МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

ПОЛЕССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Инженерный факультет

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

МАТЕРИАЛЫ

II международной научно–практической конференции “Инжиниринг: теория и практика”

**Полесский государственный университет,
г. Пинск, Республика Беларусь,
6 мая 2022 г.**

Пинск 2022

УДК 62:658
ББК 65:38
И 62

Инжиниринг: теория и практика: материалы II международной научно–практической конференции, УО “Полесский государственный университет”, г. Пинск, 6 мая 2022 г. / Министерство образования Республики Беларусь [и др.]; редкол.: В.И. Дунай [и др.]. – Пинск: ПолесГУ, 2022. – 120 с.

ISBN 978–985–516–729-8

Приведены материалы участников II международной научно–практической конференции “Инжиниринг: теория и практика”.

Материалы изложены в авторской редакции.

УДК 62:658
ББК 65:38

ISBN 978–985–729-8

УДК 681.3.06

ЗАДАЧА ОПТИМИЗАЦИИ ЧИСЛА ПРОЦЕССОРОВ В МАСШТАБИРУЕМЫХ РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

Коваленко Николай Семенович, д.ф.–м.н., профессор
Белорусский государственный университет
Павлов Павел Александрович, к.ф.–м.н., доцент
Полесский государственный университет
Kovalenko Nikolay Semenovich, D.Sc., kovalenkons@rambler.ru
Belarus State University
Pavlov Pavel Aleksandrovich, PhD, pavlov.p@polessu.by
Polessky State University

Предлагается метод оптимизации числа процессоров при распределенной обработке конкурирующих процессов в параллельных системах.

Введение. Постоянное существование задач сверхвысокой сложности: задачи проектирования сложных систем (ракетной техники, самолетов); задачи оптимизационного плана развития экономики страны или отдельного региона, сооружений, технологических процессов; задачи эффективного использования спутников Земли для развития народного хозяйства; задачи военного характера и др., характеризуются большой размерностью, десятками сотен и миллионов независимых переменных и соответствующих ограничений. Указанные задачи можно эффективно решать используя идеи распараллеливания сложных вычислительных процессов и обработки больших объемов данных и знаний с помощью параллельных многопроцессорных систем (МС) и вычислительных комплексов (ВК), “объединяя в единое целое сведения из таких областей, как архитектура компьютеров и вычислительных систем, системное программирование и языки программирования, различные методы обработки информации” [1].

Основные понятия и определения. Как и в [3–5] *процесс* будем рассматривать как последовательность блоков $Q_1, Q_2, \ldots, Q_s$, для выполнения которых используется множество процессоров. При этом процесс будем считать *распределённым*, если все блоки или часть из них выполняются на разных процессорах. Процессы, которые для ускорения выполнения обрабатываются параллельно, взаимодействуя путем обмена информацией, будем называть *кооперативными* или *взаимодействующими* процессами. Последовательность программных блоков, которую необходимо процессорам выполнять многократно, будем называть *программным ресурсом PR*, а множество соответствующих процессов – *конкурирующими***.**

Математическая модель системы распределенной обработки конкурирующих процессов включает в себя: s, $s \geq 2$ – число блоков линейно–структурированного программного ресурса $PR = (Q_1, Q_2, \ldots, Q_s)$; n, $n \geq 2$ – число распределенных относительно PR конкурирующих процессов; p, $p \geq 2$ – число процессоров многопроцессорной системы; матрицу $T = [t_{ij}]$ времен выполнения *j*–х блоков *i*–ми конкурирующими процессами $i = \overline{1,n}$, $j = \overline{1,s}$; ε – время, характеризующее дополнительные системные расходы по организации структурирования и параллельного использования блоков PR.

Определение 1. Распределенная система *n* взаимодействующих конкурирующих процессов называется *неоднородной*, если времена выполнения блоков PR зависят от объемов обрабатываемых данных и/или их структуры, т.е. разные для разных процессов.

