УДК 004.453 **DOI** 10.15622/ia.2022.21.1

К.В. Кротов

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ И АЛГОРИТМ МЕТОДА ВЕТВЕЙ И ГРАНИЦ ДЛЯ ОПТИМИЗАЦИИ РЕШЕНИЙ ПО СОСТАВАМ ПАКЕТОВ В МНОГОСТАДИЙНЫХ СИСТЕМАХ

Кротов К.В. **Математическая модель и алгоритм метода ветвей и границ для оптимизации решений по составам пакетов в многостадийных системах.**

Аннотация. Современные методы решения задач планирования выполнения пакетов заданий в многостадийных системах характеризуются наличием ограничений на их размерность, невозможностью гарантированного получения лучших результатов в сравнении с фиксированными пакетами при различных значениях входных параметров задачи. В статье автором решена задача оптимизации составов пакетов заданий, выполняющихся в многостадийных системах, с использованием метода ветвей и границ. Проведены исследования различных способов формирования порядков выполнения пакетов заданий в многостадийных системах (эвристических правил упорядочивания пакетов заданий в последовательностях их выполнения на приборах МС). Определен способ упорядочивания пакетов в последовательностях их выполнения (эвристическое правило), обеспечивающий минимизацию общего времени реализации действий с ними на приборах. На основе полученного правила сформулирован способ упорядочивания типов заданий, в соответствии с которым их пакеты рассматриваются в процедуре метода ветвей и границ. Построена математическая модель процесса реализации действий с пакетами на приборах системы, которая обеспечивает вычисление его параметров. Выполнено построение метода формирования всех возможных решений по составам пакетов заданий для заданного их количества. Решения по составам пакетов заданий разных типов интерпретируются в процедуре метода ветвей и границ с целью построения оптимальной их комбинации. Для реализации метода ветвей и границ сформулирована процедура ветвления (разбиения), предполагающая формирование подмножеств решений, включающих пакеты разных составов заданий одного типа. Построены выражения для вычисления нижних и верхних оценок значений критерия оптимизации составов пакетов для сформированных в процедуре ветвления подмножеств. Процедура отсева предполагает исключение подмножеств, нижняя оценка которых не меньше рекорда. Для поиска оптимальных решений применена стратегия поиска в ширину, предусматривающая исследование всех подмножеств решений, включающих различные пакеты заданий одного типа, полученных в результате процедуры разбиения подмножеств заданий, не исключенных из рассмотрения после реализации процедуры отсева. Разработанные алгоритмы реализованы программно, что позволило получить результаты планирования выполнения пакетов заданий в многостадийной системе, являющиеся в среднем на 30 % лучшими, чем для фиксированных пакетов.

Ключевые слова: многостадийная система, пакеты заданий, метод ветвей и границ, эвристическое правило, расписания

1. Введение. Выполнение заданий разных типов, входящих в их наборы, в многостадийных системах (МС) предполагает формирование пакетов, количество заданий в которых определяется при оптимизации решений по их составам. Выполнение в МС заданий разных типов, включенных в составы пакетов, связано с переналадкой приборов МС с выполнения заданий одного типа на выполнение заданий другого типа. Необходимость оптимизации составов пакетов вызвана неоднородностями длительностей выполнения заданий на приборах МС, а также неоднородностями длительностей переналадок приборов на выполнение заданий. Пакет – это совокупность заданий одного типа, выполняемых без переналадки приборов МС на выполнение заданий другого типа. Если в пакет включены все задания одного типа из их набора, такой пакет является фиксированным. Так как в МС выполняются ПЗ разных типов, тогда необходимо сформировать комплексные решения по составам ПЗ этих типов с учетом временных характеристик процесса реализации действий с ними в МС. Под оптимальным комплексным решением подразумевается комбинация составов ПЗ разных типов, обеспечивающая экстремум принятого показателя эффективности.

Современные методы планирования выполнения ПЗ в МС предполагают применение: целочисленного программирования (ЦЧП) для определения расписаний выполнения ПЗ [1]; метаэвристических алгоритмов (генетических алгоритмов (ГА), метода муравьиной колонии (МК)) при определении составов ПЗ и расписаний их выполнения [2-5]; эвристических правил при формировании составов ПЗ [6-8]; имитационного моделирования процесса обработки ПЗ в непрерывном производстве для оценки эффективности применения эвристик [9]. Математическая модель оптимизации решений при планировании выполнения ПЗ на параллельно функционирующих устройствах рассмотрена в [1]. Оптимизация решений с ее использованием предполагает определение значений переменных, соответствующих как составам пакетов, так и порядкам их выполнения на приборах МС. Применение данного подхода ограничено размерностью рассматриваемых задач. Применение ГА для решения задачи планирования ([2]) предполагает, что составы ПЗ определяются спросом на продукцию, построение расписания их выполнения реализуется с использованием этих алгоритмов. В [3,4] ГА применены для определения составов ПЗ (формируемые хромосомы соответствуют решениям по составам ПЗ). В [5] рассматривается метод планирования выполнения ПЗ на параллельных приборах на основе локального поиска совместно с методом МК при использовании эвристических правил. В силу стохастического характера рассмотренных алгоритмов при разных значениях входных параметров задачи они не гарантируют получение решений, приближающихся к оптимальным. В [6] эвристическое правило предусматривает включение в пакет заданий, длительности выполнения которых связаны с длительностью выполнения первого задания в этом пакете определенным образом. В [7] эвристические правила предусматривают формирование множеств отклоняемых заданий и для каждого множества не отклоненных заданий формирование пакетов. В [8] метод формирования пакетов основывается на директивных сроках окончания выполнения заданий. В [9] определение составов ПЗ реализуется на основе имитационного моделирования. Определение маршрутов движения ПЗ, расписаний их выполнения реализуется с использованием правил, а оценивается с использованием имитационной модели. Отечественные авторы (Танаев В.С., Сотсков Ю.Н., Ковалев М.Я., Лазарев А.А., Кобак В.Г., Нейдорф Р.А.) рассматривают методы построения расписаний выполнения единичных заданий (ЕЗ), не объединяемых в пакеты. В [10] рассматриваются эвристические правила определения составов пакетов и методы построения расписаний выполнения фиксированных пакетов.

Метод ветвей и границ (МВГ) [11] является одним из базовых методов дискретной оптимизации решений. Однако его применение при планировании связано лишь с решением задач построения расписаний выполнения ЕЗ в обрабатывающих системах различного вида [12-20]. Работы [12,13] посвящены применению МВГ при построении расписаний выполнения ЕЗ на одном приборе (в [12] – с учетом директивных сроков окончания выполнения ЕЗ, в [13] – с учетом формирования партий, отгружаемых клиентам в соответствии со спросом на продукцию). В [14-15] рассматривается применение МВГ при построении расписаний выполнения ЕЗ в МС (без объединения их в пакеты). В [14] вводятся выражения для нижних оценок значений критерия оптимизации при формировании частичных последовательностей выполнения ЕЗ в МС, верхние оценки определяются на основе допустимых решений, получаемых с использованием метаэвристических алгоритмов. В [15] выражения для определения нижних и верхних оценок значений критерия формируются с учетом особенностей задачи: верхние оценки – с учетом простоев приборов, нижние – с учетом времени блокировки. В [16,17] МВГ применен при формировании расписаний выполнения ЕЗ на параллельных приборах. На каждом шаге процедуры ветвления МВГ в [16,17] формируются новые частичные последовательности выполнения ЕЗ на разных параллельно функционирующих приборах, соответствующие вершинам дерева. В [17] при формировании частичных расписаний учитывается отношение порядка для выполняемых заданий. Работы [18-20] посвящены использованию МВГ для решения задач построения расписаний выполнения проектов при ограничениях на ресурсы (RCPSP-задачи).

Определение составов ПЗ связано с решением задач ЦЧП, являющихся ограниченными по размерности, с применением метаэвристических алгоритмов и эвристических правил, приводящих к решениям, не гарантирующим приближения к оптимальным. Использование МВГ связано с построением расписаний выполнения ЕЗ на одном приборе, в МС, на параллельных приборах, построением расписаний выполнения проектов. Требуется развить МВГ для решения задач оптимизации составов ПЗ и построения расписаний их выполнения в МС.

2. Математическая модель процесса выполнения ПЗ в МС. Для построения математической модели процесса выполнения ПЗ в МС, вида решений по составам ПЗ, расписаний выполнении ПЗ в МС в рассмотрение введены обозначения, представленные в Таблице 1.

Таблица 1. Обозначения параметров процесса выполнения ПЗ в МС

Обозначение	Назначение параметра
1	2
i	Идентификатор типа заданий ($i=\overline{1,n}$).
n^i	Количество заданий *i*-го типа ($i=\overline{1,n}$).
l	Номер прибора ($l=\overline{1,L}$).
m_i	Количество ПЗ *i*-го типа ($i=\overline{1,n}$) в решении по их составам.
M	Вектор количества пакетов *i*-ых типов ($i=\overline{1,n}$).
a_{ih}	Количество заданий *i*-го типа в *h*-ом пакете $(h=\overline{1,m_i})$.
A	Матрица составов ПЗ (размерность матрицы $n\times m$, где $m=\max\limits_{i=\overline{1,n}}(m_i)$).
π^l	Последовательность выполнения ПЗ на приборах МС.
n_p	Количество ПЗ в последовательностях π^l ($n_p=\sum\limits_{i=1}^{n} m_i$).
j	Номер позиции ПЗ в последовательностях π^l их выполнения на приборах МС.
P	Матрица порядка выполнения ПЗ в π^l (размерность матрицы $n\times n_p$).
R	Матрица количества заданий *i*-ых типов в пакетах, занимающих в π^l *j*-е позиции (размерность $n\times n_p$).
t_{li}	Длительность выполнения задания *i*-го типа на *l*-ом приборе МС ($i=\overline{1,n}$, $l=\overline{1,L}$).
T	Матрица длительностей выполнения заданий *i*-ых типов на приборах МС (размерность матрицы $L\times n$).

Продолжение Таблицы 1

1	2
t_{ij}^{l}	Длительность переналадки *l*-го прибора с выполнения заданий *i*-го типа на выполнение заданий *j*-го типа.
T^{l}	Матрицы переналадки *l*-ых приборов ($l=\overline{1,L}$) (размерность каждой из матриц $n \times n$).
t_{ij}^{nl}	Моменты времени начала реализации действий с ПЗ *i*-го типа в *j*-й позиции в последовательности π^{l} ($l=\overline{1,L}$) их выполнения на приборах МС.
T^{nl}	Матрицы моментов времени начала выполнения ПЗ *i*-ых типов в *j*-ых позициях в последовательностях π^{l} ($l=\overline{1,L}$) (размерность матриц $n \times n_p$).
Pr_{ij}^{l}	Простои *l*-го прибора в ожидании готовности к выполнению ПЗ *i*-го типа, в *j*-ой позиции в последовательности π^{l} ($l=\overline{1,L}$) их выполнения на приборах МС.
N_1, N_2	Множества решений *[M,A]* по составам ПЗ и расписаниям $[P,R,\{T^{nl} \mid l=\overline{1,L}\}]$ выполнения ПЗ в МС соответственно.

Порядок выполнения ПЗ на приборах МС одинаков, поэтому формируется одна матрица *P* и одна матрица *R*. Тогда расписание выполнения ПЗ на приборах МС имеет вид: $[P,R,\{T^{nl} \mid l=\overline{1,L}\}]$. В силу того, что расписание выполнения ПЗ на приборах МС формируется на основе решения *[M,A]* путем интерпретации эвристического правила, тогда оно является результатом отображения множества решений по составам ПЗ вида *[M,A]* на множество расписаний выполнения ПЗ, то есть $N_1 \to N_2([M,A])$.

