Solving Linear Differential Equations in Maple

George Labahn
Department of Computing Science
University of Waterloo, Waterloo, Ontario, Canada
e–mail: glabahn@daisy.waterloo.edu

February 2, 2000

1 Introduction

The *dsolve* command has been a major part of Maple since 1984. In most part the algorithms used and methods supported at that time offer few surprises. A simple look at any introductory text on solving ordinary differential equations would show that most of the methods presented are entirely algebraic and hence straightforward to implement in any symbolic manipulation language. For example, in the case of first order differential equations Maple tries to determine a solution via one of separation of variables, homogenous techniques, exactness of the equation or by using integrating factors in the case of linear differential equations. For higher order linear differential equations having constant coefficients, Maple tries to find the roots of the associated characteristic equation. This provides a simple way to determine the complete set of solutions for the differential equation. Linear differential equations of Euler type are converted into an associated differential equation having only constant coefficients. Solutions of the associated equation are then transformed back into solutions of the original equation. For linear nonhomogenous equations, variation of parameters is used to compute the complete solution once the homogenous solutions are known. Other examples of well known techniques used in dsolve include solving equations of Ricatti or Bernoulli type. In addition, there are special, non-algorithmic procedures which recognize and return solutions for equations such as Bessel's or Kummer's equation. For example, we have

```
> ode := (x-1)*diff( y(x), x$2) + (3/2 - x)*diff(y(x),x) + 1/2*y(x) = 0;
```

$$ode := (x-1)\left(\frac{\partial^2}{\partial x^2}\mathrm{y}(x)\right) + \left(\frac{3}{2} - x\right)\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right) + \frac{1}{2}\mathrm{y}(x) = 0$$

```
> dsolve( ode, y(x) );
```

$$\mathrm{y}(x) = \_C1\,\sqrt{x-1} + \_C2\,\mathrm{hypergeom}\left(\left[\frac{-1}{2}\right], \left[\frac{1}{2}\right], x-1\right)$$

where $\_C1$ and $\_C2$ denote arbitrary constants.

One method that cannot be found in classical texts on solving differential equations is a relatively recent one due to J. Kovacic [Kov86]. Kovacic's algorithm is a decision procedure used for second order linear differential equations that either determines if a certain specific type of solution exists (and returns this closed form solution) or else provides a proof that no such solution exists. It has been a part of dsolve since its implementation by C. Smith [Smi84] in 1984.

In this article we describe recent improvements to dsolve that were included in MapleV, Release 3. The main focus of these improvements was to extend the power of Maple when computing closed form solutions in the case of linear ordinary differential equations. The main highlight of this work is the inclusion of two new decision procedures, due S. A. Abramov [Abr89,Abr91] and M. Bronstein [Bron92a,Bron92b], that determine if there are solutions in either the domain of coefficients or of exponential extensions of the domain of coefficients of the equation. Additional improvements, such as having the ability to return *partial* solutions of equations is also included in this work.

The remainder of this article is organized as follows. In the next section we discuss and expand on the notion of decision procedures as used in both integration and linear differential equations. Section 3 presents the algorithms of Abramov and Bronstein, while section 4 discusses the problem of returning partial, rather than complete, solutions of linear differential equations. Section 5 reports on some additional improvements in the Release 3 version of dsolve, while the final section gives future directions of work.

2 Decision Procedures

Decision procedures have been part of computer algebra systems for the past two decades. The best known example is the Risch decision procedure for indefinite integration. The fundamental theorem of integral calculus implies that one can always compute a closed form solution for an indefinite integral of a continuous function, namely the area under the graph of the function starting from a particular point. Of course this is rarely what is desired. Instead one is usually interested in answers that are given in terms of well-known functions such as exponentials or logarithms, at least in the cases where the integrand itself contains only well known functions of a similar type.

The Risch decision procedure considers the case where the integrand is an elementary function - that is, an exponential, logarithmic or algebraic function (or composition of such basic functions) and either proves that the integral cannot be computed as an elementary function or, in the case that such an elementary answer exists then computes such an answer. The basis of the procedure is Liouville's Theorem which shows that, if the integral is elementary then it must have a special form. The algorithm then attempts to either construct the special form or show that such a construction is not possible. We refer the reader to the text [GCL92] for some of the details of the Risch algorithm along with references to the original papers of Risch.