Как и в [3,4] будем считать, что взаимодействие процессов, процессоров и блоков линейно–структурированного программного ресурса подчинено следующим условиям: 1) ни один из блоков PR не может обрабатываться одновременно более чем одним процессором; 2) ни один из процессоров не может обрабатывать одновременно более одного блока; 3) обработка каждого блока осуществляется без прерываний; 4) распределение блоков программного ресурса по процессорам МС для каждого из процессов осуществляется циклически по правилу: блок с номером $j = kp + i$, $j = \overline{1,s}$, $i = \overline{1,p}$, $k \ge 0$, распределяется на процессор с номером *i*; 5) отсутствуют простои процессоров при условии готовности блоков, а также невыполнение блоков при наличии процессоров; 6) для каждого из *n* процессов момент завершения выполнения *j*–го блока на *i*–м процессоре совпадает с моментом начала выполнения следующего $(j+1)$–го блока на $(i+1)$–м процессоре, $i = \overline{1, p-1}$, $j = \overline{1, s-1}$; 7) для каждого из блоков структурированного программного ресурса момент завершения его выполнения *l*–м процессом совпадает с моментом начала его выполнения $(l+1)$–м процессом на том же процессоре, $l = \overline{1, n-1}$.

Асинхронный режим взаимодействия процессоров, процессов и блоков предполагает отсутствие простоев процессоров МС при условии готовности блоков, а также невыполнение блоков при наличии процессоров и определяется условиями 1–5.

Условия 1–4, 6 определяют *первый синхронный режим*, обеспечивающий непрерывное выполнение блоков PR внутри каждого из процессов.

Второй синхронный режим, определяемый условиями 1–4, 7, обеспечивает непрерывное выполнение каждого блока всеми процессами.

Задача оптимизации числа процессоров. В [3–5] для вычисления минимального общего времени $T_н^{ac}(p,n,s,\varepsilon)$ выполнения $n \ge 2$ неоднородных распределенных конкурирующих процессов, использующих структурированный на $s \ge 2$ блоков программный ресурс в многопроцессорной системе с $p \ge 2$ процессорами с учетом параметра $\varepsilon > 0$ в случае *неограниченного параллелизма* ($2 \le s \le p$) был использован функционал задачи Беллмана–Джонсона, который имеет вид:

$$T_н^{ac}(p,n,s,\varepsilon) = \max_{1 \le u_1 \le u_2 \le \ldots \le u_{s-1} \le n} \left[\sum_{i=1}^{u_1} t_{i1}^{\varepsilon} + \sum_{i=u_1}^{u_2} t_{i2}^{\varepsilon} + \ldots + \sum_{i=u_{s-1}}^{n} t_{is}^{\varepsilon} \right],$$

где $t_{ij}^{\varepsilon} = t_{ij} + \varepsilon$, $i = \overline{1,n}$, $j = \overline{1,s}$, а u_1, u_2, ..., u_{s-1} – целые положительные числа.

Было также предложено графоаналитическое решение задачи определения $T_н^{ac}(p,n,s,\varepsilon)$, доказаны следующие теоремы.

Теорема 1. *Минимальное общее время выполнения n,* $n \ge 2$, *неоднородных распределенных конкурирующих процессов, использующих структурированный на s,* $s \ge 2$, *блоков программный ресурс с временами выполнения блоков, задаваемыми матрицей* $T^{\varepsilon} = [t_{ij}^{\varepsilon}]$, $i = \overline{1,n}$, $j = \overline{1,s}$, *в многопроцессорной системе с p,* $p \ge 2$, *процессорами в асинхронном режиме в случае* $2 \le s \le p$, *определяется длиной критического пути в сетевом вершинно–взвешенном графе* G_1^{ac} *из начальной вершины* t_{11}^{ε} *в конечную* t_{ns}^{ε}.