Для формирования информационной модели задачи *z* планирования выполнения ПЗ каждому *i*-му типу заданий поставлен в соответствие кортеж вида [21]:

$$x_i = < i, T_i, \{T_i^{l} \mid l=\overline{1,L}\} >, \tag{1}$$

где $T_i=(t_{1i},t_{2i},..,t_{Li})^{T}$ – *i*-й вектор-столбец длительностей выполнения заданий *i*-го типа на *l*-ых приборах МС (*i*-й столбец ($i=\overline{1,n}$) матрицы *T* длительностей выполнения заданий *i*-х типов на *l*-ых приборах МС), $T_i^{l}=(t^{\,l}_{\,i,1},t^{\,l}_{\,i,2},..,t^{\,l}_{\,i,n})$ – *i*-я вектор-строка матрицы T^{l} ($l=\overline{1,L}$) длительностей переналадки *l*-го прибора с выполнения заданий *i*-го типа на выполнение заданий *i'*-го типа ($t^{\,l}_{i,i}$ соответствует времени первоначальной наладки *l*-го прибора на выполнение заданий *i*-го типа). Технологические маршруты (одинаковый фиксированный порядок приборов), дисциплины обслуживания (последовательное прохождение заданиями всех приборов в соответствии с маршрутами), правила выполнения для всех типов заданий одинаковы, поэтому эти параметры не вынесены в описание (1).

Так как в качестве входных параметров для задачи *z* заданы значения $n^i > 1$ ($i = \overline{1,n}$), то каждый тип заданий характеризуется кортежем вида $< x_i, n^i >$ ($i = \overline{1,n}$). Тогда входными параметрами, на основе которых выполняется решение задачи *z*, является множество кортежей вида $< x_i, n^i >$. Моменты времени поступления всех заданий в систему являются одинаковыми, равными 0, директивные сроки выполнения заданий не заданы, приоритеты выполнения заданий одинаковы, то эти параметры не вынесены в описание входных данных задачи. Множество входных данных обозначим через *X*. Вид множества X следующий: $X = \{< x_i, n^i > | i = \overline{1,n}\}$. Через *Y* обозначим множество результатов, формируемых системой планирования выполнения заданий при решении задачи z. В силу того, что ограничения на используемые ресурсы не заданы, тогда задача *z*, решение которой выполняется в системе планирования, характеризуется видом входной информации, видом критерия Kr, используемого при выборе лучшего из множества допустимых решений. Тогда решаемой в системе задаче *z* в соответствие поставлен кортеж параметров вида $z =< X, Y, Kr >$.

В соответствие с [22] задача планирования выполнения ПЗ в МС представлена в виде: $\alpha \,|\, \beta \,|\, \gamma$, где α – тип обрабатывающей системы, β – тип решаемой задачи, γ – вид критерия оптимизации решений. Так как рассматривается многостадийная система– система *FlowShop*, то в качестве параметра α указывается *F*. Решение задачи связано с комплексным планированием выполнения ПЗ в МС (предусматривающим определение составов ПЗ всех типов и расписаний их выполнения в МС). Тогда в качестве параметра β указывается *batch* [10, 22]. Через n_p обозначено количество ПЗ в последовательностях π^l ($l = \overline{1,L}$) их выполнения на приборах МС ($n_p = \sum_{i=1}^{n} m_i$). В этом случае через t^{nL}_{i,n_p} обозначен момент времени начала выполнения последнего

n_p -го ПЗ некоторого i-го типа (занимающего последнюю n_p –ю позицию в последовательности π^L выполнения ПЗ на L-ом (последнем) приборе МС). Выражение $t^{nL}_{i,n_p} + r_{i,n_p} t_{L,i}$ определяет момент времени окончания выполнения этого ПЗ, занимающего n_p -ю позицию в последовательности π^L на L-ом приборе (то есть момент времени окончания выполнения всех ПЗ в МС).

Сформулированное эвристическое правило упорядочивания ПЗ в последовательностях π^l ($l = \overline{1, L}$) с учетом длительностей переналадок приборов является способом отображения множества решений N_1 на множество решений N_2: $N_1 \to N_2([M, A])$. Тогда сформированное с использованием правила расписание $[P, R, \{T^{nl} \mid l = \overline{1, L}\}]$ не оптимизируется и соответствует решению по составам ПЗ *[M,A]*. Значение момента времени окончания выполнения ПЗ на приборах МС, определяемое выражением $t^{nL}_{i,n_p} + r_{i,n_p} t_{L,i}$, характеризует решение по составам ПЗ вида *[M,A]*. По этой причине оно используется в качестве критерия оптимизации решений по составам ПЗ (предусматривающих, в том числе, и построение расписаний выполнения ПЗ в МС). Тогда решаемая задача планирования имеет вид: $F \mid batch \mid (t^{nL}_{i,n_p} + r_{i,n_p} t_{L,i})$.

Входными данными задачи планирования являются: количество типов заданий n, количество заданий n^i ($i = \overline{1, n}$) каждого типа, матрицы длительностей выполнения заданий на приборах МС T и длительностей переналадок приборов МС T^l ($l = \overline{1, L}$). Результатами планирования является оптимальное решение по количествам и составам ПЗ разных типов вида *[M,A]* и соответствующее ему расписание выполнения ПЗ в МС вида $[P, R, \{T^{nl} \mid l = \overline{1, L}\}]$, сформированное с использованием эвристического правила.

Задача оптимизации составов ПЗ решается в предположении, что длительности переналадок l-ых приборов ($l = \overline{1, L}$) с выполнения заданий i-го типа на выполнение заданий i'-го типа являются одинаковыми ($t^1_{ii'} = t^2_{ii'} = .. = t^L_{ii'}$), длительности выполнения заданий i-х типов ($i = \overline{1, n}$) на l-ых приборах также одинаковы ($t_{1i} = t_{2i} = .. = t_{Li}$). Значения длительностей переналадок приборов с выполнения заданий i-го типа на выполнение заданий i'-го типа и длительностей выполнения заданий на приборах МС являются не убывающими. В соответствии с введенными предположениями, связанными со значениями параметров $t^l_{ii'}$ и t_{li} ($l=\overline{1,L}$), рассмотрены следующие эвристические правила, с использованием которых реализуется формирование порядка выполнения ПЗ в π^l ($l=\overline{1,L}$) на приборах МС:

– пакеты заданий с минимальной длительностью выполнения t_{li} – в первую очередь (правило R1);

– пакеты заданий с максимальной длительностью выполнения t_{li} – в первую очередь (правило R2);

– пакеты заданий с минимальной длительностью переналадки приборов $t^l_{ii'}$ – в первую очередь (правило R3);

– пакеты заданий с максимальной длительностью переналадки приборов $t^l_{ii'}$ – в первую очередь (правило R4).

Исследования указанных способов упорядочивания ПЗ в последовательностях π^l ($l=\overline{1,L}$) осуществлены путем построения диаграмм Ганта для каждого правила (при разных составах ПЗ). Для каждого эвристического правила и соответствующих ему последовательностей π^l ($l=\overline{1,L}$) выполнения ПЗ зафиксированы моменты времени окончания реализации действий с ними в МС. Анализ рассмотренных способов упорядочивания пакетов (с точки зрения минимизации окончания выполнения ПЗ в МС) показал, что использование параметра $t^l_{ii'}$ и правила R3 для построения расписаний позволяет получить минимальную длительность выполнения ПЗ. Реализация правила R3 упорядочивания ПЗ в последовательностях π^l ($l=\overline{1,L}$) предусматривает, что: 1) в первой позиции в последовательностях π^l ($l=\overline{1,L}$) размещается ПЗ с минимальным значением $t^l_{ii'}$; 2) во второй позиции в последовательностях π^l размещается ПЗ со значением $t^l_{i'i''}$, следующим за значением $t^l_{ii'}$ для заданий в пакете в первой позиции в π^l ($l=\overline{1,L}$); 3) в следующих позициях в π^l последовательно размещаются по одному пакету заданий других типов, длительности $t^l_{ii'}$ переналадки приборов с выполнения которых упорядочены по не убыванию (до *n*-го типа включительно); 4) в π^l размещаются по одному пакету заданий разных типов (от (*i=1*)-го до (*i=n*)-го типов), длительности $t^l_{ii'}$ переналадок приборов МС с выполнения которых упорядочены по не убыванию; 5) действия, аналогичные пункту 4, продолжаются до тех пор, пока в π^l ($l = \overline{1, L}$) не будут размещены все пакеты.

На основе сформулированного эвристического правила формирования порядка выполнения ПЗ в МС реализовано упорядочивание типов заданий с учетом значений $t_{ii'}^l$. В соответствии с этим порядком типов заданий решения по составам их пакетов будут включаться с множества (подмножества) решений, формируемых при реализации процедуры разбиения (ветвления) в МВГ. В дополнение к обозначениям Таблицы 1 введены следующие обозначения: I – множество типов заданий, выполняемых в МС, упорядоченных с учетом значений $t_{ii'}^l$ ($|I| = n$); $f(t_{ii'}^l)$ – способ упорядочивания элементов из I (связывания элементов из I отношением порядка $\succ$) в соответствии со значениями параметра $t_{ii'}^l$. Способ $f(t_{ii'}^l)$ предусматривает при $t_{ii'}^l < t_{i'i''}^l$ связывание типов заданий i, i', i'', входящих в I, отношением $\succ$ следующим образом: $i' \succ i$, $i'' \succ i'$. Тогда ($i=1$)-ому типу заданий соответствует минимальное значение $t_{ii'}^l$, n-му типу – максимальное $t_{ii'}^l$.

Для построения математической модели процесса выполнения ПЗ в МС в рассмотрение введен параметр Pr_{ij}^l, характеризующий простои l-го прибора в ожидании начала выполнения ПЗ i-го типа, занимающего j-ю позицию в π^l ($i = \overline{1, n}$, $j = \overline{1, n_p}$). Простои ($l=1$)-го прибора в ожидании готовности ПЗ к выполнению отсутствуют. Данное утверждение представляется в следующем виде:

$$\mathrm{Pr}_{11}^1 = \mathrm{Pr}_{1,(n+1)}^1 = .. = 0\,,\ \mathrm{Pr}_{22}^1 = \mathrm{Pr}_{2,(n+2)}^1 = .. = 0\,,..,\ \mathrm{Pr}_{n,n}^1 = \mathrm{Pr}_{n,2n}^1 = .. = 0\,. \quad (2)$$

Значения Pr_{ij}^l определяются на основе сформированной матрицы R следующим образом:

$$\mathrm{Pr}_{i,j}^l = \max(0, (t_{i,j}^{nl-1} + r_{ij} t_{l-1,i}) - (t_{i',j-1}^{nl} + r_{i',j-1} t_{l,i'} + t_{i',i}^l)\,. \quad (3)$$

В (3) $t_{i,j}^{nl-1}$ – момент времени начала выполнения на *(l-1)*-м приборе ПЗ i-го типа, занимающего j-ю позицию в последовательности π^{l-1}, $t_{i',j-1}^{nl}$ – момент времени начала выполнения ПЗ i'-го типа, занимающего *(j-1)*-ю позицию в π^l. В соответствии с (2), (3) выражения для определения значений $t_{i,j}^{nl}$ имеют следующий вид:

– при $l=1$:

$$t_{11}^{n1}=0\,;t_{22}^{n1}=t_{11}^{n1}+r_{11}t_{11}+t_{12}^{1}\,;..;t_{ij}^{n1}=t_{i-1,j-1}^{n1}+r_{i-1,j-1}t_{1,i-1}+t_{i-1,i}^{1}$$
$$t_{nn}^{n1}=t_{n-1,n-1}^{n1}+r_{n-1,n-1}t_{1,n-1}+t_{n-1,n}^{1}\,; \tag{4}$$
$$t_{1,n+1}^{n1}=t_{n,n}^{n1}+r_{n,n}t_{1,n}+t_{n,1}^{1}\,;..;t_{n,2n}^{n1}=t_{n-1,2n-1}^{n1}+r_{n-1,2n-1}t_{1,n-1}+t_{n-1,n}^{1}\,;.$$

