

УДК 681.142.2;681.31

КОММУТИРУЕМОСТЬ И ВНУТРЕННЯЯ ЭФФЕКТИВНОСТЬ ОДНОРОДНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ В ОДНОМ ЧАСТНОМ СЛУЧАЕ

В.А.Воробьев

Рассматривается модель m -коммутируемой однородной вычислительной системы. Приводятся результаты моделирования ОВС и методика выбора параметров системы программ при заданных ограничениях на эффективность счета.

1. Постановка задачи

1.1. Как указывалось в [1], коммутационные возможности исправной структуры однородной вычислительной системы (ОВС) определяются её потенциальной и реальной m -коммутируемостью: $K(\mathcal{A}, m)$ и $L(\mathcal{A}, m, \mathcal{D}, \rho)$, соответственно, где $\mathcal{A}$ - группа автоморфизмов структуры (или любое другое её описание), m - число одновременно осуществляемых коммутаций, $\mathcal{D}$ - внутренняя дисциплина коммутаций, $\rho(\mathcal{A})$ - плотность взаимодействия, то есть вероятность связи элементарных машин (ЭМ) на расстоянии $\mathcal{A}$. Цель работы - изучить влияние m -коммутируемости (а точнее, величины m) на эффективность процесса вычислений.

Эффективность вычислений будем измерять коэффициентом счета $K_c = t_c / T$, где T - общее время работы ОВС, а t_c - время чистого счета. Аналогично вводятся коэффициенты взаимодействия k_g и простоя k_n так, что $K_c + k_g + k_n = 1$.

В дальнейшем рассматривается исправная и полностью загруженная ОВС. В этом случае простой ЭМ возможен по двум независимым причинам:

1. Нет коммутаций. Причина аналогична занятости всех обслуживающих устройств в системе массового обслуживания.

2. Есть взаимодействие между ЭМ. Причина специфична для ОВС, где заявка одной ЭМ вовлекает в обслуживание несколько других машин, и если некоторые из них уже заняты (взаимодействуют), то возникают потери.

Если имеется первая причина и только она, мы будем говорить о системном простое и измерять соответствующие потери коэффициентом системного простоя k_{nc} . Если имеется вторая причина, то независимо от наличия первой возникает алгоритмический простой, а потери характеризуются коэффициентом k_{na} . Итак, $k_n = k_{nc} + k_{na}$.

1.2. Задавая различные дисциплины $\mathcal{D}$, плотности взаимодействия $\rho(\mathcal{A})$ и структуры $\mathcal{A}$, можно получить множество постановок задачи о функционировании ОВС как замкнутой системы массового обслуживания со взаимодействием между абонентами (ЭМ) и со случайным числом обслуживающих устройств (коммутаций). Мы ограничимся следующим простейшим случаем.

1.2.1. ОВС строго m -коммутируема [1].

1.2.2. Каждая ЭМ системы последовательно решает некоторый набор простых статистически однородных задач с интенсивностью λ выдает результаты другим ЭМ.

1.2.3. Если в системе возможна запрошенная коммутация, то она реализуется и существует некоторое случайное время, необходимое для передачи и приема информации (результатов). Длительность взаимодействия распределена по экспоненциальному закону с параметром μ .

1.2.4. Очередная коммутация невозможна, если в системе уже существует m коммутаций или запрошенные ЭМ уже взаимодействуют. В этих случаях ведущая ЭМ простаивает вплоть до освобождения линий связи или ведомых ЭМ.

1.2.5. Множество ЭМ, взаимодействующих в данный момент, известно всем ЭМ ОВС. Время, необходимое для получения этой информации, пренебрежимо мало.

1.2.6. Плотность взаимодействия $\rho(\mathcal{A}) = \rho = const$, то есть каждая ЭМ может быть вовлечена во взаимодействие с вероятностью ρ , независимо от своего расположения в системе.

1.2.7. Все N машин ОВС упорядочены по приоритету.

1.2.8. Элементарная внутренняя дисциплина состоит в следующем:

1. Коммутации предоставляются, в первую очередь, более приоритетным ЭМ.

2. Процесс счета в ЭМ, вызванной на взаимодействие, прерывается до окончания обмена информацией.

3. Взаимодействие не может быть прервано.

Итак, в предлагаемую модель входят два вида характеристик: параметры системы $\{N, m, D\}$ и параметры задачи $\{\lambda, \mu, \rho\}$, решаемой на ОВС.

Описанная таким образом ОВС хорошо соответствует, например, УК — однородным системам, предложенным в [2]. Для КАИСтруктур [1] эта модель может дать полезные ориентировочные результаты.

1.3. Методом решения поставленной задачи о функционировании ОВС является статистический эксперимент. Основная особенность применения метода моделирования в этом случае состоит в том, что модель не следует за проектированием, а предшествует ему, давая проектировщику общие сведения о том классе систем, в пределах которого находится проект.

В настоящей работе применяется метод организации статистического эксперимента, предложенный в [3]. Модель системы написана на языке Р-ЛЯПАС [4] и реализована на электронной вычислительной машине (ЭВМ) "Минск-22". Применяются средства моделирования системы параллельных процессов [5], имеющиеся в архиве ПС-Р-ЛЯПАС [6]. Таким образом, настоящая статья может рассматриваться и как пример применения результатов, фиксированных в [3 - 6].

