

УДК 681.142.38:681.323

О ПОТЕРЯХ ВРЕМЕНИ ПРИ РАБОТЕ ОДНОРОДНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ В СИНХРОННО-ПРОГРАММНОМ РЕЖИМЕ

Ю.К. Дмитриев

Изучается качество функционирования однородной вычислительной системы (ОВС) [1] в синхронно-программном режиме (С-режиме). Определение С-режима и обозначения приведены в [2].

С-режим является единственным в системах матричного типа [3,4]. Нет принципиальных ограничений для организации этого режима в ОВС. Наиболее полно С-режим реализуется в управляющей линейной ОВС [5].

Основным достоинством С-режима является возможность более эффективного использования оперативной памяти (ОП) машин системы, чем в А-режиме. (В А-режиме каждая элементарная машина (ЭМ) наряду со своими операндами хранит экземпляр выполняемой программы.) Освобождающаяся при переходе к С-режиму ОП может быть использована для размещения дополнительной порции данных или программ, что ведет к уменьшению числа пересылок между различными уровнями памяти и, следовательно, к сокращению времени решения задачи^{*}). При использовании С-режима уменьшается нагрузка на ОП в ведомых ЭМ, так как источником кодов команд для них являются межмашинные связи. Освобождающиеся циклы ОП могут ис-

^{*}) С-режим не только не исключает, но предполагает разделение параллельной программы между ЭМ, как это делается, например, в [3,6]. При этом также имеет место экономия ОП. Рассмотрение особенностей работы ОВС в данных условиях составляет предмет отдельной работы.

пользоваться в машинах для совмещения счета с работой внешних устройств, с обменом с внешней памятью или другими ЭМ. Ясно, что это либо ведет к сокращению времени решения данной задачи, если освобождающиеся циклы ОП используются для ввода/вывода информации, связанной с её решением, либо повышает общую эффективность системы, если освобождающиеся циклы ОП используются для подготовки данных для других задач (при мультипрограммной работе).

Одновременное использование А- и С-режимов в одной ОВС повышает возможности операционной системы при принятии решений в сложных мультипрограммных ситуациях.

Наряду с факторами, сокращающими общее время решения задачи в С-режиме, существуют факторы, увеличивающие время её решения в А-режиме. Источниками такого увеличения являются зависимость времени выполнения команд от значений операндов, различия при выполнении ветвящихся программ в машинах и передача кодов команд из ведущей ЭМ в ведомые по межмашинным связям.

В данной работе оценивается влияние первого из указанных факторов. (Второй фактор подробно рассматривается в [2]. Влияние третьего фактора исключается методами реализации обменов [5]). Определяются меры архитектурного, программного и организационного характера, с помощью которых можно уменьшить время решения задачи в С-режиме.

Возможны два варианта организации синхронно-программной работы: С1 или С2, при которых ЭМ соответственно выполняет или не выполняет программу, выдаваемую в ведомые машины. Очевидно, не нарушая общности, можно ограничиться рассмотрением С1-режима.

Покомандная синхронизация. Покомандная синхронизация при простоте аппаратной реализации обеспечивает в вычислительной системе надежное взаимодействие машин, даже существенно отличающихся по быстродействию. При покомандной синхронизации ведущие и ведомые ЭМ работают поочередно и покомандно, останавливаясь после завершения своей части текущего акта взаимодействия. Сигналом пуска ведущей ЭМ является окончание выполнения текущей команды всеми ведомыми машинами. Сигналом пуска ведомых ЭМ является выдача кода очередной команды на межмашинные связи из ведущей ЭМ. Покомандная синхронизация использована в управляющей линейной ОВС [5].

Пусть имеется ОВС из ℓ одинаковых машин, выполняющих одну и ту же последовательность операций над различными наборами операндов. Обозначим через t_{ij} время выполнения i -й операции в j -й машине.

Современные электронные вычислительные машины строятся по асинхронному принципу, в соответствии с которым время выполнения команды зависит от вида операции и значений операндов. Например, для операций умножения с фиксированной запятой, сложения и вычитания с плавающей запятой и т.п. разница между минимальным и максимальным временем выполнения может быть значительной. Операнды являются случайными ограниченными дискретными величинами, следовательно, значения t_{ij} будут также случайными [7]. Время выполнения в А-режиме операции k -го вида ($k = 1, 2, \dots, s$) характеризуется распределением:

$$\frac{t_{kg}}{p_{kg}} \parallel \frac{t_{k1}}{p_{k1}} \parallel \frac{t_{k2}}{p_{k2}} \parallel \dots \parallel \frac{t_{kg}}{p_{kg}} \parallel \dots \parallel \frac{t_{km_k}}{p_{km_k}}, \quad (I)$$

здесь m_k — число различных значений длительности операции k -го вида, а значения t_{kg} упорядочены по возрастанию, так что для любого g имеет место $t_{k(g+1)} > t_{kg}$.

