УДК 519.854.2

ЭФФЕКТИВНЫЙ ТОЧНЫЙ ПДС-АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ О СУММАРНОМ ЗАПАЗДЫВАНИИ ДЛЯ ОДНОГО ПРИБОРА

А.А. ПАВЛОВ, Е.Б. МИСЮРА

Предложен эффективный точный алгоритм решения задачи о суммарном запаздывании при выполнении независимых заданий с директивными сроками одним прибором. Алгоритм основан на новом подходе к решению задач с директивными сроками и заключается в оптимальном использовании резервов времени незапаздывающих заданий. Его эффективность качественно превышает эффективность известных алгоритмов.

ВВЕДЕНИЕ

Рассматривается модель задачи о суммарном запаздывании для одного прибора в предположении, что известно множество независимых заданий $J = \{j_1, j_2, ..., j_n\}$, каждое из которых состоит из одной операции. При этом также известны длительность l_j и директивный срок выполнения D_j. Задания поступают в систему одновременно в момент времени $d_j = 0, j = \overline{1,n}$. Прерывания не допускаются. Необходимо построить расписание выполнения заданий для одного прибора, минимизирующее суммарное опоздание при их выполнении,

$$f = \sum_{j=1}^{n} \max\left(0, C_j - D_j\right),$$

где C_j — момент завершения выполнения задания j.

Обзор методов решения этой задачи приведен в работе [1].

Согласно Ду и Люнгу [2], задача является NP-сложной. Эммонс [3] разработал правила доминантности, которые были применены в алгоритмах ветвей и границ [4 – 6] и в алгоритмах динамического программирования [4, 7 – 10]. Алгоритм динамического программирования Шрейга и Бейкера [9] находится в категории второго поколения, так как он решает задачи с количеством заданий до 50 для $1 \le l_k \le 10$. Лоулер [11] обнаружил, что задача может быть декомпозирована на подзадачи путем определения позиции работы с наибольшим временем выполнения в оптимальной последовательности. Затем он представил правило для уменьшения списка возможных позиций. Этот список в дальнейшем укорачивается правилами Поттса и Ван Вассенгова [7], Шварца [12]. Последние алгоритмы третьего поколения Поттса и Ван Вассенгова [7, 8] комбинируют результат Лоулера [11]: сначала разбить задачу на подзадачи допустимой размерности, а затем решить каждую подзадачу динамическим алгоритмом Шрейга и Вейкера [9]. Они были способны решить задачи с количеством заданий до 100 при $1 \le l_k \le 100$. Поттс и Ван Вассенгов [8, с. 413] указывают на то, что для решения задач с более чем 100 заданиями необходим *новый* подход.

Для решения данной задачи созданы конструктивные и декомпозиционные эвристики. Наиболее известные из них исследуются в работе [13], где показано, что все они представляют очень плохие нижние границы коэффициентов аппроксимации: их нижние границы, по меньшей мере, линейно зависят от размерности задачи. В статье [1] рассматривается алгоритм четвертого поколения, который эффективно решает задачи с количеством заданий до 150. Самый современный точный метод Щварца, Гроссо и Кросе [14] может решать задачи с размерностью до 500 работ.

Мы предлагаем эффективный точный ПДС-алгоритм (алгоритм с полиномиальной и экспоненциальной составляющими) решения задачи, основанный на новом подходе к решению задач с директивными сроками, который заключается в оптимальном использовании резервов времени незапаздывающих заданий, позволяющий решать задачи с числом заданий, существенно большим 500.

Идея нашего подхода заключается в исследовании свойств рассматриваемых классов труднорешаемых задач, доказательстве положений, правил, позволяющих разработать единый принцип вычислений, и на их основе построении ПДС-алгоритмов [15, 16]. Эти алгоритмы основаны на выделении полиномиальной составляющей алгоритма и получении условий, когда экспоненциальная составляющая может декомпозировать начальную задачу на подзадачи меньшего размера. Для каждого такого алгоритма выводятся логико-аналитические условия (*p-условия*), которые проверяются в процессе решения произвольной индивидуальной труднорешаемой задачи, и в случае их выполнения данная задача точно решается полиномиальным подалгоритмом. Экспоненциальный подалгоритм содержит строго определенные логико-аналитические условия (*d-условия*). При их выполнении в процессе решения задача строго декомпозируется на подзадачи меньшего размера.

Таким образом, новым в предложенном подходе является выделение полиномиально разрешимых подклассов труднорешаемых комбинаторных задач не путем введения соответствующих ограничений в постановки задач, а на основе анализа процесса решения произвольных индивидуальных задач.

Качественное отличие разработанных алгоритмов от существующих — статистическая значимость (высокая частота получения) точных решений задач большой размерности за приемлемое время, что открывает широкие возможности для решения реальных практических труднорешаемых задач.

ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ

В статье [17] изложен новый подход к решению задачи «Минимизация суммарного взвешенного опоздания при выполнении независимых заданий с директивными сроками одним прибором» и приведен алгоритм, который позволил получить точные решения для задач с числом переменных, существенно большим 50. Рассмотрим случай, когда веса для всех заданий равны 1.

Введем следующие определения [17].

Определение 1. Резервом времени $r_{j_{[g]}}$ задания $j_{[g]}$ называется величина $r_{j_{[g]}} = D_{j_{[g]}} - C_{j_{[g]}}$.

Определение 2. Перестановкой называется процедура одновременного переноса задания $j_{[g]}$ на позицию k ($k > g$) и заданий, занимающих позиции $g+1, g+2, ..., k-1, k$ соответственно на позиции $g, g+1, ..., k-1$.

Определение 3. Интервалом перестановки задания $j_{[g]}$ на позицию k называется множество заданий, находящихся на позициях $g+1, g+2, ...$ $..., k-1, k$ до выполнения перестановки.

Определение 4. Встраиванием называется процедура одновременного переноса задания $j_{[g]}$ на позицию p ($g > p$) и заданий $p, p+1, ..., g-1$ соответственно на позиции $p+1, p+2, ..., g-1, g$.

Определение 5. Интервалом встраивания $I_{j_{[g]}}$ задания $j_{[g]}$ называется множество заданий, стоящих до встраивания на позициях $p, p+1, ..., g-1$, где p определяется из условия

$$\sum_{i=p-1}^{g-1} l_{j_{[i]}} < C_{j_{[g]}} - D_{j_{[g]}} \le \sum_{i=p}^{g-1} l_{j_{[i]}} . \qquad (1)$$

Если же условие (1) не выполняется ни для одной позиции, то $p = 1$. Таким образом, опоздание по заданию j на позиции p должно быть равно нулю или минимально.

Определение 6. Задание $j_{[g]}$ называется запаздывающим в последовательности σ, если для него выполняется условие $D_{j_{[g]}} < C_{j_{[g]}}$, где g — позиция, которую задание j занимает в последовательности σ.

Определение 7. Последовательностью $\sigma^{\text{уп}}$ (сигма упорядоченная) называется последовательность заданий множества J, $j = \overline{1, n}$, в которой задания упорядочены по неубыванию длительностей l_j, т.е. $\forall j, i : l_j \le l_i$, $j < i$.

Справедливы следующие утверждения [17].

Утверждение 1. Если в последовательности $\sigma^{\text{уп}}$ запаздывающим заданиям не предшествуют задания с резервом времени, то не существует переносов заданий, приводящих к улучшению целевой функции.

Доказательство. Пусть в последовательности $\sigma^{\text{уп}}$ для заданий, стоящих на позициях $i=\overline{1,s}$, выполняется $C_{j_{[i]}}-D_{j_{[i]}}\geq 0$, а для заданий, занимающих позиции $i=\overline{s+1,n}$, выполняется $C_{j_{[i]}}-D_{j_{[i]}}\leq 0$. Очевидно, что переносы, которые могли бы привести к уменьшению целевой функции, возможны только для заданий, занимающих позиции $i=\overline{1,s}$.

Докажем, что не существует перестановок, приводящих к уменьшению f. Пусть в последовательности σ позиции $l,g\in\overline{1,s},\ l<g$. Выполним перестановку задания $j_{[l]}$ на позицию [g]. Значение целевой функции на интервале $\overline{l,g}$ до перестановки

$$f=\sum_{i=l}^{g}\left(C_{j_{[i]}}-D_{j_{[i]}}\right).$$

После перестановки с учетом предположения, что задания, занимавшие позиции $l+1,l+2,...,g-1,g$, перестали запаздывать (т.е. пусть до перестановки выполнялось $C_{j_{[i]}}-D_{j_{[i]}}\leq l_{j_{[l]}}$, $i=\overline{l+1,g}$)

$$f'=C_{j_{[l]}}+\sum_{i=l+1}^{g}l_{j_{[i]}}-D_{j_{[l]}},$$

где $C_{j_{[l]}}$ — момент окончания выполнения задания $j_{[l]}$ до перестановки; $C_{j_{[l]}}+\sum_{i=l+1}^{g}l_{j_{[i]}}$ — момент окончания выполнения заданий $j_{[l]}$ после перестановки.

Необходимо доказать, что $f-f'\leq 0$.

$$f-f'=C_{j_{[l]}}-D_{j_{[l]}}+\sum_{i=l+1}^{g}\left(C_{j_{[i]}}-D_{j_{[i]}}\right)-\left(C_{j_{[l]}}+\sum_{i=l+1}^{g}l_{j_{[i]}}-D_{j_{[l]}}\right)=$$

$$=\sum_{i=l+1}^{g}\left(C_{j_{[i]}}-D_{j_{[i]}}\right)-\sum_{i=l+1}^{g}l_{j_{[i]}}\leq\sum_{i=l+1}^{g}\left(l_{j_{[l]}}-l_{j_{[i]}}\right)\leq 0$$

в силу того, что $l_{j_{[l]}}\leq l_{j_{[i]}}$, $i=\overline{l+1,g}$.

Докажем, что не существует встраиваний, приводящих к увеличению f. Значение целевой функции до встраивания

$$f=\sum_{i=p}^{g}\left(C_{j_{[i]}}-D_{j_{[i]}}\right),\ g,p\in[1,s].$$

После встраивания задания $j_{[g]}$ на позицию p с учетом того, что $j_{[g]}$ перестанет запаздывать, т.е. $C_{j_{[g]}}-D_{j_{[g]}}\leq\sum_{i=p}^{g-1}l_{j_{[i]}}$, значение целевой функции

$$f' = \sum_{i=p}^{g-1} \left(C_{j_{[i]}} + l_{j_{[g]}} - D_{j_{[i]}} \right),$$

где $C_{j_{[i]}}$ — момент окончания выполнения заданий $j_{[i]}$ до выполнения процедуры встраивания задания $j_{[g]}$ $\forall i, i = \overline{p, g-1}$; $C_{j_{[i]}} + l_{j_{[g]}}$ — момент окончания выполнения заданий $j_{[i]}$ после выполнения процедуры встраивания $j_{[g]}$.

Необходимо доказать, что $f - f' \le 0$.

$$f - f' = C_{j_{[g]}} - D_{j_{[g]}} + \sum_{i=p}^{g-1} \left(C_{j_{[i]}} - D_{j_{[i]}} \right) - \sum_{i=p}^{g-1} \left(C_{j_{[i]}} + l_{j_{[g]}} - D_{j_{[i]}} \right) \le$$

$$\le \sum_{i=p}^{g-1} \left(l_{j_{[i]}} - l_{j_{[g]}} \right) \le 0$$

в силу того, что $C_{j_{[g]}} - D_{j_{[g]}} \le \sum_{i=p}^{g-1} l_{j_{[i]}}$, $l_{j_{[g]}} \ge l_{j_{[i]}}$ $\forall i = \overline{p, g-1}$. ■

Утверждение 2. Встраивание запаздывающего задания $j_{[g]}$ на позицию $f < p$ не может привести к улучшению целевой функции.

