ՀԱՅԿԱԿԱՆ ՍՍՌ ԳԻՏՈՒԹՅՈՒՆՆԵՐԻ ԱԿԱԴԵՄԻԱՅԻ ՏԵՂԵԿԱԳԻՐ
ИЗВЕСТИЯ АКАДЕМИИ НАУК АРМЯНСКОЙ ССР

Ֆիզիկա-մաթեմատ. գիտություններ X, № 2, 1957 Физико-математические науки

МАТЕМАТИКА

Л. В. Канторович

О проведении численных и аналитических вычислений на машинах с программным управлением*

Данная статья посвящена информации о работах в области программирования проводимых в Ленинградском отделении Математического института (ЛОМИ) АН СССР, именно следующим двум вопросам:

1) Описанию некоторой системы программирования или, правильнее сказать, системы подготовки вычислений на электронных счетных машинах.

Сущность этой системы, которая может быть названа крупноблочной, состоит в том, что в машину вводится в закодированном виде вычислительный план работы, записанный в укрупненных элементах и операциях, который расшифровывается раз навсегда составленной универсальной программой-роботом — „прорабом". Последняя ведает также размещением данных в оперативной и внешней памяти, переадресацией и пр. Таким образом, полная программа данной работы фактически не составляется: ее получение и не ставится задачей. Целью является нахождение результатов вычисления. Если угодно, хотя это звучит парадоксально, это система беспрограммного программирования.

2) Вопросу о систематическом проведении с помощью универсальных счетных машин аналитических выкладок. Как известно, в ряде математических и прикладных исследований много труда занимают часто не численные, а аналитические преобразования: дифференцирование, подстановки и т. п., иногда требующие десятков и сотен страниц выкладок. Во многих случаях эта работа носит стандартный характер. Поэтому естественна мысль об использовании машин при ее выполнении.

Подход к обоим вопросам связан с некоторыми общими идеями, относящимися к использованию машин с программным управлением в математике. С них мы и начнем, не излагая их в полном объеме и общности.

Проведенные исследования являются результатом работы группы научных сотрудников ЛОМИ, протекавшей под общим руководством автора,— научных сотрудников Л. Т. Петровой, В. А. Булавского,

* Доклад, читанный на сессии АН Армянской ССР 28 XI 1956.

М. А. Яковлевой, И. А. Платуновой; в разработке отдельных вопросов принимали участие аспиранты О. К. Даугавет и К. В. Амбарцумян.

I. Введение математического задания в машину и его обработка

В настоящее время весьма актуальным является вопрос о проведении самых различных математических работ на электронных цифровых машинах, не только численных вычислений, но и ряда других видов их: аналитических выкладок и преобразований, логического анализа, анализа и синтеза релейных схем, обработки вычислительных планов, программ, а также некоторых нематематических (анализ текстов и пр.). Это, естественно, выдвигает задачу разработки практически удобных и универсальных методов введения такого рода заданий в машины и создания единых машинных методов выполнения этих заданий.

Для этой цели существенна разработка следующих трех вопросов:

1) Создание единообразной символики. Требования, которые естественно предъявить к такой символике, это: а) универсальность — возможность применения одной и той же символики для описания различных по содержанию математических заданий; б) точность и недвусмысленность символики; в) компактность (малая емкость записи относительно к содержащейся в ней информации), г) удобство обращения с ней на машине и достаточная наглядность для обычного восприятия.

2) Разработка единых методов обработки математического задания, безотносительно от его конкретного содержания. Выявление общих видов операций, встречающихся при такой обработке, выяснение их свойств и возможностей, а также подготовка программ, практически осуществляющих их на машинах. Иначе говоря, развитие своеобразной алгебры таких операций, так сказать, машинной алгебры.

3) Разработка удобных форм описания состава и расположения материала, вводимого и сохраняемого в машине в процессе его обработки. Эта задача, которая была мало актуальной при осуществлении математического задания человеком, с записью данных на бумаге или сохранении их в человеческой памяти, становится весьма важной при автоматическом проведении ее на машине.

