АВТОМАТИЗИРОВАННОЕ РЕШЕНИЕ ЗАДАЧИ ОБ ОПТИМАЛЬНОЙ РАССТАНОВКЕ ОБОРУДОВАНИЯ НА ПОТОЧНОЙ ЛИНИИ

А. А. Дядюшкин

Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Беларусь

Научный руководитель В. С. Мурашко

Цель данной работы автоматизировать решение задачи об оптимальной расстановки оборудования на поточной линии.

Расстановка оборудования влияет на размеры транспортных расходов, на себестоимость продукции, на капитальные вложения, на степень прямоточности, непрерывности и ритмичности производства, на уровень организации труда. В то же время система учета не позволяет выявить влияние планировки оборудования на экономику производства, представить взаимосвязи между различными параметрами в едином показателе – критерии оптимальности планировки. Среди всех параметров, характеризующих расстановку оборудования, наиболее общим и простым является грузооборот линии. Грузооборот линии влияет на размеры текущих эксплуатационных затрат (расходы на содержание, ремонт и эксплуатацию транспортных средств), на размеры внутри участкового незавершенного производства, на общую массу конвейера, следовательно, и стоимость конвейерного оснащения.

Если на линии обрабатываются предметы с разнонаправленными технологическими маршрутами их движения, размещение рабочих мест является многовариантной задачей. В общем случае число всех возможных вариантов расстановки рабочих мест, если количество единиц оборудования равно q, определяется числом возможных перестановок $q!$.

В данной работе рассматривается задача минимизация грузооборота оборудования на поточной линии, которая в общем виде формулируется следующем образом.

Пусть N_i – программа выпуска по i-му наименованию предмета за планируемый период времени; g_i – масса единицы i-го наименования предмета; $u_{i,j}$ – порядковый номер очередности обработки партии предметов i-го наименования на j-м станке ($i = 1, ..., m$; $j = 1, ..., q$), где q – число площадок, которые выделяются для размещения на них q станков. Каждый вариант размещения рабочих мест по площадкам указывает номер рабочего места, которое закрепляется за каждой площадкой, и его можно рассматривать как перестановку чисел 1, 2, ... Множество допустимых перестановок обозначим через π. В процессе обработки i-й предмет проходит в зависимости от варианта планировки π путь, длину которого обозначим через $l_i(\pi)$.

Требуется разместить рабочие места по одному на площадке так, чтобы обеспечить минимальный объем грузооборота.

Сформулированную задачу сведем к задаче о «назначениях».

Пусть $c_{a,b}$ – расстояние между a-й и b-й площадками; $g_{c,d}$ – масса продукции, идущей непосредственно от c-го к d-му рабочему месту и от d-го к c-му рабочему месту. Занумеруем числа $g_{c,d}$ в последовательность $G = (g_1, ..., g_v, ..., g_k)$, а числа $c_{a,b}$ – в последовательность $C = (c_1, ..., c_z, ..., c_k)$. Положим $g_{v,z} = g_v c_z$. Тогда необходимо решить следующую задачу. Найти минимум

$$f(x) = \sum_{v=1}^{k} \sum_{z=1}^{k} g_{v,z} x_{v,z} \tag{1}$$

при ограничениях

$$\sum_{v=1}^{k} x_{v,z} = 1, \quad \sum_{z=1}^{k} x_{v,z} = 1, \quad x_{v,z} = 0,1. \tag{2}$$

Если $x_{v,z} = x_{(c,d)(a,b)} = 1$, то при $c \neq s \neq d$, $c \neq r \neq d$, $a \neq p \neq b$, $a \neq n \neq b$ имеют место равенства:

$$x_{(c,s)(p,n)} = 0, \quad x_{(s,c)(p,n)} = 0, \quad x_{(d,s)(p,n)} = 0, \quad x_{(s,d)(p,n)} = 0, \quad x_{(s,r)(p,a)} = 0,$$

$$x_{(s,r)(a,p)} = 0, \quad x_{(s,r)(b,n)} = 0, \quad x_{(s,r)(n,b)} = 0. \qquad (3)$$

Одним из подходов к решению сформулированной задачи предлагается использовать метод возвратной рекурсии (соединение рекурсии с методом перебора с возвратом).

Решение задачи методом перебора с возвратом строится конструктивно последовательным расширением частичного решения. Если на конкретном шаге такое расширение провести не удается, то происходит возврат к более короткому частичному решению, и попытки его расширить продолжаются. Для ускорения перебора с возвратом вычисления всегда стараются организовать так, чтобы была возможность отметать как можно раньше и как можно больше заведомо неподходящих вариантов M.

При использовании возвратной рекурсии отпадает необходимость непосредственно организовывать возвраты и отслеживать правильность их осуществления. Они, как правило, становятся встроенной частью механизма выполнения рекурсивных вызовов.

В общем случае этот метод приводит к алгоритмам с экспоненциальной временной сложностью, а применяется он в основном к классу так называемых *Np*-полных задач (задача о назначениях, задача коммивояжера, задача о рюкзаке и т. д.). Задачи этого класса эквивалентны друг другу в том смысле, что все они разрешимы недетерминированными алгоритмами полиномиальной сложности. Далее, для них известно, что либо все они разрешимы, либо ни одна из них не разрешима детерминированными алгоритмами полиномиальной сложности. Иными словами, если хотя бы для одной из этих задач не существует детерминированного алгоритма, имеющего в худшем случае полиномиальную трудоемкость, то такие алгоритмы не должны существовать и для остальных задач этого класса. Наоборот, если хотя бы для одной из этих задач удалось найти детерминированный алгоритм, имеющий в худшем случае полиномиальную трудоемкость, то подобные алгоритмы существовали бы и для остальных задач этого класса и, более того, их можно было бы построить.