Совокупность случайных значений t_{ij} ($i = 1, 2, \dots, z$; $j = 1, 2, \dots, \ell$) назовем реализацией программы из z команд множеством из ℓ ЭМ. Две реализации программы различны, если они различаются хотя бы одним элементом.

Время n -й реализации (R_n) программы для А-режима определяется

$$T_{An} = \max_{1 \leq j \leq \ell} T_{nj} = \max_{1 \leq j \leq \ell} \sum_{i=1}^z t_{ij}, \quad t_{ij} \in R_n$$

а для С-режима

$$T_{Cn} = \sum_{i=1}^z \left(\max_{1 \leq j \leq \ell} t_{ij} \right), \quad t_{ij} \in R_n \quad (2)$$

Интересующие нас потери времени на покомандную синхронизацию составляют для данной реализации

$$T_{CAH} = T_{CH} - T_{AH} = \sum_{i=1}^z (\max_{1 \leq j \leq l} t_{ij}) - \max_{1 \leq j \leq l} \sum_{i=1}^z t_{ij}.$$

Переходя к вероятностным оценкам, находим математическое ожидание потерь на синхронизацию

$$M[T_{CAH}] = M[T_C - T_A] = M[T_C] - M[T_A],$$

где T_C и T_A — времена выполнения заданной программы системой в С-режиме и в А-режиме соответственно. Ясно, что $M[T_{CAH}]$ может быть вычислено, если известны все реализации программы и соответствующие им вероятности.

Отсутствие зависимости между значениями времени выполнения команд программы в С-режиме, как видно из (2), позволяет найти

$$M[T_C] = M\left[\sum_{i=1}^z t_{ci}\right] = \sum_{i=1}^z M[t_{ci}] = \sum_{k=1}^s z_k \cdot M_C[t_k],$$

где t_{ci} — время выполнения i -й по счету операции в С-режиме, а z_k — число команд k -го типа, входящих в данную программу. Для С-режима построим распределение для операции k -го вида ($k = \overline{1, s}$), используемой в программе:

$$\frac{t_{kg}}{p_{kg}} \parallel \frac{t_{k1}}{p_{k1}} \mid \frac{t_{k2}}{p_{k2}} \mid \dots \mid \frac{t_{km_k}}{p_{km_k}}, \quad (3)$$

где

$$p_{kg} = \sum_{j=1}^l \frac{l!}{(l-j)!j!} p_{kg}^j (p_{k1} + p_{k2} + \dots + p_{k(g-1)})^{l-j}, \quad (4)$$

значения t_{kg} и p_{kg} ($g = \overline{1, 2, \dots, m_k}$) взяты из распределения (1), дополненного $p_{k0} = 0$, $t_{k0} = 0$. Таким образом, вероятность того, что данная операция в С-режиме при покомандной синхронизации будет выполнена за время t_{kg} , определяется как вероятность того, что хотя бы в одной ЭМ из l операция будет выполнена за время t_{kg} и при этом не будет ни одной ЭМ с большим значением времени выполнения.

Положив $p_{k1} + p_{k2} + \dots + p_{k(g-1)} = p_{kg\Sigma}$ и приняв во внимание биномиальное разложение, перепишем (4):

$$p_{kg} = (p_{kg} + p_{kg\Sigma})^l - p_{kg\Sigma}^l = a_g^l - b_g^l. \quad (5)$$

Отсюда

$$M_C[t_k] = \sum_{g=1}^{m_k} t_{kg} \cdot p_{kg}. \quad (6)$$

Определим поведение математических ожиданий $M[T_C]$ и $M[T_A]$ как функций целочисленного аргумента l .

С-режим. Рассмотрим функцию непрерывного аргумента

$$f(x) = a^x - b^x, \quad (7)$$

аппроксимируем функцию (5) целочисленного аргумента при условиях $1 \geq a > 0$, $1 \geq b > 0$, $a + b \leq 1$, $a > b$. Можно показать, что в области положительных x для $a < 1$ (что справедливо для любых $g < m_k$) функция (7) имеет единственный экстремум (максимум):

$$x_0 = \frac{\ln\left(\frac{|\ln b|}{|\ln a|}\right)}{\ln\left(\frac{a}{b}\right)},$$

а для $a = 1$ (что справедливо для $g = m_k$) функция $f(x)$ является строго возрастающей.

Так как $\lim_{x \rightarrow \infty} (a^x - b^x) = 0$ при $a < 1$ и $\lim_{x \rightarrow \infty} (1 - b^x) = 1$ при $a = 1$, то, начиная с некоторого x (экстр), функция $y = \sum_{g=1}^{m_k} t_{kg} f_g(x)$, аппроксимирующая функцию целочисленного аргумента (6), является монотонно возрастающей, ограниченной значением t_{m_k} .