С и м в о л и к а. Вопрос об используемой символике является очень существенным при передаче машине математического задания. Очевидно, процесс передачи будет легче, если математический язык человека и язык, воспринимаемый машиной, будут приближены друг к другу. С этой точки зрения очень вероятно, что применяемый иногда ввод в машину в закодированном виде обычных формул может оказаться не наилучшим путем.

Математическая символика изменялась и развивалась с развитием новых областей математики. Естественно, что применение машин также должно оказать влияние на используемую символику, т. е. в символике нужно учесть „интересы“ и особенности машины. С другой стороны, можно и машину „научить понимать“ не только простейшие операции, но и более сложные, например матричные и др.

Схема. Первое, что нам представляется удобным, это использовать вместо аналитических и логических формул со скобками запись связей и зависимостей математических объектов в виде расчлененной схемы типа

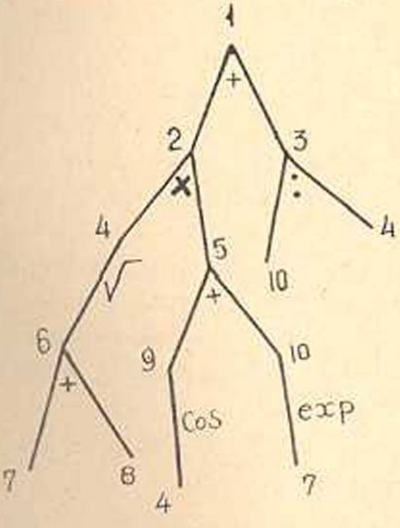
$$k_1 = (F_1, k_1', k_1''),$$
$$k_2 = (F_2, k_2', k_2'').$$

Здесь F и k — условные номера, обозначающие те или иные объекты из данного поля объектов. Каждую строку можно прочитать так: объект k_i получается с помощью оператора F_i, из объектов k_i', k_i''.

Некоторые из объектов даны, другие должны быть найдены на основании соотношений схемы.

Следует сказать, что выделение операторов является условным, так как номер, фигурирующий в качестве оператора в одной строке, в другой строке схемы может выступать в качестве аргумента или результата.

Будем называть явной схему, в которой результативные номера (т. е. номера, стоящие слева) не повторяются и в которой нельзя найти замкнутую цепочку опирающихся друг на друга номеров. Явную схему всегда можно изобразить в виде некоторого дерева, определяющего частичное упорядочение результатов. Поясним принцип записи в виде схемы на примере функции



$$\underbrace{\underbrace{y = \overbrace{\sqrt{x+a}}^{4}\,(\cos\overbrace{\underbrace{\sqrt{x+a}}_{9} + \underbrace{e^x}_{10}}^{5})}_{2} + \underbrace{\frac{e^x}{\sqrt{x+a}}}_{3}}_{1}$$

1 = (+, 2, 3)   5 = (+, 9, 10)
2 = (×, 4, 5)   9 = (cos, 4)
4 = (√, 6)   10 = (exp, 7)
6 = (+, 7, 8)   3 = (: , 10, 4)
7 = x
8 = a

Дерево подчиненности дано на фигуре.

Операции нужно было бы также обозначить цифрами. Следует обратить внимание на компактность записи в виде схемы, так как выражения, встречающиеся повторно (например, 4), не даются в развернутом виде, в отличие от обычной формульной записи.

Схему можно рассматривать как функциональное преобразование группы опорных элементов, т. е. таких, которые не встречаются слева (ни на что не опираются, например x и a), в некоторую группу результативных (например, y).

Из данной схемы естественно выделяются подсхемы — именно той же записью задается зависимость некоторых групп промежуточных аргументов от других.

Наоборот, можно рассматривать операцию объединения нескольких схем при соответствующем согласовании нумерации их элементов, в частности последовательное наложение схем, степень схемы, накопительную операцию над объектами, построенную по данной схеме, и т. п.

Некоторые упрощения и преобразования схем. Приведем несколько конкретных операций обработки абстрактных схем, безотносительно от их содержательного значения, которые встречаются при различном использовании схем.

1) Могут быть построены специальные программы упорядочения схемы. Наиболее удобно упорядочение по логической зависимости, вводящее минимальное число номеров, необходимых для нахождения данного. Например, для схемы, данной на фигуре, при логическом упорядочении, последовательно определяются результаты: 6, 4, 9, 10, 5, 2, 3, 1.

