УДК 519.853

С.В. Минухин

Харьковский национальный экономический университет имени Семена Кузнеца, Харьков

О СВОЙСТВАХ ОПТИМАЛЬНОСТИ МЕТОДА МИНИМИЗАЦИИ СУММАРНОГО ЗАПАЗДЫВАНИЯ НА ОДИНОЧНОМ УСТРОЙСТВЕ НА ОСНОВЕ РАНГОВОГО ПОДХОДА И ПРАВИЛ ДОМИНИРОВАНИЯ

Рассмотрен метод минимизации суммарного запаздывания работ на одиночном устройстве, сведенный к решению задачи нахождения кратчайшего гамильтонового пути в произвольном полносвязном графе на основе рангового подхода и правил доминирования. Рассмотрены свойства оптимальности метода на основе введенных определений локально-оптимального решения, локально-оптимального расписания на ранге и оптимальности получаемого итогового расписания. Сформулированы определения и предложения, определяющие оптимальность получаемых расписаний выполнения работ на основе доминирующих и строго доминирующих расписаний для взвешенного и невзвешенного случаев. Предложены метрики оценки улучшения результатов работы алгоритма при использовании правила доминирования. Приведены результаты вычислительного эксперимента по оцениванию улучшения работы алгоритма, подтверждающие целесообразность использования правил доминирования.

Ключевые слова: *расписание, работа, ранговый подход, правило доминирования, последовательность, метрика, граф.*

Введение

Современные информационно-коммуникационные системы характеризуются наличием большого количества различных типов ресурсов – информационных вычислительных и т.д., их распределенностью, управление которыми требует повышения эффективности и совершенствования процессов управления распределенной обработкой, в частности, на локальных ресурсах, после их нахождения для выполнения заданий метапланировщиком. Одной из важнейших задач является разработка эффективных алгоритмов построения локальных расписаний выполнения заданий в соответствии с выбранными критериями на вычислительных устройствах, таких, например, как узлы Грид-систем – вычислительные кластеры, кластеры рабочих станций (Clusters of workstations), промышленные desktop Grids, некластеризованные ресурсы и т.д. В связи с этим в последнее время получило развитие использование методов построения расписаний выполнения работ, которые широко используются в промышленном планировании (production planning). Отметим статью [1], в которой предложено для моделирования работы веб-сервера, использующего единственную очередь работ, дифференцировать QoS (Quality of Service) для различных классов запросов к веб-сервисам в соответствии с такими приоритетами как взвешенное наименьшее время выполнения (WSPT), наименьший директивный срок (EDD) и т.д. При этом веб-запрос описывается приоритетом, временем поступления на обработку, требуемой для его реализации памятью и директивным сроком выполнения. Новизна решения, рассмотренного в [2], заключается в использовании правила «наименьший зазор (Earliest Gap) – первый наименьшей директивный срок первый (Earliest Deadline First)» (EG-EDF) и табу поиска, основанного на идее заполнения «зазоров» в текущем расписании выполнения работ. Правило EG-EDF позволяет строить расписание для всех заданий последовательно, применяя технику, с помощью которой заполняются ранние имеющиеся «зазоры» (EG) в расписании для множества вновь прибывших заданий; в случае, если зазора для поступающей работы нет, то в нем используется второе правило – EDF – для включения новой работы в существующее расписание.

В данном исследовании для распределенной вычислительной системы (Грид-системы, вычислительного кластера) предлагается метод оптимизации локальных расписаний, используемый для подхода к планированию, рассмотренного в [3], и предусматривающего применение пакетного планирования: из входного потока работ формируется пул, величина которого определяется характеристиками распределенной среды и поступивших на обработку работ; далее планировщик планирует (назначает) ресурсы для их выполнения в соответствии с моделью метода о наименьшем покрытии и отправляет работы на выбранные таким образом ресурсы также в виде множества пакетов, упорядоченных и организованных в виде очередей. Если в модели описания задания используется директивный срок, то для построения локальных расписаний выполнения работ, составляющих очереди на множестве ресурсов распределенной системы, представляется возможным использование методов построения расписаний на одиночном устройстве (приборе), учитывая специфику описанного подхода к планированию. Построение локальных расписаний, с формальной точки зрения, может быть сведено к классической задаче минимизации суммарного взвешенного и невзвешенного запаздывания работ на одиночном устройстве (приборе).

