

УДК 681.31:519.3

ОБ ИССЛЕДОВАНИИ ПЕРЕСТРАИВАЕМЫХ ОДНОРОДНЫХ РЕШАЮЩИХ ПОЛЕЙ

В.В.Игнатушенко

§1. Особенности однородных решающих полей

Одним из принципиальных отличий многопроцессорных вычислительных систем с перестраиваемой структурой типа [1] от "классических" ЭВМ является их способность к предоставлению программам (или их параллельным фрагментам) стольких вычислительных ресурсов – процессорных элементов – сколько готовых к выполнению однотипных или разнотипных команд содержится в этих программах на данном шаге вычислительного процесса, – разумеется, в рамках общего объема вычислительных ресурсов системы.

Слова "готовые к выполнению команды" предполагают возможность функционального разграничения процессоров таких систем на устройства управления (УУ), которые считывают команды, дешифрируют их, обрабатывают адреса операндов, поставляют операнды из оперативной памяти или из собственных регистров, т.е. подготавливают команды к выполнению, и на собственно вычислительные ресурсы – однотипные процессорные элементы, выполняющие команды арифметических и логических действий над полученными операндами и образуящие перестраиваемое однородное решающее поле. Обобщенная блок-схема центральной части – вычислительного ядра – перестраиваемой многопроцессорной ВС представлена на рис.1.

Отметим, что свойство динамического перераспределения вычислительных ресурсов между устройствами управления, присущее системам с перестраиваемой структурой типа [1], обеспечивает им эффект динамического преобразования организации управляющих и вычислительных ресурсов, т.е. преобразования одного типа структуры в другой тип в соответствии с текущими требованиями параллельного вычислительного процесса. Так, в ВС [1], структура которой соответ-

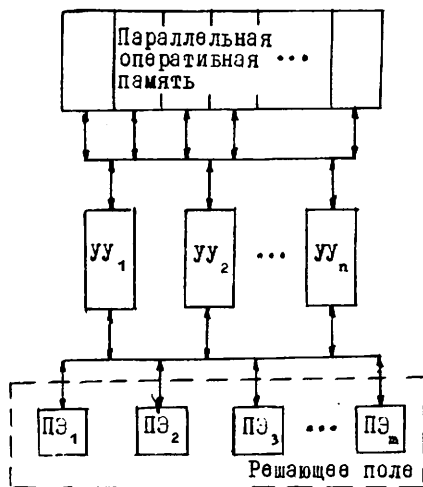


Рис. I

ствует известному классификационному признаку "множественных потоков команд и данных" (рис. I), все процессорные элементы могут быть монополизированы, например, одной высокоприоритетной программой (одним из устройств управления), осуществляющей в текущий момент времени векторные вычисления, и система в целом функционирует так, как если бы она имела структуру с "одним потоком команд и множественными потоками данных", т.е. структуру векторного процессора. У ка-

занное свойство перестраиваемых многопроцессорных ВС, очевидно, существенно расширяет возможности их проблемной ориентации.

Вообще однородные решающие поля из однотипных вычислителей различаются, согласно классификации [2], по следующим трем типам их организации:

1. Синхронное решающее поле с общим управлением, где все m процессорных элементов решающего поля в каждый момент времени выполняют одну и ту же команду, т.е. вычисления организуются по принципу "один поток команд - множественный поток данных".

2. Синхронное решающее поле с распределенным управлением. В таком поле - например, на основе однородных вычислительных структур [3], - допускается одновременное выполнение как однотипных, так и разнотипных операций.

3. Асинхронное решающее поле с распределенным (автономным) управлением. При такой организации поля, готовые к выполнению команды (в общем случае - разнотипные) из одного или нескольких $УУ$, образуют общую очередь к решающему полю. По окончании операции в каком-либо процессорном элементе первая в очереди команда передается на выполнение в него. Будем отдавать себе отчет, что при такой организации поля возникают дополнительные трудности в обмене данными между различными процессорами из-за непредсказуемости мо-

мента окончания операции в них (отметим, что длительность выполнения команды в процессорных элементах зависит не только от типа операции, но и от численных значений обрабатываемых операндов): процессорный элемент может быть занят выполнением предыдущей команды, когда на него подается результат завершившейся операции с другого элемента. В связи с этим возникает необходимость буферизации операндов и результатов на процессорных элементах и/или в УУ: каждый выходной операнд либо посылается в буфер того УУ, из которого была передана выполненная команда, либо остается в буфере того же элемента для последующего использования (обе эти возможности предусмотрены в многопроцессорных ВС [1]), либо пересылается в процессорный элемент-приемник.

Простой процессоров при асинхронной организации поля связаны, следовательно, с использованием результатов вычислений данного цикла в следующем цикле. Эти простои могут быть исключены при временном разделении работы каждого элемента по принципу "пинг-понга" [2]: если элемент в j -м цикле вычислений выполняет команду задачи (ветви) А, то в его буфер в этом же цикле передаются из другого УУ операнды и команда задачи (ветви) В, в $(j+1)$ -м цикле элемент выполняет команду задачи В, результат предыдущей операции заносится в соответствующий УУ, откуда в течение того же $(j+1)$ -го цикла передаются операнды и команда задачи А и т.д.

В [2] аналитически показано, что однородные перестраиваемые (динамически перераспределяемые между УУ) асинхронные решающие поля являются принципиально более эффективными, чем синхронные решающие поля, с точки зрения минимизации потерь производительности каждого процессорного элемента и поля в целом.

