

## ТЕОРЕТИЧЕСКИЕ ПРОБЛЕМЫ, СВЯЗАННЫЕ С ОБЪЕДИНЕНИЕМ ТИПОВЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН В ЕДИНУЮ СИСТЕМУ

Д.А. Поспелов

(Москва)

Даётся краткая характеристика основных проблем, которые возникают при создании вычислительных систем из неспециализированных машин, и указываются пути решения этих проблем.

При объединении вычислительных машин в единую систему, в которой сохраняется автономность отдельных машин, возникают две основные задачи: выбор оптимальной структуры вычислительной системы на основе анализа решаемой задачи (или класса решаемых задач) и выбор структуры записи алгоритма решения задачи при заданной структуре вычислительной системы.

Под структурой вычислительной системы мы понимаем число машин этой системы и их характеристики (быстродействие, скорость обмена информацией между различными устройствами и организация памяти), а под структурой алгоритма – нижеописанное представление его в виде специального графа.

Рассмотрим граф с ориентированными рёбрами двух типов, которые будем называть соответственно функциональными и управляющими рёбрами. Вершины графа оценим некоторыми весами. Кроме весов, каждую вершину графа сопоставим с функциональным оператором алгоритма, реализуемого на машине. Будем считать, что функциональное ребро, идущее от оператора  $F_1$  к  $F_2$ , указывает, что результаты, являющиеся выходными для  $F_1$ , играют для  $F_2$  роль аргументов. Аналогично, управляющее ребро, идущее от  $F_1$  к  $F_2$ , указывает, что после  $F_1$  может выполняться  $F_2$ . Будём рассматривать только такие графы, которые не содержат петель. При наличии циклических участков в алгоритме будем либо весь циклический участок считать одним функциональным оператором, либо итерировать граф, переходя при не-

обходимости к графу с бесконечным числом вершин и рёбер (но такому, что в нём можно выделить конечное число подграфов, для каждого из которых известна вероятность встречи их в бесконечном графе). Графы такого типа можно считать на каждом шаге итерации деревьями. Число вершин дерева является одной из важных характеристик топологии графа. Вес вершины представляет собой вектор, координатами которого служат времена выполнения данного оператора на машинах вычислительной системы. При однородной системе каждый вес является просто числом. В ряде случаев значения координат веса представляют собой некоторые вероятностные значения. Тогда, если это возможно, сообщается некоторая дополнительная информация об этих временах (например, указывается величина дисперсии). Другие характеристики функционально-временного графа можно найти в работе [13].

При решении первой из двух сформулированных выше задач структура алгоритма считается заданной и основная проблема состоит в следующем: для данной структуры алгоритма выбрать параметры вычислительной системы (её структуру), чтобы время решения задачи на системе при данной структуре алгоритма было минимальным или не превосходило некоторого заданного значения. Решение этой задачи в общем случае пока еще не получено. Некоторые обнадеживающие результаты содержатся в работе Д.Н.Кикнадзе [8], в которой указанная проблема решается для случая однородной вычислительной системы. В работе Ю.Г.Косарева [9] вводится понятие кортежа, позволяющее чётко сформулировать задачу выбора структуры системы по структуре реализуемого алгоритма. Общим для обеих работ является использование при постановке и решении задачи выбора структуры системы, описание её в виде сетевого графика специальной природы и решения вариационной задачи на таком графике.

Задача выбора оптимальной структуры алгоритма для его реализации на вычислительной системе с заданной структурой более хорошо исследована. Здесь можно выделить следующие частные задачи, тесно связанные между собой: 1) разбиение на функциональные операторы, 2) эквивалентное преобразование структуры алгоритма, 3) оптимальное распределение операторов по машинам вычислительной системы и 4) перекодирование информации в процессе обмена между машинами системы. Эти четыре задачи, хотя и могут решаться независимо, тем не менее представляют собой четыре этапа решения одной общей задачи. Поэтому оптимальность результата целиком зависит от того, будут ли опти-

мальными результаты решения всех четырех указанных задач. К сожалению, в настоящее время нет даже подходов к решению всей задачи в целом. Поэтому рассмотрим решение указанных задач, взятых вне связи друг с другом.