2. Предварительный анализ задачи

2.1. Достоинством предложенной выше постановки задачи моделирования является возможность исследования модели совместно с известными ранее аналитическими методами. В данном случае стохастическая модель ОВС со взаимодействием между машинами возникает в хорошо изученные системы массового обслуживания при плотностях взаимодействия $\rho = 0$ и $\rho = 1$. Это обстоятельство позволяет сделать некоторые априорные выводы о поведении системы, а в дальнейшем — проконтролировать эксперимент.

2.2. При $\rho = 0$ характеристики ОВС вычисляются [7] известными методами теории массового обслуживания. Заметим, что для исследования среднего k_c можно не учитывать приоритеты отдельных устройств. Хорошую оценку для рассматриваемого случая дает метод динамики средних [8]. В стационарном режиме:

$$k_c = \begin{cases} \frac{\alpha}{\alpha+1}, & \text{при } \alpha \geq \frac{N}{m}-1, \\ \frac{m}{N} \alpha, & \text{при } \alpha < \frac{N}{m}-1, \end{cases} \quad (1)$$

где $\alpha = \frac{\mu}{\lambda}$ — характеристика системы программы, решаемой на ОВС.

Нетрудно видеть, что $\frac{\alpha}{\alpha+1}$ есть собственный коэффициент счета k_{cc} отдельной ЭМ системы, то есть существуют условия, при которых k_c в ОВС без взаимодействия в точности равен k_{cc} и не зависит от m . (Строго говоря, в этом случае $\lim_{N \rightarrow \infty} k_c = k_{cc}$, но уже при $N > 10$ ошибка практически неощутима.)

2.3. При $\rho = 1$ коэффициент счета в стационарном режиме

$$k_c = \frac{\alpha}{N+\alpha}. \quad (2)$$

Таким образом, с увеличением плотности взаимодействий в ОВС значение фактора коммутруемости должно уменьшаться и постепенно сойти на нет.

2.4. Будем полагать, что системному программисту заранее известен минимальный коэффициент счета k_0 , при котором ОВС справляется со своими функциями достаточно эффективно. В этой ситуации требуется определить параметры системы задач, при которых достигается $k_c \geq k_0$.

Задачей моделирования является выяснение вида поверхности $k_c(N, m, \alpha, \rho) = k_0$, разделяющей область параметров ОВС на две части. Если $\{N, m, \alpha, \rho\}$ таковы, что $k_c \geq k_0$, то будем говорить, что система программ с характеристиками $\{\alpha, \rho\}$ эффе к т и в н о о с у щ е с т в и м а на ОВС с параметрами $\{N, m\}$ и элементарной внутренней дисциплиной.

Из формул (1) и (2) получим граничные точки искомой поверхности в сечении $N = \text{const}, m = \text{const}$.

$$\alpha(\rho=0) = \begin{cases} \frac{k_0}{1-k_0}, & \text{при } k_0 \geq 1 - \frac{m}{N}, \\ \frac{N}{m} k_0, & \text{при } k_0 < 1 - \frac{m}{N}, \end{cases} \quad (3)$$

$$\alpha(\rho=1) = \frac{k_0 N}{1-k_0} \quad (4)$$

Очевидно, что при

$$m \geq N(1-k_0) \quad (5)$$

коммутируемость ОВС не влияет на величину зоны эффективной осуществимости параллельных программ.

Кроме того, из (1) и (2) следует, что

$$\frac{\partial k_c}{\partial \alpha} \geq 0, \quad \frac{\partial k_c}{\partial \rho} < 0.$$

Таким образом, можно применить метод отслеживания сечения $k_c(N, m, \alpha, \rho) = k_0$, предложенный в [3]. Стационарные значения k_c можно получать с заданной точностью при помощи Λ -оператора слежения [3].

3. Модель

3.1. Модель ОВС представляет собой квазипараллельную программу [5]. На рис. I показана её общая структура. Весь программный материал размещен в двух заказах (нулевом и первом) ПС-Р-ЛНПАС для "Минск-22" [6]. В нулевом заказе размещен Λ -оператор модели, обеспечивающий исполнение операторов квазипараллельного программирования. В первом - собственно квазипараллельная моделирующая программа. Стыковку двух заказов в памяти ЭВМ осуществляет Λ -оператор заказа, размещенный в нулевом заказе [6].

Первоначально запускается нулевой заказ, Λ -оператор модели размечает свое рабочее поле (комплекс $\underline{B}$) и порождает следящий (нулевой) процесс. Следящий процесс реализуется, как обычная последовательная программа, и достигает Λ -оператора заказа // . Происходят вызов, трансляция, загрузка в память ЭВМ и запуск квазипараллельной программы заказа # I. Реализация этой программы также начинается следящим процессом, функции которого - организация эксперимента, развертывание и запуск всей системы процессов. На рис. I порождение и запуск процессов изображены двойной стрелкой. Обычными стрел-

