表征 Rietveld 精化进行情况的参数有剩余指数和标准偏离，其定义如下：

$$R_P = \frac{\sum \left| Y_{i观测} - \frac{1}{K} Y_{i计算} \right|}{\sum Y_{i观测}}, \qquad (12)$$

$$R_{WP} = \left[\frac{\sum W_i \left[Y_{i观测} - \frac{1}{K} Y_{i计算} \right]^2}{\sum W_i [Y_{i观测}]^2} \right]^{1/2}, \qquad (13)$$

$$\chi^2 = (R_P/R_{WP})^2, \qquad (14)$$

$$\sigma_j = \{A_{jj}^{-1} \sum W_i [Y_{i观测} - Y_{i计算}]^2/(N - P + C)\}^{1/2}, \qquad (15)$$

式中 A_{jj}^{-1} 为逆矩阵的对角元素，N，P 和 C 分别为观测点数目、修正参数目和对修正施加的约束数.

2. 一般化的结构分析系统（GSAS）[7]

进行结构精化的计算机程序很多，GSAS 则是最新的、正被广泛使用的一般化结构分析系统。这个程序能处理用X射线和中子获得单晶和粉末衍射数据，解决结构问题，能处理多相混合试样的粉末衍射数据，以精化每一相的结构参数。图 6 给出粉末衍射花样计算的 GSAS 中各程序及其相互关系。值得提及的在进行最小二乘方精化（GENLES）前使用 GSAS 编辑程序（EXPEDT）时需输入许多初始数据，并在 GENLES 过程中精化。它们包括：

（1）结构参数：空间群、点阵参数、各原子在单胞中的位置、占位几率及热振动参数（各向同性或各向异性）；

（2）衍射参数：线形函数代码和有关参数 U, V, W, X, Y, Z 等、背景函数代码和参数；

（3）实验参数：衍射仪参数——零位、波长、偏振系数，尺寸因子、吸收系数、消光参数、择尤取向参数等，以及精化计算时若干限制方程等。

（未完待续）

符号计算与物理

王稼军 韩其智

（北京大学物理系，北京 100871）

张 玫 向延育

（清华大学物理系，北京 100084） （中国科学院北京天文台，北京 100080）

符号计算同传统的数值计算相比，有其独特的优点。我们结合这些优点，以实际应用例子阐明符号计算在物理研究工作中的重要性，并希望它能在我国物理学界得到更广泛的重视及应用。

目前，在物理研究工作中使用计算机来求解物理问题已经相当普遍，但提起使用计算机，人们或许会非常自然地理解为数值求解或对物理过程进行计算机模拟。然而，现代计算机也可以代替人进行解析公式推导，并给出与人演算相同的结果。这就是所谓的符号计算。近 40 年来，这方面工作已有了巨大进展，并在国际上形成了一个新的研究领域。

一、符号计算

符号计算又称计算机代数，通俗地说就是用计算机推导数学公式，如对表达式进行因式分解、化简、微分、积分、解代数方程、求解常微分方程等。目前国际上最有代表性和最流行的

通用符号计算软件有 REDUCE, MACSYMA, mμ-MATH, MAPLE, MATHEMATICA 等. 它们的基本功能类似,但各有特点:mμ-MATH, MAPLE 可在多种微机上运行; 而 REDUCE 有多种版本，可在大、中及小型机上运行; MATHEMATICA 从一开始就是为科学工作站设计的，可在 SUN 工作站及 MACINTOSH 上运行并有效率较高的特点；功能最强的符号计算软件 MACSYMA 除了具有前面提到的功能外，还可以进行矩阵运算，对表达式作泰勒展开，拉普拉斯变换，求解部分非线性代数方程等，由于软件较大，所需计算机资源也较大，前几年最少要在 MICRO-VAXII 上运行，现在可以在 VAX-2000 上运行.

符号计算系统一般有两种运算方式：程序式和交互式. 程序式与数值计算一样，用户可运用系统提供的一些基本语句及功能函数编写程序，计算机执行程序中给出的一系列命令，最后返回结果. 交互式又称人机对话，即在终端上发出一个命令，计算机执行这个命令，返回结果，并等待接受下一个命令，再计算，再返回结果. 这与使用计算器进行数值计算的情况类似，不过现计算机进行的是符号计算. 目前多数人是通过交互式使用符号计算的.