Принципиальный вопрос организации вычислительного ресурса многопроцессорной ВС, состоящего из нескольких процессорных элементов, заключается в следующем: что дает динамическое перераспределение процессорных элементов между несколькими УУ в отличие от неперестраиваемого вычислительного ресурса, жестко зафиксированного за каждым УУ? Аналитические исследования и имитационные эксперименты по реализации реальных параллельных программ для многопроцессорной ВС [1] показывают, что за счет собственно динамической перестройки ресурсов решающего поля (связанной с возможностью доступа любого УУ ко всем элементам) обеспечивается на 22-36% большая загрузка и, следовательно, производительность решающего поля, по сравнению с жестко зафиксированным для каждого

УУ вычислительным ресурсом, состоящим из нескольких элементов (разумеется, при равном общем числе элементов в обеих версиях поля). Указанный эффект обусловлен, в первую очередь, тем, что в реальных программах присутствуют слабо распараллеливаемые участки, процедуры перегрузок УУ, команды управления, пересылок и пр.; в таких случаях ресурс, жестко зафиксированный за УУ, либо недоиспользуется, либо полностью простаивает. В перестраиваемом поле этот ресурс может быть отпущен другой ветви или, быть может, другой задаче, в которой, наоборот, в текущий момент времени процесс широко распараллелен (например, при векторных вычислениях).

Следует подчеркнуть, что динамическое перераспределение ресурсов решающего поля в темпе выполнения команд должно осуществляться аппаратно. Операционная система, из-за ее инерционности "не имеет права" быть причастной к этой процедуре. Для программиста же решающее поле может представляться как некий неограниченный вычислительный ресурс, в котором одновременно может выполняться любое количество параллельных скалярных команд или команды обработки векторов с произвольным количеством элементов. В этом смысле решающее поле может являться виртуальным ресурсом, действительный объем которого программисту неизвестен.

Виртуальное решающее поле обладает принципиальным достоинством с точки зрения живучести ВС. Поскольку любая команда из любого УУ может выполняться на любом процессоре, то отказ одного или нескольких из них может восприниматься аппаратурой ВС как уменьшение числа перераспределяемых элементов, без изменения алгоритмов функционирования УУ и алгоритма перераспределения ресурсов решающего поля, а отказавшие элементы воспринимаются как, например, постоянно занятые. Таким образом, существует принципиальная возможность сохранять работоспособность такого поля вплоть до отказа всех процессоров, кроме одного, — разумеется, с соответствующим снижением его производительности.

В последующих разделах под решающим полем будет подразумеваться однородное асинхронное решающее поле с распределенным, автономным для каждого процессорного элемента управлением. Поскольку организация таких полей является сравнительно новым структурным средством повышения производительности ВС, то библиография работ, посвященных аналитическому исследованию функционирования перестраиваемых решающих полей, их взаимодействия с устройствами управления ВС, весьма немногочисленна. Наиболее

представительные из этих работ затрагиваются в нашей статье. Описываемые аналитические исследования проводятся на основе математического аппарата систем массового обслуживания.

§2. Базовая математическая модель

Решающее поле и функциональные узлы УУ, предназначенные для подготовки команд к выполнению, вместе с системой связей между УУ и решающим полем и аппаратными средствами перераспределения его ресурсов между УУ, образуют подсистему обработки.

Наиболее близкой к исследуемой подсистеме обработки по правилам взаимодействия между УУ и решающим полем и анализируемым параметрам является модель мультипроцессора, рассмотренная в [4] и представляющая собой двухфазную систему массового обслуживания без потерь, с буфером между фазами и бесконечной очередью на входе; в первой фазе функционируют n устройств управления, во второй m одинаковых вычислителей; время генерации команд в УУ и время выполнения команд в вычислителях распределены по экспоненциальному закону.

В реальных подсистемах обработки, и в частности в ВС типа [1], имеют место существенные отличия от моделей, рассмотренных в литературе, в том числе в [4].

Исследуемые подсистемы обработки функционируют в дискретном времени, поэтому в нашем случае целесообразно использовать дискретные законы распределения времени генерации и выполнения команд с учетом возможных изменений интенсивности генерации и выполнения команд в зависимости как от типов команд (скалярные, векторные), так и от типов операций (сложение, умножение и пр.). При функционировании систем в дискретном времени начинает проявляться, как правило, неординарность потоков заявок на обслуживание и потоков обслуженных заявок. В исследуемых подсистемах обработки УУ имеют автономные (локальные) буферы, а не общий буфер для всех УУ. Рассмотренные в литературе модели игнорируют физически существующие системы связей между УУ и решающим полем, а следовательно, — и конфликты в них.

Указанные отличия удобно анализировать на основе базовой математической модели некоторой гипотетической подсистемы обработки, блок-схема которой, аналогичная блок-схеме мультипроцессора по [4], изображена на рис.2 (Б-буфер, ПЭ-процессорный элемент). В

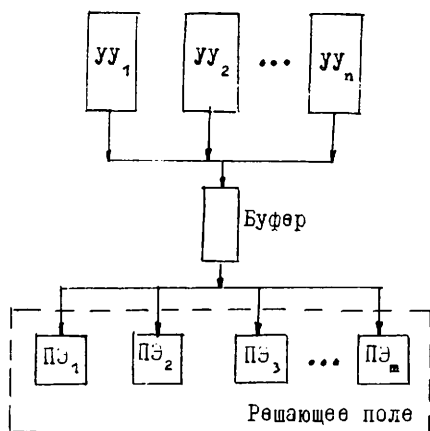


Рис. 2

этой подсистеме обработки каждое из n УУ, считывая и обрабатывая программу, генерирует последовательность готовых к выполнению команд, которые вместе со своими операндами поступают в буфер из k мест. Из буфера команда направляется на любой из m свободных процессорных элементов решающего поля и выполняется им. Устройства управления генерируют команды до тех пор, пока не заполнен буфер; после этого все УУ прекращают свою работу

(блокируются) до тех пор, пока не освободится место (ячейка) буфера. В момент появления свободного места в буфере, в соответствии с некоторой приоритетной дисциплиной, одно из УУ начинает генерацию новой команды.