– при *l>1*:

$$t_{11}^{nl}=\sum_{g=1}^{l-1}r_{11}t_{g1}\,;$$
$$t_{i,j}^{nl}=t_{i',j-1}^{nl}+r_{i',j-1}t_{l,i'}+t_{i',i}^{l}+\mathrm{Pr}_{i.j}^{l}\,(\text{при } j>2). \tag{5}$$

Выбор решения по составам ПЗ i-х типов ($i=\overline{1,n}$) осуществляется в соответствии с требованием минимизации окончания их выполнения в МС. Расписание $[P,R,\{T^{nl}\mid l=\overline{1,L}\}]$ является результатом отображения $N_1\to N_2([M,A])$, соответствующего эвристическому правилу (матрицы P и R определяются в соответствии с эвристическим правилом на основе решения *[M,A]*, расписание не оптимизируется). Тогда решение *[M,A]* может быть охарактеризовано моментом времени окончания выполнения ПЗ в МС. Выражение $t_{i,n_p}^{nL}+r_{i,n_p}t_{L,i}$ (где значение n_p ($n_p=\sum_{i=1}^{n}m_i$) и значение r_{i,n_p} (являющееся результатом отображения $f(t_{ii'}^{l})$) определяются в соответствии в решением *[M,A]*) введенное выше, определяет момент времени окончания выполнения в МС всех ПЗ. Тогда задача оптимизации составов ПЗ, выполняемых в МС, имеет следующий вид:

$$\min\ f([M,A],\ [P,R,\{T^{nl}\mid l=\overline{1,L}\}])\,, \tag{6}$$

где $f([M,A],\ [P,R,\{T^{nl}\mid l=\overline{1,L}\}])=t_{i,n_p}^{nL}+r_{i,n_p}t_{L,i}$.

Ограничения на множества N_1 и $N_2([M,A])$ имеют вид:

– ограничение на количество заданий в пакетах:

$$\sum_{h=1}^{m_i} a_{ih} = n^i \,; \ \sum_{i=1}^{n}\sum_{h=1}^{m_i} a_{ih} = \sum_{i=1}^{n} n^i ; \tag{7}$$

– ограничение на количество ПЗ i-го типа в π^l ($l = \overline{1, L}$):

$$\sum_{j=1}^{n_p} p_{ij} = m_i ; \tag{8}$$

– ограничение на общее количество ПЗ в π^l ($l = \overline{1, L}$):

$$\sum_{i=1}^{n}\sum_{j=1}^{n_p} p_{ij} = \sum_{i=1}^{n} m_i ; \tag{9}$$

– ограничение на количество заданий i-го типа в пакетах в π^l:

$$\sum_{j=1}^{n_p} r_{ij} = \sum_{h=1}^{m_i} a_{ih} ; \tag{10}$$

– ограничение на общее количество заданий i-ых типов ($i = \overline{1, n}$) в пакетах в π^l ($l = \overline{1, L}$), выполняемых на приборах МС:

$$\sum_{i=1}^{n}\sum_{j=1}^{n_p} r_{ij} = \sum_{i=1}^{n}\sum_{h=1}^{m_i} a_{ih} . \tag{11}$$

Постановка задачи оптимизации составов ПЗ, выполняемых в МС, сформулирована следующим образом. Заданными являются: количество n типов заданий; количество n^i ($i = \overline{1, n}$) заданий каждого типа; матрица T длительностей выполнения заданий в МС; матрицы T^l ($l = \overline{1, L}$) длительностей переналадок приборов. Требуется оптимизировать комплексное решение по составам ПЗ вида *[M,A]* и сформировать соответствующее ему расписание $[P, R, \{T^{nl} \mid l = \overline{1, L}\}]$.

3. Применение метода ветвей и границ при оптимизации комплексного решения по составам ПЗ**, выполняемых в МС.** Решение задачи оптимизации составов ПЗ обеспечивается построением: способа формирования решений по составам ПЗ i-х типов; способа оптимизации комплексных решений по составам ПЗ разных типов (с использованием МВГ).

С целью определения множества решений по составам ПЗ и способа формирования решений по их составам введены условия [23]:

– количество заданий i -го типа в пакетах не менее 2 ($a_{ih} \geq 2$, $h = \overline{1, m_i}$); если при формировании решения по составам m_i ПЗ i -го типа для первого пакета получено $a_{i1} < 2$, то дальнейшее построение решений по составам ПЗ i -го типа в количестве m_i прекращается;

– формирование начального решения для количества m_i ПЗ i -го типа предполагает, что: $a_{ih} = 2$ ($h = \overline{2, m_i}$), $a_{i1} = n^i - \sum_{h=2}^{m_i} a_{ih}$;

– значения параметров m_i , задаваемые первоначально для заданий всех i-х типов ($i = \overline{1, n}$), равны 2 ($m_i = 2$);

– модификация количества ПЗ i-го типа предполагает, что параметр m_i, увеличивается до тех пор, пока в начальном решении по составам m_i пакетов выполняется условие $a_{ih} \geq 2$ $(h = \overline{1, m_i})$; при $a_{i1} < 2$ формирование составов ПЗ i-го типа прекращается;

– формирование решений предполагает увеличение количества заданий в пакете с индексом $h' > 1$ и уменьшение количества заданий в пакете с *h=1*; при условии $a_{i1} \geq a_{ih}$ $(h = \overline{2, m_i})$ модификация составов h-ых ПЗ продолжается; при условии $a_{i1} < a_{ih'}$ решение по составам ПЗ в количестве m_i не рассматривается [23].

Ограничения множества решений по составам ПЗ имеют вид:

1) ограничение на количество ПЗ i-х типов ($i = \overline{1, n}$): $m_i \geq 2$; $m_i \leq \left\lceil \frac{n^i}{2} \right\rceil$, где $\lceil \cdot \rceil$ – операция округления в меньшую сторону;

2) ограничение на количество заданий i-х типов в пакетах ($i = \overline{1, n}$):

$$a_{i,h} \geq 2 \; (h = \overline{2, m_i}),\; a_{i,1} \geq 2 \;,\; a_{i,1} = n^i - \sum_{h=2}^{m_i} a_{i,h} \;,\; a_{i,1} \geq a_{i,h} \; (h = \overline{2, m_i});$$

$$a_{i,1} \leq n^i - 2(m_i - 1)\} \;;\; a_{i,h} \leq \left\lceil \frac{n^i - 2(m_i - 2)}{2} \right\rceil ;$$

3) ограничение на значения количества заданий i-х типов в пакетах:

$$a_{i,1} \in \{2,3,..,n^i - 2(m_i - 1)\},\ a_{i,h} \in \{2,3,..,\left\lceil \frac{n^i - 2(m_i - 2)}{2} \right\rceil\} \ (h = \overline{2, m_i}).$$

Для оптимизации решения по составам ПЗ с использованием МВГ в рассмотрение введены обозначения, приведенные в Таблице 2.

Значение параметра $m_i^{\max}$, соответствующего максимальному количеству ПЗ i-го типа, определяется следующим образом: $m_i^{\max} = \left\lceil \frac{n^i}{2} \right\rceil$. В матрице A_i^n хранятся все возможные решения по составам ПЗ i-го типа, сформированные в соответствии с рассмотренным ниже алгоритмом.

Значение параметра $r_{A_i^n}$ соответствует количеству строк в матрице A_i^n, определяется выражением вида: $r_{A_i^n} = \sum_{m_i=2}^{m_i^{\max}} \left\lfloor \frac{n^i + 1}{m_i} \right\rfloor$, где $\lfloor \cdot \rfloor$ – округление в большую сторону. Матрицы $A_{1i}^{буф}$ и $A_{2i}^{буф}$ предназначены для временного хранения решений по составам ПЗ i-го типа ($i = \overline{1,n}$): в $A_{1i}^{буф}$ хранятся решения, на основе которых формируются новые решения, записываемые в матрицу $A_{2i}^{буф}$. Способ определения составов ПЗ i-го типа формирует на основе решений из $q1_i$ -х строк матрицы $A_{1i}^{буф}$ новые решения, которые сохраняются в матрице $A_{2i}^{буф}$. После того, как новые решения сформированы, решения из матрицы $A_{1i}^{буф}$ переписываются в матрицу A_i^n, а решения из матрицы $A_{2i}^{буф}$ – в $A_{1i}^{буф}$ для формирования новых решений. Значения $r_{A_{1i}^{буф}}$ и $r_{A_{2i}^{буф}}$ определяются выражением $r_{A_{2i}^{буф}} = r_{A_{1i}^{буф}} = \max\limits_{2 \le m_i \le m_i^{\max}} \left(\left\lfloor \frac{n^i + 1}{m_i} \right\rfloor \right)$.

В соответствии с разработанным методом формирования составов ПЗ в каждой матрице A_i^n ($i = \overline{1,n}$) накапливаются все решения по составам пакетов i-го типа. Матрицы $A_{1i}^{буф}$ и $A_{2i}^{буф}$ используются для формирования новых решений по составам ПЗ i-го типа.

Таблица 2. Обозначения, используемые при формировании комплексного решения по составам ПЗ *i*-ых типов, выполняемых в МС

Обозначение	Назначение
1	2
A_i^n	Матрицы, предназначенные для хранения решений по составам ПЗ *i*-ых типов, размерность $r_{A_i^n} \times m_i^{\max}$.
$r_{A_i^n}$	Количество строк в матрице A_i^n.
$m_i^{\max}$	Максимальное количество ПЗ *i*-го типа.
q_i	Индекс строки в матрице A_i^n ($i = \overline{1,n}$).
$(a_{q_i,h})_i^n$	Количество заданий *i*-го типа в *(h=1)*-ом пакете для q_i-го решения по составам пакетов этого типа.
n_p^i	Счетчик количества решений по составам ПЗ *i*-го типа в матрице A_i^n ($i = \overline{1,n}$).
$A_{1i}^{буф}$, $A_{2i}^{буф}$	Матрицы, предназначенные для хранения решений по составам ПЗ *i*-го типа при их формировании (размерность матриц $r_{A_{1i}^{буф}} \times m_i^{\max}$ и $r_{A_{2i}^{буф}} \times m_i^{\max}$)
$r_{A_{1i}^{буф}}$, $r_{A_{2i}^{буф}}$	Количество строк в матрицах $A_{1i}^{буф}$, $A_{2i}^{буф}$.
$q1_i, q2_i$	Индексы строк в матрицах $A_{1i}^{буф}$ и $A_{2i}^{буф}$.
n_{p1}^i, n_{p2}^i	Счетчики решений по составам ПЗ *i*-ых типов, размещаемых в матрицах $A_{1i}^{буф}$ и $A_{2i}^{буф}$.
G_{iq_i}	Множество решений по составам ПЗ, содержащее решение по составу (*h=1*)-го ПЗ *i*-го типа, соответствующее q_i -ой строке матрицы A_i^n.
$I_{МВГ}$	Множество типов заданий, для которых при реализации МВГ выполнено формирование множеств G_{iq_i}.

На основе сформулированного способа определения составов ПЗ, в рассмотрение введены условия, позволяющие сократить количество формируемых решений [23]. Если для текущего рассматриваемого решения ($q1_i$ -й строки матрицы $A_{1i}^{буф}$) выполняется условие $(a_{q1_i,h})_{1i}^{буф} = (a_{q1_i,h+1})_{1i}^{буф}$, то увеличение на 1 значения $(a_{q1_i,h})_{1i}^{буф}$ (при неизменном значении $(a_{q1_i,h+1})_{1i}^{буф}$), а затем увеличение на 1 значения $(a_{q1_i,h+1})_{1i}^{буф}$ (при неизменном значении $(a_{q1_i,h})_{1i}^{буф}$) обуславливает получение одинаковых решений по составам ПЗ *i*-го типа.