In the case of linear, second order differential equations having coefficients from a field of rational functions, Kovacic's algorithm provides a parallel to Risch's algorithm. Formally, Kovacic's algorithm determines if a second order equation of the form

$$a(x)\frac{d^2y(x)}{dx^2} + b(x)\frac{dy(x)}{dx} + c(x)y(x) = 0$$

has closed form solutions in the class of *Liouvillian* functions. Here $a(x), b(x)$ and $c(x)$ are in $C(x)$, C the field of complex numbers. Roughly, speaking a Liouvillian function is one that can be built up from the ground field via elementary functions (logs, algebraics or exponentials) and integrals of elementary functions. Formally, we have

Definition. Let K be a differential field with derivation given by $'$. A field F is a Liouvillian extension of K if there is a tower of fields

$$K = K_0 \subset K_1 \subset \cdots \subset K_m = F$$

where each K_{i+1} is a simple extension $K_i(\theta_i)$ of K_i, such that one of the following holds:

a) θ_i is algebraic over K_i,
b) $\theta_i' \in K_i$ (that is K_i has been extended by an integral),
c) $\frac{\theta_i'}{\theta_i} \in K_i$ (that is K_i has been extended by the exponential of an integral).

A function contained in a Liouvillian extension of K is called a Liouvillian function over K.

Clearly all the elementary functions are included - algebraics in a), logs in b) and exponentials in c). The error function erf and the exponential integral Ei are examples of non-elementary Liouvillian functions since

$$erf(x) = \frac{2}{\sqrt{\pi}} \int e^{-x^2} dx \quad \text{and} \quad Ei(n,x) = -\int \frac{e^{-x}}{x^n} dx,$$

while Bessel functions are examples of functions that are not Liouvillian.

As an example of the use of Kovacic's algorithm, the equation

```
> diff(y(x),x$2) -(x^6-2*x^5+3*x^4+x^3+7/4*x^2-5*x+1)/x^4*y(x)=0;
```

$$\left(\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x)\right) - \frac{\left(x^6 - 2\,x^5 + 3\,x^4 + x^3 + \frac{7}{4}\,x^2 - 5\,x + 1\right)\mathrm{y}(x)}{x^4} = 0$$

```
> dsolve(",y(x));
```

$$\mathrm{y}(x) = \_C1\,\mathrm{e}^{(1/2\,\%1)} + \_C2\,\mathrm{e}^{(1/2\,\%1)} \int \mathrm{e}^{(-\%1)}\,dx$$

$$\%1 := \frac{x^3 - 2\,x^2 - 2 - 3\ln(x)\,x + 2\ln(x-1)\,x + 2\ln(x+1)\,x}{x}$$

computes two Liouvillian solutions, built up in this case from the ground field $K = Q(x)$ by taking exponentials, logs and integrals, while

```
> de := diff(y(x),x$2) + (x^4+1)*y(x)=0;
```

$$de := \left(\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x)\right) + (x^4 + 1)\,\mathrm{y}(x) = 0$$

```
> dsolve(de,y(x));
```

$$\mathrm{y}(x) = \mathrm{DESol}\left(\left\{\left(\frac{\partial^2}{\partial x^2}\,\_\mathrm{Y}(x)\right) + (x^4 + 1)\,\_\mathrm{Y}(x)\right\}, \{\_\mathrm{Y}(x)\}\right)$$

provides a proof that no such Liouvillian solutions exist for the equation de. The DESol structure given in the above answer can be viewed as a type of RootOf for differential equations.

Kovacic's algorithm is usually stated for linear differential equations of the form

$$\frac{d^2y(x)}{dx^2} - r(x)y(x) = 0.$$

This does not reduce its generality since equations in the general form are easily translated into one of the above form. The basis for Kovacic's algorithm is the observation that, if there are Liouvillian solutions to the above equation, then they must take one of 3 possible forms:

a) The differential equation has a solution of the form $y(x) = e^{\int w}$ where $w \in C(x)$

b) The differential equation has a solution of the form $y(x) = e^{\int w}$ where w is algebraic of degree 2 over $C(x)$ and case 1 does not hold,

c) All solutions of the differential equation are algebraic over $C(x)$ and cases 1 and 2 do not hold. In this case, the solutions are of the form $y(x) = e^{\int w}$ with w algebraic of degree 4, 6, or 12 over $C(x)$.