В соответствии с описанным принципом работы процесс выполнения программ в подсистеме обработки представим в виде системы часового обслуживания без потерь, состоящей из двух фаз, с конечным буфером между фазами (максимальное число находящихся в буфере команд равно k , где $k=0,1,2,\dots$) и бесконечной очередью на входе (последним предполагается опережающее считывание команд по отношению к их обработке, при отсутствии прерываний работы УУ, перезагрузки ветвей и пр.). В первой фазе параллельно и независимо друг от друга работают n устройств управления, во второй фазе параллельно и независимо работают m однотипных процессорных элементов. Функции распределения времени генерации и выполнения команд могут иметь различный вид. Будем считать, что время генерации команд в УУ распределено по геометрическому закону. Выбор именно такой функции распределения является здесь следствием параллелизма процессов обработки команд в каждом УУ (по крайней мере, для ВС [1]): опережающего чтения команд из оперативной памяти, расчета адресов, чтения из оперативной памяти скалярных и векторных операндов при наличии конфликтов в общей памяти, работы с внутрен-

ними буферами в УУ, промежуточных обращений к памяти при обработке косвенных адресов, пересылки результатов выполнения команд в оперативную память и пр. Как известно, закон распределения интервалов между заявками для суммарного потока, формируемого из нескольких потоков даже с неизвестными законами распределения длин интервалов, близок к геометрическому (экспоненциальному для непрерывного времени [5]).

Время выполнения команд в решающем поле вначале также будем считать распределенным по геометрическому закону: при таком распределении (т.е. при коэффициенте вариации $v = 1$) оценки загрузки решающего поля являются предельными в том смысле, что при любом другом распределении с $0 \leq v \leq 1$ показатели качества функционирования будут выше [5].

Обозначим через λ вероятность завершения выполнения команды во второй фазе системы в произвольном такте. Эта вероятность характеризует производительность решающего поля в целом, а величина λ/μ соответствует вероятности выполнения команд в одном процессорном элементе. Вероятность генерации команд всеми УУ обозначим через μ . Тогда параметр $\rho = \mu/\lambda$ характеризует сбалансированность системы. В случае $\rho > 1$ интенсивность суммарного потока команд с УУ больше интенсивности обслуживания (выполнения команд) в рабочем поле; очевидно, возрастает вероятность загрузки процессорных элементов и вероятность блокировки УУ. При $\rho < 1$ уменьшается вероятность загрузки решающего поля, т.е. уменьшается ее реальная производительность, которая и определяет, главным образом, эффективную производительность ВС в целом. Случай $\rho = 1$ соответствует сбалансированной системе, в которой вероятности загрузки УУ и решающего поля равны между собой.

Рассмотрим поведение системы при ординарных потоках генерации и выполнения команд (т.е. в каждой из фаз в отдельности), но, естественно, с учетом вероятности $\mu\lambda$ совпадения событий генерации команды в УУ и выполнения команды в рабочем поле в одном такте.

Будем рассматривать следующие $m+k+1$ состояний a_i системы ($i = 0, 1, 2, \dots, m+k$), характеризующихся числом i команд, находящихся в решающем поле и в буфере:

a_0 - работают все УУ, свободны все процессорные элементы, буфер пуст;

a_i - работают все УУ, загружены i процессорных элементов, буфер пуст; $1 \leq i < m$;

a_m - работают все УУ, загружены все m процессорных элементов, в буфере имеется $i-m < k$ команд, $m \leq i \leq m+k-1$;

a_{m+k} - загружены все процессорные элементы, в буфере находятся k команд, блокирована работа всех УУ.

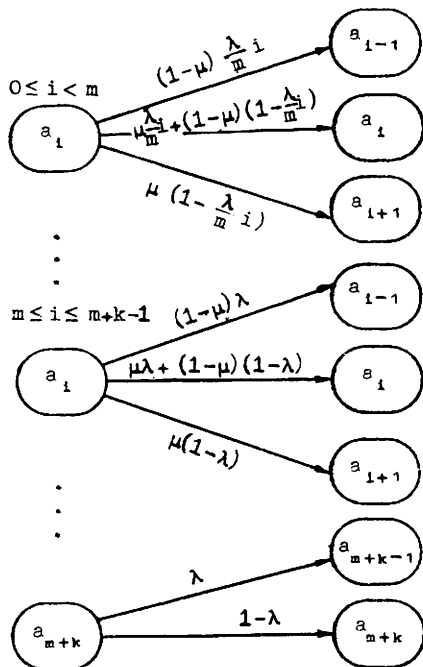


Рис. 3

Поведение такой двухфазной системы описывается неприводимой неперiodической цепью Маркова, для которой существует стационарный режим [6].

Граф переходов системы в различные состояния изображен на рис.3. По этому графу известными методами можно составить систему уравнений, связывающую вероятности пребывания системы в различных состояниях в стационарном режиме; решение этой системы уравнений приводит к следующим выражениям для стационарных вероятностей P_i состояний системы (при $0 < \lambda < 1$):

нарных вероятностей P_i состояний системы (при $0 < \lambda < 1$):

$$\left. \begin{aligned} P_i &= P_m \left(\frac{1-\rho\lambda}{\rho} \right)^{m-i} \prod_{j=1}^{m-i} \frac{m-j+1}{m-(m-j)\lambda}, \quad 0 \leq i < m, \\ P_i &= P_m \left[\frac{\rho-\rho\lambda}{1-\rho\lambda} \right]^{i-m}, \quad m \leq i \leq m+k-1, \\ P_{m+k} &= P_m \left[\frac{\rho-\rho\lambda}{1-\rho\lambda} \right]^{k-1} \cdot \rho (1-\lambda). \end{aligned} \right\} \quad (1)$$