Постановка задачи

Предположим, что на одиночное устройство (вычислительный ресурс) поступает множество независимых работ $J = \{J_1, J_2, ..., J_n\}$, каждая из которых будет непрерывно (без прерываний) выполняться на нем. При этом известны продолжительность выполнения каждого задания L_j и директивный срок его выполнения d_j. Необходимо определить такой порядок (последовательность) выполнения всех заданий, одновременно поступающих на устройство, при котором минимизируется суммарное время запаздывания всех работ входной очереди

$$TT_S = \sum_{\{j \in S\}} \max\left(0, C_j - d_j\right),$$

где C_j – время завершения работы j. Для решения этой задачи, обозначаемой $1||\sum T_i$, в [4] предложен метод на основе рангового подхода к построению кратчайшего гамильтонова пути в полносвязном графе, имеющий временную сложность $\mathbf{O}(n^3)$.

Задача минимизации суммарного взвешенного времени запаздывания на одиночном устройстве

$$TWT_S = \sum_{\{j \in S\}} \max\left(0, \left(C_j - d_j\right) \cdot w_j\right)$$

($1||\sum w_i T_i$) формулируется следующим образом. Множество работ, проиндексированных от 1 до n, должно быть выполнено без прерываний на одиночном устройстве, обрабатывающем только одну работу в определенный момент времени. Все работы поступают на устройство в момент времени t=0 (или одновременно). Работа характеризуется временем ее обработки L_i (или процессорным временем p_i), директивным сроком d_i и весом w_i. Для удобства все работы упорядочены согласно правилу EDD (Earliest Deadline First, работа с наименьшим директивным сроком первая), так что $d_i < d_j$; если $d_i = d_j$, тогда $p_i < p_j$; если $p_i = p_j$, тогда $w_i > w_j$ для всех i, j (i < j). Если работа i завершается после директивного срока d_i, результатом является взвешенный штраф за запаздывание.

Для обоснования целесообразности использования правил доминирования и ее экспериментального подтверждения рассмотрим пример, иллюстрирующий работу алгоритма на основе произвольного полносвязного графа с 4-мя вершинами, которые определяют работы тестовой последовательности, суть метода [4] и использование в нем правила доминирования.

Пусть есть множество работ J_1 (4 , 7), J_2 (5, 6), J_3 (3 , 7), J_4 (3 , 4), для которого необходимо построить расписание. Каждая работа J характеризуется двумя параметрами: длительностью L и директивным сроком d, и, таким образом, представляется как J (L, d). Граф, отображающий данное множество работ, представлен на рис. 1.

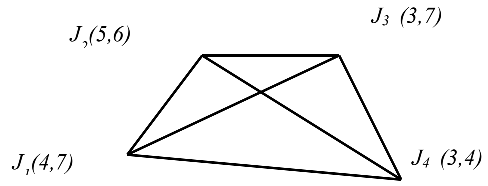

Рис. 1. Полносвязный граф,
состоящий из 4-х вершин (работ)

Для пояснения сущности и особенностей базового алгоритма [4] используем матрицу, в которой столбцам соответствуют ранги, а строкам – работы, для которых нужно построить расписание (график выполнения) (см. табл. 1).