If there are no solutions of the form a) - c) then there are no Liouvillian solutions. The result is actually not that difficult to understand once one has enough mathematical background. One can consider the group of all differential automorphisms of the differential equation leaving the ground field $C(x)$ invariant (called the differential Galois group of the equation - similar to the Galois group of an algebraic equation) . This differential Galois group can be viewed as a subgroup of the special linear group $SL(2, C)$ of nonsingular matrices having determinant equal to 1. All such subgroups are already known. By taking advantage of this classification one can determine that the only Liouvillian solutions take on certain forms (since these solutions determine the structure of the group of available automorphisms of the equation). The decision procedure itself relies on the fact that, if a Liouvillian solution does exist, then it has a certain form (specified in a) - c)) and so proceeds to either construct one of these answers or show that no such construction can occur. We refer the reader to [Kov86] or [Smi84] for further details (and to [SU92a,SU92b] for the order 3 case).

3 Higher Order Decision Procedures

There are algorithms, due to M. Singer [Sin81,Sin91], which give decision procedures for determining Liouvillian solutions of n-th order linear differential equations

$$a_n(x) \cdot \frac{d^ny(x)}{dx^n} + \cdots + a_1(x) \cdot \frac{dy(x)}{dx} + a_0(x)y(x) = b(x)$$

with $a_i(x), b(x) \in C(x)$ and n greater than 2. However, these algorithms have not been found practical for implementation and indeed have never been completely implemented. What is practical and useful is an implementation of the main subalgorithm of these algorithms, namely a decision procedure for determining rational and exponential solutions over the field of coefficients of the equation.

The RATional Linear Ordinary Differential Equation decision procedure (RATLODE), for solving linear differential equations having rational function coefficients has been included in Maple for release 3. This procedure, due to Abramov and Bronstein, determines if any closed form solutions exist in the domain of coefficients of the equation. If this is the case then the closed form solution(s) are returned. Otherwise, the algorithm provides a proof that no such solution exists. This is accomplished by returning a DESol structure. Thus in the example

```
> ode := diff(y(x),x$3)  + x*diff(y(x),x) + y(x)=0;
```

$$ode := \left(\frac{\partial^3}{\partial x^3}\mathrm{y}(x)\right) + x\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right) + \mathrm{y}(x) = 0$$

```
> dsolve( ode , y(x) );
```

$$\mathrm{y}(x) = \mathrm{DESol}\left(\left\{\left(\frac{\partial^3}{\partial x^3}\_\mathrm{Y}(x)\right) + x\left(\frac{\partial}{\partial x}\_\mathrm{Y}(x)\right) + \_\mathrm{Y}(x)\right\}, \{\_\mathrm{Y}(x)\}\right)$$

the RATLODE procedure has determined that there are no solutions of the above equation from the field of rational functions Q(x).

The EXPonential Linear Ordinary Differential Equation decision procedure (EXPLODE) of M. Bronstein [Bron92b] for linear differential equations having rational function coefficients has also been implemented for release 3. This procedure determines if any closed form solutions exist that are exponentials of integrals of functions from the domain of coefficients of the equation. As with the RATLODE procedure, either the closed form solution(s) are returned or the algorithm provides a proof that no such solution exists (again, by returning a DESol structure). For example,

```
> ode := (-6+8*x^2)*y(x)+(11+4*x-12*x^2)*diff(y(x),x)+(-6-6*x+4*x^2)*
> diff(y(x),x$2)+(1+2*x)*diff(y(x),x$3)=0;
```

$$ode := (-6 + 8x^2)\,\mathrm{y}(x) + (11 + 4x - 12x^2)\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right)$$
$$+(-6 - 6x + 4x^2)\left(\frac{\partial^2}{\partial x^2}\mathrm{y}(x)\right) + (1 + 2x)\left(\frac{\partial^3}{\partial x^3}\mathrm{y}(x)\right) = 0$$

```
> dsolve(ode,y(x));
```

$$\mathrm{y}(x) = \_C1\,\mathrm{e}^x + \_C2\,\mathrm{e}^{(2x)} + \_C3\,\mathrm{e}^x\,\mathrm{erf}(x)$$

finds two independent solutions in the domain $Q(x, e^x, e^{2x})$, while using reduction of order to determine the last independent solution.