Алгоритм решения задачи (1)–(3) состоит из следующих шагов:

Шаг 1. Ввод исходных данных: количество рабочих мест n, масса готовой продукции M_i ($i = 1, 2, ..., n$); очередность обработки на n рабочих местах $G_{i,j}$ ($i = 1, 2, ..., n$; $j = 1, 2, ..., n$) и расстояние между площадками $L_{i,j}$ ($i = 1, 2, ..., n$; $j = 1, 2, ..., n$), на которое нужно установить рабочие места.

Шаг 2. Предварительно строится матрица $U1$, значениями которой будет масса предметов, которые после c-го станка должны обрабатываться на d-м станке.

На этом шаге выполняются следующие действия.

1. Вычислить размерность матрицы грузооборота U: $m = \dfrac{n!}{2!(n-2)!}$.

2. Найти все возможные варианты перехода между рабочими местами $P_{i,j}$ ($i = 1, 2, ..., m$; $j = 1, 2$) и площадками $P1_{i,j}$ ($i = 1, 2, ..., m$; $j = 1, 2$).

3. Вычислить значения элементов матрицы $U1$.

Шаг 3. Упорядочить по убыванию элементы матриц $U1$ и P, по возрастанию – L и $P1$. В результате получаются матрицы $U1\_1$ и $L1\_1$ размерности $m \times 3$.

Шаг 4. Построить матрицу грузооборота U размерностью $m \times m$.

Шаг 5. Решить задачу (1)–(3) методом возвратной рекурсии, находящей одно из оптимальных решений задачи о назначениях при $n1$ ($n1 = m$) вариантов переходов между рабочими местами, m вариантов переходов между площадками и матрице грузооборота U.

На рис. 1 представлена опорная схема для описания рекурсии в задаче о назначениях.

Решением задачи могут служить функция *assi*($n1$, m, ne, ot, j, $U1\_1$, $L1\_1$) с рекурсией по номерам вариантов перехода между рабочими местами, функция *assi*2($U1\_1$, $L1\_1$, ne) и программа-константа *assi*.

Решение задачи возвращается в виде вектора $ot = (ot_0, ot_1,..., ot_{n1-1}, ot_{n1})^{\mathrm{T}}$, где ot_j – номер варианта перехода между площадками для варианта перехода между рабочими местами с номером j ($j = 0, 1, \ldots, n1 - 1$), а ot_{n1} – грузооборот от данного оптимального назначения.

Рабочие места (0,1, …n1 – 1) **Площадки** (0,1, …m – 1)

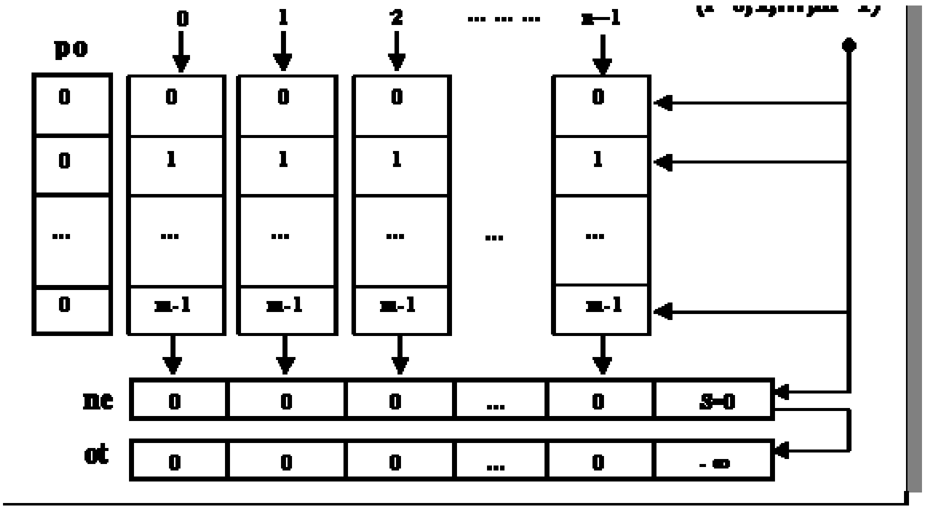

Рис. 1. Опорная схема для описания рекурсии в задаче о назначениях

Параметры функции *assi*($n1$, m, ne, ot, j, $U1\_1$, $L1\_1$) имеют следующий смысл: $ne = (ne_0, ne_1,..., ne_{n-1}, ne_n)^{\mathrm{T}}$ – вектор, в котором последовательно формируется очередное назначение: ne_j – номер варианта перехода между площадками с вариантом перехода между рабочими местами с номером j ($j = 0, 1, \ldots, n - 1$), ne_n – грузооборот от данного назначения; po – вектор меток. Если вариант перехода между площадками i в j-м рекурсивном вызове назначается варианту перехода между рабочими местами с номером j, то в векторе меток этот факт фиксируется следующим образом: $po_i = 1$. При переходе к предыдущему рекурсивному уровню или завершению построения одного из возможных назначений последний из рассмотренных вариантов перехода между площадками высвобождается: $po_i = 0$; ot – вектор, в котором при $ot_n < ne_n$ вызывается функция *assi*2($U1\_1$, $L1\_1$, ne), назначение которой проверить решение ne на выполнимость условий (3), если результат положителен, то запоминается очередное найденное назначение и соответствующий ему грузооборот: $ot = ne$; j – уровень рекурсивного вызова

Предложенный алгоритм реализован в MathCad. Разработанную программу могут использовать как студенты в учебном процессе, так и на производстве.