Таблица 1

Пример работы алгоритма без использования правила доминирования

Вершина	Ранг 1	Ранг 2	Ранг 3	Ранг 4
J_1 (4;7)	$T(1)_1 =$ $=(0 + 4) - 7 = -3$	$T(2)_{2,1} = (5 + 4) - 7 = 2$ $\mathbf{T(2)_{3,1} = (3 + 4) - 7 = 0}$ $T(2)_{4,1} = (3 + 4) - 7 = 0$	$T(3)_{3,2,1} = (8 + 4) - 7 = 5$ $T(3)_{4,3,1} = (6 + 4) - 7 = 3$ $\mathbf{T(3)_{3,4,1} = (6 + 4) - 7 = 3}$	$\mathbf{TT_{4,3,2,1} = 0 + 0 + 5 + 8 = 13}$
J_2 (5;6)	$T(1)_2 =$ $=(0 + 5) - 6 = -1$	$T(2)_{1,2} = (4 + 5) - 6 = 3$ $\mathbf{T(2)_{3,2} = (3 + 5) - 6 = 2}$ $T(2)_{4,2} = (3 + 5) - 6 = 2$	$T(3)_{3,1,2} = (7 + 5) - 6 = 6$ $\mathbf{T(3)_{4,3,2} = (6 + 5) - 6 = 5}$ $T(3)_{3,4,2} = (6 + 5) - 6 = 5$	$TT_{3,4,1,2} = 0 + 0 + 3 + 9 = 14$ $TT_{3,1,4,2} = 0 + 0 + 6 + 9 = 15$
J_3 (3;7)	$T(1)_3 =$ $=(0 + 3) - 7 = -4$	$T(2)_{1,3} = (4 + 3) - 7 = 0$ $T(2)_{2,3} = (5 + 3) - 7 = 1$ $\mathbf{T(2)_{4,3} = (3 + 3) - 7 = -1}$		
J_4 (3;4)	$T(1)_4 =$ $=(0 + 3) - 4 = -1$	$T(2)_{1,4} = (4 + 3) - 4 = 3$ $T(2)_{2,4} = (5 + 3) - 4 = 4$ $\mathbf{T(2)_{3,4} = (3 + 3) - 4 = 2}$	$\mathbf{T(3)_{3,1,4} = (7+3) - 4 = 6}$ $T(3)_{3,2,4} = (8+3)-4=7$	

Для определения величины текущего запаздывания работ на каждом ранге используем обозначение $T(J)\ j_1j_2\ldots j_n$, где J – количество рангов (столбцов) текущего пути, $j_1j_2,\ldots\ j_n$ – вершины графа (работы), входящие в текущий путь к какой-либо вершине (работе) на текущем ранге (столбце) матрицы.

Таким образом, суммарное запаздывание всех работ TT будет определяться на последнем ранге.

В начальном состоянии работы не упорядочены, запаздывание каждой из них равно 0.

В первой строке матрицы ранга 1 строятся все пути от работ J_2, J_3, J_4 к работе J_1, во второй – пути от работ J_1, J_3, J_4 к работе J_2, в третьей – пути от работ J_1, J_2, J_4 к работе J_3, в четвертой – пути от работ J_1, J_2, J_3 к работе J_4.

Далее, в соответствии с базовым алгоритмом, для каждой строки необходимо выбрать минимальные пути: для строки 1 – T(2)3,1 =3+(4-7) = 0 и T(2)4,1 = 3+(4-7) = 0; для строки 2 – T(2)3,2 = = 3+(5-6) = 2 и $T(2)_{4,2}$ = 3+(5-6) = 2; для строки 3 – $T(2)_{4,3}$ = 3+(3-7) = -1; для строки 4 – $T(2)_{3,4}$ = 3+(3-4) = = 2. После этого на ранге 3 выбираем: в первой строке – $T(3)_{4,3,1}$ = 6+(4-7) = 3 и $T(3)_{3,4,1}$ = 6+(4-7) = 3; во второй строке – $T(3)_{4,3,2}$ = 6+(5-6) = 5 и $T(3)_{3,4,2}$ = = 6+(5-6) = 5; в третьей строке – $T(3)_{4,1,3}$ = 7+(3-7) = = 3; в четвертой строке – $T(3)_{3,1,4}$ = 7+(3-4) = 6. Далее строим пути из вершин ранга 3 к вершинам ранга 4: для строки 1 – $TT_{4,3,2,1}$ = 0+0+5+8 = 13; для строки 2 – $TT_{4,3,1,2}$ = 0+0+3+9 = 12 и $TT_{4,1,3,2}$ = = 0+0+3+9 = 12. Таким образом, кратчайший гамильтонов путь в графе на рис. 1 включает последовательность вершин (работ) $J_4J_3J_1J_2$ (в табл. 1 и далее в табл. 2 выделены минимальные значения текущего запаздывания на рангах).

Из рассмотренного в табл. 1 примера следует, что на рангах матрицы имеются «конкурирующие» (имеющие одинаковое текущее запаздывание) пути, из которых требуется выбрать только один – случайным образом или при помощи правила доминирования, которое позволяет получить локально-оптимальное решение, – или же при помощи «стягивания» все сформированных путей к следующему рангу, то есть использования всех построенных на текущем ранге путей без отсекания на следующем. В данном исследовании в случае наличия одинаковых значений текущего запаздывания для нескольких путей на ранге предлагается использовать правило доминирования – например, EDD: на ранге 3 решением для путей T(3)4,3,1 = 6+(4-7) = 3 и T(3)3,4,1 = 6+(4-7) = 3 является выбор T(3)4,3,1, потому что $d_4 < d_3$. Пример использования правила EDD для графа на рис. 1 приведен в табл. 2.