Similarly, in the equation below, EXPLODE finds a single solution in $Q(x, e^x)$ while reduction of order determines the two remaining independent solutions. Notice that one of the solutions is given in an "incomplete" closed form because there is no satisfactory closed form solution to the integral.

```
> (3+6*x+20*x^2-40*x^3+16*x^4-32*x^5)*y(x)+
> (-3-15*x+44*x^3+48*x^5)*diff(y(x),x)+(9*x-26*x^2-24*x^4-16*x^5)*
> diff(y(x),x$2)+(6*x^2-4*x^3+8*x^4)*diff(y(x),x$3)=0;
```

$$(3+6x+20x^2-40x^3+16x^4-32x^5)\,\mathrm{y}(x)+(-3-15x+44x^3+48x^5)\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right)$$
$$+(9x-26x^2-24x^4-16x^5)\left(\frac{\partial^2}{\partial x^2}\mathrm{y}(x)\right)+(6x^2-4x^3+8x^4)\left(\frac{\partial^3}{\partial x^3}\mathrm{y}(x)\right)$$

```
> dsolve(",y(x));
```

$$\mathrm{y}(x)=\_C1\,\mathrm{e}^x+\_C2\,\mathrm{e}^x\,\mathrm{Ei}(1,-x)+\_C3\,\mathrm{e}^x\int \mathrm{e}^{(x^2)}\sqrt{x}\,dx$$

As with the previously mentioned decision procedures, the algorithms rely on the fact that if a particular rational or exponential solution does exist then it must have a certain form. The algorithms then either construct such a form or prove that the form of answer cannot exist. In the case of exponential solutions, the problem is converted into determining rational function solutions of an associated Ricatti equation. For example, determining an exponential solution $e^{\int w}$ of the equation

$$\frac{d^2y(x)}{dx^2}-r(x)y(x)=0$$

is equivalent to finding a rational solution w of the Ricatti equation

$$\frac{dy(x)}{dx}+y(x)^2=r(x).$$

4 Partial Solutions

It is a common occurrence that various algorithms (including those given above) succeed in finding some independent solutions of a linear differential equation, but not enough to provide a complete description of all such solutions. In this case, Maple now has the ability to return answers in terms of solutions of "reduced" equations of lower orders in addition to any closed forms. In previous versions, Maple either found a complete set of solutions or returned *null*, denoting that the complete set could not be found. For example, in the equation below, RATLODE finds a particular solution while the complete solution requires solutions of the corresponding homogeneous equation (which Maple cannot determine a closed form for). This gives

```
> ode := diff(y(x),x$3) + x*diff(y(x),x) + y(x) = (-5+2*x+x^2)/(x+1)^4;
```

$$ode := \left(\frac{\partial^3}{\partial x^3}\mathrm{y}(x)\right)+x\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right)+\mathrm{y}(x)=\frac{-5+2x+x^2}{(x+1)^4}$$

```
> dsolve(ode,y(x));
```

$$\mathrm{y}(x) = \frac{1}{x+1} + \mathrm{DESol}\left(\left\{\left(\frac{\partial^3}{\partial x^3}\_\mathrm{Y}(x)\right) + x\left(\frac{\partial}{\partial x}\_\mathrm{Y}(x)\right) + \_\mathrm{Y}(x)\right\}, \{\_\mathrm{Y}(x)\}\right)$$

As a second example, for the homogeneous equation below EXPLODE finds a single solution with the rest of the answer determined via reduction of order. In this case, there are no closed form solutions known to Maple and so an expression involving solutions to lower order equations is returned.

```
> diff(y(x),x$3) +x*diff(y(x),x$2) +(x^2+3)*diff(y(x),x) +(x^3 + x)*y(x)=0;
```

$$\left(\frac{\partial^3}{\partial x^3}\mathrm{y}(x)\right) + x\left(\frac{\partial^2}{\partial x^2}\mathrm{y}(x)\right) + (x^2+3)\left(\frac{\partial}{\partial x}\mathrm{y}(x)\right) + (x^3+x)\,\mathrm{y}(x) = 0$$

```
> dsolve( " , y(x) );
```

$$\mathrm{y}(x) = \_C1\,\mathrm{e}^{(-1/2\,x^2)} + \mathrm{e}^{(-1/2\,x^2)}\int \mathrm{DESol}\left(\left\{\left(\frac{\partial^2}{\partial x^2}\_\mathrm{Y}(x)\right) - 2\,x\left(\frac{\partial}{\partial x}\_\mathrm{Y}(x)\right) + 2\,x^2\,\_\mathrm{Y}(x)\right\}, \{\_\mathrm{Y}(x)\}\right) dx$$