На основе этого условия сформулировано следующее утверждение, позволяющее ограничить количество решений [23]: если при формировании решения по составам m_i ПЗ i-го типа на основе решения, соответствующего $q1_i$-й строке матрицы $A_{1i}^{буф}$, для элементов этой строки выполняется $(a_{q1_{i'},h})_{1i'}^{буф} = (a_{q1_{i'},h+1})_{1i'}^{буф}$, то не требуется увеличивать значение элемента $(a_{q1_i,h+1})_{1i}^{буф}$; требуется выполнить переход к элементу $(a_{q1_i,h+j})_{1i}^{буф}$ ($(h+j) \le m_i$), такому, что $(a_{q1_i,h})_{1i}^{буф} \ne (a_{q1_i,h+j})_{1i}^{буф}$, значение которого будет изменено. Если $(a_{q1_i,h})_{1i}^{буф} = (a_{q1_i,h+1})_{1i}^{буф}$ (в общем виде $(a_{q1_i,h})_{1i}^{буф} = (a_{q1_i,h+j})_{1i}^{буф}$), то на основе решения, представленного $q1_i$-й строкой матрицы $A_{1i}^{буф}$, будет получено решение, дублирующее сформированное ранее. В том случае, если для сформированного решения по составам m_i ПЗ i-го типа выполняется условие $a_{q2_i,1} < a_{q2_i,h}$ ($h = \overline{2, m_i}$), то это решение дублирует сформированное ранее, оно не рассматривается и не используется для последующего формирования новых решений [23].

Для реализации алгоритма построения решений по составам ПЗ i-го типа необходимо сформировать начальное решение для $m_i = 2$ (задать составы ПЗ (при $q1_i = 1$ и $n_{p1}^i = 1$)): $a_{q1_i,h} = 2$ ($h = \overline{2, m_i}$), $a_{q1_i,1} = n^i - \sum_{h=2}^{m_i} a_{q1_i,h}$; если $a_{q1_i,1} \ge 2$, то сформировано корректное решение по составам $m_i = 2$ ПЗ i-го типа. Если $a_{q1_i,1} < 2$, то для i-го типа заданий формируется фиксированный пакет, состав которого сохраняется в матрице A_i^n: $q_i = 1, a_{q_i,1} = n^i$, $n_p^i = 1$.

Входными данными для алгоритма формирования решений по составам ПЗ i-го типа на основе решений, находящихся в матрице $A_{1i}^{буф}$, являются: n_{p1}^i – количество решений по составам ПЗ i-го типа в матрице $A_{1i}^{буф}$ ($n_{p1}^i = 1$ в начальном решении при $m_i = 2$), матрица $A_{1i}^{буф}$. Алгоритм построения решений по составам ПЗ i-го типа на основе решений, находящихся в матрице $A_{1i}^{буф}$, имеет порядок шагов:

1. Задать значения параметров: $q1_i = 1$, $n_{p2}^i = 0$, $q2_i = 0$, $n_p^i = 0$.

2. Задать номер пакета h' в решении, соответствующем $q1_i$ –й строке матрицы $A_{1i}^{буф}$, состав которого изменяется, равным 2 ($h'=2$).

3. Модифицировать значение $q2_i$: $q2_i = q2_i + 1$; в h'-м пакете увеличить количество заданий на 1 – сформировать решение по составам ПЗ *i*-го типа ($q2_i$-ю строку матрицы $A_{2i}^{буф}$): $(a_{q2_i,h})_{2i}^{буф} = (a_{q1_i,h})_{1i}^{буф}$ ($h = \overline{2,h'-1}$ & $h = \overline{h'+1,m_i}$), $(a_{q2_i,h'})_2^{буф} = (a_{q1_i,h'})_1^{буф} + 1$,

$$(a_{q2_i,1})_{2i}^{буф} = n^i - \sum_{h=1}^{m_{i'}} (a_{q1_i,h})_{1i}^{буф} \; ; \; n_{p2}^i = n_{p2}^i + 1 .$$

4. Проверить условие $(a_{q2_i,1})_{2i}^{буф} < (a_{q2_i,h'})_{2i}^{буф}$. В случае его выполнения сформированное в виде $q2_i$-й строки матрицы $A_{2i'}^{буф}$ решение не интерпретируется, тогда $n_{p2}^i = n_{p2}^i - 1$ и $q2_i = q2_i - 1$; если $n_{p2}^i = 0$, то перейти на пункт 9; если $n_{p2}^i > 0$, то перейти на пункт 8; при $(a_{q2_i,1})_{2i}^{буф} \geq (a_{q2_i,h'})_{2i}^{буф}$ полученное решение сохранено в матрице $A_{2i}^{буф}$, выполнить задание значения параметра *j=1* (*j*–шаг изменения номера пакета *h’*), перейти на пункт 5.

5. Если $(h'+j) \leq m_i$, то проверить $(a_{q1_i,h'})_{1i}^{буф} > (a_{q1_i,h'+j})_{1i}^{буф}$; если условие не выполняется, то перейти на пункт 6. Если условие выполняется, то $h' = h' + j$, перейти на пункт 3. При $(h'+j) > m_i$ перейти на пункт 7.

6. Модифицировать параметр *j=j+1*. Перейти на пункт 5.

7. При $(h'+j) > m_i$ сформированы все решения (размещенные в матрице $A_{2i}^{буф}$) с использованием решения из $q1_i$-ой строки матрицы $A_{1i'}^{буф}$; модифицировать значение $q1_i$: $q1_i = q1_i + 1$; если $q1_i \leq n_{p1}^i$, то перейти на пункт 2; если $q1_i > n_{p1}^i$, то перейти на пункт 8.

8. Если $n_{p2}^i > 1$, то выполнить сравнение решений, хранимых в матрице $A_{2i}^{буф}$, с точки зрения дублирования ими друг друга (сравнение предполагает формирование копии матрицы $A_{2i}^{буф}$, в копии матрицы $A_{2i}^{буф}$ – упорядочивание элементов каждой $q2_i$-й строке ($q2_i = \overline{1,n_{p2}^i}$) по не возрастанию значений, поэлементное сравнение каждой $q2_i$-й строки ($q2_i = \overline{1,n_{p2}^i - 1}$) с другими $q2'_i$-ми строками

$(q2'_{i'} = \overline{q2_i + 1, n^i_{p2}})$, удаление из матрицы $A^{буф}_{2i}$ строк с индексами $q2'_i$, которые соответствуют строкам в копии матрицы $A^{буф}_{2i}$, дублирующим рассматриваемую $q2_i$-ю строку; при удалении $q2'_i$-й строки из матрицы $A^{буф}_{2i}$ реализуется изменение $n^i_{p2} = n^i_{p2} - 1$, переупорядочивание решений в $A^{буф}_{2i}$); если $n^i_{p2} = 1$, то перейти на пункт 9.

9. Перезаписать решения из матрицы $A^{буф}_{1i}$ в матрицу A^n_i: $(a_{q_i,h})^n_i = (a_{q1_i,h})^{буф}_{1i}$ при $h = \overline{1, m_i}$, $q1_i = \overline{1, n^i_{p1}}$, $q_i = \overline{n^i_p + 1, n^i_p + n^i_{p1}}$; модификация значений n^i_{p1} и $n^i_p : n^i_p = n^i_p + n^i_{p1}$, $n^i_{p1} = 0$; если $n^i_{p2} > 0$, то осуществить перезапись решений из матрицы $A^{буф}_{2i}$ в матрицу $A^{буф}_{1i}$: $(a_{q1_i,h})^{буф}_{1i} = (a_{q2_i,h})^{буф}_{2i}$ при $h = \overline{1, m_i}$, $q1_i = \overline{1, n^i_{p2}}$, $q2_i = \overline{1, n^i_{p2}}$; задать значения параметров алгоритма: $q1_i = 1$, $n^i_{p1} = n^i_{p2}$, $q2_i = 0$, $n^i_{p2} = 0$; перейти на пункт 2.

10. Если $n^i_{p2} = 0$, то реализовать модификацию значения параметра m_i (увеличить количества ПЗ *i*-го типа): $m_i = m_i + 1$; сформировать начальное решение по составам ПЗ: $q1_i = 1$, $a_{q1_i,h} = 2$ ($h = \overline{2, m_i}$), $a_{q1_i,1} = n^i - \sum_{h=2}^{m_i} a_{q1_i,h}$; $n^i_{p1} = 1$; если $a_{q1_i,1} \geq 2$, то инициализировать параметры алгоритма: $q2_i = 0$, $n^i_{p2} = 0$; перейти на пункт 2; если $a_{q1_i,1} < 2$, то начальное решение по составам ПЗ *i*-го типа не является корректным, перейти на пункт 11.

11. Останов алгоритма.

В результате реализации алгоритма каждая q_i-я строка матрицы A^n_i ($q_i = \overline{1, n^i_p}$) является решением по составам ПЗ *i*-го типа.

Для применения МВГ при решении задачи оптимизации составов ПЗ *n* типов построена математическая модель процесса их выполнения в МС. С использованием этой модели на основе матрицы *R*, соответствующей порядкам выполнения ПЗ из q_i-х строк матриц A^n_i в π^l, определяются значения t^{nl}_{ij} для ПЗ *i*-х типов, занимающих *j*-е позиции в π^l. Матрица *R* определяется для комбинации решений по составам ПЗ i-х типов из q_i-х строк ($q_i = \overline{1, n_p^i}$) матриц A_i^n (значения $r_{i,j}$ элементов R определяются на основе значений элементов $a_{q_i,h}$ из q_i-х строк матриц A_i^n в соответствии с отображением $f(t_{ii'}^l)$).

Для использования МВГ [11] с целью оптимизации комплексного решения по составам ПЗ сформулированы:

– способ разбиения множеств решений G_{iq_i}, содержащих ПЗ i-го типа, на подмножества решений $G_{(i+1)q_{(i+1)}}$, включающих ПЗ ($i+1$)-го типа ($q_{(i+1)} = \overline{1, n_p^{i+1}}$) (процедура ветвления);

– способ вычисления верхней оценки (рекорда) $H^{в}$ и способ вычисления нижних оценок $H_{i,q_i}^{н}$ для подмножеств G_{iq_i}, включающих решения по составам ПЗ i-го типа из q_i-х строк матриц A_i^n.

Применение МВГ предусматривает, что комплексное решение по составам ПЗ разных типов (дерево решений) формируется последовательно. Первоначально рассматривается множество решений G_{1q_1}, содержащее пакет заданий ($i=1$)-го типа из q_1-го решения в матрице A_i^n (корневая вершина дерева МВГ). В результате разбиения G_{1q_1} формируются подмножества решений G_{2q_2} ($q_2 = \overline{1, n_p^2}$), включающие (наряду с ПЗ ($i=1$)-го типа) ПЗ ($i=2$)-го типа. Аналогичным образом будет реализовываться процедура разбиения для каждого множества, которое не исключено из рассмотрения после реализации процедуры отсева. В соответствии с изложенной процедурой будет сформировано n_p^1 деревьев. В корне каждого находится вершина, соответствующая ($h=1$)-у ПЗ ($i=1$)-го типа в q_1-м решении ($q_1 = \overline{1, n_p^1}$) по составам ПЗ из матрицы A_1^n. Выбор лучшего решения по составам ПЗ осуществляется путем сравнения значений критерия $f(\cdot)$ для решений, полученных при реализации МВГ для каждого дерева.