Таблица 2

Пример работы алгоритма с правилом EDD

Вершина	Ранг 1	Ранг 2	Ранг 3	Ранг 4
J_1 (4;7)	$T(1)_1 = (0 + 4) - 7 = -3$	$T(2)_{2,1} = (5 + 4) - 7 = 2$ $T(2)_{3,1} = (3 + 4) - 7 = 0$ $\mathbf{T(2)_{4,1} = (3 + 4) - 7 = 0}$	$T(3)_{4,2,1} = (8 + 4) - 7 = 5$ $T(3)_{4,3,1} = (6 + 4) - 7 = 3$ $\mathbf{T(3)_{3,4,1} = (6 + 4) - 7 = 3}$	$TT_{3,4,2,1} = 0+2+5+8 = 15$
J_2 (5;6)	$T(1)_2 = (0 + 5) - 6 = -1$	$T(2)_{1,2} = (4 + 5) - 6 = 3$ $T(2)_{3,2} = (3 + 5) - 6 = 2$ $\mathbf{T(2)_{4,2} = (3 + 5) - 6 = 2}$	$T(3)_{4,1,2} = (7 + 5) - 6 = 6$ $T(3)_{4,3,2} = (6 + 5) - 6 = 5$ $\mathbf{T(3)_{3,4,2} = (6 + 5) - 6 = 5}$	$\mathbf{TT_{4,3,1,2} = 0+0+3+9 = 12}$ $TT_{4,1,3,2} = 0+0+3+9 = 12$
J_3 (3;7)	$T(1)_3 = (0 + 3) - 7 = -4$	$T(2)_{1,3} = (4 + 3) - 7 = 0$ $T(2)_{2,3} = (5 + 3) - 7 = 1$ $\mathbf{T(2)_{4,3} = (3 + 3) - 7 = -1}$	$\mathbf{T(3)_{4,1,3} = (7 + 3) - 7 = 3}$ $T(3)_{4,2,3} = (8 + 3) - 7 = 4$	
J_4 (3;4)	$T(1)_4 = (0 + 3) - 10 = -7$	$T(2)_{1,4} = (4 + 3) - 4 = 3$ $T(2)_{2,4} = (5 + 3) - 4 = 4$ $\mathbf{T(2)_{3,4} = (3 + 3) - 4 = 2}$		

Сравнительный анализ полученных в табл. 1 и табл. 2 значений суммарного запаздывания показывает, что использование правила доминирования позволяет уменьшить суммарное запаздывание: при использовании правила оно составляет 12, без – 13.

Теоретические положения обоснования целесообразности использования правила доминирования

Для обоснования целесообразности использования правила доминирования в базовом алгоритме [4] используем следующие определения и предложения.

Определение 1. Текущей последовательностью работ $\sigma(R)$ на ранге R с длиной расписания $T_\sigma(R)$ из частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ является последовательность, формируемая на шаге, соответствующем рангу R, и состоящей из работ, которые выбраны алгоритмом для ранга R-1, и работы на ранге R.

Определение 2. Текущую последовательность работ $\sigma(R)$ на ранге R с длиной расписания $T_\sigma(R)$ из частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ назовем «лучшей», чем текущая последовательность $T'_\sigma(R)$, если выполняется условие $T_\sigma(R) - T'_\sigma(R) < 0$ ($T_\sigma(R) < T'_\sigma(R)$).

Определение 3. Текущую последовательность работ $\sigma(R)$ на ранге R с длиной расписания $T_\sigma(R)$ из частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ назовем «эквивалентной» текущей последовательности $T'_\sigma(R)$, если выполняется условие $T_\sigma(R) - T'_\sigma(R) = 0$ ($T_\sigma(R) = T'_\sigma(R)$).