Задача разбиения на функциональные операторы зависит от того, какой язык выбирается в качестве входного языка вычислительной системы. Если в качестве такого языка используется язык типа АЛГОЛ, то разбиение выгодно производить на языке-посреднике, т.е. на языке, который является языком псевдокомандного типа и в котором программа имеет близкую к машинной структуру. Один из методов разбиения [10] предусматривает сначала разбиение на функциональные операторы, совпадающие с линейными участками псевдокомандной программы и небольшими циклами. Когда объем выделенных операторов больше объема оперативной памяти вычислительной машины системы происходит покомандное разбиение (разбиение на уровне функциональных связей между командами). В работе [19] дается другой подход к задаче разбиения, исходящий из условия полного заполнения памяти машин вычислительной системы. Важно подчеркнуть, что определяющее значение для успешной организации разбиения алгоритма имеет выбор входного языка системы. По мнению автора, ввод задачи в вычислительную систему, которая должна автоматически решать задачу о выборе оптимального представления алгоритма решения и его разбиения на отдельные функциональные операторы, следует производить на языке, который не является алгоритмическим, а представляет собой язык типа языка диспозиций, предложенного в работе [2]. Запись на таком языке дает возможность находить по ней наиболее оптимальную для данной вычислительной системы алгоритмическую реализацию заданной диспозиции. Заметим, что входной язык вычислительной системы необязательно должен совпадать с её внутренним языком, на котором происходит обмен информацией между машинами системы. Применение этого последнего языка обусловливается удобством перекодирования информации при передаче её внутри системы и соображениями сохранения точности вычислений.

Вторая задача заключается в эквивалентном преобразовании полученной структуры алгоритма. Задача эта практически еще не решена, хотя известны некоторые возможные подходы к её решению [9].

Третья задача — оптимальное распределение функциональных операторов по машинам вычислительной системы — находится в более хорошем состоянии. Для случая локального распределения, да-

ющего оптимум на каждом ярусе функционально-временного графа, она полностью решена в работе [11], где поставлена как минимаксная задача линейного целочисленного программирования и где для её решения построен специальный метод, дающий решение за существенно меньшее число шагов по сравнению с обычными универсальными методами. Для задачи глобального оптимального распределения известны два подхода. Один из них даёт распределение в среднем по нагрузке машин системы. Он предложен В.Г.Хорошевским и описан им в работе [17]. При другом подходе поставленная задача сводится к распределению ресурсов на сети типа PERT. Для чего строятся различные приближённые методы [4].

Задача перекодирования, как указывалось выше, тесно связана с проблемой внутреннего языка вычислительной системы. Один из возможных вариантов такого языка кратко описан в работе [18]. Однако проблема построения удобного внутреннего языка в настоящее время не только не решена, но даже чётко не поставлена. В связи с этим в теоретическом плане не решена и проблема перекодирования информации внутри системы. Имеющиеся решения носят характер эмпирических рекомендаций.

Несколько особняком стоят проблемы организации управления ВС, обменной памяти и коммутирующих устройств, а также проблемы оценки производительности и надёжности вычислительных систем.

Можно представить себе различные способы управления вычислительными системами. Однако все такие способы можно разбить на два типа управления: централизованное и децентрализованное (может быть и смешанное управление: централизованно-децентрализованное). В настоящее время достаточно хорошо исследован только вопрос о централизованном управлении, которое осуществляется с помощью либо специализированной вычислительной машины, либо управляющей программы, реализуемой на одной из машин системы [12].

Однако, с точки зрения надёжности функционирования большой системы, централизованный принцип управления является далеко не лучшим. При исследовании управления большими сетями связи [3] было обнаружено, что децентрализованное управление по целому ряду параметров оказывается эффективнее систем централизованной коммутации, рассмотренных ранее. Изучение структуры и функционирования вычислительных систем, в которых отсутствует единая центральная управляющая машина, приводит к

понятию вычислительной среды однородного или неоднородного типа. Основная проблематика таких сред рассмотрена в работах Э.В. Евреинова [6, 7] .