Теорема 2. *Минимальное общее время* $T_н^{ac}(p,n,s,\varepsilon)$ *выполнения n,* $n \ge 2$, *неоднородных распределенных конкурирующих процессов, использующих линейно структурированный на s,* $s \ge 2$, *блоков программный ресурс с временами выполнения блоков, задаваемыми матрицей* $T^{\varepsilon} = [t_{ij}^{\varepsilon}]$, $i = \overline{1,n}$, $j = \overline{1,s}$, *в многопроцессорной системе с p,* $p \ge 2$, *процессорами и до-*

полнительными системными расходами $\varepsilon > 0$, *в асинхронном режиме в случае* $s = kp + r$, $k \ge 1$, $1 \le r < p$, *определяется длиной критического пути из начальной вершины* t_{11}^{ε} *в конечную вершину* $t_{(k+1)n,(k+1)p}^{\varepsilon}$ *сетевого вершинно–взвешенного графа* G_2^{ac}.

Несомненно, время выполнения всех распределенных конкурирующих процессов $T_н^{ac}(p,n,s,\varepsilon)$ будет существенно зависеть от количества имеющихся процессоров. Задача состоит в том, чтобы при заданных *n*, *s*, ε, $[t_{ij}]$, $i = \overline{1,n}$, $j = \overline{1,s}$, и заданного директивного времени выполнения всех распределенных конкурирующих процессов *d*, найти оптимальное число процессоров p^*, обеспечивающих директивное время выполнения. Решение данной задачи рассмотрим для общего случая *асинхронного режима*, т.е. когда процессы являются *неоднородными*.

Для изложения метода решения поставленной задачи, кроме введенных выше параметров математической модели *p*, *n*, *s*, ε и *d*, нам понадобятся: M_q – двумерный массив переменной длины, составленный специальным образом из элементов матрицы $[t_{ij}^{\varepsilon}]$, где $t_{ij}^{\varepsilon} = t_{ij} + \varepsilon$, $i = \overline{1,n}$, $j = \overline{1,s}$, q, $q \in N$ – порядковый номер результирующей матрицы времен выполнения блоков (двумерного массива переменной длины M_q), а также приведенные ниже определение и теорема.

Определение 2. Число процессоров МС будем называть *достаточным* и обозначать p^s при заданных *n*, *s*, если $p^s = s$.

Обозначим через $T_н^{ac}(p^s,n,s,\varepsilon)$ минимальное общее время выполнения множества конкурирующих процессов при достаточном числе процессоров p^s, а $T_н^{ac}(p,n,s,\varepsilon)$ – минимальное общее время при исходном числе процессоров *p*. Имеет место теорема.

Теорема 3. *При заданных n, s,* ε, $[t_{ij}]$, $i = \overline{1,n}$, $j = \overline{1,s}$, *в случае достаточного* ($p^s = s$) *и ограниченного* ($p < s$) *числа процессоров МС имеет место соотношение* $T_н^{ac}(p^s,n,s,\varepsilon) \le T_н^{ac}(p,n,s,\varepsilon)$.

Теорема 3 является отправной точкой для построения метода нахождения оптимального числа процессоров p^*, обеспечивающих директивное время *d* выполнения неоднородных конкурирующих процессов при распределенной обработке в условиях асинхронного режима их взаимодействия.

Входные данные: p, $p \ge 2$ – заданное (исходное) число процессоров; n, $n \ge 2$ – число конкурирующих неоднородных распределенных процессов; s, $s \ge 2$ – число блоков линейно–структурированного программного ресурса; *M* – двумерный массив, содержащий элементы исходной матрицы с учетом дополнительных системных расходов ε $[t_{ij}^{\varepsilon}]$, $i = \overline{1,n}$, $j = \overline{1,s}$; *d* – заданное (директивное) время выполнения конкурирующих процессов.

Выходные данные: p^* – минимальное (оптимальное) число процессоров, обеспечивающих выполнение конкурирующих процессов за директивное время *d*; M_q – двумерный массив, содержащий результирующую матрицу времен выполнения блоков программного ресурса вида (1); q – порядковый номер результирующей матрицы времен выполнения блоков программного ресурса PR.

Метод.