В матрице R определяются элементы, которые соответствуют ПЗ, размещенным в π^l при реализации текущей и предыдущих итераций процедуры разбиения (ветвления) в МВГ. Алгоритм формирования матрицы R на основе значений $a_{q_i,h}$ из q_i-х строк ($q_i = \overline{1,n_p^i}$) матриц A_i^n активизируется каждый раз при реализации процедуры разбиения подмножеств для каждого нового подмножества. Сформированная матрица R используется при вычислении нижних оценок значений критерия $f(\cdot)$ для каждого из сформированных подмножеств, вычисления значений критерия $f(\cdot)$ для допустимых решений по порядку выполнения ПЗ, соответствующих рассмотренным подмножествам, и фиксированных пакетов для i-х типов заданий, соответствующих не сформированным подмножествам (при модификации рекорда [11]).

Реализация процедуры ветвления предусматривает, что каждой вершине соответствует кортеж параметров вида $< i, k_s, q_i, h_i >$, где k_s – номер вершины, которая внесена в формируемое дерево решений МВГ, q_i – номер решения (индекс строки в матрице A_i^n), которое сопоставлено с рассматриваемой вершиной, h_i – номер ПЗ i-го типа из решения, соответствующего q_i-й строке матрицы A_i^n. Корневой вершине соответствует набор вида $< 1, 1, q_i, 1 >$.

Для реализации МВГ определены три базовые процедуры [11]: процедуру поиска, процедуру отсева и процедуру ветвления. В качестве процедуры поиска использован поиск в ширину (то есть исследуются все подмножества решений G_{iq_i} ($q_i = \overline{1,n_p^i}$), включающие ($h=1$)-е ПЗ разных составов рассматриваемого i-го типа (из q_i-х строк матрицы A_i^n ($q_i = \overline{1,n_p^i}$)), сформированные на текущей итерации МВГ при реализации процедуры ветвления). Исследование сформированных в результате разбиения подмножеств G_{iq_i} предполагает вычисление оценок $H_{i,q_i}^{н}$ для каждого из них, вычисление значений оценок $H_S^{в}$ критерия $f(\cdot)$ для формируемых допустимых решений (с целью обновления рекорда при условии $H_S^{в} < H^{в}$).

Процедура разбиения предполагает, что каждому подмножеству $G_{(i-1)q_{(i-1)}}$, содержащему решение по составам ПЗ из q_{i-1}-й строки матрицы A_{i-1}^n ($q_{i-1} = \overline{1,n_p^{i-1}}$), не исключенному из рассмотрения после отсева, ставятся в соответствие подмножества G_{iq_i}, содержащие q_i-е решения по составам ПЗ i-го типа ($q_i = \overline{1,n_p^i}$). Каждому подмножеству решений G_{iq_i} ($q_i = \overline{1,n_p^i}$) соответствует вершина дерева МВГ с кортежем $< i,k_s,q_i,h_i >$. При реализации обхода дерева МВГ в глубину определяются идентификаторы решений q_i, пакеты из которых размещаются в последовательностях π^l. В результате формируется матрица R, на основе которой с использованием модели процесса выполнения ПЗ в МС реализуется вычисление значений $H^{н}_{i,q_i}$. Процедура отсева предполагает, что из рассмотрения исключаются множества G_{iq_i}, для которых $H^{н}_{i,q_i} \geq H^{в}$. Вычисление верхней оценки значений $f(\cdot)$ для множества решений G_{1q_1} осуществляется на основе сформированного в соответствии с введенным правилом расписания выполнения фиксированных ПЗ i-ых типов. Введено условие: $Pr_1^1 = Pr_2^1 = .. = Pr_n^1 = 0$, $Pr_1^l = 0$ ($l = \overline{1,L}$). Простои $(l=2)$-го прибора в ожидании готовности фиксированных пакетов ($i=2$)-го и ($i=3$)-го типов определяются в соответствии с выражениями вида:

$$Pr_2^2 = max(0,(n^1 t_{11} + t_{12}^1 + n^2 t_{12}) - (\sum_{l=1}^{2} n^1 t_{l,1} + t_{12}^2));$$

$$Pr_3^2 = max(0,(n^1 t_{11} + t_{12}^1 + n^2 t_{12} + t_{23}^1 + n^3 t_{13}) - (\sum_{l=1}^{2} n^1 t_{l,1} + t_{12}^2 +$$
$$+ Pr_2^2 + n^2 t_{22} + t_{23}^2)).$$

В общем виде формула для вычисления простоев $(l=2)$-го прибора в ожидании готовности фиксированного ПЗ i-го типа к выполнению имеет вид:

$$Pr_i^2 = max(0,(\sum_{j=1}^{i} n^j t_{1j} + \sum_{j=1}^{i-1} t_{j,j+1}^1) - (n^1 t_{11} + \sum_{j=1}^{i-1} n^j t_{2j} + \sum_{j=1}^{i-1} Pr_j^2 +$$
$$+ \sum_{j=1}^{i-1} t_{j,j+1}^1).$$

Выражение для простоев ($l=3$)-го прибора в ожидании начала выполнения фиксированных пакетов ($i=2$)-го и ($i=3$)-го типов:

$$Pr_2^3 = max(0,(\sum_{l=1}^{2} n^1 t_{l1} + t_{12}^2 + Pr_2^2 + n^2 t_{22}) - (\sum_{l=1}^{3} n^1 t_{l,1} + t_{12}^3));$$

$$Pr_3^3 = max(0,(\sum_{l=1}^{2} n^1 t_{l,1} + t_{12}^2 + Pr_2^2 + n^2 t_{22} + t_{23}^2 + Pr_3^2 + n^3 t_{23}) -$$
$$-(\sum_{l=1}^{3} n^1 t_{l,1} + t_{12}^3 + Pr_2^3 + n^2 t_{32} + t_{23}^3)).$$

Общей вид формулы для вычисления простоев l-го прибора в ожидании готовности фиксированного ПЗ i-го типа к выполнению:

$$Pr_i^l = max(0,(\sum_{q=1}^{l-2} n^1 t_{q,1} + \sum_{j=1}^{i} n^j t_{l-1,j} + \sum_{j=1}^{i} Pr_j^{l-1} + \sum_{j=1}^{i-1} t_{j,j+1}^{l-1}) -$$
$$-(\sum_{q=1}^{l-1} n^1 t_{q,1} + \sum_{j=1}^{i-1} n^j t_{l,j} + \sum_{j=1}^{i-1} Pr_j^l + \sum_{j=1}^{i-1} t_{j,j+1}^1).$$

Выражение для вычисления верхней оценки (рекорда) в корне:

$$H^в = \sum_{l=1}^{L} n^1 t_{l,1} + t_{12}^L + Pr_2^L + n^2 t_{L,2} + t_{23}^L + Pr_3^L + n^3 t_{L,3} + ..$$
$$.. + Pr_{n-1}^L + n^{(n-1)} t_{L,(n-1)} + t_{(n-1),n}^L + Pr_n^L + n^n t_{L,n}. \quad (12)$$

Выражение (12) представлено в общем виде:

$$H^в = \sum_{l=1}^{L-1} n^1 t_{l,1} + \sum_{j=1}^{n} n^j t_{L,j} + \sum_{j=1}^{n} Pr_j^L + \sum_{j=1}^{n-1} t_{j,j+1}^L. \quad (13)$$

Определение нижней оценки $H_{iq_i}^н$ осуществляется при формировании множества решений G_{iq_i}, содержащего пакет i-го типа из q_i -й строки (решения) матрицы A_i^n. Формирование множества G_{iq_i} сопровождается добавлением в последовательности π^l ($l = \overline{1,L}$) пакета i-го типа, параметры которого соответствуют вершине дерева МВГ.

Нижняя оценка $H^{н}_{iq_i}$ для множества G_{iq_i}, полученного в результате ветвления, формируется с учетом добавленных в π^{l} ПЗ рассмотренных i'-ых типов ($i'=\overline{1,i}$), размещения в π^{L} фиксированных ПЗ других типов, для решений по составам которых вершины в дереве не сформированы, размещения в π^{L} «оставшихся» заданий i'-х типов ($i'=\overline{1,i}$).

Для вычисления значения $H^{н}_{2q_2}$ синтезируется матрица R (с учетом правила формирования последовательности π^{L}), на основе которой реализуется вычисление значений t^{nl}_{ij} ($i=\overline{1,2}$, $j=\overline{1,2}$, $l=\overline{1,L}$). Оценка $H^{н}_{2q_2}$ определяется следующим образом:

$$H^{н}_{2q_2}=t^{nL}_{22}+r_{22}t_{L,2}+t^{L}_{23}+n^{3}t_{L,3}+t^{L}_{34}+n^{4}t_{L,4}+..$$
$$..+t^{L}_{(n-1),n}+n^{n}t_{L,n}+t^{L}_{n,1}+(n^{1}-r_{11})t_{L1}+t^{L}_{1,2}+(n^{1}-r_{22})t_{L2}, \quad (14)$$

где r_{22} – это количество заданий в пакете ($i=2$)-го типа в ($j=2$)-й позиции в π^{L}, соответствующее составу ($h=1$)-го пакета из q_2-го решения в матрице A^{n}_{2}. При разбиении одного из подмножеств G_{2q_2} на подмножества G_{3q_3} для каждого из них формируются вершины в дереве МВГ, которым соответствуют наборы $<3,k_3,q_3,1>$. При этом ($h=1$)-й ПЗ ($i=3$)-го типа из q_3-го решения по составам ПЗ добавляется в π^{l}. Формируется матрица R соответствующего вида, на основе которой реализуется вычисление значений t^{nl}_{ij} ($i=\overline{1,3}$, $j=\overline{1,3}$, $l=\overline{1,L}$). Оценка $H^{н}_{3q_3}$ определяются следующим образом:

$$H^{н}_{3q_3}=t^{nL}_{33}+r_{33}t_{L,3}+t^{L}_{34}+n^{4}t_{L,4}+..+t^{L}_{(n-1),n}+$$
$$+n^{n}t_{L,n}+t^{L}_{n,1}+(n^{1}-r_{11})t_{L1}+t^{L}_{1,2}+(n^{2}-r_{22})t_{L2}+(n^{3}-r_{33})t_{L3}, \quad (15)$$

где r_{33} – это количество заданий в пакете ($i=3$)-го типа в ($j=3$)-й позиции в π^L, соответствующее ($h=1$)-у пакету в q_3-м решении в матрице A_3^n. С учетом (9), (10) выражение для вычисления нижних оценок $H_{iq_i}^н$ для множеств G_{iq_i} ($q_i = \overline{1,n_p^i}$) сформировано в виде:

$$H_{iq_i}^н = t_{ij}^{nL} + r_{ij}t_{L,i} + t_{i,i+1}^L + n^{i+1}t_{L,i+1} + .. + t_{(n-1),n}^L + \\ + n^n t_{L,n} + t_{n,1}^L + (n^1 - r_{11})t_{L1} + t_{1,2}^L + (n^2 - r_{22})t_{L2} + .. + (n^i - r_{ij})t_{L,i}, \quad (16)$$

где r_{ij} – это количество заданий в пакете i-го типа в j-й позиции в π^L, соответствующее составу ($h=1$)-го пакета в q_i-м решении в A_i^n. Формирование матрицы R для вычисления значений $H_{iq_i}^н$ реализуется с использованием процедуры MATRICA_R1, рассмотренной ниже.

С целью обновления верхней оценки $H^в$ для подмножеств G_{iq_i} формируются допустимые решения по порядкам выполнения ПЗ в МС, предусматривающие, что в последовательностях π^l пакеты размещаются следующим образом (с учетом эвристики):

– первоначально в π^l ($l = \overline{1,L}$) размещаются ПЗ рассмотренных i'-х типов ($i' = \overline{1,i}$, i– тип заданий, для которого сформированы подмножества решений на текущей итерации алгоритма МВГ);

– в π^l $\left(l = \overline{1,L} \right)$ размещаются фиксированные ПЗ оставшихся i'-х типов ($i' = \overline{i+1,n}$), для которых подмножества $G_{i'q_{i'}}$ не сформированы;

– в π^l $\left(l = \overline{1,L} \right)$ размещаются «оставшиеся» задания i'-х типов ($i' = \overline{1,i}$) (типов, для которых сформированы подмножества $G_{i'q_{i'}}$).