Доказательство. В соответствии с определением 5, на позиции p опоздание по заданию $j_{[g]}$ становится равным нулю. Встраивание задания $j_{[g]}$ на любую позицию $f < p$ приводит к увеличению его резерва и перемещает задания, стоящие на позициях $f, f+1, ..., p-1$, на позиции $f+1, f+2, ..., p$. Такое перемещение приводит к увеличению целевой функции, если для какого-либо задания на $\overline{f, p-1}$ выполняется хотя бы одно из условий:

1) для $j_{[i]}$, $i \in \overline{f, p-1}$ выполняется $C_{j_{[i]}} \ge D_{j_{[i]}}$, т.е. задание $j_{[i]}$ является запаздывающим, и в результате встраивания опоздание по нему увеличивается на величину $l_{j_{[g]}}$;

2) для $j_{[i]}$, $i \in \overline{f, p-1}$ выполняется $C_{j_{[i]}} < D_{j_{[i]}}$ и $D_{j_{[i]}} - C_{j_{[i]}} < l_{j_{[g]}}$, т.е. $j_{[i]}$ до встраивания задания $j_{[g]}$ не запаздывало, но в результате встраивания станет запаздывать на величину $C_{j_{[i]}} - D_{j_{[i]}} + l_{j_{[g]}}$.

Если ни для одного из заданий $j_{[i]}$, $i \in \overline{f, p-1}$ не выполняется ни одно из перечисленных условий, то целевая функция не ухудшается, но и не улучшается в результате встраивания $j_{[g]}$ на позицию $f < p$, так как опоздание на позиции p по заданию $j_{[g]}$ равно нулю. ■

Утверждение 3. Пусть в последовательности $\sigma^{\text{уп}}$ $\forall j_{[i]}$, $i = \overline{1, p-1}$, $r_{j_{[i]}} \le 0$. Обозначим ее $\sigma^{\text{уп1}}$. Запаздывающее задание $j_{[g]}$ в последовательности $\sigma^{\text{уп1}}$ в результате выполнения встраивания ($I_{j[g]} = \overline{p, g-1}$) может занять более раннюю позицию, что улучшит целевую функцию только в том случае, если в последовательности $\sigma^{\text{уп1}}$ хотя бы у одного из заданий на интервале встраивания $I_{j[g]}$ есть резерв времени, больший нуля.

Доказательство. Выведем условия, при выполнении которых в результате процедуры встраивания значение целевой функции уменьшается, т.е. выполняется $f - f' > 0$.

Пусть для $j_{[z]}$ ($p \le z < g$) выполняется $D_{j_{[z]}} > C_{j_{[z]}}$, для остальных заданий $i \in I_{j_{[g]}}$ $D_{j_{[i]}} \le C_{j_{[i]}}$. Согласно утверждениям 1 и 2, встраивание, которое может улучшить f, должно быть выполнено на позицию z. Встроим задание $j_{[g]}$ на позицию z.

$$f = \sum_{i=p}^{z-1}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) + \sum_{i=z+1}^{g}\left(C_{j_{[i]}} - D_{j_{[i]}}\right),$$

$$f' = \max\left(0, C_{j_{[g]}} - \sum_{i=z}^{g-1} l_{j_{[i]}} - D_{j_{[g]}}\right) + \sum_{i=p}^{z-1}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) +$$

$$+ \sum_{i=z+1}^{g-1}\left(C_{j_{[i]}} + l_{j_{[g]}} - D_{j_{[i]}}\right) + \max\left(0, C_{j_{[z]}} + l_{j_{[g]}} - D_{j_{[z]}}\right),$$

$$f - f' = \sum_{i=p}^{z-1}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) + \sum_{i=z+1}^{g}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) - \sum_{i=p}^{z-1}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) -$$

$$- \sum_{i=z+1}^{g-1}\left(C_{j_{[i]}} - D_{j_{[i]}}\right) - (g - z - 1) l_{j_{[g]}} - \max\left(0, C_{j_{[z]}} + l_{j_{[g]}} - D_{j_{[z]}}\right) -$$

$$- \max\left(0, C_{j_{[g]}} - \sum_{i=z}^{g-1} l_{j_{[i]}} - D_{j_{[g]}}\right) = -(g - z - 1) l_{j_{[g]}} + \left(C_{j_{[g]}} - D_{j_{[g]}}\right) -$$

$$- \max\left(0, C_{j_{[g]}} - \sum_{i=z}^{g-1} l_{j_{[i]}} - D_{j_{[g]}}\right) - \max\left(0, C_{j_{[z]}} + l_{j_{[g]}} - D_{j_{[z]}}\right) =$$

$$= \min\left(C_{j_{[g]}} - D_{j_{[g]}}, \sum_{i=z}^{g-1} l_{j_{[i]}}\right) - (g - z - 1) l_{j_{[g]}} - \max\left(0, l_{j_{[g]}} - r_{j_{[z]}}\right). \quad (2)$$

Заметим: если существует несколько заданий $z_k \in Z$ с резервами времени на $\overline{p, g-1}$, то выражение (2) при встраивании $j_{[g]}$ на одну из позиций z_k примет вид

$$f - f' = \min\left(C_{j_{[g]}} - D_{j_{[g]}}, \sum_{i=z_k}^{g-1} l_{j_{[i]}}\right) - \sum_{i=z_k}^{g-1} \max\left[0, \min\left(l_{j_{[g]}} - r_{j_{[i]}}, l_{j_{[g]}}\right)\right], \quad (3)$$

где $\min\left(C_{j_{[g]}} - D_{j_{[g]}}, \sum_{i=z}^{g-1} l_{j_{[i]}} \right)$ — уменьшение опоздания по заданию $j_{[g]}$ в результате встраивания на позицию z. Определяется величиной опоздания и длиной интервала встраивания и не зависит от длительности выполнения самого задания; $\sum_{i=z}^{g-1} \max\left[0, \min\left(l_{j_{[g]}} - r_{j_{[i]}}, l_{j_{[g]}}\right)\right]$ — увеличение опоздания заданий на интервале встраивания в результате выполнения процедуры встраивания $j_{[g]}$ на позицию z. Определяется наличием резерва на этом интервале.

Следовательно, улучшение целевой функции возможно только в том случае, если на интервале встраивания $I_{j_{[g]}}$ есть резерв, больший нуля. В случае отсутствия резервов выражение (3) принимает отрицательное значение, так как длительность выполнения $j_{[g]}$ больше длительности заданий на интервале встраивания. ■

Пусть в последовательности $\sigma^{\text{уп1}}$ $j_{[g]}$ — первое запаздывающее задание. За ним следуют $j_{[i]}$, $i = \overline{g+1, n}$, $D_{j_{[i]}} - C_{j_{[i]}} \le 0$. Для $j_{[z]}$, $p < z < g$ выполняется $D_{j_{[z]}} > C_{j_{[z]}}$. Для остальных заданий $j_{[k]} \in I_{j_{[g]}}$, $D_{j_{[k]}} \le C_{j_{[k]}}$. Обозначим эту последовательность $\sigma^{\text{уп2}}$.

Утверждение 4. Если в последовательности $\sigma^{\text{уп2}}$ в результате процедуры встраивания $j_{[g]}$ на позицию z значение функционала уменьшилось, и для всех $j_{[k]} \in I_{j_{[g]}}$ станет справедливо $r_{j_{[k]}} \le 0$, то полученная последовательность оптимальна.

Доказательство. Для всех $j_{[s]}, s = \overline{1, p-1}, r_{j_{[s]}} \le 0$ по определению последовательности $\sigma^{\text{уп1}}$. Для всех $j_{[i]}$, $i = \overline{g+1, n}$ выполняется $l_{j_{[i]}} \ge l_{j_{[g]}}$. После встраивания $j_{[g]}$ на позицию z справедливо $\forall\, j_{[k]} \in I_{j_{[g]}}\ r_{j_{[k]}} \le 0$, следовательно, встраивание задания $j_{[i]}$, $i = \overline{g+1, n}$ на позицию z приведет, в соответствии с (3), к увеличению потерь по заданиям, смещенным на более поздние позиции в результате выполнения процедуры встраивания и, таким образом, к ухудшению значения функционала. ■

Утверждение 5. Если в последовательности $\sigma^{\text{уп2}}$ при проверке возможности встраивания задания $j_{[g]}$ на позицию z значение функционала ухудшится, то не существует перестановок и встраиваний, приводящих к уменьшению значения функционала, и последовательность $\sigma^{\text{уп2}}$ оптимальна.

Доказательство. В результате выполнения встраивания задания $j_{[g]}$ на позицию z справедливо $f - f' < 0$. Таким образом, существующих резервов на позиции z недостаточно для выполнения встраивания $j_{[g]}$. Но

$\forall\, j_{[i]}$, $i = \overline{g+1, n}$, $l_{j_{[i]}} \ge l_{j_{[g]}}$. Следовательно, не существует перестановок и встраиваний, приводящих к уменьшению значения функционала. ■

Утверждение 6. Если в последовательности $\sigma^{\text{уп}}$ ни для одного из запаздывающих заданий $j_{[g]}$ нет предшествующих заданий $j_{[i]}$, $i = \overline{1, g-1}$, для которых выполняется

$$D_{j_{[i]}} - C_{j_{[i]}} > 0, \; D_{j_{[i]}} > D_{j_{[g]}},$$

то не существует перестановок и встраиваний, приводящих к улучшению целевой функции.

Доказательство. *Встраивание.* Задание $j_{[g]}$ встроим на позицию p (см. определение 5). Учитывая условие утверждения, что не существует предшествующих заданий $j_{[s]}$, для которых $D_{j_{[s]}} - C_{j_{[s]}} > 0$ и $D_{j_{[s]}} > D_{j_{[g]}}$, получаем, что встраивание $j_{[g]}$ произойдет на интервале $\overline{p, g-1}$, на котором для любого задания $j_{[i]}$ выполняется

$$C_{j_{[i]}} - D_{j_{[i]}} \ge 0, \quad t = \overline{p, g-1}. \tag{4}$$

Пусть в выражении (4) $C_{j_{[i]}} - D_{j_{[i]}} = 0$, $t = \overline{p, g-1}$. Тогда

$$f = C_{j_{[g]}} - D_{j_{[g]}}, \quad f' = \sum_{i=p}^{g-1} \left(C_{j_{[i]}} + l_{j_{[g]}} - D_{j_{[i]}} \right),$$

$$f - f' = C_{j_{[g]}} - D_{j_{[g]}} - \sum_{i=p}^{g-1} \left(C_{j_{[i]}} + l_{j_{[g]}} - D_{j_{[i]}} \right) \le \sum_{i=p}^{g-1} \left(l_{j_{[i]}} - l_{j_{[g]}} \right) \le 0,$$

так как $l_{j_{[i]}} \le l_{j_{[g]}} \quad \forall i < g$.

Доказательство для случая $C_{j_{[i]}} - D_{j_{[i]}} > 0$ аналогично.

Перестановки. Возможны перестановки задания с нулевым и ненулевым резервами времени. Рассмотрим все возможные случаи.