А-режим. Пусть задана программа, содержащая z_k операций k -го типа ($k = \overline{1, 2, \dots, s}$), с общим числом команд $z = \sum_{k=1}^s z_k$. Пусть известно распределение времени выполнения заданной программы в А-режиме:

$$\frac{T_{A1}}{P_{A1}} \parallel \frac{T_{A2}}{P_{A2}} \mid \dots \mid \frac{T_{A1}}{P_{A1}} \mid \dots \mid \frac{T_{A2}}{P_{A2}},$$

в котором T_{A1} упорядочены по возрастанию. Легко видеть, что вероятность равенства времени реализации программы в А-режиме

некоторому значению T_{Au} может быть рассчитана по формуле (5). Это означает, что анализ, проведенный для функции $M_C[t_k] = f(\ell)$, справедлив и для $M[T_A] = f(\ell)$. В частности, время выполнения заданной программы в А-режиме является, начиная с некоторого ℓ_A (экстр), монотонно возрастающей функцией, ограниченной значением

$$T_{Az} = \sum_{k=1}^z \tau_k t_{mk} = \sum_{k=1}^z \tau_k \max_{1 \leq q \leq m_k} t_{kq},$$

(вероятность которого $P_{Az} = \prod_{k=1}^z \tau_k \cdot \rho_{mk}$), определенным для операций программы, имеющих $\tau_k \neq 0$.

Поскольку $\lim_{\ell \rightarrow \infty} M[T_C] = \lim_{\ell \rightarrow \infty} M[T_A] = T$, то существует такое $\ell_3(\Delta)$, минимальное среди q , что $M_3[T_{CA}] \neq \Delta$ для всех $\ell_q > \ell_3$ при любом наперед заданном сколь угодно малом Δ . Значение ℓ_3 представляет собой число ЭМ, при котором С-режим эффективен для данной программы.

При достаточно больших m_k и z случайную величину T_A можно считать непрерывной в замкнутом интервале $[T_{Amin} = \sum_{k=1}^z \tau_k t_{k1}, T_{Amax} = \sum_{k=1}^z \tau_k t_{mk}]$ и равной нулю вне этого интервала. Такая случайная величина удовлетворяет условию Линдберга [8], что позволяет применить к ней центральную предельную теорему. Для расчета $M[T_A]$ используем распределение $\{P_{Au}\}$, где

$$P_{Au} = P(T_{A(u-1)} < T_A < T_{Au}) = \Phi^*(T_{Au}) - \Phi^*(T_{A(u-1)}),$$

$$T_{Au} = \frac{(T_{Amax} - T_{Amin})u}{z},$$

$$u = 1, 2, \dots, z, \quad T_{A0} = T_{Amin}.$$

$\Phi^*(x)$ - нормальная функция распределения.

В качестве иллюстрации на рис. 1 и 2 приведены оценки эффективности С-режима для ОВС "Минск-222" [9]. Выбор типа ОВС определился единственно наличием в литературе [7] распределений времени выполнения операций с переменной длительностью. На рис. 1 приведены зависимости $M_C[t_k]/t_{mk}$ от числа ЭМ, занятых в С-режиме, для операций умножения с фиксированной и плавающей за-