Определение 4. Рассмотрим работы J_i и J_k, которые запланированы на предыдущем по отношению к текущему ранге. При условии $T_\sigma(R) - T'_\sigma(R) = 0$ (условие в определении 3), оптимальным является то расписание, в котором J_i предшествует J_k по типу доминирования [5]:

отношение глобального предшествования – работы располагаются в определенном порядке, который определяется по принятым правилам доминирования (невзвешенный и взвешенный случаи);

отношение безусловного предшествования – работы располагаются в таком порядке, что их перестановка увеличивает критерий или определяет возможность применения определенного правила доминирования;

отношение предшествования, зависимого от времени – работы располагаются в таком порядке, что их перестановка увеличивает критерий или определяет возможность применения определенного правила доминирования и зависит от момента начала обработки работ.

Пусть $WT_\sigma(R)$ – взвешенное запаздывание текущей последовательности работ на ранге R для частной последовательности работ σ, $C_{max}(\sigma)$ – максимальное значение времени завершения работ в частной последовательности σ. Для формулировки приведенных далее определений 5, 6 используем результаты работ [6, 7].

Определение 5. Если $C_{max}(\sigma_1) \le C_{max}(\sigma_2)$ – максимальные значения времени завершения работ в частных последовательностях σ_1 и σ_2, $WT(\sigma_1) \le WT(\sigma_2)$, σ_1, σ_2 – частные последовательности работ из последовательности $\sigma = j_1, j_2, \ldots, j_n$, то можно в любом допустимом расписании использовать последовательность σ_1 вместо последовательности σ_2 с оценкой «последовательность σ_1 не хуже последовательности σ_2 по времени завершения всех работ».

Определение 6. Если $C_{max}(\sigma_1) < C_{max}(\sigma_2)$ и $WT(\sigma_1) < WT(\sigma_2)$, σ_1, σ_2 – частные последовательности работ из последовательности $\sigma = j_1, j_2, \ldots, j_n$, то можно в любом допустимом расписании использовать последовательность σ_1 вместо последовательности σ_2 с оценкой «последовательность σ_1 лучше последовательности σ_2 по времени завершения всех работ».

Для оценки расписаний, получаемых на основе локально-оптимальных решений, сформулируем следующие предложения.

Предложение 1. Рассмотрим работы J_i и J_k, которые должны быть запланированы на произвольном ранге R. Расписание, в котором J_i предшествует J_k, является локально-оптимальным, если выполняется неравенство $T_{ik} - T_{ki} \le 0$ ($T_{ik} \le T_{ki}$).

Предложение 2 (взвешенный случай). Рассмотрим работы J_i и J_k, которые должны быть запланированы на произвольном ранге R. Расписание, в котором J_i предшествует J_k, является локально-оптимальным решением, если выполняется неравенство $WT_{ik} - WT_{ki} \le 0$ ($WT_{ik} \le WT_{ki}$).

Предложение 3. Расписание частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ на произвольном ранге R является (k,R)-оптимальным, если оно включает k ($R \ge k \ge 1$) локально-оптимальных решений.

Предложение 4. Расписание $TT(\sigma)$ частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ является (k,n)-оптимальным, если оно включает k ($n \ge k \ge 1$) локально-оптимальных решений.

Предложение 5 (взвешенный случай). Расписание $TWT(\sigma)$ частной последовательности работ $\sigma = j_1, j_2, \ldots, j_n$ является (k^w, n)-оптимальным, если оно включает k^w ($n \ge k^w \ge 1$) локально-оптимальных решений.

Предложение 6. Расписание $TT(\sigma_1)$ частной последовательности работ σ_1 доминирует над расписанием $TT(\sigma_2)$ частной последовательности работ σ_2, если выполняются условия:

расписания для последовательностей σ_1 и σ_2 являются оптимальными:

$TT(\sigma_1)$ является (k_1, n)-оптимальным, $TT(\sigma_2)$ является (k_2, n)-оптимальным;

$TT(\sigma_1) \le TT(\sigma_2); k_1 \ge k_2$.

Предложение 7. Расписание $TT(\sigma_1)$ частной последовательности работ σ_1 **строго** доминирует над расписанием $TT(\sigma_2)$ частной последовательности работ σ_2, если выполняются условия:

расписания для последовательностей σ_1 и σ_2 являются оптимальными:

$TT(\sigma_1)$ является (k_1,n)-оптимальным, $TT(\sigma_2)$ является (k_2,n)-оптимальным;

$TT(\sigma_1)< TT(\sigma_2);k_1> k_2$.

