| | |
|---|---|
| Kentucky | Paul Eakin |
| Michigan | Robert Griess |
| Minnesota | Mark Feshbach |
| North Carolina State | Michael Singer |
| †Paris | Daniel Lazard |
| Penn State | Edward Formanek |
| Penn State | George E. Andrews |
| †Pisa | Patrizia Gianni |
| Rockefeller | Joel Cohen |
| Rutgers | Eduardo D. Sontag |
| RIT | George Georgantas |
| RPI | M. S. Krishnamoorthy |
| UCSD | John J. Wavrik |
| Virginia | Leonard Scott |
| Wisconsin | Peter Orlik |
| Wisconsin | Richard Askey |
| York | Donald Solitar |

**Computer Science Departments (13)**

| | |
|---|---|
| Cornell | Dexter Kozen |
| CMU | A. Nico Habermann |
| Delaware | B. F. Caviness |
| Illinois | William Gear |
| MIT | Gail Zacharias |
| MIT | Richard Zippel |
| Rice | C. Sidney Burrus |
| Rice | Hans Boehm |
| SMU | David Y. Y. Yun |
| UC Berkeley | Richard Fateman |
| UCLA | Leonard Kleinrock |
| Waterloo | Bruce Char |
| Yale | Marina Chen |

**Government/Industrial Laboratories (7)**

| | |
|---|---|
| AT&T Bell Labs. | Andrew Odlyzko |
| Dept. of the Army | A. Brinton Cooper III |
| Dept. of the Navy Dahlgren, VA | Kenneth Bannister |
| †IMAG France | Jean Della Dora |
| Lawrence Livermore | Grant Cook |
| Los Alamos | William A. Beyer |
| Rand Corporation | A. C. Hearn |

**Engineering Departments (2)**

| | |
|---|---|
| Columbia | Morton B. Friedman |
| North Carolina State | Clarence Maday |

If you or a colleague would like to avail yourself of **Scratchpad II**, please send me a letter describing the nature of the use you would make of the system. In turn, our group will try to arrange for a joint study to be set up between IBM and your department. We would only ask that you pay for communication costs as appropriate.

Because of the limited capacity of the 4381, we cannot promise that all requests to use our system will be accepted. We will try to accommodate as many as possible.

**Richard D. Jenks**

# Algebra Snapshot: Linear Ordinary Differential Operators

Here we report on our work on operators in **Scratchpad II.** The package we describe provides linear ordinary differential operators. We decided to restrict our attention to this limited, well-understood domain for two reasons. First, this domain was a tool to be used in some work we were doing on ordinary differential equations. Second, we felt that if we could create the differential operator domain using only user-level facilities, then it could be used as an example for other operator domains.

As far as we are aware, operator algebra has not been part of the initial design of any computer algebra system. After-the-fact addition of classes of operators can be awkward, requiring either special functions for operator manipulation ("opPlus", "opTimes", etc.), which is ugly, or requiring modification of the base system, which is impossible for most users.

**Scratchpad II** addresses this issue by providing generic operations and by making application itself an operation. For example, the domain **Mapping**(A, B) of functions from A to B exports the application operation

"." : (**Mapping**(A,B), A) → B

which takes a mapping from A to B and an element of A to produce an element of B.

The notion of generic application is fundamental to the system and is used not only for function invocation but also for subscripting arrays, list element extraction, record field selection and indexing tables. Two syntaxes are provided for application, one associating to the left and another to the right:

f a b means f applied to (a applied to b)
f.a.b means (f applied to a) applied to b

To create a ring R of operators on a set S in **Scratchpad II**, one adds the operation

"." : (R, S) → S

to the constructor of R.

## *LODO*

**LODO**(A,M) is the domain of linear ordinary differential operators over an A-module M, where A is a differential ring. This includes the cases of operators which are polynomials in D acting on scalars or vectors depending on a single variable. The coefficients of the operator polynomials can be integers, rational functions, matrices or elements of other domains.