Для построения матрицы R, соответствующей последовательностям π^l, сформированным с использованием предложенного способа, разработана процедура MATRICA_R2, рассмотренная ниже. С учетом матрицы *R*, а также значений t_{ij}^{nl} ($i=\overline{1,n}$, $j=\overline{1,(2i+(n-i))}$, $l=\overline{1,L}$) вычисляются значения $f(\cdot)$, характеризующие подмножества $G_{i'q_{i'}}$: $f(\cdot)=(t_{i,n_p}^{nL}+r_{i,n_p}t_{L,i})$ (при $i=\overline{1,n}$, $j=\overline{1,(2i+(n-i))}$). Проверяется условие $H^{в}>max\, f([M,A],[P,R,\{T^{nl}\,|\,l=\overline{1,L}\}])$, при выполнении которого значение $H^{в}$ (рекорда) переопределяется.

Предварительная инициализация входных параметров алгоритма МВГ оптимизации решения по составам ПЗ *n* типов имеет вид: задать идентификатор типа заданий *i* равным 1 (*i=1*); для (*i=1*)-типа заданий задать значение индекса q_1 строки в A_1^n ($q_1=\overline{1,n_p^1}$), решение из которой будет интерпретироваться при формировании дерева; для (*i=1*)-типа заданий задать номер пакета h_1 равным 1. Алгоритм, реализующий процедуру МВГ с целью оптимизации решений по составам ПЗ разных типов, имеет следующий порядок шагов:

1. Для (*i=1*)-го типа заданий сформировать корневую вершину, сопоставить корневой вершине набор значений параметров $<i,k_s,q_i,h_i>$ в виде: $<1,1,q_1,1>$. Инициализировать $Q_1^1: Q_1^1=\{1\}$.

2. По формуле (8) вычислить значение верхней оценки $H^{в}$ значений критерия $f(\cdot)$ для рассматриваемого множества допустимых решений G_{1,q_1} (полученное значение $H^{в}$ – текущий рекорд).

3. Модифицировать тип заданий *i* и номер итерации алгоритма *s* (номер уровня дерева МВГ): $i=i+1$, $s=s+1$. Инициализировать $H_s^{в}$ значением *W* (*W*– бесконечно большое число): $H_s^{в}=W$.

4. Если $i\le n$, то инициализировать параметр h_i для рассматриваемого *i*-го типа заданий значением 1 ($h_i=1$). Для рассматриваемого *i*-го типа заданий выполнить инициализацию параметра q_i значением 1 ($q_i=1$). Задать значение счетчика sh_s количества потомков, закрепленных (зафиксированных) на *s*-ом уровне дерева МВГ, равным

0 ($sh_s = 0$). Инициализировать множество $Q_s^1 = \varnothing$. Модифицировать множество $I_{МВГ}$: $I_{МВГ} = I_{МВГ} \cup \{i\}$. Перейти на пункт 5. Если $i > n$, то инициализировать $H^в : H^в = W$. Перейти на пункт 10.

5. Если $Q_{s-1}^1 = \varnothing$, то проверить $H^в > H_s^в$. Если $H^в > H_s^в$, то инициализировать $H^в$: $H^в = H_s^в$. Перейти на пункт 3. Если $Q_{s-1}^1 \neq \varnothing$, то извлечь из $Q^1{}_{s-1}$ идентификатор родительской вершины k_{s-1}^*, для которой формируются потомки: $k_{s-1}^* = min\{ k_{s-1} \mid k_{s-1} \in Q_{s-1}^1 \}$. Модифицировать множество Q_{s-1}^1: $Q_{s-1}^1 = Q_{s-1}^1 \setminus \{ k_{s-1}^* \}$. Выполнить переход на (*s*-*1*)-ом уровне дерева МВГ к вершине с номером k_{s-1}^*.

6. Модифицировать значение sh_s счетчика вершин, зафиксированных в дереве МВГ на *s*-ой итерации алгоритма: $sh_s = sh_s + 1$. Сформировать потомка вершины с идентификатором k_{s-1}^*. Поставить ей в соответствие набор параметров вида $< i, k_s, q_i, h_i >$ (где $k_s = sh_s$).

7. Выполнить обход в глубину из корневой вершины в рассматриваемую k_s-ю вершину (в результате определяются значения параметров q_i для всех вершин, лежащих на пути ($i \in I_{МВГ}$)). Выполнить вызов процедуры MATRICA_R1 построения матрицы *R* для рассматриваемых типов заданий $i \in I_{МВГ}$ и значений q_i ($i \in I_{МВГ}$). Вычислить на основе матрицы *R* значение $H_{i,q_i}^н$ для сформированного подмножества решений G_{i,q_i} (вершины k_s). Выполнить вызов процедуры MATRICA_R2 построения матрицы *R* для типов заданий $i \in I_{МВГ}$ и полученных значений q_i ($i \in I_{МВГ}$). Вычислить на основе матрицы *R* значение критерия $f([M,A],[P,R,\{T^{nl} \mid l = \overline{1,L}\}])$.

8. Если $H_{i,q_i}^н \geq H^в$, то удалить рассматриваемую вершину из дерева МВГ (в вершине с идентификатором k_{s-1}^* удалить ссылку на эту вершину). Модифицировать значение sh_s счетчика вершин, зафик-

сированных в дереве МВГ на s-й итерации алгоритма: $sh_s = sh_s - 1$. Выполнить проверку условия $q_i < n_p^i$. Если $q_i < n_p^i$, то модифицировать значение q_i: $q_i = q_i + 1$. Перейти на пункт 6. Если $q_i > n_p^i$, то положить $q_i = 1$. Перейти на пункт 5.

9. Если $H^{н}_{i,q_i} < H^{в}$, то модифицировать множество Q_s^1 вершин, зафиксированных в дереве МВГ на s-ой итерации алгоритма: $Q_s^1 = Q_s^1 \cup \{ k_s \}$. Если $H_s^{в} > f(\cdot)$, то изменить $H_s^{в}$: $H_s^{в} = f(\cdot)$. Если $H_s^{в} \le f(\cdot)$, то проверить $q_i < n_p^i$. Если $q_i < n_p^i$, то модифицировать значение q_i: $q_i = q_i + 1$. Перейти на пункт 6. Если $q_i > n_p^i$, то положить $q_i = 1$. Перейти на пункт 5.

10. Если $Q_s^1 \ne \varnothing$, то выделить в множестве Q_s^1, соответствующем последней итерации алгоритма построения дерева МВГ, вершину k_s^*: $k_s^* = min\{ k_s \mid k_s \in Q_s^1 \}$. Модифицировать множество Q_s^1: $Q_s^1 = Q_s^1 \setminus \{ k_s^* \}$. Если $Q_s^1 = \varnothing$, то перейти на пункт 13.

11. Выполнить обход в глубину из корневой вершины в рассматриваемую k_s-ю вершину (определяются значения q_i для вершин, лежащих на пути). Вызвать процедуру MATRICA_R3 построения матрицы R для рассматриваемых типов заданий. Вычислить на основе матрицы R значение $f(\cdot)$ для рассматриваемого решения.

12. Если $f(\cdot) < H^{в}$, то зафиксировать параметры q_i вершин, соответствующих решениям по составам ПЗ i-х типов ($i = \overline{1,n}$): $q_i^{н} = q_i$ ($i = \overline{1,n}$). Модифицировать $H^{в} = f(\cdot)$. Перейти на шаг 10.

13. Останов алгоритма.

В результате получены индексы $q_i^{н}$ ($i = \overline{1,n}$) решений по составам ПЗ i-х типов (номера $q_i^{н}$ строк в матрицах $A_i^{п}$), комплексирование которых позволяет получить решение с минимальным значением $f(\cdot)$. После того, как для каждого дерева получено лучшее решение, в качестве оптимального выбирается решение с минимальным значением $f(\cdot)$. Вычислительная сложность алгоритма определена как

$O(n \cdot (n^i)^2)$. Для вычисления оценок $H^{и}_{i,q_i}$ подмножеств G_{i,q_i} формируется матрица *R* с использованием процедуры MATRICA_R1. Входными параметрами для нее являются: множество $I_{МВГ}$; значения параметров q_i для вершин ($i \in I_{МВГ}$). Порядок ее шагов следующий:

1. Выполнить инициализацию множества I_1 типов заданий, используемого при реализации алгоритма: $I_1 = I_{МВГ}$.
2. Задать значения параметров h_i ($i \in I_1$) следующим образом: $h_i = 1$, инициализировать номер *j* позиции ПЗ в π^l значением 1 (*j=1*).
3. Определить в множестве I_1 тип заданий *i'*, параметры ПЗ которого используются при инициализации значения элемента $r_{i'j}$ матрицы *R*: $i' = min(i \mid i \in I_1)$, модифицировать I_1: $I_1 = I_1 \setminus \{i'\}$.
4. Инициализировать значение элемента $r_{i'j}$ матрицы *R*: $r_{i'j} = a_{q_{i'},h_{i'}}$. Модифицировать номер *j* позиции ПЗ в π^l: *j=j+1*.
5. Если $I_1 \neq \varnothing$, то перейти на пункт 3.Если $I_1 = \varnothing$, то пункт 5.
6. Останов алгоритма.

Процедура MATRICA_R2 реализует построение матрицы *R*, используемой при вычислении значений $f(\cdot)$ на *s*-ой итерации МВГ для допустимых решений. Ее входные параметры: множество $I_{МВГ}$; значения q_i для вершин, соответствующих решениям по составам ПЗ *i*-х типов ($i \in I_{МВГ}$); множество *I* типов заданий. Порядок шагов алгоритма процедуры MATRICA_R2 следующий:

1. Выполнить инициализацию: множества I: $I = \{1, 2, .., n\}$, множества I_1: $I_1 = I_{МВГ}$, множества I_2: $I_2 = I \setminus I_{МВГ}$.
2. Задать значения параметров h_i ($i \in I_1$) следующим образом: $h_i = 1$, инициализировать номер позиции *j* ПЗ в π^l значением 1 (*j=1*).
3. Определить в множестве I_1 тип заданий *i'*, параметры ПЗ которого используются при инициализации значения элемента $r_{i'j}$ матрицы *R*: $i' = \min(i \mid i \in I_1)$, модифицировать I_1: $I_1 = I_1 \setminus \{i'\}$.
4. Инициализировать значение элемента $r_{i'j}$ матрицы *R*: $r_{i'j} = a_{q_{i'},h_{i'}}$. Модифицировать номер *j* позиции ПЗ в π^l: *j=j+1*.

5. Если $I_1 = \varnothing$, то перейти на пункт 5. Если $I_1 \neq \varnothing$, то перейти на пункт 3.

6. Если $I_2 \neq \varnothing$, то извлечь из I_2 тип заданий, фиксированный пакет которых размещается в π^l: $i' = min(i \mid i \in I_2)$. Инициализировать значение элемента $r_{i'j}$ матрицы *R*: $r_{i'j} = n^{i'}$. Перейти на пункт 6. Если $I_2 = \varnothing$, то модифицировать номер *j* позиции: $j=j+1$. Инициализировать множество I_1: $I_1 = I_{МВГ}$. Перейти на пункт 7.

7. Модифицировать номер *j*: $j=j+1$. Перейти на пункт 5.

8. Определить в I_1 тип заданий *i'*, размещаемых в π^l: $i' = min(i \mid i \in I_1)$, модифицировать I_1: $I_1 = I_1 \setminus \{ i' \}$.

9. Инициализировать значение элемента $r_{i'j}$ матрицы *R*: $r_{i'j} = n^i - \sum_{s=1}^{j-1} r_{i',s}$. Модифицировать номер *j* позиции ПЗ в π^l: $j=j+1$.