Предложение 8 (взвешенный случай). Расписание $TWT(\sigma_1)$ частной последовательности работ σ_1 доминирует над расписанием $TWT(\sigma_2)$ частной последовательности работ σ_2, если выполняются условия:

расписания последовательностей σ_1 и σ_2 являются оптимальными:

$TWT(\sigma_1)$ является $(k^w{}_1,n)$-оптимальным, $TWT(\sigma_2)$ является $(k^w{}_2,n)$-оптимальным;

$TWT(\sigma_1) \le TWT(\sigma_2);k^w{}_1 \ge k^w{}_2$.

Предложение 9 (взвешенный случай). Расписание $TWT(\sigma_1)$ частной последовательности работ σ_1 **строго** доминирует над расписанием $TWT(\sigma_2)$ частной последовательности работ σ_2, если выполняются условия:

расписания последовательностей σ_1 та σ_2 являются оптимальными:

$TWT(\sigma_1)$ является $(k^w{}_1,n)$-оптимальным, $TWT(\sigma_2)$ является $(k^w{}_2, n)$-оптимальным;

$TWT(\sigma_1) < TWT(\sigma_2);\ k^w{}_1 > k^w{}_2$.

Алгоритм минимизации суммарного запаздывания с использованием правила доминирования

Для использования правила доминирования в базовом алгоритме (алгоритме оптимизации по направлению [4]) обобщим его с учетом введенных определений и предложений.

Шаг 1. Формируем произвольную последовательность работ из исходного множества. Эта последовательность составляет работы ранга 1.

Шаг 2. Строим все пути ранга 2; на текущем ранге для каждой работы выбираем пути с минимальной величиной текущего запаздывания. Если их несколько, выбираем тот путь, для которого расписание является доминирующим, то есть включает оптимальную последовательность работ на ранге (рангах). В случае, если таких путей несколько, выбираем тот путь, для которого расписание является строго доминирующим.

Шаг 3. Строим все пути ранга 3 и т.д. до тех пор, пока не достигнем ранга n. Рассчитываем суммарное запаздывание работ для частной последовательности $\sigma = j_1,j_2,\dots,\ j_n$ и суммарное количество локально-оптимальных решений, то есть последовательности работ, построенных (выбранных) алгоритмом на ранге (рангах) по правилу доминирования.

Полученная последовательность работ σ_1 определяет строго оптимальное и доминирующее расписание, если для любых частных последовательностей σ_1 и σ_2 на разных рангах расписания являются оптимальными – $(k_{\sigma1},R_{\sigma1})$-оптимальным и $(k_{\sigma2},R_{\sigma2})$-оптимальным, причем $k_{\sigma1}\ge k_{\sigma1}$;

– для невзвешенного случая: $TT(\sigma_1)< TT(\sigma_2)$ и $k_1> k_2$, где k_1, k_2 – количество локально-оптимальных решений для последовательностей σ_1 и σ_2, соответственно, на ранге n;

– для взвешенного случая: $TWT(\sigma_1)< TWT(\sigma_2)$ и $k^w{}_1>k^w{}_2$ где $k^w{}_1$, $k^w{}_2$ – количество локально-оптимальных решений для последовательностей σ_1 и σ_2, соответственно, на ранге n.

Метрики оценки улучшения результатов работы алгоритма при использовании правила доминирования

Для оценки улучшения результатов работы базового алгоритма при использовании правила доминирования предлагаются следующие метрики:

отношение количества вершин построенного кратчайшего гамильтонова пути в графе (см. рис. 1), полученного с использованием правила доминирования, к общему количеству вершин в графе – плотность

$$\sum_{GP} v_{dr} \Big/ n ,$$

где v_{dr} – множество вершин графа, полученных с использованием правила доминирования, n – общее количества вершин в графе;

относительное уменьшение суммарного запаздывания

$$(TT - TT_{dr}) / TT ,$$

где TT_{dr} – суммарное запаздывание с использованием в алгоритме правила доминирования; TT – суммарное запаздывание без использования правила доминирования.