In the examples above the DESol structure is used as a placeholder for the part of the solutions that remains incomplete. In some cases, one can use some of Maple's other computational tools to further investigate the space of solutions. For example, in the above result, one can obtain a series expansion of the remaining solutions via

```
> rest := op(2,rhs(")):
> series(rest,x,4);
```

$$\_\mathrm{Y}(0)\,x + \frac{1}{2}\mathrm{D}(\_Y)(0)\,x^2 - \frac{1}{2}\_\mathrm{Y}(0)\,x^3 + \mathrm{O}(x^4)$$

5 Additional Improvements

The implementation of two new decision procedures and allowing for partial solutions in the output of dsolve were the two major improvements from the work that went into the dsolve command of MapleV, Release 3. However, there were additional improvements that were also added for this release. Some of these occurred simply by taking better advantage of existing tools that had been incorporated into Maple during the last 10 years. It is not surprising that such improvements were possible, given that the dsolve code itself had not undergone any major changes since 1984.

By using Maple's *RootOf* construct, it is possible to work with the roots of polynomials even if the roots themselves are not explicitly known. By taking advantage of this tool, we can let Maple return solutions for constant coefficient linear differential equations, even when the associated characteristic equation cannot be factored. For example, we have

```
> ode := diff(y(x),x$5 ) + 2*diff(y(x),x) + 2*y(x) = 0;
```

$$ode := \left(\frac{\partial^5}{\partial x^5}\,\mathrm{y}(x)\right) + 2\left(\frac{\partial}{\partial x}\,\mathrm{y}(x)\right) + 2\,\mathrm{y}(x) = 0$$

```
> dsolve(ode,y(x));
```

$$\mathrm{y}(x) = \sum_{\_R=\mathrm{RootOf}(2+2\,\_Z+\_Z^5)} \_C1_{\_R}\,\mathrm{e}^{(\_R\,x)}$$

This also works in the case of repeated roots, and can also show up as answers to partial components of solutions

```
> diff(y(x),x$5)+4*diff(y(x),x$3)+4*diff(y(x),x$2)+4*diff(y(x),x)+8*y(x)=0;
```

$$\left(\frac{\partial^5}{\partial x^5}\,\mathrm{y}(x)\right) + 4\left(\frac{\partial^3}{\partial x^3}\,\mathrm{y}(x)\right) + 4\left(\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x)\right) + 4\left(\frac{\partial}{\partial x}\,\mathrm{y}(x)\right) + 8\,\mathrm{y}(x) = 0$$

```
> dsolve(",y(x));
```

$$\mathrm{y}(x) = \_C1\,\cos\left(\sqrt{2}\,x\right) + \_C2\,\sin\left(\sqrt{2}\,x\right) + \sum_{\_R=\mathrm{RootOf}(\_Z^3+2\,\_Z+4)} \_C3_{\_R}\,\mathrm{e}^{(\_R\,x)}$$

Since equations of Euler type are transformed into differential equations with constant coefficients, these improvements also allow for a greater set of solutions for these equations. For example,

```
> ode := x^6*diff(y(x),x$6)+15*x^5*diff(y(x),x$5)+69*x^4*diff(y(x),x$4)+
> 118*x^3*diff(y(x),x$3)+75*x^2*diff(y(x),x$2)+21*x*diff(y(x),x)+4*y(x)=0;
```

$$x^6\left(\frac{\partial^6}{\partial x^6}\,\mathrm{y}(x)\right) + 15\,x^5\left(\frac{\partial^5}{\partial x^5}\,\mathrm{y}(x)\right) + 69\,x^4\left(\frac{\partial^4}{\partial x^4}\,\mathrm{y}(x)\right) + 118\,x^3\left(\frac{\partial^3}{\partial x^3}\,\mathrm{y}(x)\right) + 75\,x^2\,\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x) + 21\,x\left(\frac{\partial}{\partial x}\,\mathrm{y}(x)\right) + 4\,\mathrm{y}(x) = 0$$

```
> dsolve(ode,y(x) );
```

