

УДК 681.3.01.

## КОМПЛЕКСНЫЙ ПОДХОД К РЕАЛИЗАЦИИ МАССОВЫХ ВЫЧИСЛЕНИЙ

Л.В.Канторович, Л.Т.Петрова, Я.И.Фет

Вычислительные машины заняли большое и важное место в науке и технике, в процессах управления и обработки информации, при обработке результатов экспериментов, непосредственно в работе технологических систем. Многие сделано в развитии методов программирования, в совокупности вопросов, которые называют сейчас математическим обеспечением: разработан целый ряд универсальных и проблемно-ориентированных языков программирования, созданы различные операционные системы и т.д. Имеются значительные успехи в создании вычислительной техники. Сейчас у нас вступают в строй машины третьего поколения.

Однако развитие этих разделов далеко не закончено. Все время возникают новые грандиозные по масштабам задачи, открываются новые области применения ЭВМ, разрабатываются новые подходы в программировании, например диалоговые формы общения человека с машиной [2], теория параллельного программирования [3] и т.п. Появляются новые возможности развития машинной техники: системы машин, машины со сложной архитектурой, новые физические принципы хранения и обработки информации (голография и т.д.). В целом весь комплекс вопросов, связанный с развитием и использованием ЭВМ, необозрим. В настоящей статье имеется в виду рассмотреть лишь один из аспектов этого комплекса, касающийся реализации массовых вычислений.

Прежде всего хотелось бы отметить целесообразность и необходимость комплексного, системного подхода к проблеме. В этом состоит один из реальных резервов повышения эффективности ее

решения. Комплексный, взаимосвязанный, системный подход может достигаться как за счет хороших контактов людей, работающих на разных уровнях, так и за счет наличия некоторого числа специалистов гибридной квалификации, которые одновременно творчески владеют несколькими уровнями этого многоэтажного здания. Как показывает опыт многих других областей, внимательное изучение граничных проблем, мест стыковки и стирание этих границ является часто сильным стимулом для развития всех соприкасающихся областей.

Специально рассмотрим вопрос крупноблочного описания и машинной реализации массовых операций [1]. В больших вычислительных и информационных задачах многие операции, как правило — большинство, носят не индивидуальный характер, а относятся к определенным образом сформированным массивам данных и сами имеют однородный массовый характер.

Если сослаться на аналогию, то можно привести пример управления в армии. Распоряжения и команды там относятся не к отдельным военнослужащим, а к целым подразделениям. Можно себе представить, что если бы простейшие команды, например "поворот налево", выдавались в таком расписанном виде: "Иванов — налево, Сидоров — налево и т.д.", то это очень усложнило бы управление и сделало его весьма громоздким.

Однако если вздуматься в организацию процесса на современных вычислительных машинах, то она, как правило, именно так и осуществляется. В традиционной машине все операции сводятся в конечном счете к индивидуальным операциям над числами (машинными словами). Это положение отражается и закрепляется соответствующими конструкциями в языках программирования. Например, именно такие переменные рассматриваются в языках АЛГОЛ-60, ФОРТРАН и т.д., именно так реализуются в них циклы. В свою очередь, существующие языковые конструкции и способы описания вычислений определяют и формируют требования, предъявляемые к новым машинам. И вот уже три поколения машин имеют все ту же принципиальную структуру. Между тем остаются неиспользованными принципиальные возможности аппаратного осуществления некоторых массовых операций во вновь конструируемых машинах, особенно при использовании новых физических средств. Эти возможности, даже технически существующие, как правило, не используются, если не считать отдельных работ по микропрограммированию и аппаратной интерпретации.

Большие резервы повышения эффективности вычислительной техники заключаются в создании специальных технических средств для выполнения массовых крупноблочных операций.