2) Операция упрощения (отождествления). В случае, если соотношения, определяющие несколько номеров, одинаковы (т. е. одинаковы правые части соответствующих строк), то только одна из всех этих строк должна быть сохранена; остальные могут быть выброшены, при этом во всех прочих строках исключенные номера должны быть заменены сохраненными.

3) Пусть имеются некоторые тождества, связывающие переменные, т. е. произвольные или, в известной степени произвольные, номера объектов. Подобное тождество можеть быть записано в виде схемного тождества — некоторого правила преобразования — замены определенного вида подсхемы на другую. Может быть составлена определенная программа для преобразования схем в соответствии с этими правилами.

В качестве примера можно указать на правила преобразования, связанные со свойствами тех или иных операторов (коммутативность, ассоциативность, дистрибутивность и др.). Дистрибутивность операторов $+$, $\times$ (распределительный закон), например, записывается в виде схемного равенства $S_1 = S_2$, где

$$S_1 \begin{cases} 0 = (\times,\ 1, 4) \\ 4 = (+,\ 2, 3) \end{cases} \qquad\qquad S_2 \begin{cases} 0 = (+,\ 4, 5) \\ 4 = (\times,\ 1, 2) \\ 5 = (\times,\ 1, 3) \end{cases}$$

Здесь 0, 1, 2, 3 — основные аргументы (опорные и результативные), 4, 5 — вспомогательные, промежуточные.

С использованием подобных программ на основании данных законов действий (схемных тождеств) может быть произведено дальнейшее упрощение схем (обнаружение дополнительных тождеств между номерами), подъем или опускание некоторого оператора, исключение некоторого оператора, если дано правило его перестановки с другими и его значение для опорных элементов, максимальный подъем тех или иных аргументов (например, переменных) вверх, сокращение записи схемы за счет введения обозначений для многократно встречающихся схем и т. п.

Представляется полезным еще в ряде случаев объеденение группы номеров в систему (список). В частности, такое объединение целесообразно, если ряд номеров связан последовательным применением некоторого ассоциативного оператора, что соответствует введению свертывающих операторов типа Σa_i. Кроме того, такое объединение позволяет с удобством производить упрощение внутри группы.

Наконец, в ряде случаев, если для некоторых выражений известны правила их вычисления, схема может быть упрощена за счет выполнения этих вычислений.

Обратим внимание на то, что применение всех такого рода программ преобразования при схемной записи чрезвычайно удобно благодаря тому, что добавленные и заменяющие строки могут записываться в произвольном порядке, а также вследствие того, что возможно локальное проведение каждого преобразования, т. е. рассмотрение только ближайших логических связей данных номеров без привлечения полных формул, которыми они выражаются.

Описание расположения материала. Для этой цели удобно ввести величины и справки. Представляется удобным при схемной записи оперировать не с отдельными числами, а с более крупными объектами — *величинами*. В качестве таких числовых величин могут быть взяты вектор, матрица, трехмерная матрица, поле и т. д. Для величины наряду с ее числовым содержанием вводится *справка*, представляющая собой описание числового материала и его расположения. Рассмотрим пример такой справки для матрицы. Пусть элементы матрицы

$$K = \begin{pmatrix} a_{11} \, a_{12} \cdots \, a_{1n} \\ \cdots\cdots\cdots\cdots \\ a_{m1} \, a_{m2} \cdots a_{mn} \end{pmatrix}$$

расположены соответственно в ячейках

$$P, \; P + h_1, \; P + 2h_1, \cdots$$
$$P + h_2, \; P + h_2 + h_1, \; P + h_2 + 2h_1, \cdots$$
$$\cdots\cdots\cdots\cdots\cdots$$

Тогда справка о величине K имеет вид:

$$\begin{pmatrix} P, \; n, \; h_1 \\ \quad m, \; h_2 \end{pmatrix}$$

P — номер ячейки, в которой помещается первый элемент величины,
n — число столбцов,
m — число строк,
h_1 — расстояние между двумя соседними элементами любой строки,
h_2 — расстояние между первыми (и вообще i-тыми элементами) двух соседних строк.