$$\mathrm{y}(x) = \sum_{\_R=\mathrm{RootOf}(\_Z^3+2\,\_Z+2\,)} \_C1_{\_R}\,x^{\_R} + \sum_{\_R=\mathrm{RootOf}(\_Z^3+2\,\_Z+2\,)} \_C2_{\_R}\,x^{\_R}\ln(x)$$

A more significant improvement is that dsolve now takes advantage of a vastly improved use of reduction of order. For example, combining EXPLODE with reduction of order and Bessel's equation gives:

```
> ode := (-2*x^2+x+n^2)*y(x)+(4*x^2-2*x-n^2)*diff(y(x),x)+
>        (-3*x^2+x)*diff(y(x),x$2)+x^2*diff(y(x),x$3) =  0;
```

$$ode := (-2\,x^2 + x + n^2)\,\mathrm{y}(x) + (4\,x^2 - 2\,x - n^2)\left(\frac{\partial}{\partial x}\,\mathrm{y}(x)\right) + (-3\,x^2 + x)\left(\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x)\right) + x^2\left(\frac{\partial^3}{\partial x^3}\,\mathrm{y}(x)\right) = 0$$

```
> dsolve( ode,y(x) );
```

$$y(x) = \_C1\, e^x + \_C2\, e^x \int \mathrm{BesselJ}(n, x)\, dx + \_C3\, e^x \int \mathrm{BesselY}(n, x)\, dx$$

This in turn can be further solved in some special cases of n - for example when n = 1 we have:

```
> dsolve( subs( n = 1, ode) ,y(x) );
```

$$y(x) = \_C1\, e^x + \_C2\, e^x \mathrm{BesselJ}(0, x) + \_C3\, e^x \mathrm{BesselY}(0, x)$$

In the example below we use the RATLODE algorithm for finding a particular solution of the linear differential equation and special function solvers to determine the homogenous solution.

```
> ode := diff(y(x),x$2) + 3/x*diff(y(x),x) + (x^2-143)/x^2*y(x) = x -140/x;
```

$$ode := \left(\frac{\partial^2}{\partial x^2} y(x)\right) + \frac{3}{x}\frac{\partial}{\partial x} y(x) + \frac{(x^2 - 143)\, y(x)}{x^2} = x - 140\,\frac{1}{x}$$

```
> dsolve( ode,y(x) );
```

$$y(x) = x + \frac{\_C1\, \mathrm{BesselJ}(12, x)}{x} + \frac{\_C2\, \mathrm{BesselY}(12, x)}{x}$$

Improvements in the solving of linear differential equations also leads to improvements in determining solutions of other types of equations whose algorithms have a component that involves solving some associated linear differential equation. For example, we can now return a solution to the nonlinear differential equation

```
> ode := diff(y(x),x)+x*y(x)^2 = 1;
```

$$ode := \left(\frac{\partial}{\partial x} y(x)\right) + x\, y(x)^2 = 1$$

```
> dsolve( ode, y(x) );
```

$$y(x) = -\frac{\_C1\, \mathrm{BesselK}\left(\frac{-1}{3}, \frac{2}{3}\, x^{3/2}\right) - \mathrm{BesselI}\left(\frac{-1}{3}, \frac{2}{3}\, x^{3/2}\right)}{\sqrt{x}\left(\_C1\, \mathrm{BesselK}\left(\frac{2}{3}, \frac{2}{3}\, x^{3/2}\right) + \mathrm{BesselI}\left(\frac{2}{3}, \frac{2}{3}\, x^{3/2}\right)\right)}$$

6 Future Directions

The recent work on dsolve has not provided any improvements for solving *systems* of linear differential equations. At present there are three ways that such systems can be solved - the default method using differential operators, laplace transform method or by converting to first order and solving via the matrix exponential method. However, in all these cases the system of linear equations can only be solved if the coefficients are constant with respect to the variable of differentiation. We plan on extending Maple's capabilities in this area to include determining solutions in the case where the coefficients are rational functions. This will be done via converting these linear systems into a matrix first order system of differential equations and then computing a companion block diagonal form of such a differential operator as done in for example [Bar93,Zür94]. This reduces the problem to determining solutions for a number of independent linear differential equations, each having rational functions for coefficients.