Если $d < T_H^{ac}(p^s, n, s, \varepsilon)$, то полагаем $p^* = 0$, т.е. директивное время выполнения конкурирующих процессов d не может быть реализовано в заданных условиях ни для какого числа процессоров.

Пусть $d \ge T_H^{ac}(p^s, n, s, \varepsilon)$ и число процессоров МС является ограниченным, т.е. $s > p$. Тогда между d, $T_H^{ac}(p^s, n, s, \varepsilon)$, $T_H^{ac}(p < s, n, s, \varepsilon)$ возможны следующие случаи:

- если $T_H^{ac}(p^s, n, s, \varepsilon) \le d = T_H^{ac}(p, n, s, \varepsilon)$ или $d > T_H^{ac}(p, n, s, \varepsilon)$, то нахождение p^* осуществляется методом деления пополам отрезка $[2, p]$;
- если $T_H^{ac}(p^*, n, s, \varepsilon) \le d < T_H^{ac}(p, n, s, \varepsilon)$, то нахождение p^* осуществляется методом деления пополам отрезка $[p, p^s]$.

Пусть $s \le p$. Тогда нахождение p^* осуществляется методом деления отрезка $[2, p^s]$ пополам.

Нетрудно подсчитать, что сложность алгоритма нахождения оптимального числа процессоров p^*, базирующегося на предложенном методе, составляет величину $O((k+1)np\log_2 p)$ операций в худшем случае.

На рис.1 приводится графическая интерпретация зависимости величины $T_H^{ac}(p, n, s, \varepsilon)$ от числа процессоров p, а также указаны величины d, $T_H^{ac}(p^s, n, s, \varepsilon)$, p^* и p^s. Из рисунка видно, что величина p^* определяется либо как точка пересечения прямой d с дискретной линией, определяющей зависимость $T_H^{ac}(p, n, s, \varepsilon)$ от p, либо как ближайшая точка, которая находится ниже прямой d.

Пример. Пусть $p = 3$, $n = 3$, $s = 9$, $d = 48$, а исходная матрица времен выполнения блоков с учетом дополнительных системных расходов по организации структурирования и параллельного использования блоков PR ε имеет вид:

$$T^{\varepsilon} = M = \begin{bmatrix} 4 & 1 & 3 & 5 & 2 & 4 & 7 & 3 & 1 \\ 2 & 6 & 4 & 1 & 5 & 3 & 4 & 2 & 8 \\ 5 & 3 & 1 & 7 & 4 & 2 & 6 & 4 & 5 \end{bmatrix}.$$