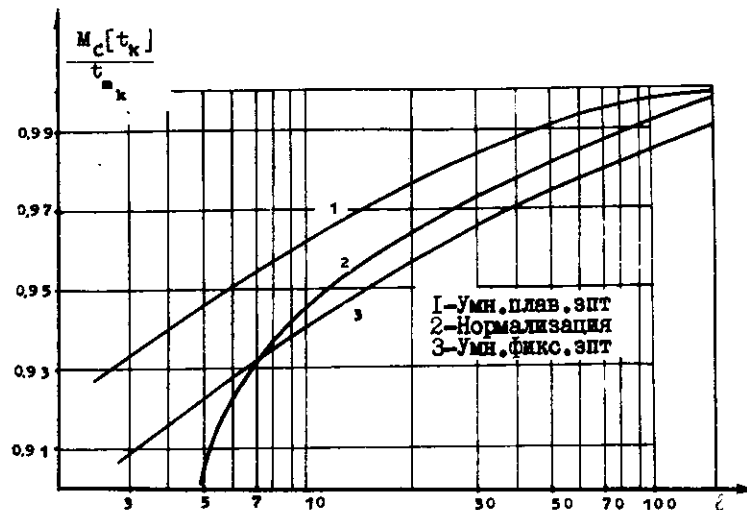


Рис. 1

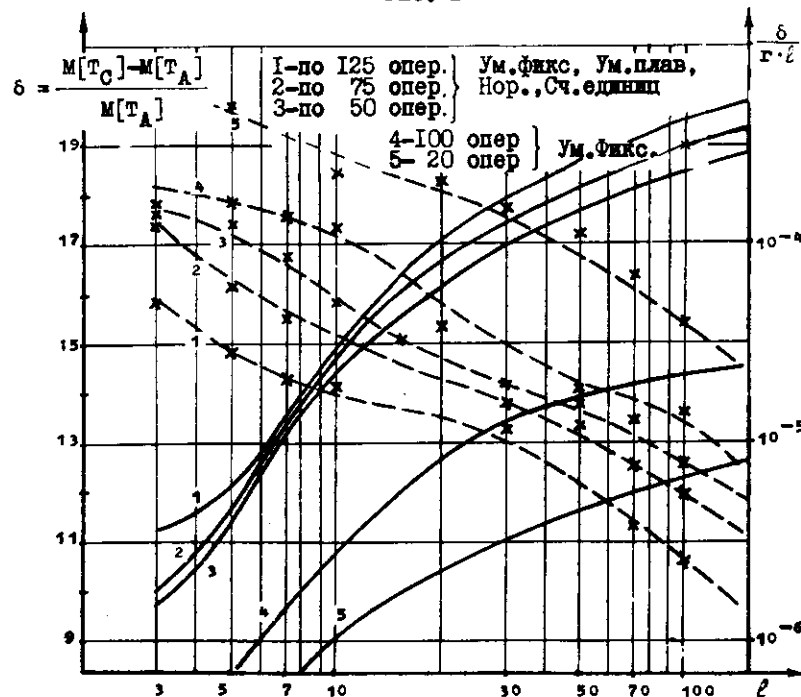


Рис. 2

пятой и для операции нормализации. На рис.2 сплошной линией показана зависимость относительных потерь $\delta = M[T_{CA}] / M[T_A]$ от числа ЭМ, выполняющих каждую из указанных на рисунке программ. Штриховой линией показана для тех же программ зависимость от ℓ потерь С-режима, приходящихся на одну операцию одной ЭМ.

Синхронизация с помощью буферной памяти. При работе в А-режиме темп выдачи команд из ОП определяется внутренней структурой вычислительного модуля элементарной машины, в частности наличием в нем сверхоперативной памяти команд (буфера команд). Блок-схема вычислительного модуля, имеющего буфер команд, при работе его в А-режиме показана на рис.3 (П-процессор, БК-буфер команд, СС-схема старшинства запросов в ОП). Здесь выделены пути обращения к ОП за командой. Блок-схема соответствует мультипроцессорной

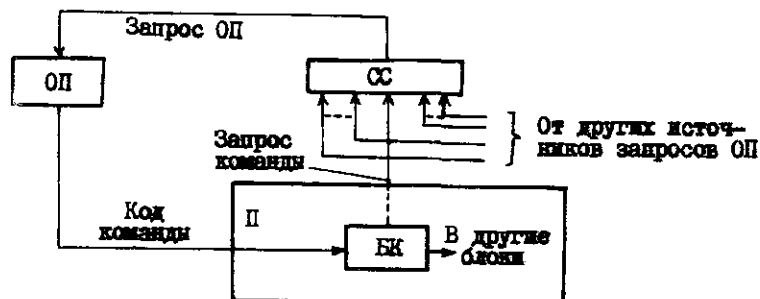


Рис.3

и/или мультипрограммной машине, отдельные устройства которой (ОП, процессоры и др.) работают асинхронно.

С точки зрения загрузки устройств вычислительного модуля, для которых буфер команд является источником информации, емкость буфера команд должна быть такой, чтобы в нем была хотя бы одна команда при значениях скорости загрузки f_I и скорости разгрузки f_O , удовлетворяющих условиям:

$$f_{I \max} \geq f_I \geq f_{I \min},$$

$$f_{O \max} \geq f_O \geq f_{O \min}.$$

В условиях указанных колебаний f_I и f_O буфер команд должен проектироваться как асинхронный блок, загрузки в который возможна лишь, если буфер команд не пуст, а разгрузки – если буфер команд не пуст. В соответствии со сказанным, сигнал "запрос команд" на рис.3, который может интерпретироваться как "готовность буфера к приему", является функцией от загруженности последнего. Этот сигнал существует до тех пор, пока хотя бы одна из ячеек буфера не будет загружена. Предполагается, что запрос ОП на выборку команд отличим от запроса ОП на выборку операндов и т.п. Это связано с особенностями обработки кодов команд и зависящей от нее загрузки процессоров.

Буфер команд тем лучше справляется с задачей сглаживания колебаний f_I и f_O и задачей сокращения числа обращений к ОП (за счет многократного использования содержащихся в буфере команд циклических участков программы), чем больше его емкость E . Вместе с тем увеличение емкости буфера увеличивает стоимость вычислительного модуля, а полному использованию буфера препятствует команды ветвления программы.

На рис.4 (СВЗК-схема выработки запроса команды, ММС-механизмы связи, В-ЭМ-ведущая ЭМ, В-ЭМ-ведомая ЭМ) приведена блок-схема ОВС, в ЭМ которой используется буфер команд при работе в С-режиме. Как и на рис.3, здесь выделены только пути обращения к ОП за кодом команды. Развернутая блок-схема ЭМ такой ОВС, определяющая направление выдачи кодов команд для различных режимов работы, показана на рис.5 (В-признак ведущей ЭМ, В-признак ведомой ЭМ, С-признак работы ЭМ в С-режиме, $\bar{C}$ -признак работы ЭМ в А-режиме, $\cdot$ – логическое умножение, $+$ – логическое сложение). Линии задержки, показанные на рис.4 пунктиром, определяют задержки распространения сигналов по межмашинным связям. В данном рассмотрении эти задержки считаются пренебрежимо малыми. Схема выработки запросов команды определит зависимость выдачи заявки в ОП В-ЭМ на очередную команду от выработки соответствующего сигнала в каждой С-машине. Таким образом, схема выработки запросов команды в логическом отношении представляет собой многоходовую схему И.