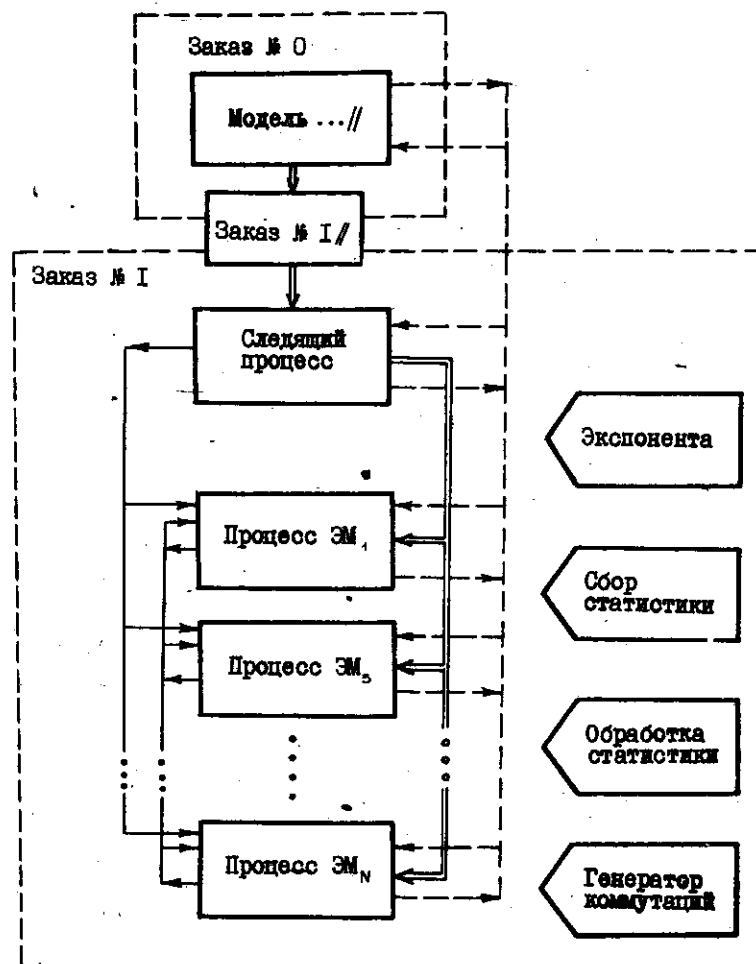


Рис. I

ками изображаются явные (для программиста) взаимодействия между процессами, штрих-стрелками - неявные взаимодействия между процессами и Λ -оператором модели, необходимые для реализации операторов квазипараллельного программирования.

3.2. Представленная ниже программа рассчитана на моделирование ОВС, содержащих не более 32-ЭМ. Для облегчения её чтения дадим описание всех операндов.

3.2.1. Локальные операнды:

ρ - приоритет процесса (минимальный приоритет равен 1, максимальный - $N+1$ - имеет следующий процесс);

α - полное время счета данной ЭМ;

β - полное время взаимодействия данной ЭМ;

ϵ - полное время простоя данной ЭМ;

ζ - коммутация, необходимая данной ЭМ в некоторый момент времени (булевский вектор);

δ - полное время алгоритмических потерь ЭМ;

m - число коммутаций, заказанное всеми ЭМ;

A - рабочее поле ЭМ (в настоящей модели оно пустое:

$A_0 = 0, B_0 = 0$).

Список ρ ($\alpha \beta \epsilon \zeta \delta m$) есть сообщение.

3.2.2. Глобальные операнды:

α' - вектор, отмечаящий своими единичными элементами взаимодействующие в данный момент ЭМ;

ℓ - число коммутаций, которые еще можно организовать в системе;

t - текущее время.

Статистика накапливается в переменных: $h, i, j, k, m, z, s, x, y, u, w$ и комплексе $\mathcal{D}$, смысл которых для понимания программы несуществен.

Слежение за ходом реализации программист может осуществлять, задавая индексы i и j :

i - текущее время, начиная с которого модель выдает отладочную информацию;

j - признак выдачи результатов L -оператора слежения и т.д.

Точность эксперимента задается операндами:

z' - число коммутаций в одном интервале слежения;

e' - точность слежения (число правых разрядов, которыми можно пренебречь);

f - точность задания плотности взаимодействия (см. вектор [3]).

В организации моделирования участвуют также следующие операнды:

B - рабочий комплекс L -оператора модель ($A_1 = 10000, B = 1000$);

N - число шагов по координатам $\{\alpha, \rho, N, m\}$ ($A_7 = 11017, B_7 = 4$);

G - условные координаты очередного варианта ($A_8 = 11012, B_8 = 4$);

K - истинные координаты начальной точки пространства координат ($A_{12} = 11024, B_{12} = 4$);

F - величины шагов по координатам ($A_5 = 11005, B_5 = 4$);

E - истинные координаты $\{\alpha, \rho, N, m\}$ очередного варианта ($A_4 = 11000, B_4 = 4$);

K' - вектор знаков производных;

ϵ' - нижний уровень оценки;

α' - верхний уровень оценки [3].

3.2.2. На вход программы подаются: $H, K, F, z, e, f, i, j, k, c, d, \alpha'$. После окончания эксперимента в очередной точке печатается $G, E, k_c, k_g, k_c + k_g, k_n, k_{na}, k_{nc}$ и комплекс, i - й элемент которого содержит число коммутаций, реализованных по заявке ЭМ с приоритетом $i+1$. Печать восьмеричная, все коэффициенты увеличены в 1000_{10} раз.

3.2.4. Сечение поверхности $k_c = k_0$ строится вручную, при этом точка со значением k_0 получается линейной интерполяцией соседних значений $k_c > k_0$ и $k_c < k_0$.

3.3. Ниже следует полный текст моделирующей программы. Для обеспечения восприятия программа разбита на нумерованные блоки, и далее следуют соответственно пронумерованные комментарии.