### *Differential operators with constant coefficients*

We begin by making some type assignments and declaring *Dx* to be a linear differential operator. Please refer to "Some Scratchpad II Constructor Name Abbreviations" on page 8 for expansions of the type abbreviations.

```
PZ := UPDR(P[x]I, I);

PQ := UPDR(P[x]RN,RN);

Dx: LODO(RN, PQ)
```

**PZ** is the domain of univariate polynomials in *x* over the integers with some additional properties that make it into a differential ring. Similarly, **PQ** is a domain of univariate polynomials over the rationals. (Within a few months one should be able to use **P[x] I** and **P[x] RN** directly, without resorting to the **UPDR** constructor.)

Operators are created as polynomials in *D()*.

```
Dx := D()

   (5)   "D"

a  := Dx  + 1

   (6)   D + 1

b  := a + 1/2*Dx**2 - 1/2

          1  2         1
   (7)   (---)D  + D + ---
          2            2

c := (1/9)*b*(a + b)**2

   (8)
        1    6      5    5      13    4      19    3
      (----)D  + (----)D  + (----)D  + (----)D
        72         36         24         18
   +
        79   2      7          1
      (----)D  + (----)D + ---
        72         12        8
```

To apply the operator *a* to the value *p* the usual function call syntax is used.

```
p: PQ := 4*x**2 + 2/3;

a p

              2          2
   (10)   4x  + 8x + ---
                         3
```

Operator multiplication is defined by the identity $(a \times b)p = a(b(p))$ .

```
(a*b) p = a b p

              2           37       2           37
   (11)   2x  + 12x + ---- = 2x  + 12x + ----
                           3                   3
```

Operator expressions may be applied directly.

```
(a**2 - 3/4*b + c) (p + 1)

             2      44         541
   (12)   3x  + (----)x + -----
                     3          36
```

When the operator coefficients are rational, it is possible to factor.

```
factor c

             1            2        4
   (13)   (----)(D + 3) (D + 1)
            72
```

### *Differential operators with rational function coefficients*

We clear the values of most of the variables, but retain those of **PZ** and **PQ**.

```
)clear value Dx a b p e f

(Dx, a, b): LODO(QF PZ, QF PZ)


Dx := D()

   (16)   "D"

b := 3*x**2*Dx**2 + 2*Dx + 1/x

             2 2           1
   (17)   3x D  + 2D + ---
                           x

a := b*(5*x*Dx + 7)

              3 3         2         2          7
   (18)   15x D  + (51x  + 10x)D  + 29D + ---
                                               x

p: QF PZ := x**2 + 1/x**2

            4
           x  + 1
   (19)   --------
              2
             x
```

Since the operator coefficients depend on *x* the operator mulitplication is not commutative.

```
(a*b - b*a) p

                4
        - 75x  + 540x - 75
   (20) ---------------------
                  4
                 x
```

When the coefficients come from a field it is possible to define left and right division of the operators. The results of ***ldiv*** and ***rdiv*** are quotient/remainder pairs.

```
ldiv(a,b)

   (21) [5x*D + 7,0]

-- "%" means the result of the previous expression

a - (b * %.quotient + %.remainder)

   (22)  0

rdiv(a,b)

                               5
   (23) [5x*D + 7,10D + ---]
                               x

a - (%.quotient * b + %.remainder)

   (24)  0
```

The divisions allow the computation of left and right greatest common divisors via remainder sequences, and consequently the computation of left and right least common multiples.

```
e := lgcd(a,b)

           2 2            1
   (25)  3x D  + 2D + ---
                          x
```

A GCD doesn't necessarily divide a and b on both sides—here the left GCD does not divide $a$ on the right.

```
ldrem(a, e)    -- the remainder from left division

   (26)  0

rdrem(a, e)    -- the remainder from right division

                 5
   (27)  10D + ---
                 x
```

Likewise, an LCM is not necessarily divisible from both sides.

```
f := rlcm(a,b)

   (28)
                        4       3
           5 5      684x  + 80x      4
        20x D  + (----------------)D
                           3

   +
            3        2
       5832x  + 1656x  + 80x    3
      (------------------------)D
                  9

   +
                2
         3672x  + 2040x + 352    2
       (------------------------)D
                   9

   +
        172        - 28
      (-----)D + ------
        9x            2
                    9x

rdrem(f, b)

   (29)  0

ldrem(f, b)

          - 1176x + 160          312x - 80
   (30)  (---------------)D + -----------
               9x                   2
                                  9x
```