Решение системы уравнений (I) с учетом условия нормировки

$\sum_{i=0}^{m+k} P_i = 1$ позволяет определить вероятность P_{m+k} блокировки УУ:

$$P_{m+k} = \frac{\rho(1-\lambda)}{\left(\frac{1-\rho\lambda}{\rho-\rho\lambda}\right)^{k-1} \left[\sum_{i=0}^m \left(\frac{1-\rho\lambda}{\rho}\right)^{m-i} \prod_{j=1}^{m-i} \frac{m-j+1}{m-(m-j)\lambda} - \frac{1-\rho\lambda}{\rho-1} \right] + \frac{\rho^2(1-\lambda)}{\rho-1}} \quad (2)$$

и, следовательно, вероятность H загрузки УУ, т.е. вероятность события, что УУ не простаивает из-за их блокировки:

$$H = 1 - P_{m+k}. \quad (3)$$

Очевидно, вероятность H здесь можно интерпретировать как

$H = \frac{n_{cp}}{n}$ (n_{cp} - среднее число неблокированных УУ), поскольку для нашей модели, с учетом (I), имеет место $n_{cp} = n \sum_{i=0}^{m+k-1} P_i = n(1 - P_{m+k})$.

Ключевым параметром в нашем анализе является вероятность E загрузки второй фазы (решающего поля); эта вероятность имеет здесь следующую физическую интерпретацию.

Среднее число m_{cp} процессорных элементов, занятых выполнением команд, выражается, с учетом (I), в виде $m_{cp} = \sum_{i=0}^{m-1} i P_i + m \sum_{i=m}^{m+k} P_i$.

Значение E соответствует численному выражению доли решающего поля, занятой обслуживанием заявок, т.е. $E = m_{cp}/m = \sum_{i=0}^{m-1} \frac{i}{m} P_i +$

$+ \sum_{i=m}^{m+k} P_i$. С другой стороны, E определяется как параметр, характеризующий эффективную производительность Π решающего поля:

$$\Pi = \frac{\lambda}{m} m_{cp} = \lambda E.$$

В [6] показано, что для двухфазных систем в стационарном режиме должно выполняться соотношение $\frac{\Pi}{n} n_{cp} = \frac{\lambda}{m} m_{cp}$, откуда

$$E = \rho H. \quad (4)$$

Из непосредственного анализа выражений (2)-(4) следует, что $dE/dk > 0$, т.е. с увеличением емкости буфера производительность решающего поля возрастает. Но наиболее важный практический вывод из анализа этих же выражений заключается, по аналогии с [4], в том, что $dE/dm > 0$ при $1 \leq m < \infty$ и при фиксированных остальных па-

раметрах модели, в том числе λ . Это означает, что если в решающем поле заменить систему процессорных элементов на другую систему с той же суммарной производительностью, но состоящую из большого числа менее быстродействующих процессорных элементов, то эффективная производительность такого решающего поля увеличится. В частности, наименьшее значение вероятности E имеет место для решающего поля, состоящего из единственного процессора с ожидаемой производительностью λ .

Указанный вывод аналитически подтверждает, что одним из принципиальных достоинств организации перестраиваемых решающих полей является возможность достижения весьма высокой эффективной производительности многопроцессорной ВС при реализации решающего поля на сравнительно медленнодействующих, а следовательно - более надежных и помехоустойчивых, недорогих процессорных элементах.

Конечно же, указанный вывод не означает, что решающее поле, состоящее из одного процессорного элемента, всегда будет уступать по эффективной производительности решающему полю из нескольких процессорных элементов. Во-первых, можно показать (как это сделано в [4] для аналогичной модели), что "недостающую" производительность можно компенсировать увеличением емкости буфера. Во-вторых, любое увеличение числа элементов требует усложнения схемы коммутации между УУ и решающим полем, фактическое существование которой вообще не учитывалось в математической модели (этот вопрос рассматривается в §5).

Из анализа базовой математической модели следуют и другие практические выводы [7,8]. В частности, по аналогии с [4], исследованы поведения сбалансированной базовой модели, т.е. при $\rho=1$, - режим, к которому обычно стремятся при проектировании подсистем обработки. Показано, что при увеличении емкости буфера $k \rightarrow \infty$ вероятности загрузки УУ и решающего поля стремятся к своим предельным значениям: $H=1$ и $E=\rho$ при $\rho \leq 1$; $E=1$ и $H=1/\rho$ при $\rho \geq 1$ (этот результат в точности совпадает с [4]). Наименьшие же значения вероятности E и H в сбалансированной системе имеют [8] при $\lambda \rightarrow 0$ и равны $E=H=\frac{k+1}{k+2}$. Зависимость вероятности E от емкости k буфера приведена на рис.4. Из этой зависимости следует практиче -

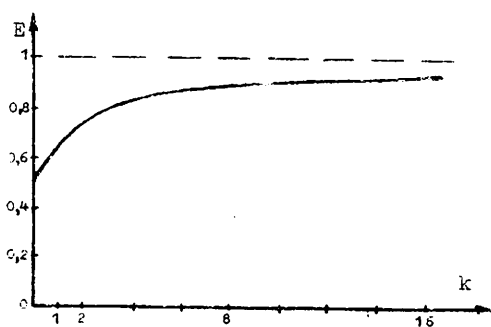


Рис. 4

При реальных значениях λ вероятность E будет выше и зависит, как мы показали, и от числа m процессорных элементов. Зависимость величины E от m в сбалансированной системе при $k = 8$ и $\lambda = 0,3$ приведена в табл. I.