1. Пусть $D_{j_{[s]}} - C_{j_{[s]}} > 0$, $D_{j_{[s]}} \le D_{j_{[g]}}$ и задание $j_{[g]}$ после перестановки запаздывает, т.е. $C_{j_{[g]}} - l_{j_{[s]}} > D_{j_{[g]}}$. В этом случае $f = C_{j_{[g]}} - D_{j_{[g]}}$,

$$f' = \left(C_{j_{[g]}} - l_{j_{[s]}} > D_{j_{[g]}} \right) + \left(C_{j_{[g]}} - D_{j_{[s]}} \right),$$

$$f - f' = \left(C_{j_{[g]}} - D_{j_{[g]}} \right) - \left(C_{j_{[g]}} - l_{j_{[s]}} - D_{j_{[g]}} \right) - \left(C_{j_{[g]}} - D_{j_{[s]}} \right) =$$

$$= l_{j_{[s]}} - \left(C_{j_{[g]}} - D_{j_{[s]}} \right) \le l_{j_{[s]}} - \left(C_{j_{[g]}} - D_{j_{[g]}} \right) \le 0, \text{ так как } C_{j_{[g]}} - D_{j_{[g]}} > l_{j_{[s]}}.$$

2. Пусть $D_{j_{[s]}} - C_{j_{[s]}} > 0$, $D_{j_{[s]}} \le D_{j_{[g]}}$ и задание $j_{[g]}$ после перестановки не запаздывает, т.е. $C_{j_{[g]}} - D_{j_{[g]}} \le l_{j_{[s]}}$.

$$f' = C_{j_{[g]}} - D_{j_{[s]}}.$$

$$f - f' = \left(C_{j_{[g]}} - D_{j_{[g]}}\right) - \left(C_{j_{[g]}} - D_{j_{[s]}}\right) \le 0, \text{ так как } D_{j_{[s]}} \le D_{j_{[g]}}.$$

3. Пусть $D_{j_{[s]}} - C_{j_{[s]}} = 0$, $D_{j_{[s]}} \le D_{j_{[g]}}$ и задание $j_{[g]}$ после перестановки запаздывает, тогда

$$f - f' = C_{j_{[g]}} - D_{j_{[g]}} - \left(C_{j_{[g]}} - l_{j_{[s]}} - D_{j_{[g]}}\right) - \sum_{i=s+1}^{g} l_{j_{[i]}} = l_{j[s]} - \sum_{i=s+1}^{g} l_{j_{[i]}} =$$

$$= l_{j_{[s]}} - l_{j_{[g]}} - \sum_{i=s+1}^{g-1} l_{j_{[i]}} < l_{j_{[s]}} - l_{j_{[g]}} \le 0, \text{ так как } l_{j_{[s]}} \le l_{j_{[g]}}, \quad s < g.$$

4. Пусть $D_{j_{[s]}} - C_{j_{[s]}} = 0$, $D_{j_{[s]}} \le D_{j_{[g]}}$ и задание $j_{[g]}$ после перестановки не запаздывает, т.е. $C_{j_{[g]}} - D_{j_{[g]}} \le l_{j_{[s]}}$. В этом случае

$$f - f' = C_{j_{[g]}} - D_{j_{[g]}} - \sum_{i=s+1}^{g} l_{j_{[i]}} \le l_{j_{[s]}} - \sum_{i=s+1}^{g} l_{j_{[i]}} \le 0$$

аналогично п. 3.

Следовательно, так как для запаздывающего задания $j_{[g]}$ нет предшествующих заданий, имеющих резервы на интервале переноса $j_{[g]}$, то не существует перестановок, приводящих к уменьшению значения функционала. ■

Следствие 1. Если в последовательности $\sigma^{\text{уп}}$ на интервале встраивания запаздывающего задания $j_{[g]}$, определяемом позициями $\overline{p, g-1}$, резервы отсутствуют, то $j_{[g]}$ сможет занять более раннюю позицию, что приведет к уменьшению значения функционала только в том случае, если в последовательности $\sigma^{\text{уп}}$ на интервале $\overline{1, p-1}$ есть задания $j_{[k]}$, для которых справедливо $D_{j_{[k]}} - C_{j_{[k]}} > 0$, $D_{j_{[k]}} > D_{j_{[g]}}$. В этом случае резервы на интервале встраивания $j_{[g]}$ могут быть образованы в результате перестановок заданий $j_{[k]}$ на более поздние позиции.

Следствие 2. Если в последовательности $\sigma^{\text{уп}}$ для каждого $j_{[k]}$, принадлежащего множеству R заданий с резервами, и каждого $j_{[g]}$, принадлежащего множеству Y запаздывающих заданий, выполняется $l_{j_{[k]}} \le l_{j_{[g]}}$, $D_{j_{[k]}} \le D_{j_{[g]}}$, то не существует перестановок и встраиваний, приводящих к уменьшению значения функционала.

Утверждение 7. Если в последовательности $\sigma^{\text{уп}}$ для запаздывающего задания $j_{[g]}$ существует предшествующее $j_{[k]}$ такое, что выполняются условия

$$D_{j_{[k]}} > C_{j_{[g]}} - l_{j_{[g]}}, \quad \min\left(l_{j_{[k]}}, C_{j_{[g]}} - D_{j_{[g]}}\right) > C_{j_{[g]}} - D_{j_{[k]}},$$

то перестановка $j_{[k]}$ на позицию g приводит к уменьшению целевой функции.

Доказательство. Для доказательства утверждения выведем условия перестановки, при выполнении которых уменьшается целевая функция, т.е. $f - f' > 0$.

$$f = \sum_{i=k}^{g} \max\left(0, C_{j_{[i]}} - D_{j_{[i]}}\right).$$

После перестановки $j_{[k]}$ на позицию g

$$f' = \max\left(0, C_{j_{[g]}} - D_{j_{[k]}}\right) + \sum_{i=k+1}^{g} \max\left(0, C_{j_{[i]}} - l_{j_{[k]}} - D_{j_{[i]}}\right),$$

$$f - f' = \sum_{i=k}^{g} \max\left(0, C_{j_{[i]}} - D_{j_{[i]}}\right) - \left(C_{j_{[g]}} - D_{j_{[k]}}\right) -$$

$$- \sum_{i=k+1}^{g} \max\left(0, C_{j_{[i]}} - l_{j_{[k]}} - D_{j_{[i]}}\right) = -\left(C_{j_{[g]}} - D_{j_{[k]}}\right) +$$

$$+ \sum_{i=k+1}^{g} \max\left(0, C_{j_{[i]}} - D_{j_{[i]}}\right) - \sum_{i=k+1}^{g} \max\left(0, C_{j_{[i]}} - l_{j_{[k]}} - D_{j_{[i]}}\right).$$

Пусть все задания $i = \overline{k+1, g-1}$ не запаздывают до переноса, т.е. $C_{j_{[i]}} - D_{j_{[i]}} < 0$. Тогда

$$f - f' = -\left(C_{j_{[g]}} - D_{j_{[k]}}\right) + \left(C_{j_{[g]}} - D_{j_{[g]}}\right) - \max\left(0, C_{j_{[g]}} - l_{j_{[k]}} - D_{j_{[g]}}\right) =$$

$$= -\left(C_{j_{[g]}} - D_{j_{[k]}}\right) + \min\left(l_{j_{[k]}}, C_{j_{[g]}} - D_{j_{[g]}}\right) > 0 .$$

Таким образом, если выполняется условие

$$\min\left(l_{j_{[k]}}, C_{j_{[g]}} - D_{j_{[g]}}\right) > C_{j_{[g]}} - D_{j_{[k]}},$$

то при перестановке $j_{[k]}$ на позицию g происходит уменьшение целевой функции. ■

На основании доказанных утверждений сформулируем необходимые условия для перемещения запаздывающих заданий на более ранние позиции, при котором значение функционала уменьшается путем использования существующих резервов.

Утверждение 8. Пусть в последовательности $\sigma^{\text{уп}}$ $j_{[g]}$ — запаздывающее задание. Уменьшение значения функционала при перемещении $j_{[g]}$ на более ранние позиции в результате перестановок и встраиваний возможно только при выполнении хотя бы одного из следующих условий:

1. $\exists\, j_{[i]}$, $P_{\min} \le i \le g \,|\, r_{j[i]} > 0$, $D_{j_{[i]}} > D_{j_{[g]}}$. На интервале встраивания $j_{[g]}$ есть задания $j_{[i]}$ с резервами времени, для которых $D_{j_{[i]}} > D_{j_{[g]}}$, $P_{\min}$ — позиция, где запаздывание по $j_{[g]}$ минимально (или равно нулю).

2. $\exists\, j_{[q]}$, $q < g$, $D_{j_{[q]}} > C_{j_{[g]}}$. В последовательности $\sigma^{\text{уп}}$ на позиции q, предшествующей позиции g, есть задание с резервом времени, перестановка которого после $j_{[g]}$ уменьшает опоздание по $j_{[g]}$. Задание $j_{[q]}$ остается незапаздывающим.

3. $\exists\, j_{[q]}$, $q < g$, $C_{j_{[g]}} - l_{j_{[g]}} < D_{j_{[q]}} < C_{j_{[g]}}$. Существует незапаздывающее задание $j_{[q]}$, директивный срок которого больше момента начала выполнения $j_{[g]}$, но меньше момента окончания $j_{[g]}$. При этом

$$\min\left(C_{j_{[g]}} - D_{j_{[g]}}, l_{j_{[q]}}\right) - \left(C_{j_{[g]}} - D_{j_{[q]}}\right) > 0.$$

Следовательно, перестановка задания $j_{[q]}$ после $j_{[g]}$ приведет к уменьшению значения функционала за счет использования резерва $j_{[q]}$.

4. $\forall i \,|\, P_{\min} \le i \le g \,|\, r_{j_{[i]}} \le 0$, но $\exists\, j_{[k]}$, $k < P_{\min} \,|\, D_{j_{[k]}} > D_{j_{[g]}}$, $r_{j_{[k]}} > 0$. На интервале перестановки задания $j_{[g]}$ резервы отсутствуют, но существует задание $j_{[k]}$, $k < P_{\min}$, директивный срок которого больше $D_{j_{[g]}}$ и резерв больше нуля.

Выполнение одного из условий $1-4$ означает, что на интервале встраивания задания $j_{[g]}$ существуют резервы (условия 1–3) или они будут образованы в результате перестановок (условие 4). При невыполнении условия 4 перестановка заданий, занимающих позиции $\overline{1, P_{\min} - 1}$, на интервал встраивания задания $j_{[g]}$ приведет к запаздыванию этих заданий, и так как они имеют меньшую длительность, чем $j_{[g]}$, то встраивание задания $j_{[g]}$ приведет к увеличению значения функционала. Следовательно, если не выполняется ни одно из условий $1-4$, то не существует встраиваний задания $j_{[g]}$ на более ранние позиции, улучшающих целевую функцию. ■

Следствие. Пусть в последовательности $\sigma^{\text{уп}}$ число запаздывающих заданий $n_{\text{з}} > 1$. Тогда, если в последовательности $\sigma^{\text{уп}}$ ни для одного из запаздывающих заданий $j_{[g]}$ нет предшествующих заданий $j_{[s]}$, $s<g$, для которых выполняется хотя бы одно из условий $1-4$, то последовательности $\sigma^{\text{уп}}$ отвечает оптимальное значение функционала.

Пусть в последовательности $\sigma^{\text{уп}}=\{j_{[1]},j_{[2]},\dots,j_{[n]}\}$ запаздывающее задание $j_{[g]}$ в результате встраивания заняло позицию $P_{\min}=m$. Пометим его $j^*_{[m]}$, а полученную последовательность — $\sigma(j_{[g]})$.