Рассмотрим на примере использование буфера команд заданной емкости E , найденной из условий заданной загрузки ЭМ в А-режиме, при выполнении программы в С-режиме. Положим для просто-

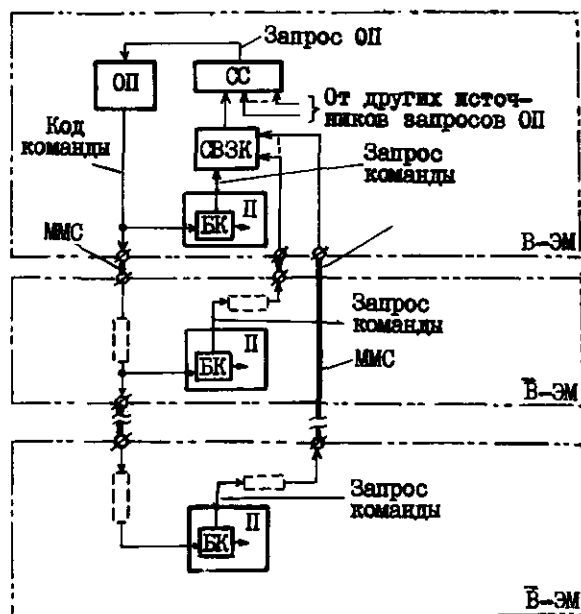


Рис. 4

ти, что программа из $z = II$ команд выполняется на двух ЭМ, имеющих $E = 2$. ЭМ, имеющую наибольшее значение времени выполнения данной программы в А-режиме, назовем определяющей машиной. Выполнение программы начинается после полного заполнения буфера команд. Для упрощения считаем, что обращение к ОП за очередной командой совмещено в В-ЭМ с выполнением текущей и $\frac{1}{f_I} = 0$. Последнее условие не нарушает общности. В самом деле, все простоем процессора определяющей машины из-за конечного значения f_I (из-за отсутствия команд в буфере) можно учесть в общем времени выполнения рассматриваемой программы. Это достигается путем увеличения времени выполнения команд на величины простоев, которые имеют место после каждой из них. Случайная величина потеря из-за конечности f_I не зависит от того, какая ЭМ является определяющей, какова величина времени выполнения программы

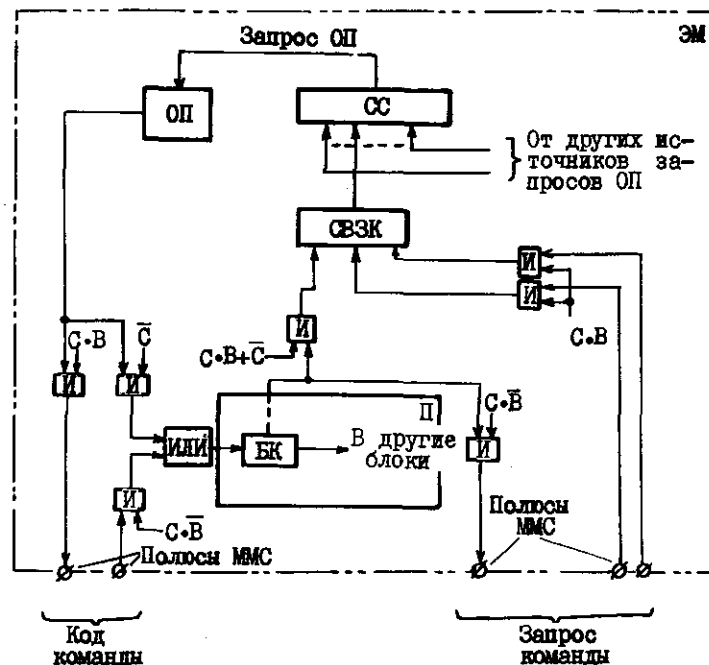


Рис. 5

и каков режим работы машин. Вместе с тем, условие $\frac{1}{f_I} = 0$ позволяет упростить рассмотрение.

Распределение времени выполнения команд рассматриваемой программы задано для А-режима на рис. 6, а^{*}.