Использованное в модели геометрическое распределение времени выполнения команд в процессорном элементе и решающем поле в целом приводит к нижней оценке производительности поля, т.е. при других законах распределения (при коэффициентах вариации $0 \leq v < 1$) прогнозируемая оценка производительности будет выше. Известно, что верхнюю оценку качества функционирования дает система обслуживания с постоянным временем обслуживания, при котором коэффициент вариации $v = 0$ [5]. Для вычислительных систем анализ детерминированного времени выполнения команд представляет и другой практический интерес, ибо соответствует анализу характеристик решающего поля, состоящего из процессоров с арифметико-магистральной обработкой.

Т а б л и ц а I

m	E	E_d
1	0,926	0,965
2	0,931	0,968
4	0,935	0,970
8	0,940	0,972
10	0,943	0,974

ская рекомендация для выбора объема буфера B : $k = 8-16$, и дальнейшее увеличение емкости буфера приводит лишь к незначительному росту E . Так, при $k \rightarrow \infty$ значение E отличается менее чем на 6% от величины E , полученной для $k = 16$ (речь идет, разумеется, о сравнении аналитических оценок).

В [8] показано, что в решающем поле с детерминированным временем выполнения команд вероятность P_{m+k} блокировки системы (а следовательно, и простоя поля, так как при $\rho = 1$ имеет место $E = H$) приблизительно вдвое меньше, чем при геометрическом распределении, при одинаковых значениях ρ, k, λ и m . Значения вероятности загрузки для этого случая, обозначенные E_d , приведены в табл. I.

Таким образом, на основе анализа рассмотренной математической модели с достаточно большой точностью - в пределах, задаваемых значениями E и E_d (см. пример по табл. I) - может быть оценена производительность решающего поля при любом реальном законе распределения времени выполнения команд (кроме гипергеометрического), а также обоснованно выбран объем буфера готовых к выполнению команд.

§3. Подсистема обработки с локальными буферами

В рассмотренной выше модели подсистемы обработки общий буфер образует последовательный канал доступа от УУ к решающему полю. Реально же, даже с точки зрения живучести, целесообразно организовывать собственные буферы готовых к выполнению команд в каждом из УУ; эти буферы названы локальными. Блок-схема такой подсистемы обработки приведена на рис. 5.

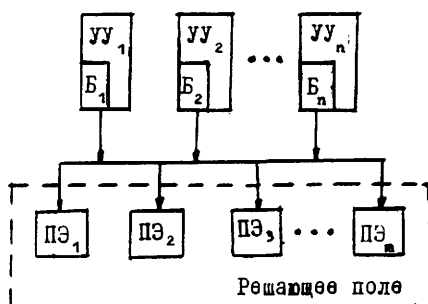


Рис. 5

Система функционирует следующим образом. Каждое УУ генерирует последовательность готовых к выполнению команд, которые поступают в соответствующий локальный буфер. При наличии свободных процессорных элементов, очередная команда поступает в решающее поле от одного из УУ (или из его локального буфера) в соответствии с некоторой дисциплиной доступа различных УУ к

полю. Работа каждого УУ организована так, что она генерирует команды с интенсивностью μ/n до заполнения собственного локального буфера. Как только буфер заполнится, УУ прекращает свою работу (блокируется) до освобождения одного места в буфере.

Система с любым числом n локальных буферов описывается n -мерной марковской цепью, которая рассчитывается численными методами. Проиллюстрируем ход исследования таких систем при $n=2$.

Состояния системы a_{ij} характеризуются здесь двумя числами $0 \leq i \leq k$, $0 \leq j \leq k$, где k - объем локального буфера. Как и в базовой модели, предполагается, что в произвольном такте с вероятностью λ заканчивается выполнение команды в решающем поле, и но-

вая команда поступает из какого-либо буфера на выполнение. Выбор команды из того или иного буфера происходит с одинаковой вероятностью, в данном случае равной $1/2$ (тем самым предполагается случайный выбор УУ для доступа к полю). Если один из буферов пуст, то команды выбираются из непустого буфера.

Состояния системы определяются следующим образом:

- a_1 - работают оба УУ, загружены 1 из m процессорных элементов, буферы B_1 и B_2 пусты; $0 \leq 1 < m$;

- a_{ij} - работают оба УУ, загружены все m процессорных элементов, в буферах B_1 и B_2 содержится i и j команд соответственно; $0 \leq i < k$, $0 \leq j < k$;

- a_{ik} , a_{kj} - работает одно УУ, другое УУ заблокировано, загружено все поле, в буферах B_1 и B_2 содержатся i и k (k и j) команд соответственно; $0 \leq i \leq k-1$, $0 \leq j \leq k-1$;

- a_{kk} - оба УУ заблокированы, все процессорные элементы загружены, в буферах B_1 и B_2 содержится по k команд.