Для экспериментального исследования влияния правил доминирования на время работы базового алгоритма и величину суммарного запаздывания проведены вычислительные эксперименты на исходных данных с таким количеством работ, которое встречается на практике. Для этого использован алгоритм [8] и сгенерированы пакеты из 75 заданий (instances, наблюдений) по 20 –160 работ в каждом и рассчитаны относительное уменьшение суммарного времени запаздывания (рис. 2) и количество локально-оптимальных решений, полученных с использованием правил доминирования (рис. 3).

Полученные результаты показывают, для некоторых характеристик заданий относительное уменьшение суммарного запаздывания может составлять до 25%, а количество получаемых локально-оптимальных решений может достигать 50%, что подтверждает целесообразность использования правил доминирования в базовом алгоритме.

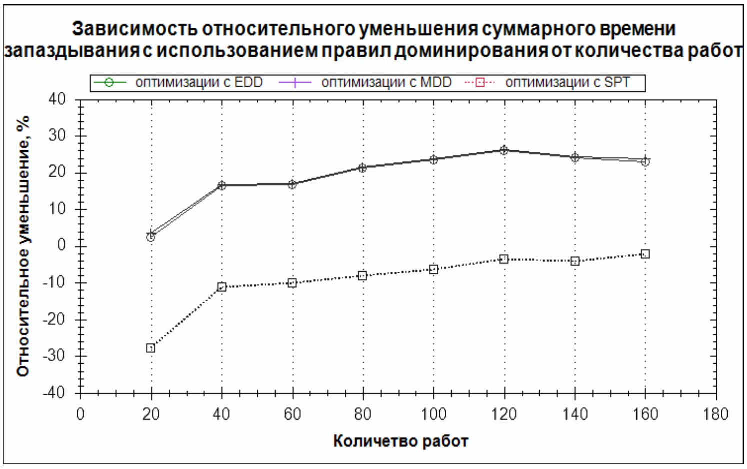

Рис. 2. Зависимость относительного уменьшения суммарного времени запаздывания при использовании правил доминирования EDD, MDD, SPT по отношению к базовому алгоритму от количества работ, %

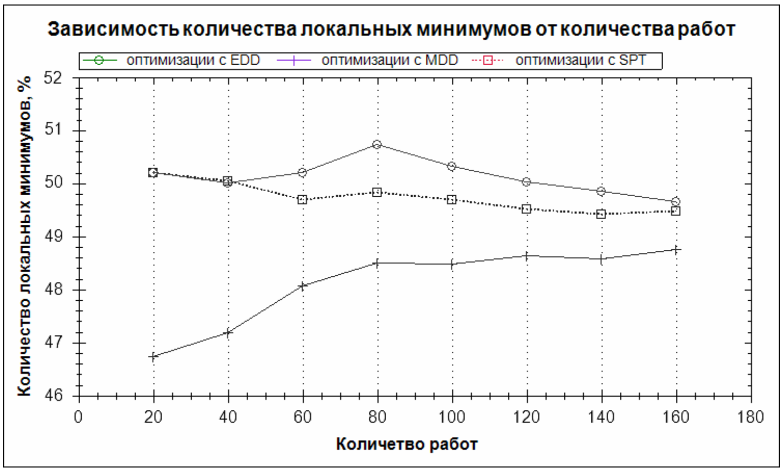

Рис. 3. Зависимость количества локально-оптимальных решений, полученных с использованием правил доминирования EDD, MDD, SPT, от количества работ, %

Выводы

Рассмотрен метод минимизации суммарного запаздывания на одиночном устройстве на основе рангового подхода и правил доминирования, имеющий малую временную сложность, что позволяет использовать его для построения локальных расписаний в распределенных вычислительных системах. Обоснована целесообразность использования в данном методе правила доминирования, позволяющего улучшить получаемые расписания выполнения работ. Сформулированы определения и предложения, в целом, определяющие свойства оптимальности метода при использовании правила доминирования для выбора работы строящегося расписания на каждом ранге. Приведены результаты экспериментального исследования алгоритма на основе предложенных метрик оценки улучшения его результатов, подтверждающие целесообразность применения правил доминирования в исследуемом базовом алгоритме.

Дальнейшие исследования предполагается провести в направлении автоматического выбора правила доминирования в зависимости от характеристик поступающих в систему работ.