Утверждение 9. Запаздывающее задание $j_{[k]}$, $k=\overline{m+1,g}$ в последовательности $\sigma(j_{[g]})$ может быть встроено на более раннюю позицию, что приведет к улучшению целевой функции только в том случае, если хотя бы у одного предшествующего задания есть резерв времени либо (в случае отсутствия резервов) только в результате перестановки $j^*_{[m]}$ после $j_{[k]}$.

Доказательство. Доказательство в случае наличия резервов приведено в утверждениях 3 и 8. Рассмотрим случай, когда в последовательности $\sigma(j_{[g]})$ нет резервов.

Очевидно, что среди заданий на интервале $\overline{1,k-1}$ перестановка на позицию k возможна только для задания $j^*_{[m]}$, так как $\forall\, i=\overline{1,k-1}$, $D_{j[i]}-C_{j[i]}\le 0$, а для всех непомеченных заданий $l_{j[i]}\le l_{j[k]}$, $i=\overline{1,k-1}$.

Выполним перестановку на позицию k задания $j^*_{[m]}$.

Значение целевой функции до перестановки

$$f=\sum_{i=m}^{k}\max\left(0,C_{j[i]}-D_{j[i]}\right)=\sum_{i=m}^{k}\left(C_{j[i]}-D_{j[i]}\right).$$

После переноса $j^*_{[m]}$ на позицию k

$$f'=C_{j[m]}+\sum_{i=m+1}^{k}l_{j[i]}-D_{j_{[m]}}+\sum_{i=m+1}^{k}\max\left(0,C_{j[i]}-D_{j[i]}-l_{j[m]}\right),$$

где $C_{j[m]}$ — момент окончания выполнения задания $j^*_{[m]}$ до перестановки.

Докажем, что $f-f'$ может быть больше нуля.

$$f-f'=C_{j[m]}-D_{j[m]}+\sum_{i=m+1}^{k}\left(C_{j[i]}-D_{j[i]}\right)-$$

$$-\left[C_{j[m]}+\sum_{i=m+1}^{k}l_{j[i]}-D_{j_{[m]}}+\sum_{i=m+1}^{k}\max\left(0,C_{j[i]}-D_{j[i]}-l_{j[m]}\right)\right]=$$

$$=C_{j[m]}-D_{j[m]}-C_{j[m]}-\sum_{i=m+1}^{k}l_{j[i]}+D_{j[m]}+$$

$$+\sum_{i=m+1}^{k}\Big[C_{j[i]}-D_{j[i]}-\max\Big(0, C_{j[i]}-D_{j[i]}-l_{j[m]}\Big)\Big]=$$

$$=\sum_{i=m+1}^{k}\Big[\min\Big(l_{j[m]}, C_{j[i]}-D_{j[i]}\Big)-l_{j[i]}\Big]. \qquad (5)$$

Проанализируем выражение (5). Разобьем сумму (5) на две части.

Пусть P — множество заданий на $\overline{m+1,k}$, которые после перестановки $j^*_{[m]}$ уже не запаздывают, следовательно,

$$P:\Big\{l_{j[m]} \ge C_{j[i]} - D_{j[i]}\Big\} \quad ;$$

Q — множество заданий на $\overline{m+1,k}$, которые после перестановки $j^*_{[m]}$ остались запаздывающими и, следовательно,

$$Q:\Big\{l_{j[m]} < C_{j[i]} - D_{j[i]}\Big\} \quad .$$

Выражение (5) примет вид

$$\sum_{j[i]\in P}\Big[C_{j[i]}-D_{j[i]}-l_{j[i]}\Big]+\sum_{j[i]\in Q}\Big[l_{j[m]}-l_{j[i]}\Big]. \qquad (6)$$

Так как справедливо $\forall\, j_{[i]}$, $i=\overline{m+1,k}$, $l_{j_{[i]}} \le l_{j^*_{[m]}}$, то полученное значение выражения (6) может быть больше нуля и перестановка задания $j^*_{[m]}$ на позицию k может привести к уменьшению f. ■

Утверждение 10. Пусть в последовательности $\sigma(j_{[g]})$ задания $j_{[l]}$, $l=\overline{g+1,n}$ запаздывающие, и у всех предшествующих заданий $j_{[i]}$, $i=\overline{1,g}$, $r_{j_{[i]}} \le 0$. В этом случае в оптимальной последовательности задания $j_{[l]}$ останутся на занимаемых позициях на интервале $\overline{g+1,n}$.

Справедливость этого утверждения основывается на том факте, что $\forall\, j_{[l]}$, $l=\overline{g+1,n}$ выполняется $l_{j[l]} \ge l_{j[g]}$. Резервы последовательности $\sigma^{\text{уп}}$ (исходной последовательности до встраивания задания $j_{[g]}$ на позицию m) в последовательности $\sigma(j_{[g]})$ использованы заданием меньшей длительности $j_{[g]}$. Следовательно, встраивание $j_{[l]}$, $l=\overline{g+1,n}$ на более ранние позиции приведет к увеличению потерь по заданиям, перемещаемым на более поздние позиции, и к ухудшению целевой функции. ■

Определение 8. Процедурой свободной перестановки называется процедура перестановки задания $j_{[k]}$ на позицию q ($k<q$) такую, что $D_{j[k]} \ge C_{j[q]}$, $D_{j[k]} < C_{j[q+1]}$, если хотя бы для одного задания на интервале $\overline{k+1,q}$ выполняется

$$D_{j_{[i]}} < C_{j_{[i]}}, \quad i = \overline{k+1, q}\ .$$

Таким образом, свободная перестановка k выполняется на позицию с максимальным номером, на которой k не будет запаздывать. Очевидно, что в результате выполнения всех свободных перестановок в последовательности $\sigma^{\text{уп}}$ значение целевой функции уменьшается.

При выполнении одной свободной перестановки значение целевой функции уменьшается на величину $\sum_{i \in z} \min\left(l_{j_{[k]}}, C_{j_{[i]}} - D_{j_{[i]}}\right)$, где

$$Z = \left\{ j_{[i]} \,\middle|\, C_{j_{[i]}} > D_{j_{[i]}},\ i = \overline{k+1, q} \right\}.$$

Примечание к определению 8.

При выполнении нескольких свободных перестановок во избежание зацикливания начинать следует с задания, имеющего максимальный директивный срок, т.е. просматривать незапаздывающие задания по убыванию их директивных сроков.

Определение 9. Последовательность заданий, полученную в результате выполнения всех свободных перестановок в последовательности $\sigma^{\text{уп}}$, назовем $\sigma^{\text{сп}}$.

Утверждение 11. Утверждения 1, 2, $4-8$ справедливы для последовательности $\sigma^{\text{сп}}$.

Доказательство. Выделим различия между последовательностями $\sigma^{\text{уп}}$ и $\sigma^{\text{сп}}$. Покажем, что эти различия не нарушают доказательств приведенных выше утверждений. При доказательстве утверждений 1, 2, $4-8$ основными приемами являлись перестановки и встраивания.

Для перестановок доказательства утверждений аналогичны, если вместо $\sigma^{\text{уп}}$ использовать $\sigma^{\text{сп}}$, по следующим причинам:

- Все задания на интервале перестановки сдвигаются на более ранние позиции и, следовательно, их запаздывание уменьшается. При этом задания, нарушающие последовательность $\sigma^{\text{уп}}$ (перенесенные в результате свободных перестановок), имеют запаздывание, равное нулю, и поэтому могут быть исключены из рассмотрения как в f, так и в f'.

- Для запаздывающих заданий в последовательности $\sigma^{\text{сп}}$ упорядоченность в соответствии с длительностями сохраняется и, следовательно, длительность задания, которое переставляется, меньше длительности любого из заданий на интервале перестановки.

Процедура встраивания для последовательности $\sigma^{\text{сп}}$ аналогична процедуре встраивания для $\sigma^{\text{уп}}$, так как длительность встраиваемого задания в $\sigma^{\text{сп}}$ всегда больше или равна длительности любого из заданий на интервале встраивания, что обусловливает необходимость для выполнения процедуры встраивания наличия резервов у заданий, занимающих более ранние позиции.

Таким образом, утверждения 1, 2, 4–8 верны и для последовательности $\sigma^{\text{сп}}$. ■

На интервале встраивания в последовательности $\sigma^{\text{сп}}$ могут находиться задания, перенесенные в результате свободных перестановок и нарушающие упорядоченность в соответствии с длительностями. Поэтому все задания, следующие за встраиваемым, необходимо переупорядочить в соответствии с их длительностями, и для запаздывающих заданий в этой последовательности, в свою очередь, проверять возможность переноса на более ранние позиции за счет резервов незапаздывающих заданий.

Таким образом, условие встраивания (3), справедливое для последовательности $\sigma^{\text{уп}}$, в $\sigma^{\text{сп}}$ не выполняется.

Утверждение 12. Для последовательности $\sigma^{\text{сп}}$ не существует перестановок и встраиваний, приводящих к улучшению целевой функции, если выполняется хотя бы одно из условий:

а) $D_{j_{[i]}} - C_{j_{[i]}} \geq 0,\ \forall i = \overline{1,n}$;

б) $D_{j_{[i]}} - C_{j_{[i]}} \leq 0,\ \forall i = \overline{1,l}$, $D_{j_{[i]}} - C_{j_{[i]}} \geq 0,\ \forall i = \overline{l+1,n}$;

в) $D_{j_{[i]}} - C_{j_{[i]}} \leq 0,\ \forall i = \overline{1,n}$;

г) для каждого задания $j_{[k]}$, принадлежащего множеству заданий с резервами (R), и каждого задания $j_{[g]}$, принадлежащего множеству запаздывающих заданий (Y), выполняется

$$D_{j_{[k]}} \leq D_{j_{[g]}},\ \ l_{j_{[k]}} \leq l_{j_{[g]}}\ \ \forall\, k \in R,\ \ g \in Y\,.$$

Пункт а) не требует доказательства, так как $f = 0$. Доказательство пунктов б), в), г) аналогично доказательству утверждений 1, 3 и 6. ■

Следствие. При выполнении хотя бы одного из условий, сформулированных в утверждении 12, в последовательности $\sigma^{\text{сп}}$ достигается оптимальное значение функционала.

Определение 10. Запаздывающее задание $j_{[g]}$ в последовательности $\sigma^{\text{сп}}$ называется конкурирующим, если в ней найдется хотя бы одно предшествующее задание $j_{[l]}$, для которого выполняются условия

$$D_{j_{[l]}} > D_{j_{[g]}} \text{ и } D_{j_{[l]}} - C_{j_{[l]}} > 0\,.$$

Определение 11. Последовательностью σ^{k} называется последовательность, полученная из $\sigma^{\text{сп}}$ в результате выполнения перестановок и встраиваний с целью оптимизации использования резервов времени незапаздывающих заданий.

Используются следующие типы перестановок и встраиваний.

Перестановки заданий:

а) для которых резерв времени больше нуля;

б) использовавших резервы в результате перестановок и встраиваний.

Перестановки осуществляются, если при их выполнении значение функционала уменьшается.

Встраивания:

а) при наличии резервов на интервале встраивания;

б) при образовании резервов на интервале встраивания в соответствии с условием 4, сформулированным в утверждении 8.

Правила выполнения перестановок и встраиваний в последовательности σ^k приведены ниже.

Введем правила пометки заданий в последовательности σ^k при выполнении перестановок и встраиваний.