为使读者对符号计算有一初步了解，我们以 MACSYMA 为例，给出几个交互式运行的例子. 下面是一个对多项式进行因式分解的实例，尽管该多项式有 21 项，但在 MACRO-VAXII 上计算，仅花费 3916ms 就完成了因式分解.

```
(c4) FACTOR (-22*z^3
        + 203*y*z^2
        - 35*x*z^2 + 63*z^2
        - 328*y^2*z
        - 254*x*y*z - 88*y*z
        + 382*x^2*z - 190*x*z
        + 16*z - 105*y^3
        - 107*x*y^2 - 311*y^2
        + 81*x^2*y - 238*x*y
        - 163*y + 35*x^3
        + 229*x^2 - 115*x
        - 21);
Time = 3916msec.
```

(d4) $-(z-7y+5x-3)(2z-5y-7x-1)$
$(11z+3y+x+7)$

其中 (c4) 为 MACSYMA 的输入行提示符，(d4) 为输出行标记符，FACTOR 是 MACSYMA 的因式分解命令. 还可以用 MACSYMA 求微分命令 DIFF 及积分命令 INTEGRATE 来对表达式求导和积分.

(c5) DIFF (X^X^X,X);

Time = 116msec.

(d5) $x^{x^x} \quad (x^x \quad \log(x) \quad (\log(x)+1)+x^{x-1})$

(c6) INTEGRATE (SQRT(A^2 + X^2),X);

Time = 900 msec.

(d6) $$\frac{a^2 \operatorname{asinh}\left(\frac{x}{\operatorname{abs}(a)}\right)}{2}+\frac{x\operatorname{sqrt}(x^2+a^2)}{2}$$

例中 SQRT(x) 是平方根函数，asinh(x) 是反双曲函数. 当从 (c7) 行输入计算定积分

$$\int_0^\infty \frac{\cos px}{x^2+a^2}\,dx$$

的命令后，由于积分值与参量 p 和 a 有关，所以机器提问 p 和 a 是正、负还是零，只有在用户作出明确回答之后，机器才给出计算结果. 例如从终端键入 'POS;' (即输入 p 和 a 均为正的信息)，计算结果在 (d7) 行给出. 其中 %pi 和 %e 分别代表常数 π 和 e.

(c7) INTEGRATE (COS(P*X)/(X^2 + A^2),X,0,INF);

Is p positive, negative, or zero?

POS;

Is a positive, negative, or zero?

POS;

Time = 28116 msec.

(d7) $$\frac{\%pi\ \%e^{-ap}}{2a}$$

又如从 (c8),(c9) 和 (c10) 行输入线性方程 EQ1,EQ2,EQ3，并在 (c11) 行输入解这三个方程构成的线性方程组命令 LINSOLVE,线性方程组的解便在 (d11) 行给出.

(c8) EQ1:3 * X + 4 * Y + 5 * Z = 6;

Time = 33 msec.

(d8) $$5z+4y+3x=6$$

(c9) EQ2: A * X + B * Y + Z = 3;

Time = 33 msec.

(d9) $$z+by+ax=3$$

(c10) EQ3: (A - 1) * X - B * Y - A * Z = 0;

Time = 66 msec.

(d10) $$-az-by+(a-1)x=0$$

(c11) LINSOLVE ([EQ1,EQ2,EQ3],[X,Y,Z]);

Time = 1183 msec.

(d11) $$\left[x=\frac{(6a+9)b-12a}{(13a-8)b-4a^2-4a+4}\right.$$
$$y=-\frac{6a^2-18a+9}{(13a-8)b-4a^2-4a+4}$$
$$\left.z=\frac{(12a-15)b-12a+12}{(13a-8)b-4a^2-4a+4}\right]$$

其他符号计算软件，除输入行提示符和输出行标记符不同之外，具体命令形式和数学功能也不尽相同，但交互式运行方式是一样的.

现在，符号计算的各种软件已经在科学、技术、工程和教育等方面得到许多应用，并正日益引起重视.

二、符号计算是物理研究的有力工具

同传统的数值计算相比，符号计算具有一些独特之处，其主要特点可以归纳为：公式推导、精确计算和带有"智能"性质．正是这些特点，使它成为物理研究的有力工具.

符号计算能在计算机上进行公式推导，这给物理学家带来很多方便.