10. Если $I_1 = \varnothing$, то перейти на пункт 10. Если $I_1 \neq \varnothing$, то перейти на пункт 7.

11. Останов алгоритма.

Процедура MATRICA_R3 реализует построение матрицы *R* для получения окончательного значения критерия $f(\cdot)$ для решений, получаемых для отдельных деревьев МВГ. Перед реализацией процедуры множества I_1 и I_2 заданы следующим образом: $I_1 = I_2 = I$, где *I* – упорядоченное множество типов заданий. Порядок шагов алгоритма:

1. Задать значения параметров h_i ($i = \overline{1,n}$): $h_i = 1$ при ($i = \overline{1,n}$), инициализировать номер позиции *j* ПЗ в π^l значением 1 ($j=1$).

2. Определить в множестве I_2 тип заданий *i'*, параметры ПЗ которого используются при инициализации значения элемента r_{ij} матрицы *R*: $i' = min(i \mid i \in I_2)$, модифицировать I_2: $I_2 = I_2 \setminus \{ i' \}$.

3. Инициализировать значение элемента $r_{i'j}$ матрицы R: $r_{i'j} = a_{q_{i'},h_{i'}}$, Если $\sum_{h=1}^{n_p} r_{i'h} < n^i$, то $h_{i'} : h_{i'} = h_{i'} + 1$, модифицировать *j*:

$j=j+1$. Перейти на пункт 4. Если $\sum_{h=1}^{n_p} r_{i'h} = n^i$, то $I_1 = I_1 \setminus \{ i' \}$, модифицировать j: $j=j+1$. Перейти на пункт 4.

4. Если $I_2 \neq \varnothing$, то определить тип заданий i', пакет которых размещается в $\pi^l : i' = min(i \mid i \in I_2)$. Перейти на пункт 3. Если $I_2 = \varnothing$, то проверить условие $I_1 \neq \varnothing$. Если $I_1 \neq \varnothing$, то инициализировать I_2: $I_2 = I_1$. Перейти на пункт 2. Если $I_1 = \varnothing$, то перейти на пункт 5.

5. Останов алгоритма.

Для исследования эффективности планирования с использованием рассмотренной реализации МВГ использованы параметры: неоднородность длительностей выполнения заданий на приборах МС – $max(t_{li}) / min(t_{li})$; неоднородность длительностей переналадок приборов МС – $max(t_{ij}^l) / min(t_{ij}^l)$. Значения $max(t_{ij}^l) / min(t_{ij}^l)$ задавались равными 1, 2, 4, 8, 16; значения $max(t_{li}) / min(t_{li})$ – равными 1, 2, 4, 8, 16. Значения параметра n заданы равными 3 и 5, параметра n^i равными 6,8,12. Значение L задано равным 5. В качестве параметра, характеризующего снижение времени на выполнение ПЗ, использовалось отношение $f_{эмосп} = (f^{фикс} - f^{мосп}) / f^{фикс}$, где $f^{фикс}$ – значение критерия f для фиксированных пакетов (ФП), $f^{мосп}$ – значение критерия f для решения с оптимизированными составами ПЗ. Графики, отображающие зависимости снижения времени выполнения заданий при формировании ПЗ с использованием МВГ по сравнению с фиксированными пакетами для n =5, $n^i \in \{ 6,8,12 \}$, представлены на рисунках 1-3. Аналогичные результаты получены при исследовании зависимости $f_{эмосп}$ от входных параметров при n =5, $n^i \in \{ 6,8,12 \}$.

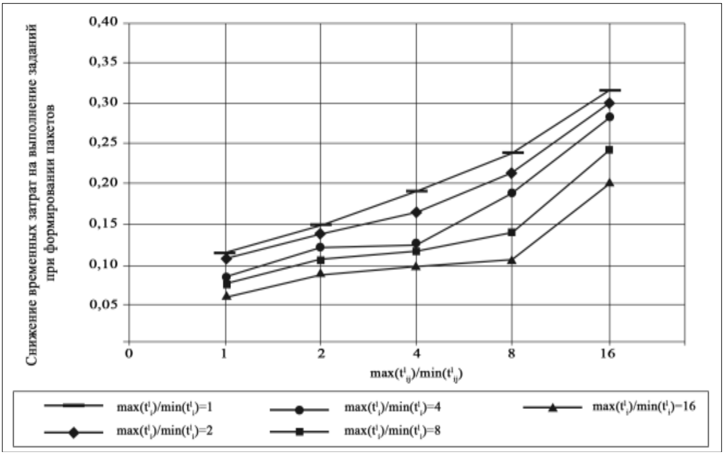

Рис. 1. Снижение времени выполнения заданий в МС при оптимизации составов ПЗ по сравнению с ФП ($n = 5$, $n^i = 6$)

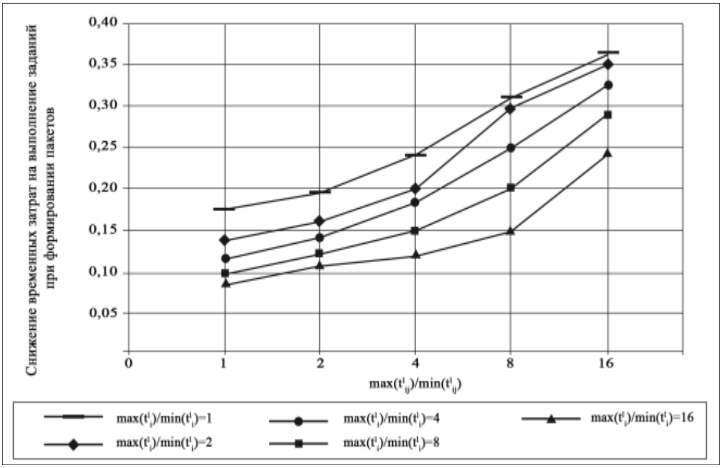

Рис. 2. Снижение времени выполнения заданий в МС при оптимизации составов пакетов по сравнению с ФП ($n = 5$, $n^i = 8$)

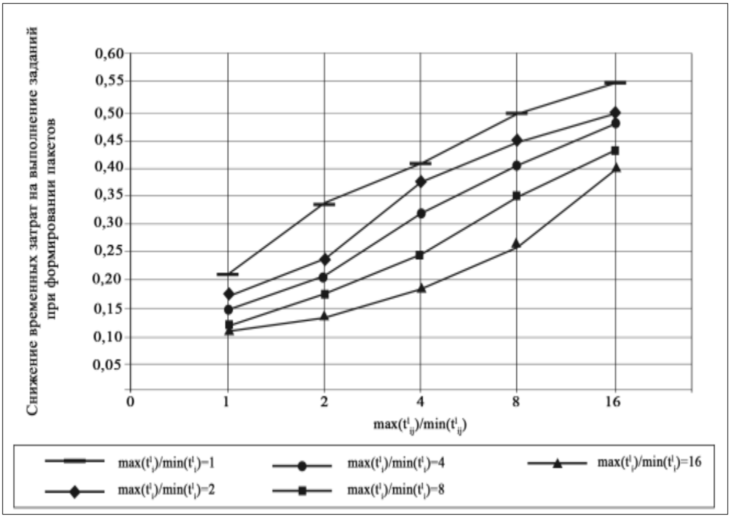

Рис. 3. Снижение времени выполнения заданий в МС при оптимизации составов пакетов по сравнению с ФП ($n = 5$, $n^{i} = 12$)

Результаты исследований показали, что использование МВГ при планировании позволяет максимально на 45% (в среднем на 30%) снизить временные затраты на выполнение ПЗ в МС (в сравнении с фиксированными ПЗ).

4. Заключение. Существующие методы планирования выполнения ПЗ в МС позволяют определять решения задач по составам ПЗ и расписаниям их выполнения малой размерности, не гарантируют получение решений, приближающихся к оптимальным. В то же время с использованием МВГ реализуется построения расписаний выполнения ЕЗ в системах разного вида. Для использования МВГ с целью оптимизации составов ПЗ разработано эвристическое правило упорядочивания пакетов в последовательностях их выполнения в МС. Разработана математическая модель процесса выполнения ПЗ в МС. Построены выражения для вычисления нижних и верхних оценок значений критерия для каждого из подмножеств решений, сформированных в результате реализации процедуры разбиения МВГ. Сформулирован алгоритм формирования решений по составам ПЗ разных типов и алгоритм МВГ, позволяющий определить оптимальное решение по составам ПЗ. Программная реализация алгоритмов показала, что их использование позволяет увеличить эффективность планирования в среднем на 30% по сравнению с ФП. Результатами, представленными в работе и обладающими элементами научной новизны, являются: математическая модель процесса выполнения ПЗ в МС; эвристическое правило упорядочивания ПЗ в последовательностях их выполнения в МС; алгоритм МВГ, адаптированный для решения задачи оптимизации составов ПЗ, выполняемых в МС; выражение для расчета нижних оценок и верхней оценки; способ построения последовательностей выполнения ПЗ в МС.

Литература

1. Ogun B., Cigdem A.-U. Mathematical Models for a Batch Scheduling Problem to Minimize Earliness and Tardiness. Journal of Industrial Engineering and Management. JIEM, 2018. № 11(3). pp. 390-405.
2. Chaudhry I.A., Elbadawi I. A-Q., Usman M., Chugtai M.T. Minimising Total Flowtime in a No-Wait Flow Shop (NWFS) using Genetic Algorithms. Ingeniería e Investigación. 2018. Vol. 38. № 3. pp. 68-79.
3. Tan Y., Huangi W., Sun Y., Yue Y. Comparative Study of Different Approaches to Solve Batch Process Sheduling and Optimisation Problems. Proceedings of the 18th International Conference on Automation & Computing. Loughborough University. Leicestershire. UK. 2012. pp 424–444.
4. Кротов К.В. Использование аппарата генетических алгоритмов при формировании решений по составам партий данных в двухуровневой задаче построения комплексных расписаний их обработки. Автоматизированные технологии и производства // Международный научно-технический журнал. 2017. №2 (16). С. 23-34.
5. Li X.L., Wang Y. Scheduling Batch Processing Machine Using Max–Min Ant System Algorithm Improved by a Local Search Method. Mathematical Problems in Engineering. 2018. Vol. 2018.
6. Li Sh., Cheng T.C.E., Ng C.T., Yuan J. Single-machine batch scheduling with job processing time compatibility. Theoretical Computer Science. 2015. Vol. 583. pp. 57-66.
7. Jin M., Liu X., Luo W. Single-Machine Parallel-Batch Scheduling with Nonidentical Job Sizes and Rejection. Mathematics. 2020. Vol. 8.
8. Surjandari I., Rachman A., Purdianta, Dhini A. The batch scheduling model for dynamic multi-item, multi-level production in an assembly job shop with parallel machines. International Journal of Technology. 2015. № 1. pp. 84-96.
9. Joglekar G. Using Simulation for Scheduling and Rescheduling of Batch Processes. Processes. 2017. № 5.
10. Ковалев М.Я. Модели и методы календарного планирования. Курс лекций. Минск: БГУ. 2004. 63 с.
11. Morrison D.R., Jacobson Sh.H., Sauppe J.J., Sewell E.C. Branch-and-bound algorithms: A survey of recent advances in searching, branching and pruning. Discrete Optimization. 2016. № 19. pp. 79-102.
12. Dawd S.T., Ayvaz B. A branch and bound approach for single machine scheduling problem. Istanbul Commerce University. Journal of Science. 2017. № 16 (31). pp. 43-55.
13. Rasti-Barzoki M., Hejazi S.R. A branch and bound algorithm to minimize the total weighted number of tardy jobs and delivery costs with late deliveries for a supply chain-scheduling problem. Journal of Industrial and Systems Engineering. 2017. Vol. 10. № 1. pp 50- 60.