В этой последовательности * обозначены задания, которые были запаздывающими в последовательности $\sigma^{\text{сп}}$ и в результате выполнения перестановок и встраиваний переместились на более ранние позиции, использовав резервы времени предшествующих заданий; ** помечены задания, ранее помеченные *, которые в результате последующих перестановок переместились на более поздние позиции (но эти позиции не соответствуют упорядоченности по длительностям этих заданий). За заданиями, помеченными * или **, в последовательности σ^k следуют запаздывающие задания с меньшей длительностью.

Рассмотрим структуру последовательности σ^k. Каждому заданию в ней соответствует номер задания и номер позиции. В исходной последовательности $\sigma^{\text{уп}}$ задания пронумерованы по неубыванию их длительностей. На следующих итерациях присвоенная нумерация сохраняется (меняются только номера позиций, занимаемых заданиями). Следовательно, в последовательности σ^k чем меньше номер задания, тем меньше его длительность. Упорядоченность в соответствии с длительностями нарушают задания, помеченные * или **, которые в результате перестановок и встраиваний использовали резервы времени предшествующих заданий и переместились с позиций, соответствующих их длительностям, на более ранние позиции, что позволило улучшить значение целевой функции.

Упорядоченность в соответствии с длительностями в последовательности σ^k нарушают также задания, которые в результате свободных перестановок переместились на более поздние позиции. Все остальные задания упорядочены в соответствии с их длительностями.

Таким образом, структуру последовательности σ^k от последовательности $\sigma^{\text{сп}}$ отличают только задания, помеченные * или **, которые переместились на более ранние позиции.

Сформулируем необходимые условия для перемещения в последовательности σ^k запаздывающих заданий на более ранние позиции, при котором значение функционала уменьшается за счет существующих и освободившихся резервов.

Утверждение 13. Пусть в последовательности σ^k $j_{[g]}$ — запаздывающее задание. Уменьшение значения функционала при перемещении $j_{[g]}$

на более ранние позиции возможно только при выполнении хотя бы одного из следующих условий:

1. $\exists j_{[i]}, P_{\min} \le i \le g \; || r_{j_{[i]}} > 0, D_{j_{[i]}} > D_{j_{[g]}}$. На интервале встраивания задания $j_{[g]}$ есть задания $j_{[i]}$, для которых $D_{j_{[i]}} > D_{j_{[g]}}$; $P_{\min}$ — позиция, где запаздывание по заданию $j_{[g]}$ минимально (или равно нулю).

2. $\exists j_{[q]}, \; q < g, \; D_{j_{[q]}} > C_{j_{[g]}}$.

3. $\exists j_{[l]}, \quad l < g, \; D_{j_{[l]}} > C_{j_{[g]}} - l_{j_{[g]}}, \quad \min\left(l_{j_{[l]}}, C_{j_{[g]}} - D_{j_{[g]}}\right) > C_{j_{[g]}} - - D_{j_{[l]}}$.

4. $\forall i \,|\, P_{\min} \le i \le g \,|\, r_{j_{[i]}} \le 0$, но $\exists j_{[k]}, \; k < P_{\min} \,|\, D_{j_{[k]}} > D_{j_{[g]}}$.

5. $\forall i \,|\, i = \overline{1, g-1}, \; r_{j_{[i]}} \le 0$, но $\exists j^{*}_{[m]} \; (j^{**}_{[m]}), \; m < g$.

Аналогично утверждению 8, выполнение одного из условий 1–4 означает, что на интервале встраивания задания $j_{[g]}$ существуют резервы (условия 1–3) или они будут образованы в результате перестановок (условие 4). Выполнение условия 5 означает, что в последовательности σ^k присутствуют задания, использовавшие резервы на интервале встраивания $j_{[g]}$, и перемещение их на более поздние позиции приведет к образованию резервов на этом интервале. Если не выполняется ни одно из условий 1–5, то перемещение $j_{[g]}$ на более ранние позиции приведет к увеличению значения функционала. ■

Следствие. Пусть в результате выполнения алгоритма построена оптимальная подпоследовательность на интервале $\overline{1, k}$. Если для каждого из запаздывающих заданий на интервале $\overline{k+1, n}$ не выполняется ни одно из условий 1–5, то последовательность σ^k оптимальна.

Число итераций предлагаемого алгоритма определяется количеством конкурирующих заданий. На каждой итерации проверяется возможность использования резервов времени предшествующих заданий очередным конкурирующим заданием и строится оптимальное расписание рассматриваемой последовательности. На первой итерации строится оптимальное расписание для заданий на интервале $\overline{1, g_1}$, где $j_{[g_1]}$ — первое запаздывающее конкурирующее задание в последовательности $\sigma^{\text{сп}}$. На следующей итерации рассматривается последовательность заданий на интервале $\overline{1, g_2}$, где $j_{[g_2]}$ — следующее запаздывающее конкурирующее задание. Пусть уже выполнена $k-1$ итерация и построено оптимальное расписание для последовательности заданий на интервале $\overline{1, k-1}$. Переходим к очередному запаздывающему конкурирующему заданию $j_{[k]}$ и строим оптимальное расписание для подпоследовательности σ^k, включающей задания на интервале

$\overline{1,k}$. Проверим возможность уменьшения значения функционала за счет использования заданием $j_{[k]}$ существующих резервов времени или резервов, полученных в результате перестановок заданий с метками (или заданий с большей длительностью) на более поздние позиции. На каждой итерации значение функционала уменьшается или остается неизменным.

Примечание. Необходимо различать конкурирующие запаздывающие задания, определенные в последовательности $\sigma^{\text{сп}}$, и задания, запаздывание которых относительно их директивных сроков возникло в процессе выполнения перестановок и встраиваний.

Следующие утверждения, доказательство которых очевидно, позволят определить и исключить бесперспективные перестановки и встраивания.

Утверждение 14. Запаздывающее задание $j_{[g]}$ на k-й итерации может занять более раннюю позицию, что приведет к улучшению целевой функции только в том случае, если хотя бы у одного задания $j_{[k]}$ на интервале $\overline{1,g-1}$ есть резерв и выполняется $D_{j_{[k]}} > D_{j_{[g]}}$ либо (в случае отсутствия резервов) на интервале $\overline{1,g-1}$ есть задания с метками. В противном случае $j_{[r]}$ исключается из множества конкурирующих.

Утверждение 15. Пусть в последовательности σ^k для запаздывающих заданий $j_{[k]}, j_{[r]} \in J$ выполняется

$$l_{j_{[k]}} \le l_{j_{[r]}}, \;\; D_{j_{[k]}} \le D_{j_{[r]}}.$$

Тогда в оптимальном расписании задание $j_{[k]}$ будет предшествовать $j_{[r]}$ [7].

Утверждение 16. Неконкурирующие задания в последовательности σ^k не могут занять более ранние позиции, чем позиции, занимаемые ими в $\sigma^{\text{сп}}$.

Утверждение 17. Пусть уже построена оптимальная подпоследовательность на интервале $\overline{1,g-1}$. Запаздывающее конкурирующее задание $j_{[g]}$ на k-й итерации не может быть перемещено на более раннюю позицию и исключается из числа конкурентоспособных, если на интервале $\overline{1,g-1}$ ни у одного из заданий нет резервов, и для всех помеченных заданий выполняется

$$l_{j^*_{[i]}} \le l_{j_{[g]}}, \;\; i = \overline{1,g-1}.$$

Утверждение 18. Если в последовательности σ^k конкурирующее задание $j_{[g]}$ в результате выполнения перестановок и встраиваний не использовало существующие резервы, минимальное значение функционала соответствует позиции g, занимаемой этим заданием, то задания $j_{[r]}$, $r = \overline{g+1,n}$, для которых $D_{j_{[r]}} \ge D_{j_{[g]}}$, в оптимальной последовательности не могут занять позицию более раннюю, чем g.

Утверждение 19. Если в последовательности σ^k конкурирующее задание $j_{[g]}$ в результате выполнения перестановок и встраиваний не использовало существующие резервы, минимальное значение функционала соответствует позиции g, занимаемой этим заданием, а $\forall j_{[r]}$, $r = \overline{g+1, n}$, $D_{j_{[r]}} \ge D_{j_{[g]}}$ и $C_{j_{[r]}} \ge D_{j_{[r]}}$, то задания $j_{[r]}$ исключаются из множества конкурирующих, и последовательность σ^k оптимальна.

Утверждение 20. Пусть на итерации k, выполняющейся для очередного конкурирующего запаздывающего задания $j_{[g]}$, для всех $j_{[l]}$, $l = \overline{1, g-1}$ справедливо $D_{j[l]} \le C_{j[l]}$, для всех $j^*_{[l]}$ выполняется $l_{j^*_{[l]}} \le l_{j_{[g]}}$, и на интервале $\overline{g, n}$ резервы отсутствуют. Тогда запаздывающие задания, занимающие позиции $\overline{g, n}$, исключаются из множества конкурирующих, а последовательности σ^k отвечает оптимальное значение функционала.

Утверждение 21. Пусть на итерации k в последовательности σ^k конкурирующее задание $j_{[g]}$ в результате выполнения перестановок и встраиваний заняло позицию s. Если на интервале $\overline{1, s}$ резервы отсутствуют, то конкурирующие задания $j_{[r]}$, $r = \overline{g+1, n}$, для которых $D_{j_{[r]}} \ge D_{j_{[g]}}$, в оптимальном расписании не смогут занять более раннюю позицию, чем s.

Утверждение 22. Пусть на итерации k в последовательности σ^k конкурирующее задание $j_{[g]}$ в результате выполнения перестановок и встраиваний заняло позицию s, оставаясь запаздывающим, и пусть для всех $j_{[r]}$, $r = \overline{g+1, n}$ выполняется $D_{j_{[r]}} \ge D_{j_{[g]}}$ и $D_{j_{[r]}} \le C_{j_{[r]}}$. Тогда эти задания в оптимальном расписании не смогут занять более раннюю позицию, чем s, и если на интервале $\overline{s, n}$ резервы отсутствуют, то последовательность σ^k оптимальна.

Утверждение 23. Если на интервале встраивания очередного запаздывающего задания $j_{[g]}$ есть задание $j_{[r]}$, для которого $D_{j_{[r]}} < D_{j_{[g]}}$, $l_{j_{[r]}} < l_{j_{[g]}}$, то интервал встраивания $j_{[g]}$ определится позициями $\overline{r+1, g-1}$.

Утверждение 24. Если в последовательности $\sigma^{\text{уп}}$ $\forall j_{[i]}$, $i = \overline{1, g-1}$, где $j_{[g]}$ — первое запаздывающее задание, выполняется $r_{j_{[i]}} < l_{j_{[g]}}$ и задания $j_{[i]}$, $t = \overline{g+1, n}$ — запаздывающие, то задача решается посредством полиномиальной составляющей предлагаемого алгоритма.

Утверждение 25. Пусть $j_{[g]}$ — первое запаздывающее задание в последовательности $\sigma^{\text{сп}}$. Если на интервале встраивания $j_{[g]}$ все задания упорядочены в соответствии с длительностями и $\forall j_{[s]}$, $s = \overline{g+1, n}$, выполняется $D_{j_{[s]}} < C_{j_{[s]}}$ и $D_{j_{[s]}} > D_{j_{[g]}}$, то задача решается посредством полиномиальной составляющей предлагаемого алгоритма.

ОПИСАНИЕ АЛГОРИТМА

Предлагаемый алгоритм решения рассматриваемой задачи состоит из двух этапов — предварительного (алгоритм А0) и оптимизации (алгоритм А1).