由于符号计算的处理对象是数学表达式，得到的结果也是数学表达式，这有可能比数值计算更有利于总结规律．因为从一个含参量的公式总结规律，往往比由浮点数构成的数据表总结规律要容易．一般说来，人们只在没有解析表达式或求不出解析表达式时，才采用数值计算．这里有一个求积分的例子．在光学辨别问题中[1]，考虑在地球漫散射基础上，用能流打到物体上，从不同角度观察它的辐射亮度，来辨别这个物体．需要计算积分.

$$\int \cos p \operatorname{arctg}\left(\frac{b}{\cos p}\right) dp.$$

这个积分结果，还要作为被积函数，在一个重积分中出现．如果用计算机作数值计算，需要计算的点是很多的，有人在 CYBER 机上花费了

三个多小时[1]，算出 15 个点. 而用符号计算 MACSYMA，得到积分的解析结果：

$$-\sqrt{b^2+1}\ \operatorname{arctg}\left(\frac{b\operatorname{tg}p}{\sqrt{b^2+1}}\right)$$
$$+2b\operatorname{arctg}\left(\frac{\sin p}{\cos p+1}\right)$$
$$+\sin p\operatorname{arctg}\left(\frac{b}{\cos p}\right).$$

利用这个解析公式，仅花费了两分钟[1]，就可以算出数百个点.所以在这个问题上，采用符号计算的优越性是显而易见的.

一般情况下，手做计算如没有好的核对系统，其准确性是难以保证的. 而找一个好的核对系统相当困难，并且计算量一般与重新计算一遍相差不多. 符号计算可以作为一个好的核对系统为人们做大量重复性的和繁复的解析推导. 例如 1963 年到 1965 年期间，Klerer 和 Grossman 用符号计算核对了八本积分表[2]，发现每本均有错误，平均错误达 10%，最差的达 25%，并且不像是误排的. 这说明即使是熟练的专家，手算的结果，其准确性也难与计算机相比.

有的物理问题不一定有解析结果，但直接进行数值计算又有一定困难. 在这种情况下，还可以把符号计算与数值计算结合起来使用，即先用符号计算完成部分推导工作，得到中间结果，然后再进行数值计算. 北京大学、清华大学有些学生在经过符号计算课学习后，采用这种办法来做科研工作，取得了较好的效果.

符号计算可提供精确的计算. 一般在数值计算中，数被表示为浮点数，如 1/3 和 2/3 各用浮点数 0.333333 和 0.666667 表示，计算 (2/3 －1/3) 的结果是 0.333334.说明数值计算中，数 1/3 可以表示为 0.333333，也可以表示为 0.333334. 这就引起了数值计算的舍人误差，这种由数的表示和相应的数据存储方式引起的舍人误差，即便使用最先进的计算机，也是不可避免的. 另外在数值计算中，对大的整数的位数也有限制，如要准确计算 200!，将很困难. 而在符号计算中，存储的数据是准确的. 如以 LISP 语言写的符号计算软件为例，数据用表的形式存储，表由圆括号及包括在括号内的元素组成. 如 1/3 和 2/3 用表

(QUOTIENT 1 3) 和 (QUOTIENT 2 3)

表示，计算 (2/3 － 1/3) 用表

```
(DIFFERENCE (QUOTIENT 2 3)
            (QUOTIENT 1 3))
```

表示，计算后得到准确结果 1/3. 同样 200! 用表

(FACTORIAL 200)