Нужно сказать, что у нас накоплен некоторый опыт в этом отношении. В начале 60-х годов в Институте математики СО АН СССР был разработан специализированный процессор для массовых вычислений [4]. Такой процессор используется в комплексе с универсальной ЭВМ, на него передаются групповые операции. Необходимая высокая скорость вычислений достигается за счет применения специального "роторного" арифметического устройства [5] и конвейерного принципа обработки информации. Этот принцип состоит в том, что аргументы подаются из оперативного запоминающего устройства машины непрерывным потоком с той скоростью, с какой память может их выдавать. Аргументы поступают на конвейер, состоящий из цепочки сверхоперативных регистров. На каждой позиции конвейера выполняется определенная операция из той последовательности операций, которую нужно выполнить над каждым набором аргументов данного массива. Поток результатов с выхода конвейера может сниматься с той же предельной скоростью, с которой подаются аргументы. Испытания экспериментального образца конвейерного процессора, проведенные в Вычислительном центре СО АН СССР, показали, что такой процессор обеспечивает скорость выполнения групповых операций на порядок более высокую, чем универсальная ЭВМ (при тех же логических элементах). Следует отметить, что некоторые архитектурные решения, аналогичные положенным в основу рассмотренной выше разработки, были впоследствии использованы в мощных вычислительных машинах фирм *IBM* и *СДС*.

Другое техническое направление, представляющее особый интерес для решения массовых информационно-логических задач (упорядочение массивов, поиск информации и т.д.), связано с матричными ассоциативными процессорами. Применение специализированных ассоциативных матриц (например, описанных в [6]), позволяет увеличить скорость решения информационно-логических задач на два порядка по сравнению с универсальными машинами. Ассоциативные принципы массовой крупноблочной обработки информации используются в проектируемых новых машинах как в СССР, так и за рубежом.

Наличие крупных операций, а не отдельных команд, знание их эффективности и времени реализации позволяют заранее оценить время, необходимое для вычислений, оценить алгоритмы, выделить крупные независимые части вычислительного плана, преобразовать вычислительный план, оптимизировать его.

Естественная массовость, крупноблочность вычислительных процессов нашла некоторое отражение в системах программирования и языках. Примером могут служить описания массивов в таких языках, как АЛГОЛ-60, ФОРТРАН и т.п. (целой совокупности данных присваивается одно имя). Начиная с 1954 года, крупноблочность особенно пропагандировалась в работах ленинградских математиков [1]. Под известным влиянием этого направления массовые операции нашли некоторое отражение во входном языке системы АЛЬФА-АЛГОЛ [7]. Такие подходы впоследствии получили развитие в некоторых других системах программирования (например, в языках *APL*, *PL-I*, АЛГОЛ-68). Однако систематически они развиты еще не достаточно.

В настоящее время представляется необходимой работа по анализу структуры массовой информации в различных задачах, по изучению и типизации различных способов обработки этой информации, по исследованию характера и особенностей массовых операций. Эти особенности вычислений с учетом технических средств их реализации должны найти отражение в языковых конструкциях. Представляется, что именно эти структурные особенности массовой информации и массовых операций могут стать определяющими и в характеристике задачи, и при выборе описывающего ее языка.

Здесь имеется известная аналогия с развитием других областей математики. Так, возникший универсальный аппарат дифференциального исчисления вызвал появление и исследование многих разнообразных уравнений и систем, описывающих те или иные конкретные процессы. При этом отдельная система исследовалась сначала, хотя и на базе универсального аппарата, но индивидуально, именно для данного процесса. Затем внутреннее развитие аппарата выявило характерные черты и специальные средства исследования целых разделов: сформировались теории уравнений эллиптических, гиперболических и т.д. В дальнейшем развитие функционального анализа выявило новые глубинные связи и особенности, привело к новым обобщениям.

Представляется, что нечто подобное имеет место и в программировании. Вычислительные задачи в общем имеют неуниверсальный

характер, они обладают некоторыми специальными чертами и особенностями, которые, естественно, нужно использовать. Без учета этих особенностей весьма проблематичной становится возможность лаконичного описания задач и эффективного использования ЭВМ. Это обстоятельство нашло отражение в появлении многих проблемно-ориентированных языков. В результате этого накоплен значительный опыт описания и представления различных видов информации, опыт описания и реализации различных вычислительных процессов. Как уже говорилось, в настоящее время представляется оправданным и целесообразным анализ вычислительных процессов относительно структуры массовой информации и массовых операций в них, переход от проблемной - к структурной ориентированности задач, структурной ориентированности языков.

Элементы подобного структурного подхода имеются в разрозненных работах по программированию, например в [3]. Примером структурно-ориентированных языковых конструкций могут служить и укрупненные конструкции языков [8] и [9].