Алгоритм А0 (предварительный этап)

1. Упорядочить задания по неубыванию их длительностей l_j. Полученную последовательность обозначить $\sigma^{\text{уп}}$. Проверить условия утверждения 8. Если в последовательности $\sigma^{\text{уп}}$ ни для одного из запаздывающих заданий нет предшествующих заданий, для которых выполняется хотя бы одно из условий $1-4$, то последовательность $\sigma^{\text{уп}}$ оптимальна. Конец алгоритма. Иначе перейти на п.2.

2. Выполнить свободные перестановки в последовательности $\sigma^{\text{уп}}$, начиная с задания с максимальным директивным сроком.

2.1. Определить множество заданий *W*.

$$W = \{k : D_{j_{[k]}} \ge C_{j_{[q]}};\ D_{j_{[k]}} \le C_{j_{[q+1]}};\ k, q = \overline{1, n},\ k < q,$$

$$\exists\ j_{[i]} : D_{j_{[i]}} < C_{j_{[i]}},\ i = \overline{k+1, q}\ \}.$$

2.2. Найти задание $j_{[k']}$ с максимальным директивным сроком из множества W. Если $W = \varnothing$, то получена последовательность $\sigma^{\text{сп}}$. Перейти на п.2.4.

2.3. Выполнить перестановку задания $j_{[k']}$ на соответствующую ему позицию q' и исключить $j_{[k']}$ из множества W. Перейти на п. 2.2.

2.4. Проверить условия а), б), в), г) утверждения 12. Если выполняется хотя бы одно из этих условий, то последовательность $\sigma^{\text{сп}}$ оптимальна. Конец алгоритма. Иначе определить множество конкурирующих заданий K и перейти к этапу оптимизации.

Алгоритм А1 (этап оптимизации)

Алгоритм состоит из k однотипных итераций, где k — количество конкурирующих заданий. На каждой итерации проверяется возможность использования резервов времени незапаздывающих заданий очередным конкурирующим заданием и порожденными в результате его перестановок новыми запаздывающими заданиями. В результате выполнения каждой итерации значение функционала уменьшается (или остается неизменным).

Пусть $g^{\text{пр}}$ — позиция предшествующего задания, которое рассматривалось на предыдущей итерации (на первой итерации $g^{\text{пр}} = 0$).

1. Найти очередное запаздывающее задание $j_{[g]}$ на интервале $\overline{g^{\text{пр}}+1, n}$. Если запаздывающее задание не найдено, то конец. Иначе если это задание неконкурирующее — перейти на п.2, а если конкурирующее — проверить условия утверждений 14, 17, 18–20, 22, позволяющие исключить задание $j_{[g]}$ из множества конкурирующих заданий. Если в результате задание $j_{[g]}$ не исключено из множества конкурирующих, перейти на п.2. Иначе установить $g^{\text{пр}} = g$. Перейти на п.1. Если $g = n$, то конец алгоритма, текущая последовательность оптимальна.

2. Выполнить все свободные перестановки на интервале $\overline{1, g}$, начиная с задания с максимальным директивным сроком.

3. Если в результате выполнения п.2 задание $j_{[g]}$ перестало запаздывать, переместившись на позицию $g^{\text{н}}$, то установить $g^{\text{пр}} = g^{\text{н}}$ и перейти на п.1, иначе на п.4.

4. Найти задание $j_{[l]}$ ($l < g$), для которого выполняется

$$D_{j_{[l]}} \geq C_{j_{[g]}} - l_{j_{[g]}},$$

$$C_{j_{[g]}} - D_{j_{[l]}} < \min\left(l_{j_{[l]}}, C_{j_{[g]}} - D_{j_{[l]}}\right).$$

Если найдено хотя бы одно задание, отвечающее этим условиям, перейти на п.5, иначе на п.7.

5. Выполнить перестановку задания $j_{[l]}$ на позицию g. Если $j_{[g]}$ перестало запаздывать, то перейти на п.6, иначе на п. 2.

6. Если после п.2 выполнялись пп.4, 5 для задания $j_{[g]}$ более одного раза, то задания, следующие за $j_{[g]}$, упорядочить в соответствии с их длительностями, уничтожив все метки. Установить $g^{\text{пр}} = g^{\text{н}}$ и перейти на п.1.

7. Найти позицию *p*, на которой запаздывание по заданию $j_{[g]}$ станет минимальным или равным нулю. Если такая позиция найдена, перейти на п.8, иначе на п.13.

8. Если на интервале встраивания задания $j_{[g]}$, определяемом позициями $\overline{p, g-1}$, есть задание $j_{[s]}$, для которого $D_{j_{[s]}} < D_{j_{[g]}}$, $l_{j_{[s]}} < l_{j_{[g]}}$, то интервал встраивания задания $j_{[g]}$ определится позициями $\overline{s+1, g}$.

9. Проверить, есть ли на интервале $\overline{1, g-1}$ задания $j_{[s]}$, для которых $D_{j_{[s]}} - C_{j_{[s]}} > 0$, $D_{j_{[s]}} > D_{j_{[g]}}$. Если да, то перейти на п.10, иначе установить $g^{\text{пр}} = g$ и перейти на п.1.

10. Встроить задание $j_{[g]}$ на позицию p (или при выполнении п.8 на позицию $s+1$). Если эту позицию занимает задание, помеченное * или **,

встроить $j_{[g]}$ на позицию $p+1$ и перейти на п.11. Если в результате выполнения процедуры встраивания $j_{[g]}$ осталось на занимаемой позиции g, перейти на п.13.

11.Пометить задание $j_{[g]}$, занявшее в результате выполнения встраивания новую позицию, знаком *. Задания, следующие за ним, упорядочить в соответствии с их длительностями, уничтожив все метки.

12.Если $r_{j_{[g]}} \geq 0$, установить $g^{\text{пр}} = g^{\text{н}}$, иначе $g^{\text{пр}} = g^{\text{н}} - 1$. Перейти на п.1.

13.Найти задание $j^*_{[m]}$ ($j^{**}_{[m]}$) на интервале $\overline{1, g-1}$. Если $j_{[g]}$ — конкурирующее, то неконкурирующие задания $j^*_{[m]}$ ($j^{**}_{[m]}$) не рассматриваются. Не рассматриваются также задания, для которых $l_{j_{[g]}} > l_{j^*_{[m]}}$ ($l_{j^{**}_{[m]}}$) и $D_{j_{[g]}} >$ $> D_{j^*_{[m]}}$ ($D_{j^{**}_{[m]}}$). В первую очередь анализируется задание, помеченное * или ** последним, затем предпоследним и т.д. Если помеченное задание не найдено, перейти на п.18, иначе на п.14.

14.Проверить условие

$$C_{j_{[k]}} - D_{j^*_{[m]}} < \sum_{i=m}^{g} \max\left(0, C_{j_{[i]}} - D_{j_{[i]}}\right), \tag{7}$$

где $k = g$, если $l_{j_{[g]}} \leq l_{j^*_{[m]}}$, иначе $k = s$, где $s+1$ — позиция первого непомеченного задания, длительность которого больше $j^*_{[m]}$ ($j^{**}_{[m]}$), $s > m^*$.

В случае выполнения условия (7) перейти на п.15, иначе найти следующее задание, помеченное * или **, для которого выполняется это условие. Если такие задания не найдены, перейти на п.18.

15.Запомнить текущую последовательность заданий, обозначив ее $\sigma^{\text{ф}}$.

16.Выполнить перестановку задания $j^*_{[m]}$ ($j^{**}_{[m]}$):

а) после задания $j_{[g]}$, если $l_{j_{[g]}} \leq l_{j^*_{[m]}}$;

б) на позицию k, определенную в п.14, если $l_{j_{[g]}} > l_{j^*_{[m]}}$.

Установить $g^{\text{пр}} = g^{\text{н}} - 1$, где $g^{\text{н}}$ — новая позиция $j_{[g]}$ после перестановки $j^*_{[m]}$. Перейти на п.1 и переупорядочить задания на интервале $\overline{1, g}$ по изложенным правилам.

17.Определить значение функционала полученной последовательности $\sigma^{\text{н}}$ и сравнить его со значением функционала последовательности $\sigma^{\text{ф}}$. Если значение целевой функции не уменьшилось, возвратиться к $\sigma^{\text{ф}}$ и перейти на п.13. В случае уменьшения значения целевой функции в последовательности $\sigma^{\text{н}}$ обозначение * снимается с задания $j^*_{[m]}$, если в результате перестановок это задание заняло позицию в соответствии с его длительностью. Иначе это задание пометим **.

18. Установить $g^{\text{пр}} = g$. Перейти на п.1. Если $g^{\text{пр}} = n$, то конец.

Алгоритм построен таким образом, что на каждой итерации строится оптимальное расписание для последовательности σ^k заданий $j_{[i]}$, $i = \overline{1, g_k}$, где $j_{[g_k]}$ — очередное конкурирующее задание. На первой итерации рассматривается подпоследовательность $\sigma^1 \subset \sigma^{\text{сп}}$, содержащая задания, занимающие в последовательности $\sigma^{\text{сп}}$ позиции $\overline{1, g_1}$, где g_1 — позиция первого конкурирующего запаздывающего задания. На второй итерации полученная оптимальная последовательность дополняется заданиями $j_{[l]}$, которые в последовательности $\sigma^{\text{сп}}$ занимают позиции $l = \overline{g_1 + 1, g_2}$, и строится оптимальное расписание для этой последовательности. На каждой итерации оптимизация осуществляется за счет использования резервов времени незапаздывающих заданий очередным конкурирующим заданием и порожденными в результате его перестановок новыми запаздывающими заданиями. Оптимизация выполняется путем перестановок и встраиваний, направленных на улучшение значения функционала на рассматриваемом подмножестве заданий. Исключаются только те перестановки, которые заведомо ухудшают значение функционала. Таким образом строится оптимальное расписание для всего множества заданий.

В процессе решения задачи проверяются условия утверждений 8, 12, 13, 24, 25.Если 8 или 13 не выполняются или выполняются условия хотя бы одного из утверждений 12, 24, 25, то задача решается с помощью полиномиальной составляющей предложенного алгоритма, иначе алгоритм работает по декомпозиционной составляющей. В этом случае для каждого запаздывающего задания $j_{[g]}$ определяется интервал встраивания $I_{j_{[g]}}$, ограниченный позициями $\overline{p, g-1}$, на котором возможно уменьшение значения функционала за счет резервов заданий на этом интервале или резервов, которые могут быть образованы заданиями, занимавшими позиции $\overline{1, p-1}$, в результате выполнения перестановок. В этом случае необходимо выполнение условий утверждения 6. Таким образом, задача декомпозируется на подзадачи меньшего размера, а в результате проверки условий утверждений 14, 17 – 20, 22 часть конкурирующих заданий исключается из множества конкурирующих, что существенно сокращает число выполняемых итераций.

ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ

Для оценки работы алгоритма были вручную просчитаны и исследованы тестовые примеры размерности $n \le 150$, взятые на интернет-странице [18], щедро предоставленной Бахар Йетис Кара. Оптимальные значения функционала, полученные нашим алгоритмом, совпали с описанными в [18] значениями функционала. Ниже приведены результат и анализ хода решения тестового примера, одного из наиболее трудоемких для размерности $n = 150$.

Исходные данные приведены в табл. 1. В этой и последующих таблицах T_j обозначено запаздывание по заданию j: $T_j = C_j - D_j$.