На рис. 6, б построена зависимость времени T_x работы ЭМ от числа x выполненных команд: $T_x = \sum_{i=1}^x t_i$, где $x = 1, 2, \dots, z$, а t_i — время выполнения i -й команды. Определяющей машине на этом графике соответствует наибольшее значение ординаты при $x = z$. Горизонталь с номером x на рис. 6, б отмечает момент по-

* Представленные на рис. 6, а величины являются временами выполнения команд для однопроцессорных ЭМ без совмещения операций. В общем случае эти величины представляют собой промежутки времени между последовательными обращениями к буферу и находятся в интервале $\left[\frac{1}{f_{\min}} - \frac{1}{f_{\max}} \right]$.

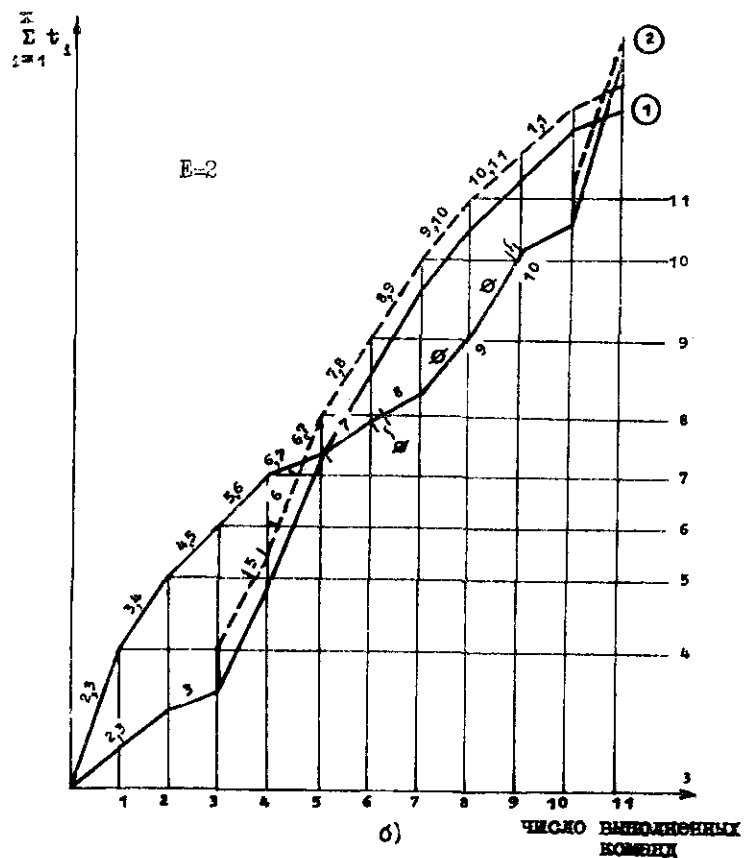
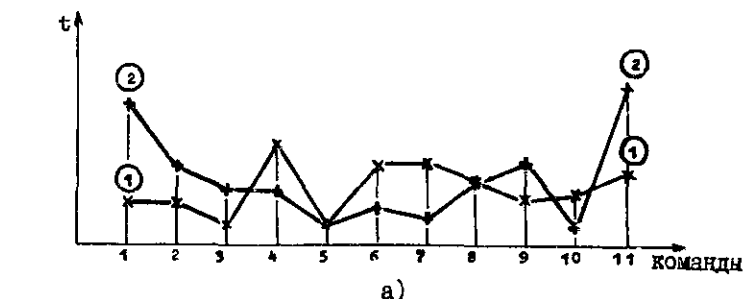


Рис. 6
80

ступления кода x -й команды из В-ЭМ на входы буферов команд ЭМ ОВС в С-режиме. Номера команд, находящихся в буфере команд, для каждой ЭМ выписаны над соответствующими участками графика на рис. 6, б.

Как можно видеть, различие скоростей разгрузки в С-машинах может привести к увеличению времени решения задач в С-режиме. После выполнения третьей команды ЭМ-1 простаивает в ожидании поступления кода четвертой команды, вследствие чего график T_x для неё смещается в положение, указанное пунктиром. Аналогичным образом ЭМ-2 простаивает в ожидании кода одиннадцатой команды.

Рассмотрим две ЭМ, имеющие взаимопересекающиеся графики T_x . Назовем петлей фрагмент графиков T_x , заключенный между соседними точками их пересечения, а сами точки пересечения назовем узлами петли. На рис. 6 выделены две такие петли.

Из рассмотренного примера можно видеть, что время выполнения программы в С-режиме равно времени T_0 её выполнения в А-режиме, если ни в одной из С-ЭМ буфер команды не становится пустым.

Динамика загрузки и разгрузки буфера команд в С-ЭМ такова, что в узлах любой (вновь образуемой) петли буфер команд машины полон. В самом деле, буфер С-ЭМ полон в левом узле первой петли (начало координат на графиках T_x), и содержимое его во всех машинах одинаково по условию. Буфер команд машины, образующей верхнюю часть петли, полон в любой момент времени в силу условий выработки запроса на очередную пересылку из ОП в буфер команд (в том числе и при достижении правого узла петли). Содержимое буфера команд машины, образующей нижнюю часть петли, в момент достижения правого узла, совпадает с содержимым буфера команд машины, образующей верхнюю часть петли, так как номера команд, выполняемых обеими машинами, одинаковы. Поэтому буфер команд машин, образующей нижнюю часть петли, полон в правом её узле. В силу совпадения правого узла i -й петли ($i = 1, 2, 3, \dots$) с левым узлом $(i+1)$ -й петли утверждение справедливо для произвольной петли.