3.3.1. Заказ № 0

0100 0000 1000	I
0137 1400 0000	
0000 0000 0000	
$\S 0 \ 6 < 6 \Rightarrow g \ C_{22} \Rightarrow f$	2
м о д е л ь # 101 020 t p (α β ε ζ δ m)	3
A 41 41 41 g B f 0 1 i //	
з а к а з # 101 050 1 //	4
$\S 1$.	5

Комментарий

I. Шапка заказа: режим, список фиксированных комплексов, номер заказа (см. [6]).

разрядов, слежение закончится; иначе все действия повторяются. При $j \neq 0$ на печать выдаются промежуточные значения x, y, z . После стабилизации значений x, y, z следящий процесс продолжит свою работу с §10.

7. Отсюда начинают свою работу все N процессов ЭМ. Обращение к датчику случайных чисел (§6) и накопление времени счета в локальной переменной α .

8. Задержка процесса ЭМ на время счета (пассивная фраза).

9. Процесс ЭМ, кончивший счет, вырабатывает случайную коммутацию $\underline{c}$ плотности $\rho = \underline{c}$, (см. [3] в е к т о р).

10. Засечка времени начала простоя.

11. Алгоритмический простой. Процесс ЭМ ждет освобождения от взаимодействий всех вызванных в $\underline{c}$ ЭМ. Взаимодействующие ЭМ помечены в $\underline{c}$.

12. Засечка времени алгоритмического простоя и накопление его в локальной переменной ℓ .

13. Системный простой, если коммутационные возможности ОВС исчерпаны.

14. Засечка полного времени простоя и накопление в локальной переменной ϵ .

15. Процесс ЭМ, получивший предварительную возможность взаимодействия, объявляет себя ведущим. В этом состоянии он находится до тех пор, пока аналогичную заявку не сделают все остальные процессы ЭМ, достигшие Л-оператора в е д у щ и й в тот же момент времени.

Действительно, ведущими станут в первую очередь процессы, имеющие высший приоритет. (Л-оператор в е д у щ и й дополняет средства, описанные в [5].)

16. Ведущий процесс ЭМ проверяет, не нарушили ли более приоритетные ведущие процессы условий взаимодействия. Если да, то статус ведущего теряется и ожидание начинается с начала (§5).

17. Процесс, получивший возможность взаимодействия, вырабатывает случайную величину $\underline{c}'$ (время взаимодействия), отмечает (захватывает) ведомые ЭМ в векторе $\underline{a}'$; занимает линии связи (уменьшает ℓ) и пересылает свою коммутацию $\underline{c}$ в глобальную переменную $\underline{c}'$, доступную всем другим процессам.

18. Прерывание ведомых на время взаимодействия, задержка их и ведущего на это время и накопление времени взаимодействия в соответствующих локальных переменных $\underline{b}$.

19. Ведущий процесс, завершивший взаимодействие, возвращает ведомые в предыдущие состояния, освобождает вектор $\underline{a}'$ от соответствующих отметок, освобождает линии связи, накапливает в переменной m число своих выдач и переходит на счет (§3).

20. Генератор случайных чисел (см. [3]).

21. После окончания эксперимента в заданной точке пространства $\{\alpha, \rho, N, m\}$ следящий процесс объявляет себя ведущим и запускает все N процессов ЭМ с §12, независимо от их состояния. Сам следящий процесс продолжит свою работу с §13.

22. Все процессы ЭМ проходят процедуру сбора статистики и удаляются из модели (ф и н и ш).

23. Следящий процесс задерживается на время сбора статистики и, оставшись один, запускает подпрограмму обработки, оценивает результат (о ц е н к а вырабатывает вектор оценки $\underline{h}'$) и печатает его.

24. Переход на вычисление очередного варианта.

25. Сбор статистики.

26. Обработка статистики.

27. Процедура сбора и обработки статистики, которую Л-оператор слежение навязывает процессу ЭМ номер 1.

28. Конец.

Объем машинной программы заказа № 1 для ЭВМ "Минск-22" - 1644, команд. Таким образом, с учетом рабочего поля ПС-Р-ЛЯПАС [6] описанная модель занимает 6134, ячейки 1-го блока памяти ЭВМ.

4. Результаты моделирования

4.1. С помощью описанной программы исследован вид функции $k_c(\alpha, \rho, N, m) = 0,725$ при $N = 4, 8, 12, 16$; $m = 1 \div \lfloor \frac{N}{2} \rfloor$. Точность эксперимента, при которой было получено большинство результатов, составляет $\pm 0,025 k_c$. Это позволяет произвести достаточно хорошую количественную оценку поведения модели ОВС, однако некоторые тонкости, возможно, были упущены при сглаживании кривых.

4.2. На рис.2 показано семейство сечений поверхности $k_c = 0,725$ при $m = \lfloor \frac{N}{2} \rfloor$, $N = 4, 8, 12, 16$ в координатах $\{\alpha, \rho\}$. Если параметры задачи $\{\alpha, \rho\}$ таковы, что эта точка окажется для