Исчисление данных в [8] вводится конструкцией  $\Delta$ -список. Элементарными объектами этого исчисления являются списки языка ЭПСИЛОН - одномерные массивы, заданные фактически или в виде специальной процедуры, вычисляющей элемент списка по его номеру. Определяются двуместные операции над списками. Если рассматривать списки как конечные упорядоченные множества элементов (кодов), то введенные операции являются теоретико-множественными операциями объединения ( $\cup$ ), прямого произведения ( $\times$ ) и поэлементной конкатенации ( $\uparrow$ ).  $\Delta$ -список задается формулой в таком исчислении данных. Формула эта записывается в виде схемы, так что описание  $\Delta$ -списка имеет, например, такой вид:

$$\Delta\text{-список} \quad a = b \times c, \quad b = \text{процедура}, \quad c = d \uparrow e, \\ d = f \cup h;$$

здесь  $e$ ,  $f$ ,  $h$  - исходные списки, фактически заданные в машине, а  $b$  - процедура, описанная в тексте обычным образом и выдающая при обращении к ней элемент списка по его номеру. Формулы такого исчисления позволяют алгоритмически конструировать весьма сложные структуры данных. Возможны разные способы использования этих структур, в частности поэлементное их использование. При этом не требуется одновременно хранить всю структуру целиком. В другой конструкции языка, укрупнен-

ном операторе процедуры, отражен конвейерный способ обработки информации. Следует отметить, что различные конкретные формы алгоритмического хранения информации нередко рассматриваются в программировании. В частности, они встречались в программах решения больших математико-экономических задач, например в [10], они весьма характерны и для аналитических машинных вычислений различного рода.

В настоящее время для обоснованного дальнейшего развития математического обеспечения и вычислительной техники, по нашему мнению, целесообразно поставить более систематически анализ математических и технических требований к процессам вычислений. В этом отношении представляют интерес следующие направления работ:

1. Характеристика наиболее важных областей применения, которые порождают сложные и громоздкие задачи.
2. Классификация и типизация этих задач, методов и алгоритмов их решения.
3. Анализ методов и алгоритмов с точки зрения тех операций, которые являются наиболее массовыми и предъявляют наибольшие требования к машинам.
4. Изучение технических возможностей различных аппаратных средств массовой обработки информации.
5. Согласованная формулировка требований к методам, языкам и техническим средствам.

## Л и т е р а т у р а

1. КАНТОРОВИЧ Л.В. Перспективы работы в области автоматизации программирования на базе крупноблочной системы. - Тр. Матем. ин-та АН СССР, 1968, № 6, с. 5-15.
2. ГЛУШКОВ В.М. Диалог с вычислительной машиной: современные возможности и перспективы. - "УСМ", 1974, № 1, с.3-8.
3. КОТОВ В.Е. Теория параллельного программирования. Прикладные аспекты. - "Кибернетика", 1974, № 1, с. 1-17.
4. КАНТОРОВИЧ Л.В., ФЕТ Я.И. Вычислительная система, состоящая из универсальной цифровой вычислительной машины и малой цифровой вычислительной машины. Бюл. изобр., 1965, № 13.
5. КАНТОРОВИЧ Л.В., ФЕТ Я.И., ИЛОВАЙСКИЙ И.В. Арифметическое устройство цифровой вычислительной машины. Бюл. изобр., 1968, № 4.

6. ФЕТ Я.И. Обработка информации в простых вычислительных средах. - В кн.: Информационные методы в системах управления измерения и контроля. (Доклады II Всес.сем.), Владивосток, 1972, том II.
7. ЕРШОВ А.П. (ред.) АЛЬФА-система автоматизации программирования. Новосибирск, "Наука", Сиб. отд., 1967.
8. ПЕТРОВА Л.Т. Схемно-списочная символика в языке ДЕЛЬТА-ЭПСИЛОН.- В кн.: Оптимизация, 1972, № 6 (23), Новосибирск, с. 65-79.
9. ПЕТРОВА Л.Т., ПИВКИНА Н.Н. Схемно-блочный язык программирования. В кн.: Оптимизация, Новосибирск, 1972, № 7 (24), с. 71-103.
10. МАКАРОВ В.Л., МАРШАК В.Д., ФЕДЕЛОВ В.Ф. Алгоритм формирования оптимальных динамических моделей "затраты-выпуск". В кн.: Алгоритмы и программы реализации народно-хозяйственных моделей. Новосибирск, 1971, с. 116-133.

Поступила в ред.-изд. отд.  
27. III. 1974 г.