The decision procedures discussed in Section 3 work only in the case of linear differential equations having coefficients that are rational functions. Recent work by M. Mohrenschildt [Moh94] shows how such a decision procedure can also extend to the case where the coefficients include *discontinuous* or *piecewise* functions (for example differential equations having *abs* or *Heaviside* functions as part of their coefficient set). We plan on implementing and extending this work in future versions.

At present the dsolve command has the problem that it appears to most users as a black box. It takes in differential equations as input and returns closed form solutions, if possible, as output. There is little control over how such closed form solutions are determined. In many cases, however, a user would prefer to solve these equations using alternate methods. In these cases the user would simply like some tools (such as say a user level reduction of order or variation of parameters command) to determine some or all of the other solutions of a particular equation. One of the major projects that are planned for future versions of Maple is a comprehensive set of tools for solving differential equations. Examples of such tools include such simple operations such as reduction of order or computing the indicial equation, to more sophisticated tools that allow for algebraic manipulation of differential operators. The latter set of tools takes advantage of the fact that the algebra of such operators has already been formalized by Ore [Ore33]. The algebraic operations include algorithms to find right or left greatest common divisors and even right or left factors of linear differential operators.

References

[Abr89] S.A. Abramov, Rational solutions of linear differential and difference equations with polynomial coefficients, USSR Comput. Maths. Math. Phys. 29, 7-12 (translated from Zh. vychisl. mat. mat. fiz. 29, 1611-1620) (1989)

[Abr91] S.A. Abramov and K.Yu. Kvashenko, Fast Algorithms to Search for the Rational Solutions of Linear Differential Equations with Polynomial Coefficients, Proc. of ISSAC'91, Bonn, pp. 267-270, (1991)

[Bar93] M.A. Barkatou, An Algorithm for Computing a Companion Bock Diagonal Form for a System of Linear Differential Equations, AAECC, 4, pp. 185-195, (1993)

[Bron92a] M. Bronstein, On Solutions of Linear Differential Equations in their Coefficient Field. J. of Symbolic Computation, 13, pp. 413-439, (1992)

[Bron92b] M. Bronstein, Linear Ordinary Differential Equations: breaking through the order 2 barrier. Proc. of ISSAC'92 (P. Wang ed.), pp. 42-48, (1992)

[GCL92] K.O. Geddes, S. Czapor and G. Labahn, Algorithms for Computer Algebra, Kluwer Academic Publishers, (1992) [Kov86] J. Kovacic, An Algorithm for Solving Second Order Linear Homogenous Differential Equations. J. of Symbolic Computation, 2, pp. 3-43, (1986)

[Moh94] M. von Mohrenschildt, Symbolic Solutions of Discontinuous Differential Equations, Ph.D Thesis, Computer Science , ETH, Zürich, (1994)

[Ore33] O. Ore, Theory of Non-commutative Polynomials. Annals of Math., 34, (1933)

[Sin81] M. Singer, Liouvillian Solutions of n-th order Homogenous Linear Differential Equations, American Journal of Mathematics, 103, pp. 661-682, (1981)

[Sin91] M. Singer, Liouvillian Solutions of Linear Differential Equations with Liouvillian Coefficients, J. of Symbolic Computation, 11, pp. 251-273, (1991)

[SU92a] M. Singer and F. Ulmer, Galois Groups of Second and Third Order Linear Differential Equations, J. of Symbolic Computation, 16, pp. 9-36, (1992)

[SU92b] M. Singer and F. Ulmer, Liouvillian and Algebraic Solutions of Second and Third Order Linear Differential Equations, J. of Symbolic Computation, 16, pp. 37-73, (1992)

[Smi84] C. Smith, A Discussion and Implementation of Kovacic's Algorithm for Ordinary Differential Equations. Tech. Report #CS 84-35, Department of Computer Science, Univ. of Waterloo, Waterloo, (1984)

[Zür94] B. Zürcher, Rationale Normalform von pseudo-linearen Abbildungen, Diplomarbeit, Mathematics, ETH Zürich