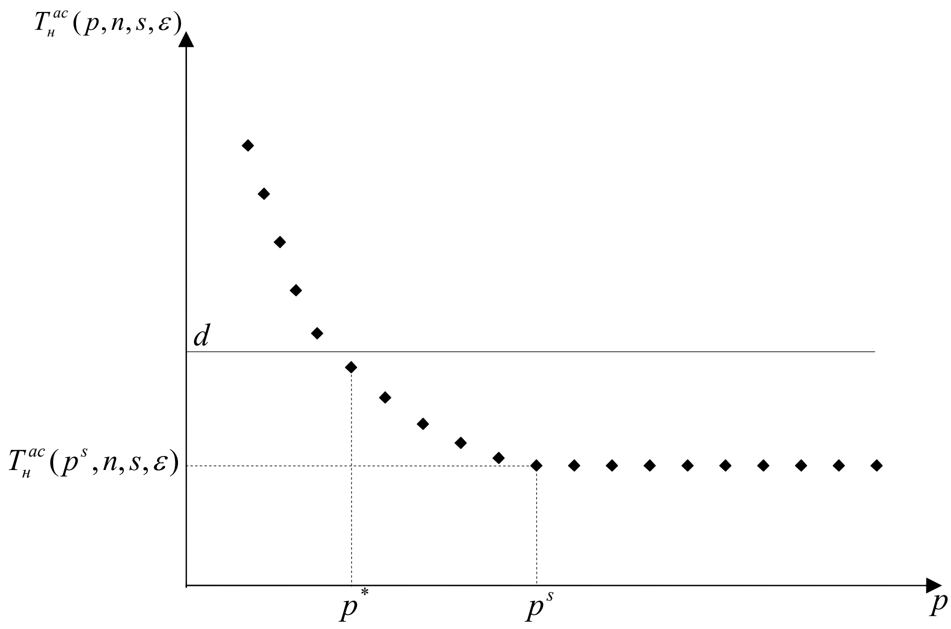

Рисунок 1. – Зависимости $T_н^{ac}(p,n,s,\varepsilon)$ от числа процессоров

В данном случае достаточное число процессоров $p^s = 9$.

- По исходной матрице M строим вершинно–взвешенный граф G_1^{ac} (Рис.2) и находим величину $T_H^{ac}(p^s = 9, n, s, \varepsilon) = 45$.

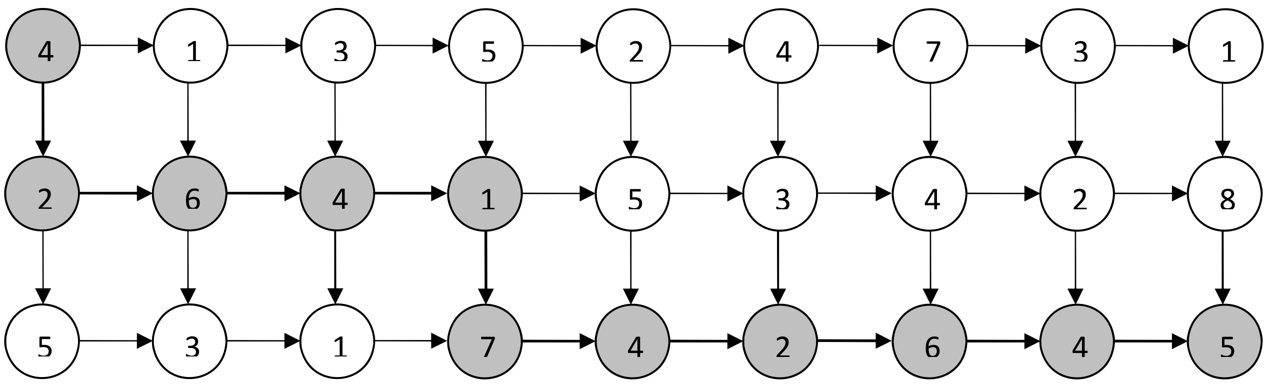

Рисунок 2. – Вершинно–взвешенный граф G_1^{ac}

По исходным данным p, n, s и M строим результирующую матрицу T^* вида:

$$T^* = \begin{bmatrix} \begin{bmatrix} \hat{4} & 1 & 3 \\ \hat{2} & \hat{6} & \hat{4} \\ 5 & 3 & 1 \end{bmatrix} & \begin{bmatrix} 5 & 2 & 4 \\ \hat{1} & 5 & 3 \\ \hat{7} & 4 & 2 \end{bmatrix} & \begin{bmatrix} 7 & 3 & 1 \\ 4 & 2 & 8 \\ 6 & 4 & 5 \end{bmatrix} \\ \begin{bmatrix} 5 & 2 & 4 \\ 1 & 5 & 3 \\ 7 & 4 & 2 \end{bmatrix} & \begin{bmatrix} \hat{7} & 3 & 1 \\ \hat{4} & 2 & 8 \\ \hat{6} & \hat{4} & \hat{5} \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 7 & 3 & 1 \\ 4 & 2 & 8 \\ 6 & 4 & 5 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{bmatrix}.$$

С помощью этой матрицы определяем величину $T_H^{ac}(p = 3, n, s, \varepsilon) = 50$, которая и будет определять минимальное общее время выполнения неоднородных распределенных конкурирующих процессов в асинхронном режиме на $p = 3$ процессорах. Оно совпадает со значением времени выполнения процессов в совмещенной диаграмме Ганта (Рис.3).