Отметим, что справка о транспонированной матрице К* легко получается из данной и имеет вид:

$$\begin{pmatrix} P, m, h_2 \\ n, h_1 \end{pmatrix}.$$

Отделение справки от числового материала тем более оправдано, что несколько величин могут опираться на один и тот же числовой материал.

II. Схемное или крупноблочное программирование

Развитие физических и технических дисциплин требует численного решения все большего числа различных математических задач и по возможности в короткие сроки.

Реализация огромных возможностей, которые несет за собой использование быстродействующих машин для решения этих задач, в известной степени затруднена при применении обычной системы программирования, так как процесс подготовки программы и отладки ее для реальной задачи оказывается довольно сложным и длительным.

В связи с этим и у нас и за границей были предприняты работы по упрощению и автоматизации программирования. Известен метод стандартных подпрограмм.

Другим путем является использование специальной программирующей программы (П.П.). Этот метод, в особенности, разработан в Математическом институте АН СССР Любимским и Камыниным под руководством А. А. Ляпунова и М. Р. Шура-Бура.

Мы опишем тот путь решения этого вопроса, по которому пошли в ЛОМИ, который использует в известной степени общие приемы, описанные в I части.

Вычислительный план машине может быть задан в виде описанной выше схемы, т. е. в виде ряда строк (блоков), элементами которых могут быть: 1) числовые величины, а также программы, матрицы программ, схемы и 2) операции над этими величинами. В основном должны применяться операции универсальные, заранее подготовленные и используемые в различных работах.

Среди операций отметим *типовые*, которые могут выполняться над числами или поэлементно над величинами (это все машинные операции $+, -, \times, \cdots$ и операции, выполняемые по стандартным подпрограммам, рассчитанным на подачу аргументов в стандартные ячейки: e^x, $\sqrt{x}$, x^y), *специальные*, которые опираются на некоторую программу, записанную в условных аргументах (например, операция умножения двух комплексных чисел) и, наконец, *операции, рассчитанные специально на укрупненные аргументы*, например такая

$$X = (R;\ A,\ B),$$

которая по матрице A и вектору B выдает вектор X — решение системы $AX = B$.

Кроме операций над числовыми величинами, могут быть операции над схемами и программами: упростить схему, по данной схеме (формуле) составить программу, провести итерацию схемы до выполнения некоторого условия.

Рассмотрим в качестве примера блоксхему для решения интегрального уравнения

$$\varphi(x) - \int_0^1 K(x,\ y)\ \varphi\ (y)\ dy = f(x)$$

путем сведения его к алгебраической системе

$$\tilde{\varphi}(x_i) - \sum_{k-1}^{n} A_k K\ (x_i,\ x_k)\ \tilde{\varphi}(x_k) = f(x_i) = f_i\ \ (i = 1, 2, \cdots,\ n)$$

с помощью квадратурной формулы

$$\int_0^1 f(x)\ dx \simeq \sum A_k f\ (x_k).$$

Участвующие в ней величины будут $A = \{A_k\}$, $X = \{x_k\}$, $F = \{f_k\}$, $\Phi = \{\tilde{\varphi}_k\}$ — векторы, $B = \{K\ (x_i,\ x_k)\}$, $C = \{A_k K\ (x_i,\ x_k)\}$, D, E—матрицы.

Тогда блоксхема задачи запишется так:

1) $\Phi = (R;\ D,\ F)$ (R — решение системы);

2) $D = (-\ ;\ E,\ C)$ ($-$ поэлементное вычитание матриц);

3) $C = (\cdot,\ A,\ B)$ (умножение столбцов матрицы на вектор);

4) $B = (K,\ x,\ x)$ (K — програма соотв. $K\ (x,\ y)$; применяется поэлементно к парам: $(x_i,\ x_k)$;

5) $F = (f,\ x)$ (f — программа соотв. $f\ (x)$; применяется поэлементно к x).

В машину вводится только это задание: справки о величинах и операции с ними специально связанные (например, функции $K\ (x, y)$ и $f(x)$, записанные в виде схем); некоторые общие операции предполагаются введенными раз и навсегда.

Как уже говорилось, проведение вычислительного плана осуществляется с помощью специально построенной универсальной программы — *прораба*.