表示，其中 FACTORIAL 是用 LISP 语言定义的阶乘函数. 下面是用 MACSYMA 软件计算的 200!结果.

```
(c12) 200!;
Time = 333 msec.
(d12)  7 8 8 6 5 7 8 6 7 3 6 4 7 9 0 5 0 3 5 5 2 3 6 3 2 1 3 9 3 2 1 8 5 0 6 2 2 9 5 1 3 5 9 7 7 6 8
       7 1 7 3 2 6 3 2 9 4 7 4 2 5 3 3 2 4 4 3 5 9 4 4 9 9 6 3 4 0 3 3 4 2 9 2 0 3 0 4 2 8 4 0 1 1 9
       8 4 6 2 3 9 0 4 1 7 7 2 1 2 1 3 8 9 1 9 6 3 8 8 3 0 2 5 7 6 4 2 7 9 0 2 4 2 6 3 7 1 0 5 0 6 1
       9 2 6 6 2 4 9 5 2 8 2 9 9 3 1 1 1 3 4 6 2 8 5 7 2 7 0 7 6 3 3 1 7 2 3 7 3 9 6 9 8 8 9 4 3 9 2
       2 4 4 5 6 2 1 4 5 1 6 6 4 2 4 0 2 5 4 0 3 3 2 9 1 8 6 4 1 3 1 2 2 7 4 2 8 2 9 4 8 5 3 2 7 7 5
       2 4 2 4 2 4 0 7 5 7 3 9 0 3 2 4 0 3 2 1 2 5 7 4 0 5 5 7 9 5 6 8 6 6 0 2 2 6 0 3 1 9 0 4 1 7 0
       3 2 4 0 6 2 3 5 1 7 0 0 8 5 8 7 9 6 1 7 8 9 2 2 2 2 2 7 8 9 6 2 3 7 0 3 8 9 7 3 7 4 7 2 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

符号计算能给出精确结果，这一特点为用户提供了良好的使用环境，可避免由舍人误差引起的“病态问题”. 一个有代表性的例子是英国[3] Oxfordshire 的 Culham 实验室的 W. Arter 等人，通过 MACSYMA 进行推导及分析等离子不稳定性，发现了以前用超级计算机

1） 这里给出的时间为中央处理机计算时间，简称“CPU”时间.

作许多小时的数值模拟所得结果，竟然是错误的。可见在一些"病态问题"中，采用符号计算，有可能使情况大为改善。又如在进行原子核壳模型波函数计算时，考虑零级近似，核子在平均中心力场中作独立运动，所以单核子运动近似波函数是已知的。对有多个核子的原子核，其波函数还必须满足交换对称和转动不变，即波函数要反对称化并具有确定的总角动量。现在流行的壳模型程序[4]都先用递归办法，得到超完备空间母分系数，再经过解$SP(N)$群Casimir算子的本征值问题和正交化，得到正交归一完备基下母分系数，从而求出波函数[5]。这就存在一个从超完备态空间中，选出独立的线性无关的态矢量，构成完备态空间的问题。而对这类问题，数值计算的舍入误差在大量计算中就会引进困难。如经过大量计算后，用浮点数给出两个向量

$$0.332238|a\rangle - 0.403782|b\rangle$$

$$-0.332245|a\rangle + 0.403762|b\rangle$$

由于存在舍入误差，难于判断它们是否线性相关。而符号计算给出向量的精确值，如两个向量

$$\sqrt{15/154}\,|a\rangle - 4\sqrt{51/5005}\,|b\rangle$$

$$-\sqrt{15/154}\,|a\rangle + 4\sqrt{51/5005}\,|b\rangle$$

可以准确判断它们是线性相关的，而向量

$$\sqrt{16999/154001}\,|a\rangle - 4\sqrt{51001/5004998}\,|b\rangle$$

与前两个向量线性无关。这说明利用符号计算可以克服舍入误差造成的混乱。

符号计算为用户编写自己需要的符号处理程序提供了强有力的语言环境。值得注意的是，它的语言带有一定"智能"性质。如符号计算允许递归，即在函数定义中可以调用所定义的函数本身。这在 FORTRAN 程序中是禁止的。利用递归可以使符号计算程序编写容易。例如已知勒让德函数

$$P_0(x) = 1,\quad P_1(x) = x,$$

并有递推关系

$$lP_l(x) - (2l-1)xP_{l-1}(x) + (l-1)P_{l-2}(x) = 0 \quad (l > 0).$$

利用 MACSYMA 容易写出求任意阶勒让德函数的程序，调用该程序可以计算出所需的勒让德函数，如(c13)到(c14)所示。(d14)给出 $P_{11}(x)$。