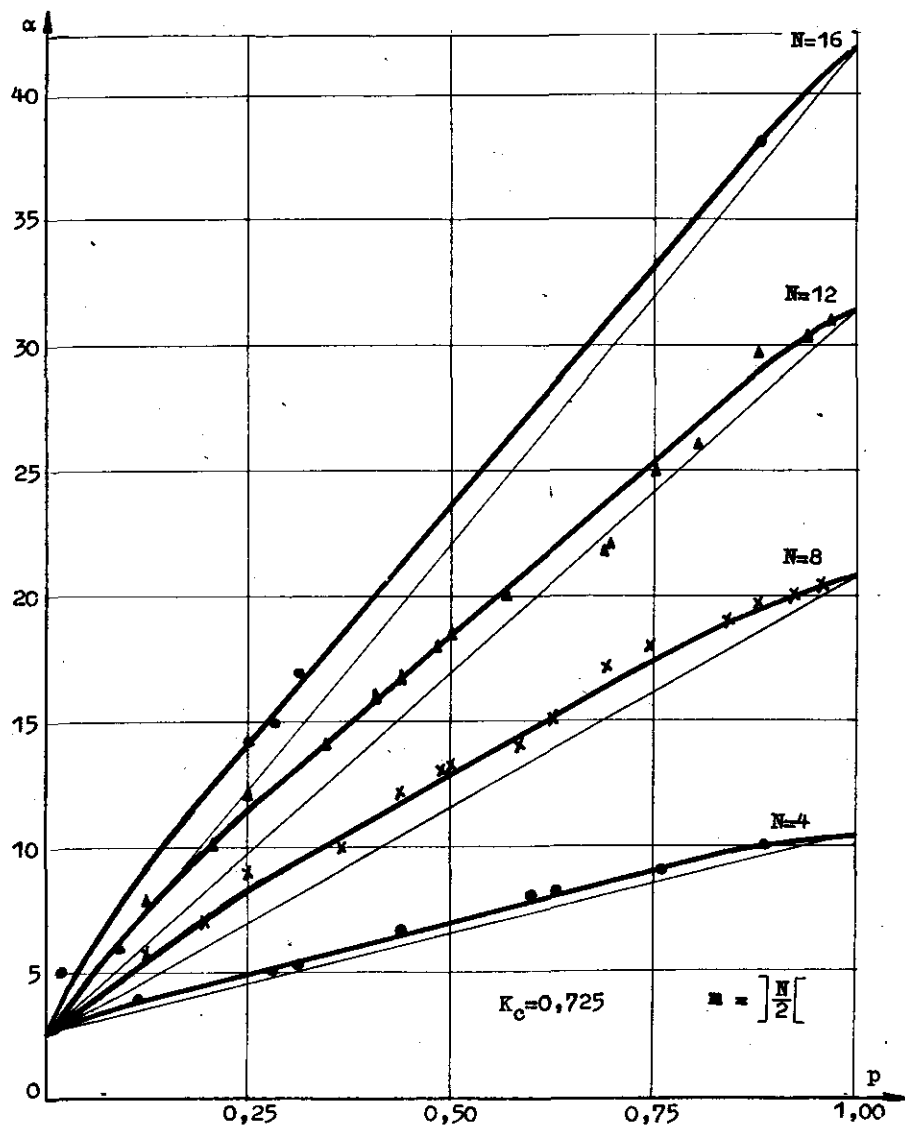


Рис. 2. Сечения поверхности $K_c = 0.725$ при $m = \frac{N}{2}$

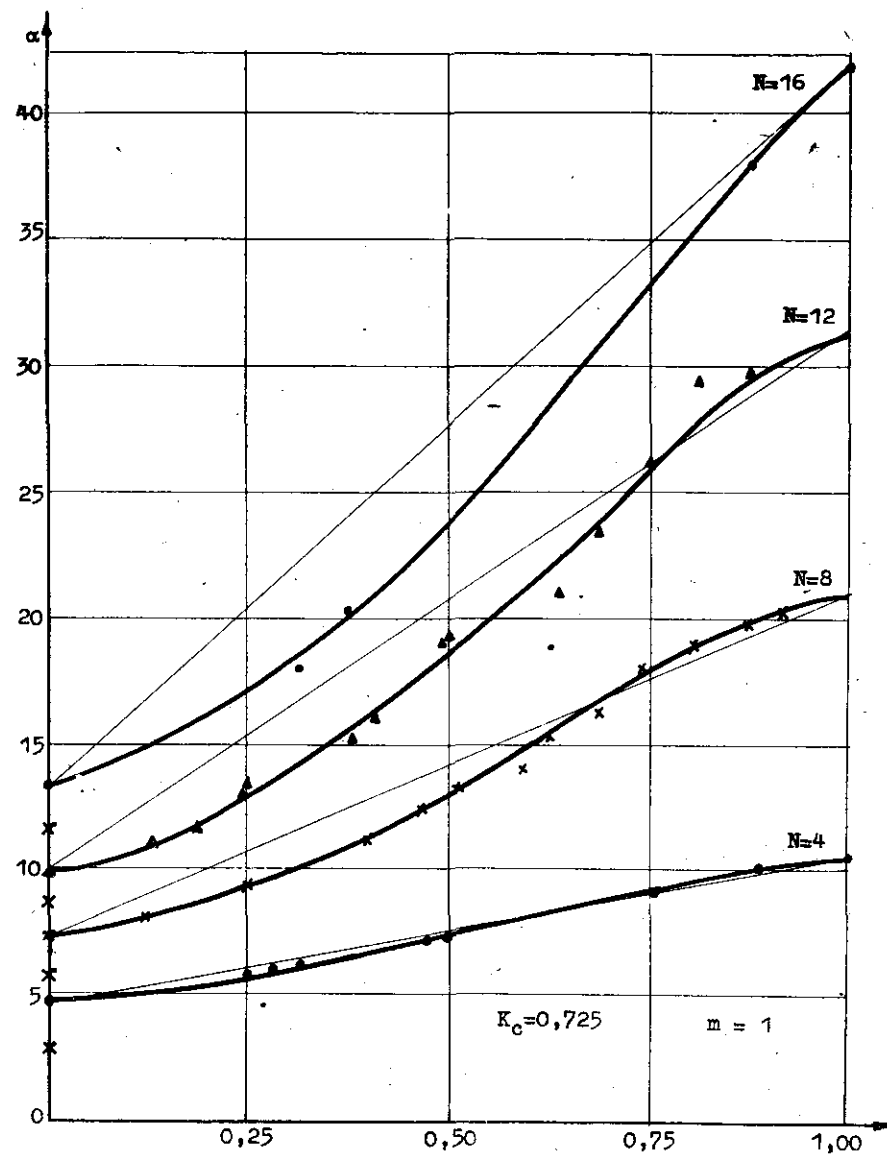


Рис. 3. Сечения поверхности $K_c = 0.725$ при $m = 1$

заданного N выше соответствующей кривой, то достигается коэф-
фициент счета $k_c > 0,725$.