Было создано несколько типов такого рода программ. Пусть в машину введены блок-схема и исходные данные. *Прораб первого типа* (разработан Л. Т. Петровой) „сам определяет“ порядок выполнения работ. Работа прораба заключается в последовательной логически упорядоченной выписке блоков схемы. Если для данного блока все аргументы оказываются известными, подаются данные о них и включается операция, если какой-либо аргумент неизвестен, предварительно выписывается блок, определяющий этот аргумент, и он, в свою очередь, подвергается исследованию.

Помимо выполнения операции, стоящей в данной строке, прорабом вырабатывается справка результата, если она не предусмотрена заранее. При выполнении операции, в соответствии со справками аргументов и результата, прорабом производится переадресация; возможны два способа ее — в зависимости от того, как построена программа операции — в расчете на фиксированные ячейки или на условные аргументы.

Включение переадресации в работу прораба делает программы большинства универсальных и специальных операций весьма короткими.

Прорабом производится и размещение результатов.

Результаты могут размещаться на места, предусмотренные в плане, или на свободные со стиранием ненужных. Такая безадресная система особенно удобна, если объем получаемых результатов и фактическая схема процесса заранее полностью неизвестны. Возможна также система, в которой используется ряд полей, и положение результатов фиксируется относительно начала поля, но поля целиком можно сдвигать, стирать или переносить в магнитную память. При наличии отметки о печати полученный результат печатается.

Основным достоинством этой системы является то, что вводимое задание максимально приближено к обычному вычислительному плану, отличаясь от него по существу только тем, что он надлежаще закодирован. Более того, при введении в машину подобного плана можно обойтись даже без предварительного выписывания всех формул задания, оформляя непосредственно словесное или математическое описание метода в виде блоксхемы. Это делает такой план наглядным и удобным для математика.

В частности, удобным является введение необходимых изменений в него, контрольных операций; облегчается также отладка работы на машине.

Недостатками в работе этого прораба являются прежде всего значительные „накладные расходы“, так как при выполнении вычислений с помощью него, наряду с необходимыми действиями над числами, выполняется ряд машинных операций, связанных с работой самого прораба — по развертыванию схемы и справок, перемещению и стиранию данных и т. п. Хотя эти операции просты, но довольно многочисленны и, в особенности на машинах со стандартным операционным циклом (типа БЭСМ), дают заметное увеличение машинного времени.

Кроме того, в некоторых случаях такой порядок заставляет иногда фиксировать в памяти лишние промежуточные величины, что приводит к увеличению загрузки оперативной памяти. Прораб сам также занимает значительное место, до 500—600 ячеек, зато, правда, получается выигрыш в месте, затрачиваемом на ввод программы. Увеличение расхода машинного времени не играет существенной роли при единичных работах небольшой продолжительности, в частности экспериментальных, и оно вполне окупается упрощением работы, сокращением времени на подготовку задачи, отладку программы и возможностью применения более сложных, но экономных по времени схем счета. Однако в других случаях это существенно.

В связи с этим были предприняты работы по составлению других типов прорабов, более практичных. *В прорабе второго типа* схема пишется в порядке ее выполнения, но могут быть скачки перехода и вообще вставлены отдельные машинные строчки. Таким образом, схема, по которой работает этот прораб, несколько ближе к обычной программе, отличаясь от нее в основном тем, что использует укрупненные операции и величины.

В настоящее время отлажен прораб № 5 (разработан В. А. Булавским и М. А. Яковлевой), который решает задачи весьма экономно по времени.

За счет чего достигнуты эти усовершенствования и сокращения?

В ряде случаев некоторые действия (порядок работы, размещение, стирание, подготовка справки) оказывается выгоднее осуществлять не через „прораба", а непосредственно при составлении плана.

Упрощено составление справок.

Для некоторых участков схемы оказалось целесообразным фиксировать ту программу, по которой фактически ведется вычисление на этом участке, и определенное время сохранять ее в памяти машины.

Использованы некоторые специальные приемы (условные номера, возбуждение). Кроме того, накопленный опыт позволил рационализировать решение некоторых вопросов.

С помощью прорабов проводился ряд работ, относящихся к задачам линейной алгебры, численному решению дифференциальных и интегральных уравнений, уравнений эллиптического типа, численному осуществлению конформного преобразования и др.