## *Differential operators with elementary function coefficients*

**Problem:** find the first few coefficients of $e^x x^{-i}$ in $L_3\,\phi$, where

$$L_3 = D^3 + \frac{G}{x^2}D + \frac{H}{x^3} - 1$$

and

$$\phi = \sum_{i=0}^{\infty} \frac{s_i e^x}{x^i}$$

We will compute the first five coefficients. We start by defining the domain **EDF** to be the elementary functions viewed as a differential ring in the variable $x$.

```
)clear value n Dx L3 phi rho num xnum

n := 5

   (32)  5

EDF := ODR(SE, EF P I, x);
```

Next we assign the differential operator $L$ and the first terms of the function $\phi$.

```
(Dx, L3): LODO(EDF, EDF)

Dx := D()

   (35)  "D"

L3 := Dx**3 + G/x**2*Dx + H/x**3 - 1

                                    3
            3      G           H - x
   (36)  D  + (----)D + --------
                    2            3
                   x            x
```

```
phi := +/[ s[i]*exp(x)/x**i for i in 0..n ]
```

$$(37)\quad \frac{(s_5 + x s_4 + x^2 s_3 + x^3 s_2 + x^4 s_1 + x^5 s_0)\exp(x)}{x^5}$$

```
rho: EF P I := L3 phi
```

(38)

$$\Big( (H + (x - 5)G - 15x^2 + 90x - 210)s_5$$
$$+ (x*H + (x^2 - 4x)G - 12x^3 + 60x^2 - 120x)s_4$$
$$+ (x^2 H + (x^3 - 3x^2)G - 9x^4 + 36x^3 - 60x^2)s_3$$
$$+ (x^3 H + (x^4 - 2x^3)G - 6x^5 + 18x^4 - 24x^3)s_2$$
$$+ (x^4 H + (x^5 - x^4)G - 3x^6 + 6x^5 - 6x^4)s_1$$
$$+ (x^5 H + x^6 G)s_0 \Big) * \exp(x) \,/\, x^8$$

We extract the numerator and remove the common factor of *exp(x)*.

```
-- ending the next line with ";" suppresses output
num: P I := rho / exp(x) * x**(n+3);
```

To collect terms with like powers, make $x$ the main variable.

```
xnum: P[x] P I := num;
```

The answer is obtained by reading off the coefficients corresponding to the powers $\frac{1}{x^0}$ to $\frac{1}{x^n}$.

```
[ coef(xnum, n+3 - i) for i in 0..n ]
```

(41)

$$[0, 0, -3s_1 + s_0 G, -6s_2 + s_1 G + 6s_1 + s_0 H,$$
$$-9s_3 + s_2 G + 18s_2 + s_1 H - s_1 G - 6s_1,$$
$$-12s_4 + s_3 G + 36s_3 + s_2 H - 2s_2 G - 24s_2]$$

## ***Differential operators with matrix coefficients acting on vectors***

Define the differential ring of 3 by 3 matrices of polynomials in $x$ over the integers and the three dimensional vector space of these polynomials.

```
)clear value Dx a b t p0 p1 p2 p q m

Mat  := SMDR(3, PZ);

Vect := DPMM(3, PZ, Mat, PZ);
```

The matrix $m$ will be used as a coefficient and the vectors $p$ and $q$ will be operated upon.

```
t: PZ := x**2;

m := zero(3,3)$Mat;

m(0,0) := t;
m(0,1) := 1;
m(1,0) := 1;
m(1,1) := t**2;
m(2,2) := 4*t;

m
```

$$(52)\quad \begin{bmatrix} x^2 & 1 & 0 \\ 1 & x^4 & 0 \\ 0 & 0 & 4x^2 \end{bmatrix}$$

```
p0: PZ := 3*x**2 + 1;
p1: PZ := 2*x;
p2: PZ := 7*x**3 + 2*x;

p: Vect := [p0, p1, p2]$V(PZ)
```

$$(56)\quad [3x^2 + 1, 2x, 7x^3 + 2x]$$

```
q: Vect := m * p
```

$$(57)\quad [3x^4 + x^2 + 2x, 2x^5 + 3x^2 + 1, 28x^5 + 8x^3]$$

The operation

"*": **(Mat, Vect) → Vect**

is defined so it makes sense to define operators with coefficients in ***Mat*** acting on members of ***Vect***

```
(Dx, a, b): LODO(Mat, Vect)

Dx := D()
```