Рассмотрим теперь множество из $\ell > 2$ С-ЭМ, которому соответствует множество графиков $\tau = \{T_{x1}, T_{x2}, \dots, T_{x\ell}\}$. Построим для

них огибающую (жирная линия на рис.7 для $\tau = \{T_{x1}, T_{x2}, \dots, T_{x7}\}$), образуемую из участков T_{xj} ($j = 1, 2, \dots, \ell$), для которых значения времени выполнения x команд (значения ординат на рис.7) максимальны. Машины, участки T_{xj} которых входят в состав огибающей, назовем существенными. Выделим из

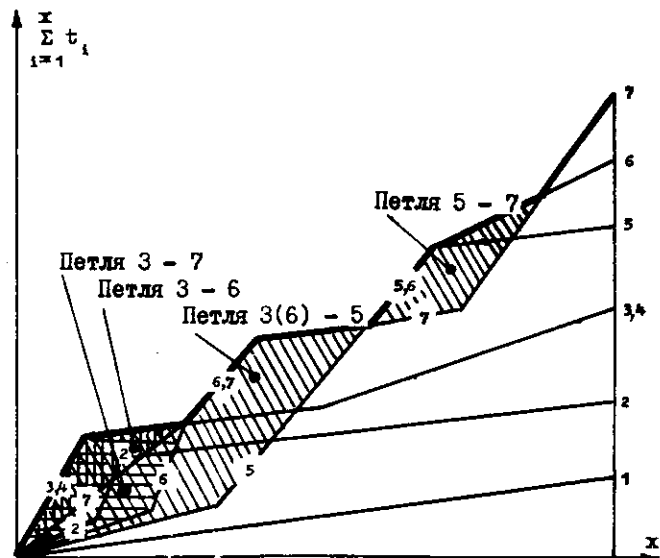


Рис. 7

множества петель, образуемых попарным пересечением T_{xi} и T_{xj} ($i \neq j$), петли, содержащие участки огибающей. На рис.7 показано несколько подобных петель, называемых существенными.

Фактически для реализации за время T_0 программы в С-ре- жме можно вместо выполнения условия "буфер команд не пуст" для всех ЭМ (или для всех возможных петель) потребовать выполнения этого условия только для существенных петель.

Рассмотрим произвольную петлю, показанную на рис.8. Пусть для неё известны два параметра: угол развешивания β , образован-

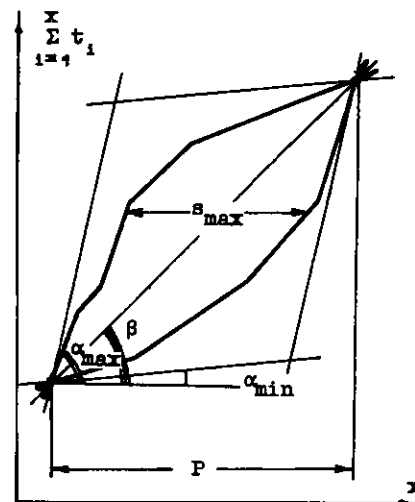


Рис. 8

с T_{xj} , образующей петлю, сечением петли. Выделим среди множества $\{s_i\}$ возможных сечений s_i данной петли сечение s_{max} , имеющее наибольшее значение. Значение s_{max} берем округленным до большего целого.

Можно показать, что для петли, заданной значениями β и P , условие "буфер команд не пуст" обеспечивается при $E = s_{max} + 1$, округленном до большего целого (необходимое условие эффективности буфера). В самом деле, произвольное сечение s_i , проведенное из точки i нижней части петли, определяет для этой точки отставание определяющей машины от существенной ЭМ по числу выполненных команд. Очевидно, для точки i буфер команд не пуст, если его емкость больше s_i , т.е. если $E > s_i + 1$. Отсюда и следует, что условие "буфер команд не пуст" достигается при $E \geq s_{max} + 1$.

Найденное для заданных β и P значение s_{max} не превосходит сечения максимального параллелограмма, образованного прямыми, проведенными через узлы петли под углами α_{min} и α_{max} . Эти углы соответствуют минимальному и максимальному временам выполнения операций в ЭМ. Сечение максимального параллелограм-

ный прямой, проведенной через узлы петли, с осью абсцисс, и длина P проекции отрезка этой прямой на ось абсцисс, выраженная в единицах измерения для оси абсцисс (число команд, составляющих петлю).