Таблица 1. Исходные данные для примера (последовательность $\sigma^{\text{уп}}$)

j	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
L_j	1	1	1	1	2	3	4	4	4	5
D_j	880	1160	1944	2079	1494	754	815	1211	1698	991
C_j	1	2	3	4	6	9	13	17	21	26
T_j	0	0	0	0	0	0	0	0	0	0
j	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
L_j	5	5	5	6	6	7	7	8	8	8
D_j	1508	1760	1795	1299	1579	1370	1861	955	1117	1140
C_j	31	36	41	47	53	60	67	75	83	91
T_j	0	0	0	0	0	0	0	0	0	0
j	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
L_j	8	8	8	9	9	10	10	11	11	12
D_j	1156	1709	1890	970	1386	1023	1516	1361	1749	1116
C_j	99	107	115	124	133	143	153	164	175	187
T_j	0	0	0	0	0	0	0	0	0	0
j	**31**	**32**	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**
L_j	12	13	13	14	14	14	14	15	16	20
D_j	1553	1566	2013	715	883	1281	2066	1246	1230	717
C_j	199	212	225	239	253	267	281	296	312	332
T_j	0	0	0	0	0	0	0	0	0	0
j	**41**	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**
L_j	21	22	22	24	24	25	25	26	27	27
D_j	1366	1554	1589	1175	1902	773	1655	1155	1046	1744
C_j	353	375	397	421	445	470	495	521	548	575
T_j	0	0	0	0	0	0	0	0	0	0
J	**51**	**52**	**53**	**54**	**55**	**56**	**57**	**58**	**59**	**60**
L_j	28	29	29	31	31	32	34	38	38	39
D_j	891	990	1865	1416	1556	1684	1516	710	1826	992
C_j	603	632	661	692	723	755	789	827	865	904
T_j	0	0	0	0	0	0	0	117	0	0
J	**61**	**62**	**63**	**64**	**65**	**66**	**67**	**68**	**69**	**70**
L_j	39	40	40	41	42	42	43	43	46	46
D_j	1535	933	1816	1051	728	1722	1268	1551	1762	1774
C_j	943	983	1023	1064	1106	1148	1191	1234	1280	1326
T_j	0	50	0	13	378	0	0	0	0	0
J	**71**	**72**	**73**	**74**	**75**	**76**	**77**	**78**	**79**	**80**
L_j	46	47	48	48	50	51	52	52	52	52
D_j	1813	1753	1007	2019	1148	1776	788	810	851	999
C_j	1372	1419	1467	1515	1565	1616	1668	1720	1772	1824
T_j	0	0	460	0	417	0	880	910	921	825
j	**81**	**82**	**83**	**84**	**85**	**86**	**87**	**88**	**89**	**90**
L_j	52	52	53	54	54	55	55	55	55	56
D_j	1221	1630	1558	852	1995	716	800	962	1495	909
C_j	1876	1928	1981	2035	2089	2144	2199	2254	2309	2365
T_j	655	298	423	1183	94	1428	1399	1292	814	1456
j	**91**	**92**	**93**	**94**	**95**	**96**	**97**	**98**	**99**	**100**
L_j	56	57	57	58	58	58	59	59	60	61
D_j	1066	1123	1498	1623	1706	1963	1338	1702	2010	1807
C_j	2421	2478	2535	2593	2651	2709	2768	2827	2887	2948
T_j	1355	1355	1037	970	945	746	1430	1125	877	1141

Окончание табл. 1

j	**101**	**102**	**103**	**104**	**105**	**106**	**107**	**108**	**109**	**110**
L_j	62	62	63	64	65	67	68	68	71	71
D_j	1297	1583	2084	1357	1544	852	779	1016	1213	1542
C_j	3010	3072	3135	3199	3264	3331	3399	3467	3538	3609
T_j	1713	1489	1051	1842	1720	2479	2620	2451	2325	2067
j	**111**	**112**	**113**	**114**	**115**	**116**	**117**	**118**	**119**	**120**
L_j	71	72	72	72	72	73	73	73	74	76
D_j	1847	842	1590	1928	2060	824	1677	1699	943	1475
C_j	3680	3752	3824	3896	3968	4041	4114	4187	4261	4337
T_j	1833	2910	2234	1968	1908	3217	2437	2488	3318	2862
j	**121**	**122**	**123**	**124**	**125**	**126**	**127**	**128**	**129**	**130**
L_j	77	78	78	80	80	81	81	82	84	84
D_j	1619	1373	1789	916	1255	1874	1958	1761	840	1986
C_j	4414	4492	4570	4650	4730	4811	4892	4974	5058	5142
T_j	2795	3119	2781	3734	3475	2937	2934	3213	4218	3156
j	**131**	**132**	**133**	**134**	**135**	**136**	**137**	**138**	**139**	**140**
L_j	85	85	86	86	89	89	90	91	93	95
D_j	900	1053	1333	1533	906	1614	1137	1376	1737	990
C_j	5227	5312	5398	5484	5573	5662	5752	5843	5936	6031
T_j	4327	4259	4065	3951	4667	4048	4615	4467	4199	5041
j	**141**	**142**	**143**	**144**	**145**	**146**	**147**	**148**	**149**	**150**
L_j	95	95	96	96	96	97	97	99	100	100
D_j	1250	1719	721	996	1665	2028	2080	1005	1191	1215
C_j	6126	6221	6317	6413	6509	6606	6703	6802	6902	7002
T_j	4876	4502	5596	5417	4844	4578	4623	5797	5711	5787

Значение функционала по последовательности $\sigma^{\text{уп}}$ составляет 197658. В табл. 2 приведена последовательность, полученная после выполнения свободных перестановок, — $\sigma^{\text{сп}}$.

Таблица 2. Последовательность $\sigma^{\text{сп}}$

j	**1**	**2**	**6**	**7**	**8**	**10**	**14**	**18**	**19**	**20**
L_j	1	1	3	4	4	5	6	8	8	8
D_j	880	1160	754	815	1211	991	1299	955	1117	1140
C_j	1	2	5	9	13	18	24	32	40	48
T_j	0	0	0	0	0	0	0	0	0	0
j	**21**	**24**	**26**	**30**	**34**	**35**	**36**	**38**	**39**	**40**
L_j	8	9	10	12	14	14	14	15	16	20
D_j	1156	970	1023	1116	715	883	1281	1246	1230	717
C_j	56	65	75	87	101	115	129	144	160	180
T_j	0	0	0	0	0	0	0	0	0	0
j	**44**	**46**	**48**	**49**	**51**	**52**	**58**	**60**	**62**	**64**
L_j	24	25	26	27	28	29	38	39	40	41
D_j	1175	773	1155	1046	891	990	710	992	933	1051
C_j	204	229	255	282	310	339	377	416	456	497
T_j	0	0	0	0	0	0	0	0	0	0
j	**65**	**73**	**75**	**77**	**78**	**79**	**80**	**81**	**82**	**67**
L_j	42	48	50	52	52	52	52	52	52	43
D_j	728	1007	1148	788	810	851	999	1221	1630	1268
C_j	539	587	637	689	741	793	845	897	949	992
T_j	0	0	0	0	0	0	0	0	0	0

Продолжение табл. 2

j	**28**	**41**	**16**	**25**	**54**	**5**	**11**	**27**	**57**	**61**
L_j	11	21	7	9	31	2	5	10	34	39
D_j	1361	1366	1370	1386	1416	1494	1508	1516	1516	1535
C_j	1003	1024	1031	1040	1071	1073	1078	1088	1122	1161
T_j	0	0	0	0	0	0	0	0	0	0
j	**68**	**31**	**42**	**55**	**32**	**15**	**43**	**47**	**56**	**9**
L_j	43	12	22	31	13	6	22	25	32	4
D_j	1551	1553	1554	1556	1566	1579	1589	1655	1684	1698
C_j	1204	1216	1238	1269	1282	1288	1310	1335	1367	1371
T_j	0	0	0	0	0	0	0	0	0	0
j	**22**	**66**	**29**	**50**	**72**	**12**	**69**	**70**	**76**	**13**
L_j	8	42	11	27	47	5	46	46	51	5
D_j	1709	1722	1749	1744	1753	1760	1762	1774	1776	1795
C_j	1379	1421	1432	1459	1506	1511	1557	1603	1654	1659
T_j	0	0	0	0	0	0	0	0	0	0
j	**71**	**63**	**59**	**17**	**53**	**23**	**45**	**3**	**83**	**33**
L_j	46	40	38	7	29	8	24	1	53	13
D_j	1813	1816	1826	1861	1865	1890	1902	1944	1558	2013
C_j	1705	1745	1783	1790	1819	1827	1851	1852	1905	1918
T_j	0	0	0	0	0	0	0	0	347	0
j	**74**	**84**	**37**	**4**	**85**	**86**	**87**	**88**	**89**	**90**
L_j	48	54	14	1	54	55	55	55	55	56
D_j	2019	852	2066	2079	1995	716	800	962	1495	909
C_j	1966	2020	2034	2035	2089	2144	2199	2254	2309	2365
T_j	0	1168	0	0	94	1428	1399	1292	814	1456
j	**91**	**92**	**93**	**94**	**95**	**96**	**97**	**98**	**99**	**100**
L_j	56	57	57	58	58	58	59	59	60	61
D_j	1066	1123	1498	1623	1706	1963	1338	1702	2010	1807
C_j	2421	2478	2535	2593	2651	2709	2768	2827	2887	2948
T_j	1355	1355	1037	970	945	746	1430	1125	877	1141
j	**101**	**102**	**103**	**104**	**105**	**106**	**107**	**108**	**109**	**110**
L_j	62	62	63	64	65	67	68	68	71	71
D_j	1297	1583	2084	1357	1544	852	779	1016	1213	1542
C_j	3010	3072	3135	3199	3264	3331	3399	3467	3538	3609
T_j	1713	1489	1051	1842	1720	2479	2620	2451	2325	2067
j	**111**	**112**	**113**	**114**	**115**	**116**	**117**	**118**	**119**	**120**
L_j	71	72	72	72	72	73	73	73	74	76
D_j	1847	842	1590	1928	2060	824	1677	1699	943	1475
C_j	3680	3752	3824	3896	3968	4041	4114	4187	4261	4337
T_j	1833	2910	2234	1968	1908	3217	2437	2488	3318	2862
j	**121**	**122**	**123**	**124**	**125**	**126**	**127**	**128**	**129**	**130**
L_j	77	78	78	80	80	81	81	82	84	84
D_j	1619	1373	1789	916	1255	1874	1958	1761	840	1986
C_j	4414	4492	4570	4650	4730	4811	4892	4974	5058	5142
T_j	2795	3119	2781	3734	3475	2937	2934	3213	4218	3156
j	**131**	**132**	**133**	**134**	**135**	**136**	**137**	**138**	**139**	**140**
L_j	85	85	86	86	89	89	90	91	93	95
D_j	900	1053	1333	1533	906	1614	1137	1376	1737	990
C_j	5227	5312	5398	5484	5573	5662	5752	5843	5936	6031
T_j	4327	4259	4065	3951	4667	4048	4615	4467	4199	5041

Окончание табл. 2

j	141	142	143	144	145	146	147	148	149	150
L_j	95	95	96	96	96	97	97	99	100	100
D_j	1250	1719	721	996	1665	2028	2080	1005	1191	1215
C_j	6126	6221	6317	6413	6509	6606	6703	6802	6902	7002
T_j	4876	4502	5596	5417	4844	4578	4623	5797	5711	5787

Значение функционала по последовательности $\sigma^{\text{сп}}$ составляет 191643. В табл. 3 приведена оптимальная последовательность $\sigma^{\text{опт}}$.