Точки $\{\alpha_0, 0\}$ и $\{\alpha_1, 1\}$ подсчитаны по формулам (3), (4)
и хорошо согласуются с данными эксперимента.

На рис.3 дано то же семейство при $m=1$. Крестиками отме-
чены исходные точки $\{\alpha_0, 0\}$, даваемые формулой (3). Видно, что
при $m=1$ метод динамики средних дает слишком оптимистичную
оценку и здесь необходимо пользоваться более точной формулой.

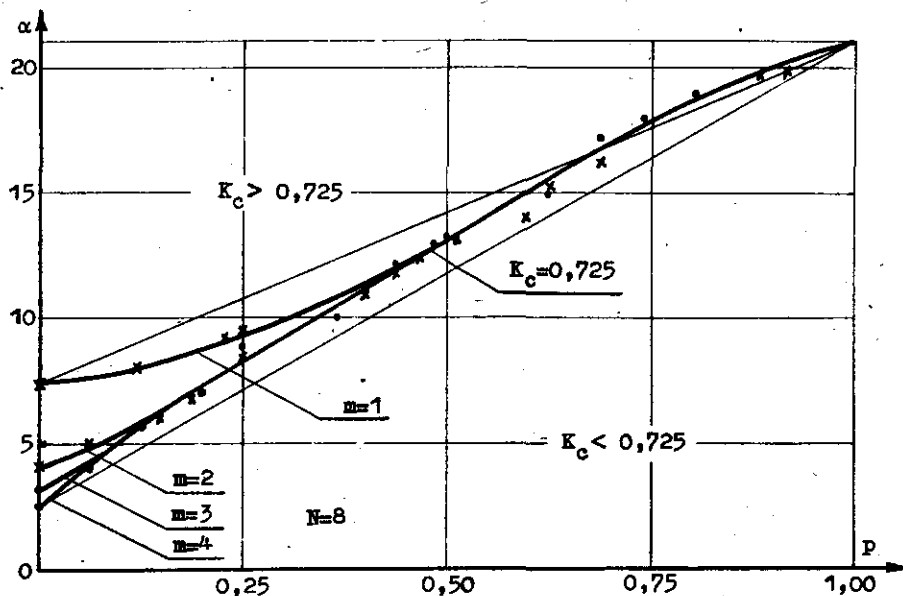


Рис. 4

На рис. 4 дано семейство сечений поверхности $k_c = 0,725$
для $N=8$; $m=1,2,3,4$. Аналогично выглядят и семейства для
 $N=4,12,16$. Видно, что с увеличением m показатели качества
систем быстро стремятся к предельным значениям (при полной ком-
мутационности), а при $m=N(1-k_c)$, как и ожидалось, почти не от-
личаются от предельных.

Из анализа рис. 2-4 следует, что требования, предъявляе-
мые к программе или системе программ для полностью коммутируе-
мой ОВС, чуть больше, чем линейная зависимость:

$$\alpha = \alpha_0 + (\alpha_1 - \alpha_0)\rho. \quad (6)$$

Аналогично, при $m=1$ они чуть меньше, чем

$$\alpha(m=1) = \alpha_0(1) + [\alpha_1(1) - \alpha_0(1)]\rho. \quad (7)$$

Итак, если системный программист вынужден добиваться вы-
сокого коэффициента счета ($k_c > 0,7$), то ему следует оценить па-
раметры системы программ и построить прямые (6) и (7). Если
точка $\{\alpha, \rho\}$ лежит выше обеих прямых, то его задача осущест-
вима без дополнительных организационных усилий; между прямыми-
положение может быть улучшено специальным размещением задач в
элементарных машинах ОВС; ниже - осуществимость системы про-
грамм становится проблематичной (конечно, если нет более силь-
ных ограничений на коммутации).

Если число M задач в системе значительно больше числа N
машин, то плотность взаимодействия между ЭМ

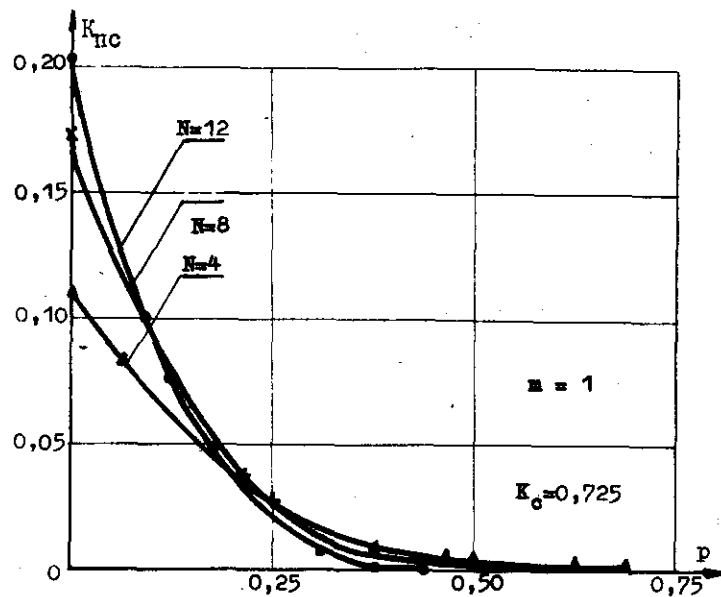
$$\rho = 1 - (1 - \rho')^{M/N},$$

где ρ' - плотность взаимодействия между задачами.