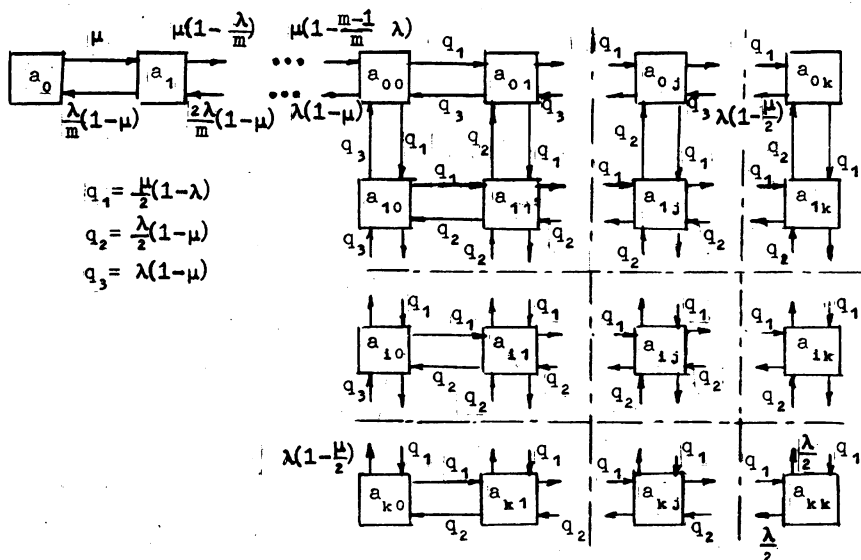


Рис. 6

Здесь по-прежнему суммарный поток генерации заявок обоими УУ (с суммарной интенсивностью μ) считается ординарным. Для простоты на рис.6 не приведены тривиально рассчитываемые вероятности перехода системы из каждого состояния a_{ij} в то же самое состояние a_{ij} .

Производительность такой системы сравнивалась с производительностью базовой модели, имеющей общий буфер объемом $2k$, при $\rho = 1$. Показано [7], что, как и следовало ожидать, производительность (вероятность загрузки) такой системы с общим буфером выше, чем производительность рассмотренной системы с локальными буферами, а равная производительность обеих систем достигается при объеме a общего буфера, равном $a = r \cdot 2k$, где $r = 0,75$, при одинаковых прочих параметрах моделей. Этот коэффициент пропорциональности $r = 0,75$ оказывается весьма стабильным по отношению к параметрам моделей (при $\rho = 1$). При численных расчетах систем с различными $\mu, \lambda = \mu$ и $k \geq 8$ значения r колебались только в пределах $0,73-0,76$. При малых значениях k величина r уменьшается (например, при $k = 4$ — приблизительно на 10%).

Аналогичным образом можно провести исследования для подсистемы обработки с тремя и четырьмя УУ, общий объем буферов которых равен соответственно $3k$ и $4k$. Содержательный результат такого исследования заключается в получении значений коэффициента r ; он оказался равным около $2/3$ и $1/2$ соответственно при $k \geq 8$.

Таким образом, полученные значения r позволяют заменять громоздкий анализ систем с локальными буферами при различных значениях параметров моделей более простым и наглядным анализом базовой математической модели, выбирая для нее соответствующий объем общего буфера.

6.4. 0 генерации команд с переменной интенсивностью

При функционировании реальных многопроцессорных ВС и их подсистем обработки, интенсивность генерации команд устройствами управления и интенсивность выполнения команд в решающем поле изменяется во времени в зависимости от типов обрабатываемых и выполняемых команд.

Так, длительности обработки векторной и скалярной команды (в пересчете на одну выполняемую в решающем поле операцию) существенно различны из-за различий во временах чтения программы, времени расчета адресов, различия в способах буферизации операндов и

результатов и пр. Это приводит к скачкообразному изменению интенсивности генерации команд в УУ при переходе от обработки векторной команды к скалярным командам, и наоборот.

Исследование подсистемы обработки с локальными буферами при переменной интенсивности потоков генерируемых команд проведено в [8]. Однако для аналитической оценки вероятностей загрузки УУ и рабочего поля по [8] необходимо знать соотношение между числом операций, выполняемых в решающем поле по векторным командам, и числом скалярных команд в каждой версии выполнения каждой программы, что практически непредсказуемо.

Здесь нам приходит на помощь то обстоятельство, что по крайней мере во многих реальных задачах "удельный вес" векторных вычислений весьма велик. Имитационное моделирование потоков генерации команд таких программ для ВС типа [1] показывает, что суммарный поток команд от УУ при $n = 4$ имеет почти постоянную интенсивность с колебаниями не более 12% от ее среднего значения. Такие изменения интенсивности потока команд уже при объеме $k = 8$ локальных буферов приводят к изменению коэффициента загрузки решающего поля (соответствующего вероятности E загрузки в аналитической модели) лишь на 2-3%. Разумеется, при "отключении" какого-либо УУ от решающего поля вследствие прерываний или перезагрузки ветвей, а также при выполнении в УУ команд пересылок массивов информации, интенсивность суммарного потока готовых к выполнению в решающем поле команд изменяется на десятки процентов и производительность поля может заметно снижаться. Чтобы избежать этого нежелательного эффекта, реально приходится идти на создание систем с $\rho > 1$.

Влияние изменений интенсивности λ выполнения команд в решающем поле на параметры загрузки системы оценивается в следующем разделе.

§5. Анализ конфликтов в подсистеме обработки

В предыдущем разделе исследование подсистемы обработки проводилось в предположении, что суммарный поток команд, генерируемых различными УУ, как и суммарный поток команд, выполненных различными процессорными элементами, является ординарным. Предполагалось также, что команды покидают систему по завершению их выполнения. В действительности же результаты вычислений должны передаваться в память или, как в [1], в соответствующие УУ. Вопрос о

параллельных (при неординарных потоках) обменах информацией в подсистеме обработки тесно связан с организацией структуры связей между УУ и процессорными элементами решающего поля.

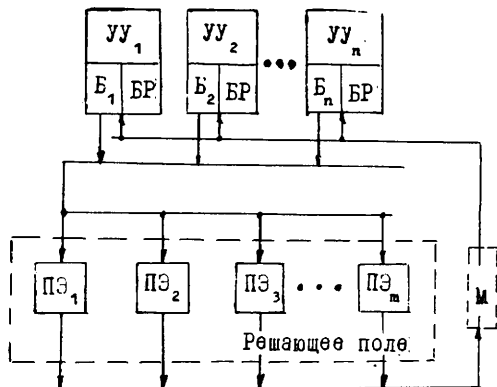


Рис. 7

Наиболее простым решением системы связей является магистральная структура для передачи команд из УУ в решающее поле и передачи результатов выполнения команд из решающего поля в УУ — в буферы результатов (БР) последних, рис. 7. Поскольку такая структура является и наименее худшей, с точки зрения параллелизма пе-

редачи информации, то ей присущи наибольшие потери производительности подсистемы обработки из-за конфликтов между УУ и между процессорными элементами при обменах информацией между УУ и решающим полем, которые мы и оценим.