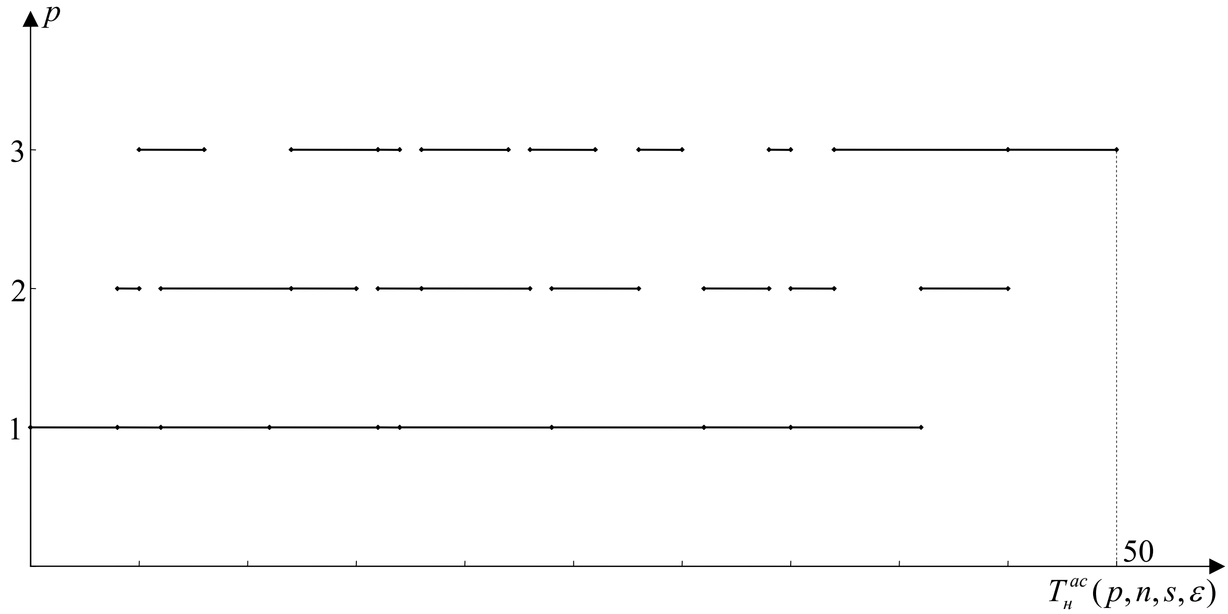

Рисунок 3. – Совмещенная диаграмма Ганта

Учитывая, что $T_H^{ac}(p^s = 9, n, s, \varepsilon) = 45 \le d = 48 < T_H^{ac}(p = 3, n, s, \varepsilon) = 50$, рассмотрим отрезок $[3,9]$.

2. Методом деления отрезка [3,9] пополам находим $p_1 = 6$ и строим по заданным n, s и полученному $p_1 = 6$ результирующую матрицу M_1 вида:

$$M_1 = \begin{bmatrix} \begin{bmatrix} \hat{4} & 1 & 3 & 5 & 2 & 4 \\ \hat{2} & \hat{6} & \hat{4} & \hat{1} & 5 & 3 \\ 5 & 3 & 1 & \hat{7} & \hat{4} & \hat{2} \end{bmatrix} & \begin{bmatrix} 7 & 7 & 1 & 0 & 0 & 0 \\ 4 & 2 & 8 & 0 & 0 & 0 \\ \hat{6} & \hat{4} & \hat{5} & 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 7 & 7 & 1 & 0 & 0 & 0 \\ 4 & 2 & 8 & 0 & 0 & 0 \\ 6 & 4 & 5 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{bmatrix}.$$

С помощью матрицы M_1 вычисляем величину $T_н^{ac}(p_1 = 6, n, s, \varepsilon) = 45$. Так как $T_н^{ac}(p_1 = 6, n, s, \varepsilon) = 45 \le d = 48$, то рассматриваем отрезок $[3,6]$.