Остановимся более подробно на оценке производительности вычислительной системы. Нам кажется целесообразным оценивать производительность вычислительных систем следующим образом. Среди функционально-временных графов выбирается некоторая совокупность представителей, характеризующих тот или иной структурный тип алгоритма, и для каждого эталонного представителя берётся отношение времени его реализации на вычислительной системе рассматриваемого типа к времени его реализации на эталонной вычислительной машине (или на наиболее быстродействующей из машин, образующих исследуемую систему). Если число выделенных эталонных структур алгоритмов равно 1, то в результате мы получим 1-мерный вектор, характеризующий производительность вычислительной системы. Если как-то определены веса эталонных задач в общем потоке задач, решаемых на системе, то за оценку производительности системы можно взять сумму попарных произведений весов эталонных задач на компоненты найденного вектора. Такой подход к оценке производительности использован в работах [14, 15] . В работе [5] использована оценка производительности системы на основе расчета загрузки машин системы. Такая оценка, по существу, является частным случаем общей оценки.

Наконец, большую самостоятельную проблему составляет организация сети связи в вычислительной системе. Из теоретических исследований, посвящённых этой проблеме, можно указать на работу Ю.Г. Решетняка [16] , в которой содержится, по существу, отрицательный ответ на вопрос, можно ли аналитически найти оптимальный путь информации в системе. Поэтому основные результаты, полученные в этом направлении носят модельный характер и наиболее полно отражены в работах [1] и [3] .

Как показывает этот краткий обзор проблематики, связанной с построением вычислительных систем, в настоящее время число нерешённых задач достаточно велико, а для решённых задач возможно нахождение более изящных решений.

## Л и т е р а т у р а

1. И.А. Башмаков. О некоторых задачах управления сетью вычислительных центров. Данный сборник, стр. 63-71.
2. В.Б. Борщёв, Ю.А. Шрейдер. Алгоритмы, языки программирования и диспозиции. - Кибернетика, 1965, № 4.
3. А.В. Бутрименко, В.Г. Лазарев. Система поиска оптимальных путей передачи сообщений. - Проблемы передачи информации. 1965, вып. I.
4. Е.Г. Гольштейн, Д.Б. Юдин. Новые направления в линейном программировании. М, "Советское радио", 1966.
5. Э.В. Евреинов, Ю.Г. Косарев. О вычислительных системах высокой производительности. - Известия АН СССР, Техническая кибернетика, 1963, № 4, стр. 3-25.
6. Э.В. Евреинов. Теоретические основы построения универсальных вычислительных сред. - Вычислительные системы, Новосибирск, "Наука", 1965, вып. 16, стр. 3-72.
7. Э.В. Евреинов. Универсальные вычислительные системы с частично переменной структурой. - Вычислительные системы, Новосибирск, "Наука", 1965, вып. 17, стр. 3-60.
8. Д.Н. Кикнадзе. Метод анализа структуры параллельных алгоритмов, реализуемых на однородной вычислительной системе. Данный сборник, стр. 89-96.
9. Ю.Г. Косарев. О методике решения задач на универсальных вычислительных системах. - Вычислительные системы, Новосибирск, "Наука", 1965, вып. 17, стр. 61-99.
10. Я. Малушински, Д.А. Поспелов. Об одном методе выделения в схеме программы независимых кусков. - Труды МЭИ, 1964, вып. 53.
11. К. Мацячик, Д.А. Поспелов. Оптимальное поярусное распределение программы на параллельно работающие вычислительные устройства. - Сети передачи информации и их автоматизация, 1965.
12. Д.А. Поспелов. О некоторых математических проблемах, возникающих при совместной работе нескольких вычислительных машин. - Труды МЭИ, 1964, вып. 53.
13. Д.А. Поспелов. О некоторых соотношениях структуры алгоритмов решения и структуры вычислительной машины. *Výzkumný ústav matematických strojů. Praha, 1964.*
14. Д.А. Поспелов. Об одном способе построения вычислительной системы. - Семинар по теории автоматов, Киев, 1964.

15. Д.А. Пospelov. Оценка производительности вычислительной системы, состоящей из обычных вычислительных машин. - Известия АН СССР, Техническая кибернетика (в печати).
16. Ю.Г. Решетняк. О задаче соединения элементов вычислительной системы. - Вычислительные системы, Новосибирск, "Наука", 1962, вып. 3, стр. 17-30.
17. В.Г. Хорошевский. Об алгоритмах распределения задач по ЭЦВМ. - Труды Сибирского ФТИ, Томск, Изд. Томского университета, 1965, вып. 47, стр. 29-34.
18. E.Nuding. A Language designed for communication between computers of different types. Symbolic Languages Data Process, 1962.
19. Y.Vlček, E.Outrata. Elements of segment programming in a system of automatic programming. Výzkumný ústav matematických strojů, Praha, 1964.