Подсистема обработки представляется теперь в виде трехфазной системы обслуживания без потерь. В первой фазе работают УУ, каждое из которых имеет локальный буфер объемом k для готовых к выполнению команд и буфер для результатов вычислений, рис. 7. Команды, находящиеся в буферах, образуют общую очередь к решающему полю. Во второй фазе независимо функционируют m процессорных элементов (как и ранее, будем предполагать время генерации и время выполнения команд распределенными по геометрическому закону с параметрами μ и λ соответственно). Третьей фазой является магистраль М с фиксированным временем обслуживания (передачи) результатов вычислений, равным времени записи результатов в буфер УУ; это время принимается равным одному такту работы системы.

Такая модель выглядит упрощенной: в ней отсутствует фаза, соответствующая магистрали передачи команд из УУ к решающему полю. Это упрощение мотивируется следующим образом. Будем считать процессорный элемент свободным только после того, как он выдает

результат предыдущей команды в магистраль М. Это означает, что в течение одного такта в решающем поле может оказаться свободным не более одного процессорного элемента (при наличии команд в буферах). Предполагается, что передача команды от какого-либо УУ к освободившемуся процессорному элементу выполняется в том же такте, в котором этот элемент передает результат через магистраль в соответствующий буфер результатов. В этом случае для интересующей нас оценки простоев процессорных элементов достаточно учитывать конфликты из-за доступа только к магистрали М.

При одновременном завершении выполнения команд в $j > 1$ процессорных элементах, магистраль М в первом такте равновероятно представляется для передачи результатов одному из этих элементов; остальные $j - 1$ процессорных элементов блокируются, т.е. не принимают команды из УУ. В последующих тактах магистраль поочередно используется другими процессорными элементами, завершившими выполнение команд.

Аналитическое исследование такой системы весьма громоздко [8]. Содержательный результат этого исследования заключается в том, что вероятность блокировки процессорных элементов из-за конфликтов монотонно сходится с увеличением $k \geq 8$ к некоторому предельному значению, к вычислению которого мы переходим.

Вследствие конфликтов между процессорными элементами на магистрали, вероятность загрузки решающего поля уменьшается на некоторую величину z , под которой и имеется в виду вероятность блокировки элементов. Вывод оценки влияния конфликтов на загрузку элементов покажем для решающего поля, состоящего из двух элементов, т.е. $m = 2$, в этом случае интенсивность выполнения команд на одном процессорном элементе равна $\lambda_1 = \lambda/2$.

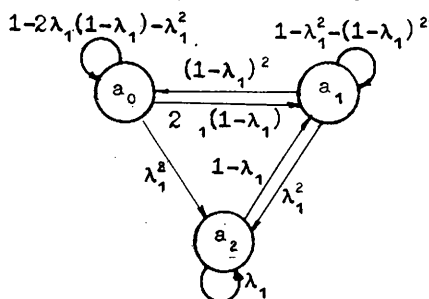


Рис. 8

Такое решающее поле может находиться в следующих трех состояниях: a_0 - оба элемента не обращаются к магистрали; a_1 - к магистрали обращается только один из элементов; a_2 - к магистрали обращаются оба элемента, одному из них (равновероятно) предоставляется доступ к магистрали -

ли, а другой элемент блокируется и повторяет обращение в следующем такте. Граф переходов системы в указанные состояния изображен на рис.8. Нетрудно убедиться, что выражения для стационарных вероятностей состояний имеют вид:

$$P_0 = (1 - \lambda_1)^3 / (1 - \lambda_1 + \lambda_1^2),$$

$$P_1 = [\lambda_1(1 - \lambda_1)(2 - \lambda_1)] / (1 - \lambda_1 + \lambda_1^2),$$

$$P_2 = \lambda_1^2 / (1 - \lambda_1 + \lambda_1^2).$$

Вероятность блокировки равна $z = P_2 / 2$, поскольку в состоянии a_2 блокирован один из двух элементов, т.е. $z = \lambda_1^2 / 2(1 - \lambda_1 + \lambda_1^2)$ и, следовательно, $0 \leq z \leq 1/2$ во всем диапазоне значений λ_1 .

Т а б л и ц а 2

λ	m		
	2	3	4
0,1	0,001	0,001	0,001
0,2	0,005	0,005	0,005
0,3	0,013	0,012	0,011
0,5	0,038	0,037	0,036
0,7	0,079	0,079	0,075
0,9	0,135	0,137	0,132
1,0	0,167	0,161	0,164
1,2	0,237	0,243	0,239
1,4	0,310	0,317	0,315
1,6	0,381	0,386	0,386
1,8	0,445	0,448	0,448
2,0	0,500	0,501	0,501
2,4	-	0,563	0,583
3,0	-	0,667	0,667

Аналогичным образом могут быть определены оценки вероятности z для решающего поля при $m > 2$. В табл.2 приведены значения z для $2 \leq m \leq 4$ при одной и той же суммарной интенсивности $\lambda = m\lambda_1$.

Поскольку полученные оценки потерь производительности решающего поля из-за конфликтов являются верхними (при иных, "более параллельных" структурах связей эти потери будут заведомо меньше), то уже из этих оценок можно извлечь некоторые практические рекомендации по организации связей между УУ и решающим полем. В частности, при достаточно больших m все процессорные элементы целесообразно разделить на группы — например, по числу УУ — с магистральными связями между УУ и элементами

одной группы и с радиальными связями между УУ и группами элементов (см. рис.9, где изображены связи только между выходами процессорных элементов и входами УУ); число процессорных элементов в группе выбирается в зависимости от значений z , считаемых проектировщиком допустимыми.