Эксперимент был несколько затруднен тем, что нам приходилось пользоваться московскими машинами.

Вследствие этого, нам также трудно провести сопоставление этого метода с методом П.П.

Последний имеет свои преимущества, но и свои трудности, так как использование П.П. требует предварительной подготовки, довольно большой, и не очень наглядной информации, вывода и хранения полученной программы.

При этом в работах со сложной вычислительной и логической схемой необходимо детальное составление программы для всех логически возможных случаев, в то время как в крупноблочной системе при разветвлениях процесса развертываются только те ветви, которые фактически реализуются в данных условиях.

В общем нам кажется, что оба пути являются перспективными. По-видимому, для однократно проводимых и экспериментальных работ должна оказаться более выгодной крупноблочная система. Для других работ это менее ясно, но, как мы уже говорили, ввиду недостаточного опыта в применении нашей системы, это заключение является предварительным.

Следует еще обратить внимание на то, что наличие этой системы программирования полезно учесть при проектировании электронных машин с программным управлением. Например, нежелательна унификация времени на разные команды, полезно наличие полуоперативной памяти (для размещения прораба) и др.

III. Аналитические вычисления

Та же схемная символика и упоминавшиеся операции по преобразованию схем весьма удобны для проведения на машине аналитических выкладок и преобразований.

Прежде всего следует отметить, что имеет большие преимущества запись функции в виде схемы. Мы уже отмечали некоторые из них: отчетливое выявление характера зависимости, экономия записи и др., которые имеют значение при введении функций в машину. Однако эти преимущества вскрываются в полном виде, когда машина не только использует эту запись для вычислений, но и сама преобразует и составляет такие записи, как это имеет место при аналитических вычислениях, подобно тому, как преимущество двоичной системы записи чисел становится особенно заметным при осуществлении действий с ними на машине.

В качестве первого примера аналитического преобразования приведем упрощение рациональных функций. Рациональная функция это любая схема с операциями $+, -, \times, :$, с опорными элементами x, y, $z, \cdots$ (независимые, переменные) и c_1, $c_2, \cdots$ (постоянные). Записав в качестве схемных тождеств правила действий с дробями и раскрытия скобок, используя выполнение вычислений в случае действий с постоянными, а также применяя упоминавшуюся запись групп и их упрощение (что, впрочем, не обязательно), в результате последовательного применения перечисленных операций, получим преобразование рациональной функции к канонической форме отношения двух многочленов.

Более подробно мы остановимся на аналитическом выполнении дифференцирования на машине.

Пусть функция k записана в виде схемы S, состоящей из ряда строк вида

$$\left.\begin{array}{l} k=f(k_1,\ k_2) \\ k_1=f_1(\cdots) \\ \cdots\cdots\cdots\cdots \end{array}\right\} S.$$

Тогда запись в виде схемы S' для ее производной по x мы получаем формально, присоединив к ней одну строку:

$$\left.\begin{array}{l} S \\ k'=(D;\ k,\ x) \end{array}\right\} S'.$$

Однако, чтобы раскрыть эту запись и получить схему, не содержащую символа D, мы должны использовать ряд правил, а именно

$(D\,;c,\ x)=0$

$(D\,;z,\ x)=0$ (z — другая независимая переменная),

$(D\,,x,\ x)=1$

а также таблицу производных, записанную как правила перестановки оператора дифференцирования с остальными операторами; например, правило дифференцирования произведения $k_1=k_2\,k_3$

$$k_1'=(k_2\ k_3)'=k_2'\,k+k_2\ k_3'$$

дает такое схемное тождество $A=B$:

$$\left.\begin{array}{l} k=(D,\ k_1,\ x) \\ k_1=(\times,\ k_2,\ k_3) \end{array}\right\} A \qquad\qquad \left.\begin{array}{l} k=(+\,,\ n_1,\ n_2) \\ n_1=(\times,\ n_3,\ k_3) \\ n_2=(\times,\ k_2,\ n_4) \\ n_3=(D,\ k_2,\ x) \\ n_4=(D,\ k_3,\ x) \end{array}\right\} B$$

(n_1, $n_2\cdots$ вспомогательные аргументы). Тогда, опуская последовательно знак D на основе этих тождеств и пользуясь для опорных элементов отмеченными правилами, получим схему для k'. Эта схема может быть еще упрощена за счет использования некоторых правил алгебраических преобразований, а также выполнения вычислений, если встретятся действия над числами.