(59) "D"

```
a := Dx + m
```

$$(60)\quad D + \begin{bmatrix} x^2 & 1 & 0 \\ 1 & x^4 & 0 \\ 0 & 0 & 4x^2 \end{bmatrix}$$

```
b := m*Dx + 1
```

$$(61)\quad \begin{bmatrix} x^2 & 1 & 0 \\ 1 & x^4 & 0 \\ 0 & 0 & 4x^2 \end{bmatrix} D + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
a * b
```

(62)

$$\begin{bmatrix} x^2 & 1 & 0 \\ 1 & x^4 & 0 \\ 0 & 0 & 4x^2 \end{bmatrix} D^2 + \begin{bmatrix} x^4 + 2x + 2 & x^4 + x^2 & 0 \\ x^4 + x^2 & x^8 + 4x^3 + 2 & 0 \\ 0 & 0 & 16x^4 + 8x + 1 \end{bmatrix} D + \begin{bmatrix} x^2 & 1 & 0 \\ 1 & x^4 & 0 \\ 0 & 0 & 4x^2 \end{bmatrix}$$

```
(a+b) (p + q)
```

(63)

$$[3x^6 + 14x^5 + 17x^4 + 22x^3 + 10x^2 + 18x + 6, \; 2x^9 + 10x^8 + 3x^6 + 10x^5 + 16x^4 + 12x^3 + 7x^2 + 18x + 6, \; 112x^7 + 560x^6 + 88x^5 + 320x^4 + 23x^3 + 53x^2 + 2x + 2]$$

**Stephen M. Watt**
**Jean Della Dora**

# Current Visitors to the Scratchpad II Group

**Martin Brock**
A recent M.S. graduate in Electrical Engineering from M.I.T.. System: 5/85-7/86

**Rüdiger Gebauer**
Wiss. Ang., Inst. f. Angewandte Mathematik, Universität Heidelberg, Im Neuenheimer Feld 294, D 6900 Heidelberg. Interface: 12/85-11/86.

**Professor Patrizia Gianni**
Dipartimento di Matematica, Universita Di Pisa, Pisa, Italy. Algebra: 1/86.

**Vladimir A. Grinberg**
Graduate Student, Columbia University, New York, New York. Interpreter: 12/85-6/86.

**Michael Lucks**
An M.S. graduate in Computer Science from the University of Hawaii. Interpreter: 2/85-9/86.

**Mohamed Mobarak**
Undergraduate student, Department of Computer Science, Cornell University, Ithaca, New York. System: 1/86.

**Professor Moss E. Sweedler**
Department of Mathematics, Cornell University, Ithaca, New York, Interface: 9/85-6/86.

# Recent Demonstrations of Scratchpad II

| | |
|---|---|
| August, 1985 | IJCAI '85, Los Angeles, California. |
| August, 1985 | AMS Summer Conference in Computational Number Theory, Humboldt State University, Arcata, California. |
| September, 1985 | CIRM Conference on Computers in Algebraic Geometry, Trento, Italy. |
| October, 1985 | NSF Workshop on Comp. in Algebra and Number Theory, University of California, Berkeley. |
| November, 1985 | Computer Science Department Chairmen's Conference, Yorktown Heights, New York. |
| December, 1985 | Future Directions in Computing, Grenoble, France. |

*For your information ...*

The **Scratchpad II** Computer Algebra System currently runs only under the IBM **VM/SP** operating system on mainframe computers. A three megabyte segment is shared among all users on a given computer, and each user requires at least a four megabyte virtual machine.

At last count, the **Scratchpad II** library consisted of 53 category constructors, 102 domain constructors and 59 package constructors.