решения задачи непрерывной резки нуждается в дальнейшем исследовании.

## Восстановление удалённых контуров

После того как маршрут инструмента для обхода контуров, не содержащих внутри себя других, построен, как описано выше, достроим его до полного маршрута, посещающего все исходные контуры, причём так, чтобы соблюдалось ограничение предшествования.

Обратим внимание, что полученный на предыдущем шаге маршрут

пересекает все исходные контуры , потому что оставленные на первом этапе контуры он посещает по построению на шагах 2 и 3, а все внешние контура он пересекает согласно теореме Жордана, потому что начальная и конечная точки и лежат снаружи всех контуров .

Таким образом для каждого (внешнего) контура , ещё не включённого в маршрут, мы находим все точки пересечения с ним маршрута, построенного на предыдущем шаге и если таких точек оказывается несколько (как правило), то выбираем из них самую последнюю (посещаемую маршрутом позже всех остальных), см. Рис. 3. После добавления полученных таким образом точек врезки мы получаем уже полный маршрут, который, с одной стороны, посещает все исходные контуры, и при этом внешние контуры всегда посещаются позже, чем содержащиеся в них внутренние. Длина маршрута, очевидно, при этом не изменяется.

Mi

Mi-1

Mi-2

Mi+1

Mi+2

Внешний контур

Добавочная точка врезки

Рис 3. Построение дополнительных точек врезки

Легко понять, что свойство оптимальности маршрута при этом также не нарушается. Действительно, если бы существовал более короткий маршрут, обходящий все контуры, мы могли бы простым удалением из него точек врезки, лежащих на внешних контурах получить маршрут обхода только внутренних контуров, имеющий ту же, то есть меньшую длину. Тем самым мы бы построили более короткий маршрут для подзадачи без ограничений предшествования, что по предположению невозможно.

Таким образом, мы выполняем ограничение предшествования почти автоматически за линейное время .

# Оптимальность решения задачи непрерывной оптимизации

# Заключение

# Библиографический список