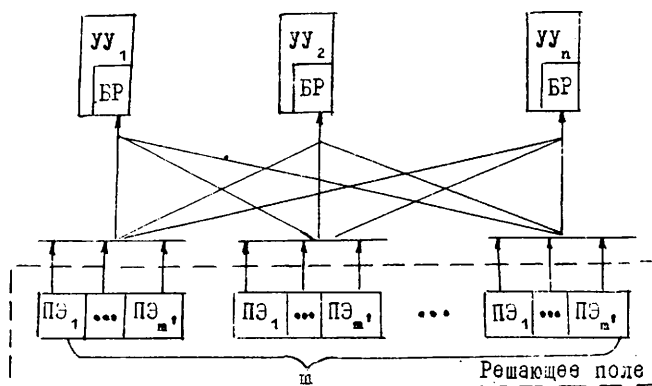


Рис. 9

Подробное исследование структур связей с анализом их сложности и пропускной способности, с учетом заданной производительности решающего поля для рассматриваемых подсистем обработки приведено в [9].

До сих пор нами предполагалось, что все процессорные элементы обращаются к магистрали передачи результатов с одинаковой интенсивностью λ_1 . Однако на этих элементах могут выполняться разнотипные команды с существенно различными временами выполнения, тогда длительность выполнения команд в решающем поле аппроксимируется гипергеометрическим законом распределения (с коэффициентом вариации $v > 1$). Оценка производительности такой системы может быть дана при подстановке предельных — максимальных и минимальных — значений параметров ρ и λ в выражения (I)–(4). Что же касается значения z , то эта величина может заметно уменьшаться. Покажем это на примере решающего поля, состоящего из $m=2$ процессорных элементов с интенсивностью выполнения команд λ_1' и λ_1'' и с суммарной интенсивностью $\lambda = \lambda_1' + \lambda_1''$, имеющей такую же величину, как и в предыдущей модели.

По аналогии с анализом предыдущей модели можно показать, что в рассматриваемом случае вероятность блокировки z' рассчитывается по формуле:

$$z' = \frac{1}{2} \cdot \frac{\lambda_1' \lambda_1''}{1 - \frac{\lambda_1' + \lambda_1''}{2} + \lambda_1' + \lambda_1''}.$$

Поскольку $\lambda_1' \cdot \lambda_1'' < \lambda^2$, то $z - z' > 0$, т.е. вероятность блокировки процессорных элементов в системе с неодинаковой интенсивностью выполнения команд меньше, причем величина этой вероятности уменьшается с увеличением разницы между λ_1' и λ_1'' . Аналогичный вывод справедлив и для решающего поля с $m > 2$ и с большим количеством разнообразных значений λ_1 [8].

Таким образом, при одновременном выполнении команд различных типов в решающем поле (а вероятность таких событий велика при динамическом перераспределении процессорных элементов поля между несколькими УУ) реальная производительность поля увеличивается за счет уменьшения простоев элементов из-за конфликтов в системе между решающим полем и УУ.

Л и т е р а т у р а

1. ПРАНГИШВИЛИ И.В., ИГНАТУШЕНКО В.В. и др. Организация вычислений на однородной управляющей вычислительной системе с перестраиваемой структурой. - В кн.: Тезисы докладов VII Всесоюзного совещания по проблемам управления, Минск, 1977. М., 1977, кн. 2, с. 382-384 (Ин-т проблем управления АН СССР).
2. ИГНАТУШЕНКО В.В., ТОЧЕВА К.Т. Организация однородных решающих полей и оценка их эффективности. - В кн.: Вопросы кибернетики. Вычислительные машины и системы с перестраиваемой структурой. М., 1978, вып. 43, с. 83-90.
3. ПРАНГИШВИЛИ И.В., ИГНАТУШЕНКО В.В. и др. Перестраиваемая управляющая вычислительная система на основе однородных структур. - В кн.: VI Всемирный конгресс IFAC, Бостон, США, 1975, ч. I/A. (Препринт, гер. 13.3).
4. СОКОЛ Ю.М., КАЗАНЦЕВ П.Н. Некоторые вопросы эффективности процессора с разделением вычислительных ресурсов. - Управляющие системы и машины, 1973, № 2, с. 81-88.
5. Основы теории вычислительных систем /Под ред. Майорова С.А. - М.: Высшая школа, 1978. - 408 с.
6. КОФМАН А., КРЮН Р. Массовое обслуживание. Теория и приложения. - М.: Мир, 1965. - 302 с.
7. ЗАВОЛОТНЫЙ А.А., ИГНАТУШЕНКО В.В., НЕДЗЕЛЬСКИЙ Д.А. Исследования эффективности однородных мультипроцессорных систем с общими вычислительными ресурсами. - М., 1980. - 21 с. - Рукопись представлена Ин-том проблем управления АН СССР. Деп. в ВИНТИ 23 февр. 1981, № 812-81.

8. ЗАВОЛОТНЫЙ А.А. Организация и исследование структур однородных асинхронных решающих сетей в многопроцессорной вычислительной системе с перестраиваемой структурой: Автореф. дис. канд. техн. наук. - М., 1981. - 20 с.

9. ЗАВОЛОТНЫЙ А.А. Анализ организации общих вычислительных ресурсов в однородных мультипроцессорных вычислительных системах. - Автоматика и телемеханика, 1981, № 2, с.164-174.

Поступила в ред.-изд.отд.
21 июня 1982 года