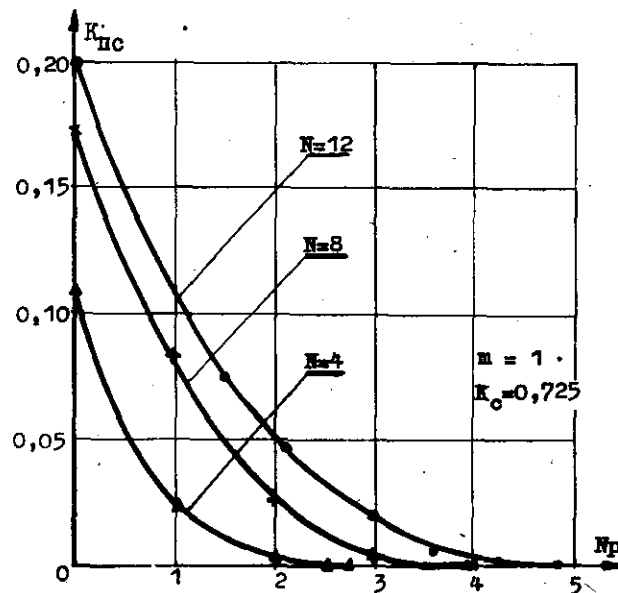
4.3. Как и ожидалось, с ростом плотности взаимодействия
значение фактора коммутируемости быстро уменьшается. Коммути-
руемость полностью определяет коэффициент системных потерь k_{nc} .
Представление о нем дает рис.5, где изображено семейство кри-
вых $k_{nc} = k_{nc}(\rho)$ и $k_{nc} = k_{nc}(N\rho)$ для $N=4,8,12$ при $m=1$.

При высоких коэффициентах счета ($k_c > 0,7$) и плотности
взаимодействия $\rho > 0,25$, системные потери настолько малы, что
вряд ли выгодно решать задачу загрузки ЭМ во время ожидания
коммутаций.

4.4. Интересна зависимость коэффициента алгоритмических
потерь $k_{па}$ от параметров ОВС и системы задач, показанная для
 $N=4,8,12$ на рис. 6. Во-первых, $k_{па}$ практически не зависит
от коммутируемости ОВС; во-вторых, он невелик ($k_{па} < 0,04$), име-
ет максимум и уменьшается с ростом плотности взаимодействия.
Этот результат не будет неожиданным, если вспомнить, что при
 $\rho = 1$ коэффициент $k_{па} = 0$, так как в момент взаимодействия
не может возникать новых заявок.



a)



б)

Рис. 5. Кривые системных потерь

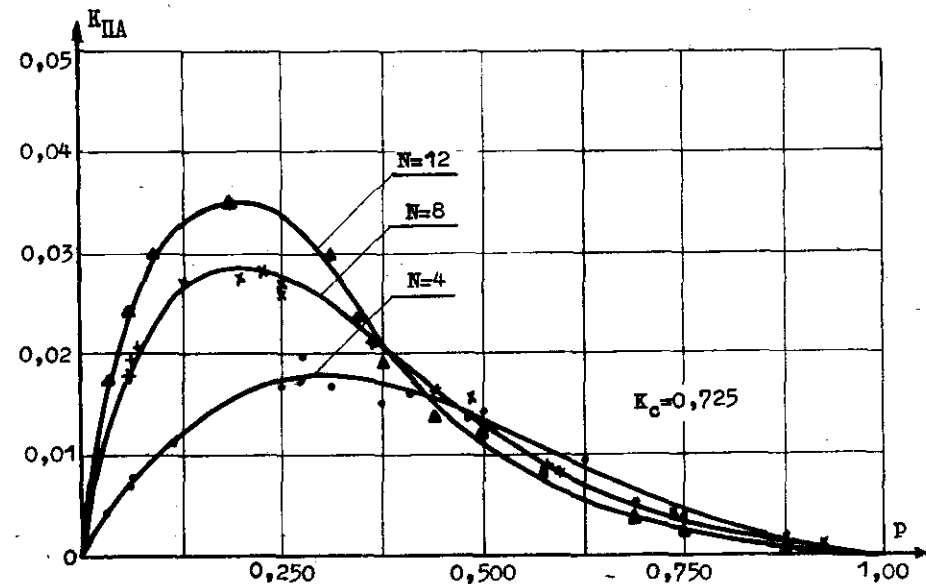


Рис. 6

Таким образом, асинхронность порождения заявок на комму-
тации при высоких k_c не играет роли. С ростом ρ происходит как
бы самоорганизация системы. Это, по-видимому, в
значительной мере должно снижать заботы системного программиста о синхронизации взаимодействий в ОВС.