```
(c13) LEGENDRE(L,X):= IF L = 0 THEN 1
                      ELSE IF L = 1 THEN X
                        ELSE ((2*L - 1)*X*LEGENDRE(L - 1,X)
                             - (L - 1)*LEGENDRE(L - 2,X))/L$
(c14) RATSIMP (LEGENDRE(11,X));
Time = 26883 msec.
(d14) (88179x^11 - 230945x^9 + 218790x^7 - 90090x^5 + 15015x^3 - 693x)/256
```

另外，可以进行模式匹配也属于"智能"性质。粗略地说，模式是把具有某些特征的表达式归成一类，一类叫做一个模式。而匹配则是判定一个表达式是否属于某个模式。所谓模式匹配就是对表达式分类并按类对表达式进行处理。这种处理方法可以把对表达式一个一个处理的过程变成对表达式一类一类地处理。例如，在量子力学二次量子化表象中考虑费米子问题，根据泡利原理，任何两个单粒子态不能相同，以 a_α^+ 表示产生算符，a_α 表示湮灭算符，则有

$$a_\alpha^+ a_\alpha^+ = 0,\qquad a_\alpha a_\alpha = 0.$$

在符号计算中，可以用模式匹配来定义替换规则，使整个计算过程中，自动地把 $a_\alpha^+ a_\alpha^+$ 与 $a_\alpha a_\alpha$ 替换成零。

智能性质还包括可以建立数学对象的关系数据库，如可以给某个函数赋以对称或反对称性质，给某个数赋以偶数性质等，以后的计算可以反映出这些性质。符号计算的这些特点使其程序比较接近人的思维。编写程序相对容易些，为处理各种复杂的数学问题，编写各种专用程序包提供了极大的便利。例如，在高能物理、天

体物理等方面已写成了一些专用的软件包，正在物理研究工作中起着重要的作用。

可以说，符号计算在广阔的物理学领域如光学、电磁学、统计物理学、固体、等离子体、原子、分子和原子核物理、高能物理和天体物理等方面都有应用。这些方面的应用例子可在文献[1]中找到许多。这说明符号计算正日益发展成为物理学研究的重要工具。

三、符号计算的发展概况及展望

符号计算已经发展成为一个新的学科，不少国家成立了对符号计算特别有兴趣组织，每年定期召开国际性符号计算学术会议，并出版了符号计算的专门杂志Jour. Symbolic Computation。符号计算也已受到物理学家的重视。有些物理学家是符号计算最热心的支持者，少数更是最积极的参与者。最突出的是 REDUCE 软件作者 A. C. Hearn，60 年代他在美国斯坦福大学做理论物理工作，在研究正负电子相互作用过程中开始用计算机进行公式推导[6]，后来致力于符号计算，写成了著名的符号计算软件 REDUCE。

为了推广应用符号计算，国外从 1982 年起就陆续在大学中开设符号计算课程[3,7]。从 1982 年起奥地利的 Johannes Kepler 大学就开设了全面介绍符号计算的课程，到 1985 年欧美许多大学都相继开设了符号计算课，如美国的 Delaware大学，加拿大的Waterloo大学，英国的 Queen Mary 学院，Liverpool 大学等，并出版了有关符号计算的教材[8]。我国物理学界，如中国科学院理论物理研究所、中国科学院物理研究所和北京大学、清华大学、中国科学技术大学研究生院、南开大学等院校物理系，都开展了符号计算的推广工作，包括设置符号计算课，举办讲习班等，也编写了一些符号计算方面教材[9]。由此看来，国内也已经有推广符号计算的知识条件。

当然符号计算还正处在发展阶段，因此它并不是尽善尽美的。目前的符号计算软件，大多数建立在 LISP 和C语言基础之上，所以与数值计算相比，符号计算需要较大的内存和较长的运行时间。解决这个问题一方面有赖于计算机技术的发展，另一方面是希望用新的语言如 Scratchpad 和 Newspad 等代替 LISP 作为符号计算软件的基础，从而摆脱冗长的 LISP 程序及由此带来的弱点

要进一步应用符号计算，物理学家必须针对要解决的物理问题，提出合适的算法，因为与数值计算一样，算法同样是符号计算的核心。因此发展符号计算，更好地在物理领域中应用符号计算，有着广阔的天地和光明的前景。我们相信符号计算会带来一场变革，首先是对数学和理论物理的研究方式会有较大冲击。我国物理学界应当及时抓住时机，利用我们现有的条件，在符号计算的应用方面走到世界前列。

[1] R. Pavelle, Applications of Computer Algebra, Kluwer Academic Publishers, Boston, (1985).

[2] 斯英古，数值计算与计算机应用，**10**(1989)，49，121.

[3] M. Maccallum, *New Scientist*, **23**(1986), 52.

[4] D. Zwart, GENESIS Code, R. J. Van de Graaff Laboratory Internal Publication, (1984).

[5] A. de-Shalit and I Talmi, Nuclear Shell Theory, Academic, New York, (1963)

[6] Y S Tsai et al., *Phys. Rev. B*, **140**(1965), 721; A. C. Hearn, *Phys. Rev.*, **187**(1969), 1950.

[7] B. F. Caviness, *Lecture Notes in Computer Science*, **203**(1985), 1.

[8] R. H. Rand, Computer Algebra in Applied Mathematics: an Introduction to MACSYMA, Pitman Publishing, (1984); B. Buchberger et al., Computer Algebra: Symbolic and Algebraic Computation, Version 2, Springer-Verlag, Vienna, (1983).

[9] 陈元江等，REDUCE 语言简明教程，南开大学出版社，(1989).