Список литературы

1. *Nong Ye, Esma S. Gel, Xueping Li, Toni Farley, Ying-Cheng Lai. Web server QoS models: applying scheduling rules from production planning // Computers & Operations Research, 32 (2005), p. 1147–1164.*
2. *Klusacek D., Rudova H. Improving QoS in computational Grids through schedule-based approach. // In Scheduling and Planning Applications Workshop at the Eighteenth International Conference on Automated Planning and Scheduling (ICAPS 2008). 2008: Sydney, Australia.*
3. *Листровой С.В. Модель и подход к планированию распределения ресурсов в гетерогенных ГРИД-системах / С.В. Листровой, С.В. Минухин. // Проблемы управления и информатики: Международный научно-технический журнал. – 2012. – № 5. – С. 120–133.*
4. *Мінухін С.В. Метод мінімізації часу виконання завдань з директивними строками на некластеризованому ресурсі обчислювальної системи. / С.В. Мінухін // Інформаційно-керуючі системи на залізничному транспорті. – 2009. – № 3. – С. 47 – 53.*
5. *Шевченко К.Ю. Алгоритм гілок та меж для статистичних досліджень нового ПДС-алгоритму розв'язання задачі мінімізації сумарного зваженого запізнення виконання робіт на одному приладі / К.Ю. Шевченко // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: Зб. наук. пр. – К.: Век+. – 2012. – № 56. – С.59–69.*

6. Chu C. Efficient heuristics to minimize total flow time with release dates. // Operations Research Letters. – 1992. – №12. – p. 321–330.

7. Jouglet A. Dominance Rules for the Parallel Machine Total Weighted Tardiness Scheduling Problem with Release Dates. / A. Jouglet, D. Savourey. // Computers & Operations research. – 2011. – V. 38. – Issue 9. – p. 1259–1266.

8. OR-Library [Электронный ресурс]. – Режим доступа : http://people.brunel.ac.uk/~mastjjb/jeb/orlib/wtinfo.html.

Поступила в редколлегию 10.01.2014

Рецензент: д-р техн. наук, проф. О.Г. Руденко, Харьковский национальный университет радиоэлектроники, Харьков.

ПРО ВЛАСТИВОСТІ ОПТИМАЛЬНОСТІ МЕТОДУ МІНІМІЗАЦІЇ СУМАРНОГО ЗАПІЗНЮВАННЯ НА ОДИНОЧНОМУ ПРИСТРОЇ НА ОСНОВІ РАНГОВОГО ПІДХОДУ І ПРАВИЛ ДОМІНУВАННЯ

С.В. Мінухін

Розглянуто метод мінімізації сумарного запізнювання робіт на одиночному пристрої, зведений до вирішення задачі знаходження найкоротшого гамільтонового шляху в довільному повнозв'язному графі на основі рангового підходу і правил домінування. Розглянуто властивості оптимальності методу на основі введених визначень локально-оптимального рішення, локально-оптимального розкладу на ранзі і оптимальності одержуваного підсумкового розкладу. Сформульовано визначення та пропозиції, що визначають оптимальність одержуваних розкладів виконання робіт на основі домінуючих і строго домінуючих розкладів для зваженого і незваженого випадків. Запропоновано метрики оцінки поліпшення результатів роботи алгоритму при використанні правила домінування. Наведено результати обчислювального експерименту з оцінювання поліпшення роботи алгоритму, які підтверджують доцільність використання правил домінування.

***Ключові слова**: розклад, робота, ранговий підхід, правило домінування, послідовність, метрика, граф.*

ON THE OPTIMALITY PROPERTIES OF A METHOD TO MINIMIZE TOTAL TARDINESS ON A SINGLE MACHINE BASED ON THE RANK APPROACH AND THE DOMINANCE RULES

S.V. Minukhin

The method of minimizing the total delay work on the single device pivoted to the problem of finding the shortest Hamiltonian path in a random fully connected graph based on the rank approach and the dominance rules is considered. The properties of an optimal method based on the introduced definitions of locally optimal solutions for the optimal schedule on the rank and the optimality of the final schedule are considered. Definitions and propositions that define the optimality of job schedules on the basis of strictly dominant and dominant schedules for the weighted and unweighted cases are formulated. Metrics evaluation to improve the results of the algorithm using the dominance rules are proposed. The results of computer simulation for estimating improvement of the algorithm, confirming the expediency of the dominance rules.

***Keywords**: schedule, job, rank approach, dominance rule, metric, graph.*