3. Методом деления отрезка [3,6] пополам находим $p_2 = 4$, причем в качестве p_2 берем величину, которая является наименьшим целым полусуммы чисел 3 и 6. Далее, по заданным n, s и полученному значению $p_2 = 4$ строим результирующую матрицу M_2 вида:

$$M_2 = \begin{bmatrix} \begin{bmatrix} \hat{4} & 1 & 3 & 5 \\ \hat{2} & \hat{6} & \hat{4} & \hat{1} \\ 5 & 3 & 1 & \hat{7} \end{bmatrix} & \begin{bmatrix} 2 & 4 & 7 & 7 \\ 5 & 3 & 4 & 2 \\ \hat{4} & \hat{2} & \hat{6} & \hat{4} \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 \\ \hat{5} & 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 2 & 4 & 7 & 7 \\ 5 & 3 & 4 & 2 \\ 4 & 2 & 6 & 4 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{bmatrix}.$$

С помощью матрицы M_2 вычисляем величину $T_н^{ac}(p_2 = 4, n, s, \varepsilon) = 45$. Таким образом, директивное время $d = 48$ выполнения $n = 3$ процессов реализуется при $p_2 = 4$, так как $d = 48 > T_н^{ac}(p_2 = 4, n, s, \varepsilon) = 45$ и не реализуется при $p = 3$, так как $d = 48 < T_н^{ac}(p = 3, n, s, \varepsilon) = 50$. Следовательно, $p^* = 4$.

Заключение. Проведенные исследования позволяют давать практические рекомендации по оптимальной организации параллельных процессов, конкурирующих за использование общих программных ресурсов в различных режимах их взаимодействия применительно к многопроцессорным вычислительным системам и вычислительным комплексам, что является отправной точкой для решения ряда практических задач при проектировании сетевых многопроцессорных вычислительных систем и вычислительных комплексов, вычислительных систем с технологией клиент–сервер и кластерного типа, при создании системного и прикладного программного обеспечения. Предложенные методы и математические модели позволяют решать проблемы эффективного отображения параллельных алгоритмов и соответствующих программных реализаций с учетом архитектурных особенностей МС и ВК, проблемы разработки и математического обоснования приемов ускорения вычислений на базе принципов распараллеливания, конвейеризации.

Список использованных источников

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – СПб.: БХВ–Петербург, 2002. – 608 с.

2. Капитонова Ю.В., Летичевский А.А. Математическая теория проектирования вычислительных систем. – М.: Наука, 1988. – 296 с.

3. Коваленко Н.С., Павлов П.А. Математическое моделирование параллельных процессов. LAP Lambert Academic Publishing GmbH, Saarbrucken, Germany, 2011. – 246 с.

4. Коваленко Н.С., Павлов П.А. Алгоритм построения оптимальной компоновки одинаково распределенных систем / Н.С. Коваленко, П.А. Павлов // Программирование. – 2012. – №3.– С. 3–10.

5. Kovalenko N.S., Pavlov P.A. Optimal Grouping Algorithm of Identically Distributed Systems / N.S. Kovalenko, P.A. Pavlov // Programming and Computer Software. – 2012. – Vol.38, №3. – PP. 143–150.

СОДЕРЖАНИЕ

ЭКОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ СРЕДЫ: ДОСТИЖЕНИЯ, ИННОВАЦИИ И ПЕРСПЕКТИВЫ РАЗВИТИЯ

УСТОЙЧИВОЕ РАЗВИТИЕ И КЛИМАТИЧЕСКИЙ МЕНЕДЖМЕНТ

Научное издание

МАТЕРИАЛЫ

II международной заочной научно–практической конференции
“Инжиниринг: теория и практика”

Полесский государственный университет,
г. Пинск, Республика Беларусь,
6 мая 2022 г.

За содержание и достоверность информации
в материалах сборника отвечают авторы

Формат 60×84/8 Гарнитура Times
Усл. печ. л. 13,95. Уч.–изд.л. 8,65.