Таблица 3. Последовательность $\sigma^{\text{опт}}$

j	1	2	6	7	10	18	24	26	34	35
L_j	1	1	3	4	5	8	9	10	14	14
D_j	880	1160	754	815	991	955	970	1023	715	883
C_j	1	2	5	9	14	22	31	41	55	69
T_j	0	0	0	0	0	0	0	0	0	0

j	40	46	49	51	52	58	60	62	64	77
L_j	20	25	27	28	29	38	39	40	41	52
D_j	717	773	1046	891	990	710	992	933	1051	788
C_j	89	114	141	169	198	236	275	315	356	408
T_j	0	0	0	0	0	0	0	0	0	0

j	78	79	80	65	73	86	87	84	90	88
L_j	52	52	52	42	48	55	55	54	56	55
D_j	810	851	999	728	1007	716	800	852	909	962
C_j	460	512	564	606	654	709	764	818	874	929
T_j	0	0	0	0	0	0	0	0	0	0

j	91	30	21	19	75	92	20	48	44	8
L_j	56	12	8	8	50	57	8	26	24	4
D_j	1066	1116	1156	1117	1148	1123	1140	1155	1175	1211
C_j	985	997	1005	1013	1063	1120	1128	1154	1178	1182
T_j	0	0	0	0	0	0	0	0	3	0

j	38	39	81	36	14	67	28	41	16	25
L_j	15	16	52	14	6	43	11	21	7	9
D_j	1246	1230	1221	1281	1299	1268	1361	1366	1370	1386
C_j	1197	1213	1265	1279	1285	1328	1339	1360	1367	1376
T_j	0	0	44	0	0	60	0	0	0	0

j	54	5	11	27	57	61	31	42	55	32
L_j	31	2	5	10	34	39	12	22	31	13
D_j	1416	1494	1508	1516	1516	1535	1553	1554	1556	1566
C_j	1407	1409	1414	1424	1458	1497	1509	1531	1562	1575
T_j	0	0	0	0	0	0	0	0	6	9

j	15	43	68	47	9	56	22	50	29	12
L_j	6	22	43	25	4	32	8	27	11	5
D_j	1579	1589	1551	1655	1698	1684	1709	1744	1749	1760
C_j	1581	1603	1646	1671	1675	1707	1715	1742	1753	1758
T_j	2	14	95	16	0	23	6	0	4	0

j	13	66	59	17	53	23	45	3	63	69
L_j	5	42	38	7	29	8	24	1	40	46
D_j	1795	1722	1826	1861	1865	1890	1902	1944	1816	1762
C_j	1763	1805	1843	1850	1879	1887	1911	1912	1952	1998
T_j	0	83	17	0	14	0	9	0	136	236

Окончание табл. 3

j	**33**	**70**	**37**	**4**	**71**	**72**	**74**	**76**	**82**	**83**
L_j	13	46	14	1	46	47	48	51	52	53
D_j	2013	1774	2066	2079	1813	1753	2019	1776	1630	1558
C_j	2011	2057	2071	2072	2118	2165	2213	2264	2316	2369
T_j	0	283	5	0	305	412	194	488	686	811
j	**85**	**89**	**93**	**94**	**95**	**96**	**97**	**98**	**99**	**100**
L_j	54	55	57	58	58	58	59	59	60	61
D_j	1995	1495	1498	1623	1706	1963	1338	1702	2010	1807
C_j	2423	2478	2535	2593	2651	2709	2768	2827	2887	2948
T_j	428	983	1037	970	945	746	1430	1125	877	1141
j	**101**	**102**	**103**	**104**	**105**	**106**	**107**	**108**	**109**	**110**
L_j	62	62	63	64	65	67	68	68	71	71
D_j	1297	1583	2084	1357	1544	852	779	1016	1213	1542
C_j	3010	3072	3135	3199	3264	3331	3399	3467	3538	3609
T_j	1713	1489	1051	1842	1720	2479	2620	2451	2325	2067
j	**111**	**112**	**113**	**114**	**115**	**116**	**117**	**118**	**119**	**120**
L_j	71	72	72	72	72	73	73	73	74	76
D_j	1847	842	1590	1928	2060	824	1677	1699	943	1475
C_j	3680	3752	3824	3896	3968	4041	4114	4187	4261	4337
T_j	1833	2910	2234	1968	1908	3217	2437	2488	3318	2862
j	**121**	**122**	**123**	**124**	**125**	**126**	**127**	**128**	**129**	**130**
L_j	77	78	78	80	80	81	81	82	84	84
D_j	1619	1373	1789	916	1255	1874	1958	1761	840	1986
C_j	4414	4492	4570	4650	4730	4811	4892	4974	5058	5142
T_j	2795	3119	2781	3734	3475	2937	2934	3213	4218	3156
j	**131**	**132**	**133**	**134**	**135**	**136**	**137**	**138**	**139**	**140**
L_j	85	85	86	86	89	89	90	91	93	95
D_j	900	1053	1333	1533	906	1614	1137	1376	1737	990
C_j	5227	5312	5398	5484	5573	5662	5752	5843	5936	6031
T_j	4327	4259	4065	3951	4667	4048	4615	4467	4199	5041
j	**141**	**142**	**143**	**144**	**145**	**146**	**147**	**148**	**149**	**150**
L_j	95	95	96	96	96	97	97	99	100	100
D_j	1250	1719	721	996	1665	2028	2080	1005	1191	1215
C_j	6126	6221	6317	6413	6509	6606	6703	6802	6902	7002
T_j	4876	4502	5596	5417	4844	4578	4623	5797	5711	5787

Значение функционала $f^{\text{опт}} = 186307$.

В результате решения задачи было выполнено 10 итераций (по заданиям 83–92, см. табл. 2). Задания 93–150 в процессе решения были исключены из множества конкурирующих, и для них перестановки не выполнялись. Полученная последовательность (табл. 3) оптимальна.

Анализ предложенного алгоритма позволил определить, что если число запаздывающих заданий и число заданий с резервами в последовательности $\sigma^{\text{сп}}$ приблизительно равны, и при этом с возрастанием длительностей запаздывающих заданий их директивные сроки уменьшаются, то возможно существенное увеличение числа итераций.

ВЫВОДЫ

1. Предложен новый эффективный ПДС-алгоритм для решения NP-сложной задачи составления расписаний по критерию минимизации суммарного запаздывания выполнения заданий одним прибором относительно директивных сроков. Алгоритм основан на новом подходе к решению задач комбинаторной оптимизации — разработке алгоритмов с полиномиальной и декомпозиционной составляющими. Эти алгоритмы базируются на выделении полиномиальной составляющей алгоритма и получении условий, когда экспоненциальная составляющая декомпозирует начальную задачу на подзадачи. Если условия утверждений 8 или 13 не выполняются или выполняются условия хотя бы одного из утверждений 12, 24, 25, то задача решается с помощью полиномиальной составляющей предложенного алгоритма. В противном случае алгоритм работает по декомпозиционной составляющей. В этом случае для каждого запаздывающего задания определяется интервал встраивания, на котором возможно уменьшение значения функционала. Таким образом, задача декомпозируется на подзадачи меньшего размера.

Новым в предложенном подходе является выделение полиномиально разрешимых подклассов труднорешаемых комбинаторных задач не путем введения соответствующих ограничений в постановки задач, а на основе анализа процесса решения произвольных индивидуальных задач.

2. Выявлены и обоснованы новые свойства исследуемой задачи, что позволяет получать эффективные решения для задач большой размерности. Расширена область, для которой задача может быть оптимально решена полиномиальной составляющей нашего алгоритма. На базе предложенного алгоритма может быть построен приближенный быстродействующий алгоритм.

В дальнейшем предполагается определить вычислительную сложность предложенного алгоритма с использованием известной схемы генерации исходных данных [7, 8, 19], выявить новые полиномиально разрешимые подклассы исследуемой задачи и разработать ПДС-алгоритм для случая взвешенного запаздывания.

ЛИТЕРАТУРА

1. *Shwarz W. and Mukhopadhyay S.K.* Decomposition of the single machine total tardiness problem // Oper. Res. Lett. — 1996. — № 19. — P. 243–250.
2. *Du J. and Leung J.Y.-T.* Minimizing total tardiness on one processor is NP-hard // Math. Oper. Res. — 1990. — № 15. — P. 483–495.
3. *Emmons H.* One machine sequencing to minimize certain functions of job tardiness // Oper. Res. — 1969. — № 17. — P. 701–715.
4. *Baker K.R. and Shrage L.E.* Finding an optimal sequence by dynamic programming: an extension to precedence-related tasks // Oper. Res. — 1978. — № 26. — P. 111–120.
5. *Fisher M.L.* A dual algorithm for the one-machine scheduling problem // Math. Programming. — 1976. — № 11. — P. 229–251.
6. *Rinnooy Kan A.H.G., Lageweg B.J. and Lenstra J.K.* Minimizing total costs in one-machine scheduling // Oper. Res. — 1975. — № 23. — P. 908–927.

7. *Potts C.N.and Wassenhove Van L.N.* A decomposition algorithm for the single machine total tardiness problem // Oper. Res. Lett. — 1982. — № 1. — P. 177–181.
8. *Potts C.N. and Wassenhove Van L.N.* Dynamic programming and decomposition approaches for single machine total tardiness problem // European J. Oper. Res. — 1987. — № 32. — P. 405–414.
9. *Shrage L. and Baker K.R.* Dynamic programming solution of sequencing problems with precedence constraints // Oper. Res. — 1978. — № 26. — P. 444–449.
10. *Srinivasan V.* A hybrid algorithm for the one-machine sequencing problem to minimize total tardiness // Naval Res. Logist. Quart. — 1971. — № 18. — P. 317–327.
11. *Lawler E.L.* A pseudopolynomial algorithm for sequencing jobs to minimize total tardiness // Ann. Discrete Math. — 1977. — № 1. — P. 331–342.
12. *Szwarc W.* Single machine total tardiness problem revisited in: Y.Ijiri (ed.), Creative and Innovative Approaches to the Science of Management, Quorum Books, 1993. — P. 407–419.
13. *Delia Croce F., Grosso A., Paschos V.T.* Lower bounds on the approximation ratios of leading heuristics for the single-machine total tardiness problem // Journal of Scheduling. — 2004. — № 7. — P. 85–91.
14. *Szwarc W., Grosso A. and Delia Croce F.* Algorithmic paradoxes of the single machine total tardiness problem // Journal of Scheduling. — 2001. — № 4. — P. 93–104.
15. *Pavlov A., Pavlova L.* PDC-algorithms for intractable combinatorial problems. Theory and methodology of design. — Uzhhorod: Karpatskij region, 1998. — 320 p.
16. *Павлов А.А., Теленик С.Ф.* Информационные технологии и алгоритмизация в управлении.– Киев: Техніка, 2002. — 344 с.
17. *Павлов А.А., Мисюра Е.Б.* Новый подход к решению задачи «Минимизация суммарного взвешенного опоздания при выполнении независимых заданий с директивными сроками одним прибором» // Системні дослідження та інформаційні технології. — 2002. — № 2.— С. 7–32.
18. http://www.bilkent.edu.tr/~bkara/start.html.
19. *Tansel B., Kara B. and Sabuncuoglu I.* An Efficient Algorithm for the Single Machine Total Tardiness Problem // IIE Transactions. — 2001. — **8**, № 33. — P. 661–674.

Надійшла 14.05.2004