14. Прилуцкий М.Х., Власов В.С. Метод ветвей и границ с эвристическими оценками для конвейерной задачи теории расписаний // Математическое моделирование. Оптимальное управление. Вестник Нижегородского университета им. Н.И. Лобачевского. 2008. № 3. 147-153 с.
15. Takano M.I., Nagano M.S. A branch-and bound method to minimize the makespan in a permutation flow shop with blocking and setup times. Cogent Engineering. 2017.
16. Григорьева Н.С. Алгоритм ветвей и границ для задачи составления расписания на параллельных процессорах // Вестник Санкт-Петербургского университета. серия 10. 2009. выпуск 1. 44-55 с.
17. Mazda Ch.N., Kurniawati D.A. Branch and Bound Method to Overcome Delay Delivery Order in Flow Shop Scheduling Problem. IOP Conf. Series: Materials Science and Engineering. 2020.
18. Watermeyer K., Zimmermann J. A branch-and-bound procedure for the resource-constrained project scheduling problem with partially renewable resources and general temporal constraints. OR Spectrum. 2020. № 42. pp. 427–460.
19. Hu Sh., Wang S., Kao Y., Ito T., Sun X. A Branch and Bound Algorithm for Project Scheduling Problem with Spatial Resource Constraints. Hindawi Publishing Corporation. Mathematical Problems in Engineering. Vol. 2015. P. 9.
20. Могилев А.А. Обзор методов решения задач теории расписаний // Информатика, вычислительная техника и инженерное образование. 2019. № 4 (37). 19-32 с.
21. Кротов К.В. Информационная модель многоуровневой системы выполнения конвейеризированных программ // Международный научно-технический журнал «Проблемы управления и информатики». 2014. № 3. 89-101 с.
22. Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B. Sequencing and Scheduling: Algorithms and Complexity. Handbook in Operations Research and Managment Science. North-Holland, Amsterdam. 1993. Vol. 4. pp. 445-522.
23. Кротов К.В. Комплексный метод определения эффективных решений по составам партий данных и расписаниям их обработки в конвейерных системах // Журнал «Вычислительные технологии». Новосибирск. Изд-во Института вычислительных технологий СО РАН. 2018. № 3. 58-76 с.
24. Кротов К.В., Скатков А.В. Организация web-ориентированного сервиса мониторинга окружающей среды с использованием данных дистанционного зондирования Земли и конвейеризации обработки данных // Труды учебных заведений связи. 2021. Т. 7. №1. 105-121 с.

Кротов Кирилл Викторович — канд. техн. наук, доцент, Кафедра информационных систем ФГАОУ ВО Севастопольский Государственный университет (СевГУ). Область научных интересов: моделирование процессов в вычислительных системах, оптимальное планирование процессов в вычислительных системах, иерархические игры. Число научных публикаций — 110. krotov_k1@mail.ru; ул. Университетская, 33, 299053, Севастополь, Россия; р.т.: +7 (978) 730-38-19.

DOI 10.15622/ia.2022.21.1

K. KROTOV

MATHEMATICAL MODEL AND ALGORITHM OF BRANCH AND BOUNDARY METHOD FOR OPTIMIZING SOLUTIONS FOR PACKAGE COMPOSITIONS IN MULTI-STAGE SYSTEMS

Krotov K.V. **Mathematical model and algorithm of branch and boundary method for optimizing solutions for package compositions in multi-stage systems**

Abstract. Modern methods for solving problems of planning of task packages execution in multi-stage systems are characterized by the presence of restrictions on their dimension, the impossibility of obtaining guaranteed best results in comparison with fixed packages for different values of the input parameters of tasks. The problem of optimizing the composition of task packages executed in multi-stage systems using the method of branches and borders is solved in the paper. Studies of various ways of forming the order of execution of task packages in multi-stage systems (heuristic rules for ordering task packages in the sequences of their execution on MS devices) have been carried out. The method of ordering packets in the sequence of their execution (a heuristic rule), which minimizes the total time for implementing actions with them on the devices, is defined. The method of ordering the types of tasks, according to which their packages are considered in the procedure of the method of branches and borders, is formulated on the basis of the obtained rule. A mathematical model of the process of implementing actions with packages on the system devices, which provides the calculation of its parameters, has been built. The construction of a method for forming all possible solutions for the composition of task packages for a given number of them has been completed. Decisions on the composition of task packages of different types are interpreted in the procedure of the method of branches and borders in order to build the optimal combination of them. To implement the method of branches and borders, a branching (splitting) procedure is formulated, which assumes the formation of subsets of solutions that include packages of different compositions of tasks of the same type. Expressions for calculating the lower and upper estimates of the values of the optimization criterion for the composition of packages for subsets formed in the branching procedure are constructed. The dropout procedure involves the exclusion of subsets whose lower estimate is not less than the record. To find optimal solutions, a breadth-first search strategy is applied, which provides for the study of all subsets of solutions that include various packages of tasks of the same type obtained as a result of the procedure for splitting subsets of tasks that are not excluded from consideration after the implementation of the dropout procedure. The developed algorithms are implemented programmatically, which allowed to obtain the results of planning the execution of task packages in a multi-stage system, which are on average 30 % better than fixed packages.

Keywords: multi-stage system, task packages, method of branches and boundaries, heuristic rule, schedules.

Krotov Kirill — Ph.D., Assistant Professor, Information systems department Sevastopol State University. Research interests: modeling of processes in computing systems, optimal planning of processes in computing systems, hierarchical games. The number of publications — 110. krotov_k1@mail.ru; 33, St. University, 299053, Sevastopol, Russia; office phone: +7 (978) 730-38-19.

References

1. OgunB., Cigdem A.-U. Mathematical Models for a Batch Scheduling Problem to Minimize Earliness and Tardiness. Journal of Industrial Engineering and Management. JIEM, 2018. № 11(3). pp. 390-405.
2. Chaudhry I.A., Elbadawi I. A-Q., Usman M., Chugtai M.T. Minimising Total Flowtime in a No-Wait Flow Shop (NWFS) using Genetic Algorithms. Ingeniería e Investigación. 2018. Vol. 38. № 3. pp. 68-79.
3. Tan Y., Huangi W., Sun Y., Yue Y. Comparative Study of Different Approaches to Solve Batch Process Sheduling and Optimisation Problems. Proceedings of the 18th International Conference on Automation & Computing. Loughborough University. Leicestershire. UK. 2012. pp 424–444.
4. Krotov K.V. [The use of the apparatus of genetic algorithms in the formation of decisions on the composition of data batches in the two-level task of constructing complex schedules for their processing]. Avtomatizirovannye tekhnologii i proizvodstva. Mezhdunarodnyj nauchno-tekhnicheskij zhurnal. – Automated technologies and production. International scientific and technical journal. 2017. № 2 (16), pp. 23-34 (In Russ).
5. Li X.L., Wang Y. Scheduling Batch Processing Machine Using Max–Min Ant System Algorithm Improved by a Local Search Method. Mathematical Problems in Engineering. 2018. Vol. 2018.
6. Li Sh., Cheng T.C.E., Ng C.T., Yuan J. Single-machine batch scheduling with job processing time compatibility. Theoretical Computer Science. 2015. Vol. 583. pp. 57-66.
7. Jin M., Liu X., Luo W. Single-Machine Parallel-Batch Scheduling with Nonidentical Job Sizes and Rejection. Mathematics. 2020. Vol. 8.
8. Surjandari I., Rachman A., Purdianta, Dhini A. The batch scheduling model for dynamic multi-item, multi-level production in an assembly job shop with parallel machines. International Journal of Technology. 2015. № 1. pp. 84-96.
9. Joglekar G. Using Simulation for Scheduling and Rescheduling of Batch Processes. Processes. 2017. № 5.
10. Kovalev M.M. [Models and methods of calendar planning: a course of lectures]. Minsk: Izdatelslvo BGU. 2004. P. 63. (In Russ).
11. Morrison D.R., Jacobson Sh.H., Sauppe J.J., Sewell E.C. Branch-and-bound algorithms: A survey of recent advances in searching, branching and pruning. Discrete Optimization. 2016. № 19. pp. 79-102.
12. Dawd S.T., Ayvaz B. A branch and bound approach for single machine scheduling problem. Istanbul Commerce University. Journal of Science. 2017. № 16 (31). pp. 43-55.
13. Rasti-Barzoki M., Hejazi S.R. A branch and bound algorithm to minimize the total weighted number of tardy jobs and delivery costs with late deliveries for a supply chain-scheduling problem. Journal of Industrial and Systems Engineering. 2017. Vol. 10. № 1. pp 50- 60.
14. Prilutsky M.H., Vlasov V.S. [The method of branches and boundaries with heuristic estimates for the pipeline problem of the theory of schedules.] Matematicheskoe modelirovanie. Optimal'noe upravlenie. Vestnik Nizhegorodskogo universiteta im. N.I. Lobachevskogo. [Mathematical modeling. Optimal control. Bulletin of the Nizhny Novgorod University named after N. I. Lobachevsky]. 2008. № 3. pp. 147-153 (In Russ).
15. Takano M.I., Nagano M.S. A branch-and bound method to minimize the makespan in a permutation flow shop with blocking and setup times. Cogent Engineering, 2017. 4:1. 1389638. DOI: 10.1080/23311916.2017.1389638.

16. Grigorieva N.S. [Algorithm of branches and boundaries for the task of scheduling on parallel processors]. Vestnik Sankt-Peterburgskogo universiteta, seriya 10.–Vestnik of St. Petersburg University, series 10, 2009, issue 1, pp. 44-55(In Russ).
17. Mazda Ch.N., Kurniawati D.A. Branch and Bound Method to Overcome Delay Delivery Order in Flow Shop Scheduling Problem. IOP Conf. Series: Materials Science and Engineering. 2020. IOP Publishing. DOI:10.1088/1757-899X/1003/1/012129.
18. Watermeyer K., Zimmermann J. A branch-and-bound procedure for the resource-constrained project scheduling problem with partially renewable resources and general temporal constraints. OR Spectrum. 2020. № 42. p. 427–460, DOI: 10.1007/s00291-020-00583-z.
19. Hu Sh., Wang S., Kao Y., Ito T., Sun X. A Branch and Bound Algorithm for Project Scheduling Problem with Spatial Resource Constraints. Hindawi Publishing Corporation, Mathematical Problems in Engineering. Volume 2015. Article ID 628259. 9 pages. DOI: 10.1155/2015/628259.
20. Mogilev A.A. [Review of methods for solving problems in the theory of schedules]. Informatika, vychislitel'naya tekhnika i inzhenernoe obrazovanie. [Informatics, Computer Engineering and Engineering education]. 2019. №. 4 (37), pp. 19-32 (In Russ).
21. Кротов К.В. Информационная модель многоуровневой системы выполнения конвейеризированных программ // Международный научно-технический журнал «Проблемы управления и информатики». 2014. № 3. 89-101 с.
22. Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B. Sequencing and Scheduling: Algorithms and Complexity. Handbook in Operations Research and Managment Science. North-Holland, Amsterdam. 1993. Vol. 4. pp. 445-522.
23. Krotov K.V. [A complex method for determining effective solutions for the composition of data batches and schedules of their processing in conveyor systems]. Zhurnal «Vychislitel’nye tekhnologii», Izd-vo Instituta vychislitel’nyh tekhnologij SO RAN. [Computational technologies, Publishing house Of the Institute of computational technologies SB RAS]. Vol. 23. № 3. 2018. pp. 58-76 (In Russ).
24. Krotov K.V., Skatkov A.V. [Organization of a web-based environmental monitoring service using Earth remote sensing data and data processing pipelining]. Trudy uchebnyh zavedenij svyazi. [Proceedings of educational institutions of Communications]. 2021. Vol. 7. No. 1. pp. 105-121 (In Russ).