После того, как упрощение выполнено, таким же образом может быть найдена на основании схемы первой производной схема k'' и т. д. (так находятся не только обычные, но также и частные производные).

Если потребуется численное значение производной при конкретном значении переменных, то оно может быть легко найдено с помощью прораба непосредственно в машине, без вывода этой схемной записи и какой-либо внешней обработки ее.

Итак, машина „умеет дифференцировать". Наличие возможности аналитического дифференцирования позволяет сделать многое, что было бы невозможно без этого.

Например, мы можем не только провести численное интегрирование, но и выполнить на машине строгую оценку погрешности по остаточному члену. Наличие такого аналитического решения позволяет производить ряд исследований и оценок, которые нельзя сделать с помощью одного численного решения, изучать зависимость от параметра и пр.

Приведем два примера аналитического построения решения дифференциального уравнения.

Пример 1. Разложение решения в степенной ряд

$$y' = f(x, y), \qquad y(x_0) = y_0,$$

$$y(x) = y_0 + \frac{y_0'}{1!}(x - x_0) + \frac{y_0''}{2!}(x - x_0)^2 + \cdots$$

Для нахождения последовательных производных мы должны описанным выше образом по схеме

$$y' = f(x, y) \mid S$$

строить схему производной $y'' = (y')'$, присоединяя к ней только еще одно соотношение:

$$(y' = (D, y, x)$$

(вместо $(D, y, x) = 0$; она включает, как указывалось выше, схему S, определяющую y').

Точно так же по y'' находятся y''' и т. д., после чего вычисляются значения производных в точке x_0 и составляется степенной ряд.

Пример 2. В качестве другого примера упомянем о возможности построения известного разложения решения дифференциального уравнения по степеням параметра (метод Пуанкаре — Ляпунова):

$$y'' + y = F(\sigma)\, y'^2 \qquad (\text{либо} = F(x, y, y', \sigma))$$

$$y(0) = 1, \; y'(0) = 0; \qquad F(\sigma) = c_1\sigma + c_2\sigma^2 + \cdots$$

Решение ищется в виде

$$y(\sigma, x) = \varphi_0(x) + \varphi_1(x)\sigma + \varphi_2(x)\sigma^2 + \cdots$$

φ_k последовательно находятся из дифференциальных уравнений

$$\ddot{\varphi}_0 + \varphi_0 = 0; \qquad (\varphi_0 = \cos x)$$

$$\ddot{\varphi}_1 + \varphi_1 = \dot{\varphi}_0^2 c_1 = f_1(x);$$

$$\ddot{\varphi}_2 + \varphi_2 = \cdots \equiv f_2(x);$$

$$\cdots\cdots\cdots\cdots$$

$$\varphi_k(x) = \int_0^x f_k(t) \sin(x - t)\, dt;$$

$$f_k(t) = \left[\frac{1}{k!}\frac{d^k}{d\sigma^k}\left\{ y_{k-1}'^2(t)\, F(\sigma)\right\}\right]_{\sigma=0}.$$

Здесь f_k и φ_k — выражения типа

$$\sum_{k,\nu}(a_{k,\nu}\ t^k \sin \nu t + b_{k,\nu}\ t^k \cos \nu t).$$

Операции, которые здесь требуются, кроме дифференцирования, что мы уже умеем, осуществляются посредством простых схемных тождеств. Именно нужны преобразования произведений синусов и косинусов в суммы (формулы, обратные формулам приведения к виду, удобному для логарифмирования) и формулы интегрирования по частям выражений вида

$$\int x^l \sin kx\ dx.$$

Существенную роль играют в этой задаче также различные действия над группами номеров.

Этот метод широко используется в различных задачах небесной механики и теории регулирования, при этом осуществление его связано с весьма громоздкими выкладками. Поэтому их проведение с помощью машины было бы весьма полезным.

Для данной задачи, хотя она в основном запрограммирована, фактическое решение на машине еще не осуществлялось, но решение ряда других задач, в частности аналитическое вычисление производных различных порядков, было уже реализовано (Л. Т. Петровой).