4.5. До сих пор речь шла только о среднем по всем ЭМ коэф-
фициенте счета k_c . На самом же деле он будет различным для
ЭМ, имеющих разный приоритет доступа к системе связей. Очеви-
дно, что это имеет место только для ОВС с ограниченными комму-
тационными возможностями. Поправку k_c для данной ЭМ можно
оценить по той доле n заявок от общего их числа, которая прихо-
дится на эту машину при заданных N, m, ρ .

На рис.7 дано семейство кривых $n=n(\pi)$, где π - приори-
тет, $N=12, m=1$, при различных плотностях взаимодействия ρ .
Видно, что с ростом ρ вклад, вносимый каждой ЭМ в общую рабо-
ту, быстро стремится к среднему.

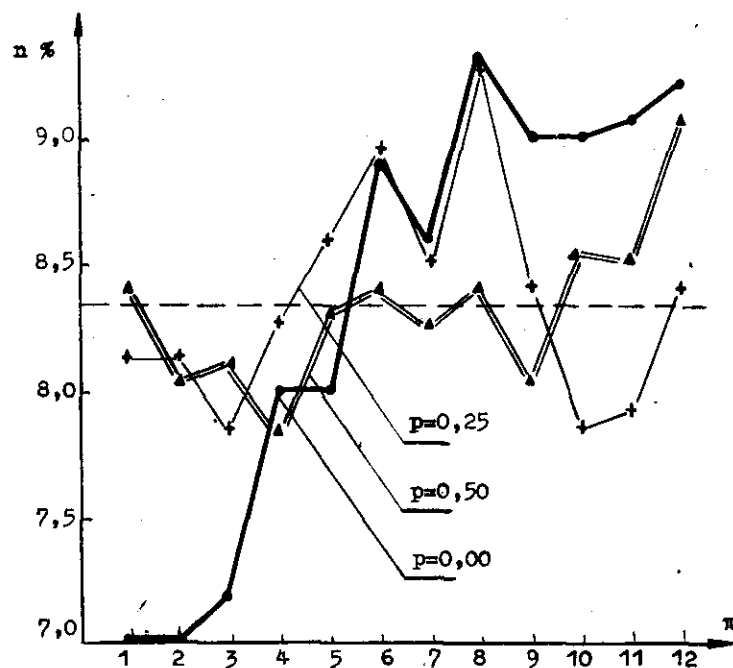


Рис. 7

Итак, ОВС имеет то преимущество, что количество процессоров в ней может быть большим и все они будут равномерно загружены даже при элементарной внутренней дисциплине.

Итак, рассмотрен довольно частный и, скорее всего, трудный случай функционирования ОВС, когда коэффициент счета достаточно высок ($\geq 0,7$), а взаимодействия между ЭМ наименее упорядочены (плотность взаимодействия не зависит от расположения ЭМ в системе).

Предложена программа моделирования, позволяющая определить те ограничения, которым должна удовлетворять система параллельных задач, решаемых на ОВС, чтобы достичь заданного уровня эффективности вычислений. Результаты моделирования показали, что даже при высоких требованиях на эффективность ОВС и элементар-

ной внутренней дисциплине эти ограничения вполне соответствуют реальным возможностям программиста при числе ЭМ около 20.

Предложена методика приближенной аналитической оценки эффективности системы параллельных программ на ОВС.

Моделирование показало, что если система программ осуществима на ОВС с эффективностью 0,7 и более, то:

1) коммутруемость ОВС влияет на эффективность вычислений (и, соответственно, на требования к программам) только при малых плотностях взаимодействия;

2) в случае, когда коммутации охватывают (в среднем) более четверти всех ЭМ, можно использовать даже I-коммутуемые ОВС без существенных системных потерь;

3) при этих же условиях пренебрежимо малы и алгоритмические потери.

Таким образом, в рассмотренной ситуации вполне достаточна элементарная внутренняя дисциплина.

Л и т е р а т у р а

1. ВОРОБЬЕВ В.А., КОРНЕЕВ В.В. Некоторые вопросы теории структур однородных вычислительных систем. - Настоящий сборник, стр. 3-16.

2. КОСАРЕВ Ю.Г. О структурах вычислительных систем, устойчивых к изменению числа машин. - В кн.: Вычислительные системы, Вып. 42. Новосибирск, 1970, стр. 59-73.

3. ВОРОБЬЕВ В.А., КОРНЕЕВ В.В. Метод статистических испытаний при исследовании характеристик осуществимости. - Настоящий сборник, стр. 70-83.

4. ВОРОБЬЕВ В.А. Р-ЛЯПАС - базовый язык моделирования дискретных устройств. - В кн.: Вычислительные системы, Вып. 39. Новосибирск, 1970, стр. 67-81.

5. ВОРОБЬЕВ В.А. Моделирование системы параллельных процессоров на Р-ЛЯПАСе. - В кн.: Вычислительные системы, Вып. 51. Новосибирск, 1972, стр. 82-97.

6. ВОРОБЬЕВ В.А., ПЕТРОВА Э.А. Программирующая система ЛЯПАС-Р-ЛЯПАС "Минск-22". Инструкция по эксплуатации. - В кн.: Вычислительные системы, Вып. 59, Новосибирск, 1974, стр. 162-179.

7. ХОРОШЕВСКИЙ В.Г. Исследование функционирования однородных вычислительных систем. Диссертация на соискание ученой степени доктора технических наук. Л., 1973, (ЛЭТИ).

8. ВЕНЦЕЛЬ Е.С. Исследование операций. М., "Сов. радио", 1972.

Поступила в ред.-изд. отд.
23 октября 1973 года