Для того, чтобы буфер команд был не пуст, достаточно, чтобы он имел емкость $E = P$. Улучшим эту оценку. Проведем из произвольной точки участка, входящего в данную петлю, прямую, параллельную оси абсцисс, и назовем длину отрезка, заключенного между этой точкой и точкой пересечения прямой

ма, найденное как функция значений P, β, α_{min} и α_{max} , является достаточным условием для выполнения требования "буфер команд не пуст":

$$E = P \left[1 - \frac{tg \beta}{tg \alpha_{max}} \right] + 1. \quad (8)$$

Выделяя для заданного множества τ все возможные существенные петли, найдем по формуле (3) набор $(E_1, E_2, \dots, E_x)$ значений емкости буфера команд. Значение $E_m = \max(E_1, E_2, \dots, E_x)$ обеспечивает выполнение данной программы в С-режиме за время T_0 .

Выше описан способ повышения эффективности решения задачи в С-режиме, заключающийся в увеличении емкости буфера команд. Другим фактором, приводящим к той же цели, является уменьшение времени обращения к ОП за командой в ведущей ЭМ.

В приведенных на рис. 3-5 схемах ОП представляет собой систему массового обслуживания без потерь. В системе запросов на ОП запрос на команду обладает определенным приоритетом, не зависящим от режима работы ОВС. Эффективность работы ОВС в С-режиме может быть повышена, если запросам на ОП (за командами и операндами) в С-режиме будет выделен собственный - более высокий, чем для А-режима, - приоритет. Данное решение может быть использовано для (частичной) компенсации потерь времени на решение задачи в С-режиме, поскольку при прочих равных условиях время решения задачи тем меньше, чем выше скорость загрузки буфера команд.

В работе рассмотрены особенности функционирования ОВС в режиме "Один поток Команд-Много потоков Данных" [10] (С-режим), определены качественные преимущества этого режима сравнительно с работой в режиме "Много потоков Команд-Много потоков Данных" [10] (А-режим). Выявлены источники, ухудшающие время решения задачи в С-режиме по сравнению с временем А-режима.

Найдены выражения для расчета времени выполнения заданной линейной программы в автономном и синхронно-программном режимах как функции от числа заданных в работе машин ОВС. Установлено, что при достаточно большом числе машин время выполнения программы в С-режиме близко к времени её выполнения в А-режиме.

При выполнении линейных участков программы, содержащих до 200 команд с переменной длительностью (команды "Счет Единиц", "Нормализация", "Умножение фиксированное и Плавачее" в различных сочетаниях) на ОВС "Минск-22" (18 машин) величина потерь времени С-режима сравнительно с А-режимом представляет собой возрастающую функцию и не превосходит 17%. Эта величина уменьшается с уменьшением числа С-машин и длины программы.

Потери времени, вызываемые различием операндов, для С-режима могут быть уменьшены с помощью сверхоперативной памяти команд. Найдено аналитическое выражение для оценки сверху емкости такой памяти, при которой время выполнения линейной программы в С-режиме не хуже времени её выполнения в А-режиме. Оно может быть использовано при построении модели расчета емкости сверхоперативной памяти команд для С-режима.

Л и т е р а т у р а

1. ЕВРЕЙНОВ Э.В., КОСАРЕВ Ю.Г. Однородные универсальные вычислительные системы высокой производительности. Новосибирск, "Наука", 1966.
2. ДИМИТРИЕВ Ю.К. О способе реализации ветвящихся программ при работе однородной вычислительной системы в синхронно-программном режиме. - В кн.: Вычислительные системы. Вып. 60. Новосибирск, 1974, с. 103-114.
3. SLOTHICK D.L. and oth. The SOLOMON computer. - "Proc. EJCC", 1962, p. 97-107.
4. BOUKHOUGH W.L. and oth. The ILLIAC system. - "Proc. ICCC", 1972, vol. 60, April, p. 369-388.
5. ДИМИТРИЕВ Ю.К., ШУМ Л.С., ТОМИНОВ Ю.Ф., ПОТАПОВА Ю.Н. Управляющая линейная однородная вычислительная система. - В кн.: Вычислительные системы. Вып. 39. Новосибирск, 1970, с. 107-118.
6. ГАМИДОВ В.М., МЯГЕНКОВ Н.Н. Сегментирование параллельных программ. - Настоящий сборник, с. 54-68.
7. КОСАРЕВ Ю.Г., НАГАЕВ С.В. О потерях времени на синхронизацию в однородных вычислительных системах. - В кн.: Вычислительные системы. Вып. 24. Новосибирск, 1967, с. 21-40.
8. ВЕНТИЛЬ Е.С. Теория вероятностей. М., "Наука", 1964.
9. ЕВРЕЙНОВ Э.В., ЛОПАТО Г.П. Универсальная вычислительная система "Минск-222". - В кн.: Вычислительные системы. Вып. 23. Новосибирск, "Наука", 1966, с. 13-20.
10. FLYNN M.J. Very High-Speed Computing Systems. - "Proc. ICCC", 1966, vol. 54, N 12, Dec., p. 1901-1909.

Поступила в ред.-изд. отд.
5 ноября 1973 года