Подобным же образом может выполняться и ряд других аналитических выкладок.

При этом преобразования и упрощения схем (формул) могут выполняться двояким образом. Либо вводя подходящее поле объектов, в котором все участвующие в схеме операции определены. При этом можно последовательно включить значения элементов схемы в этом поле. Либо схему можно привести к каноническому виду за счет последовательного применения к ней, пока это возможно, группы преобразований, определяемых некоторыми тождествами (подобно нормальному алгорифму).

Сложнее дело обстоит с задачами, которые не всегда имеют решения и для решения которых нет определенного алгорифма, например аналитическое интегрирование в общем случае. Однако намечаются некоторые перспективы разработки довольно общих методов их анализа и систематизации возможных попыток их решения.

Те же приемы, помимо аналитических выкладок, могут использоваться также для преобразования, упрощения, приведения к канонической форме и других видов математических заданий, например:

а) преобразования вычислительных планов с целью обеспечения более удобного вычисления по ним: вынос наверх выражений, зависящих от большего числа параметров, подъем более сложных операций, исключение повторного вычисления выражений и т. д.;

б) преобразование логических и теоретико-множественных формул: вынос кванторов наружу, исключение некоторых операций (отрицания, дополнения) и т. п.;

в) в вопросах анализа и синтеза релейно-контактных схем;

г) различные вычисления и преобразования в алгебре и топологии;

д) формальный анализ дедуктивных теорий (нахождение параллелизмов, частей, допускающих аксиоматическое построение, и т. д.).

В заключение мы полагаем, что нужно считать уже в самое ближайшее время осуществимым на машинах, наряду с обычными численными вычислениями, полное автоматическое проведение определенного математического метода решения задачи, с выполнением всех необходимых числовых и аналитических выкладок, а также логического анализа.

Наряду с использованием в приближенных вычислениях и прикладной математике машинный анализ должен получить систематическое применение и в ряде так называемых теоретических областей математики.

Ленинградское отделение
Математического института
АН СССР

Поступило 21 I 1957

Լ. Վ. Կանտորովիչ

ԾՐԱԳՐԱՅԻՆ ՂԵԿԱՎԱՐՈՒՄ ՈՒՆԵՑՈՂ ՄԵՔԵՆԱՆԵՐԻ ՎՐԱ ԹՎԱՅԻՆ ԵՎ ԱՆԱԼԻՏԻԿ ՀԱՇՎՈՒՄՆԵՐ ԿԱՏԱՐԵԼՈՒ ՄԱՍԻՆ

Ա Մ Փ Ո Փ Ո Ւ Մ

Հոդվածը բաղկացած է երեք մասից: Առաջին մասում մշակվում են խնդիրը ավտոմատիկ հաշվիչ մեքենայի մեջ մտցնելու ընդհանուր հարմար եղանակ, ինչպես նաև մաթեմատիկական ու տրամաբանական բանաձևերը սիմվոլիկ կերպով գրելու եղանակ, որպիսի եղանակները կիրառվում են մաթեմատիկական հաշվումներում և, մասնավորապես, ընդհանուր բանաձևերը հատուկ սխեմաներով փոխարինելու, այդ սխեմաները ձևափոխելու և այլ հարցերում:

Երկրորդ մասում նկարագրվում է մեքենայի վրա հաշվումների նախապատրաստման արագ սխեմ, որի էությունն այն է, որ կարճագրված սեղմով մեքենայի մեջ է մտցվում աշխատանքի հաշվողական սխեման, որը գրված է խոշորացված էլեմենտներով և գործողություններով: Այդ սխեման վերծանվում և կատարվում է մեկ ընդմիշտ կազմված ծրագրով՝ «աշխողեկով», որը ղեկավարում է նաև մեքենայի օպերատիվ և արտաքին հիշողության տվյալների տեղավորումը, հրամանների վերահասցեագրման ամբողջ սխեման և այլն:

Վերջապես, երրորդ մասը նվիրված է անալիտիկ հաշվումների (օրինակ՝ ածանցման, ինտեգրումների և այլն) ավտոմատիկ կատարմանը մեքենաների օգնությամբ: