Algorithms of Computer Analysis for Approximate Solutions of Linear ODEs with Polynomial Coefficients

K. Hantzschmann and O. Becken

University of Rostock

1 Adaptive Approximation Method

Algorithms in Computer Algebra base on algebraic concepts and aim at finding exact solutions. Computer Analysis gives priority to algorithms using computer algebra systems (CAS) for finding controlled analytic approximate solutions for non closed solvable or only with not acceptable expense solvable problems. In this sense Computer Analysis is the pendant to Numerical Mathematics. The analytic approximate solutions should reflect inherent properties of the given problem and the possible influence of parameters. All formula expressions should be simple and transparent, while maintaining an appropriate level of precision. Approximate solutions without adequate error statements are worthless for the practical application. Therefore it is important to control the accuracy by error estimations, which are computed entirely by the computer program, the same way as approximations are determined without requiring user interaction. Efficiency and quality of the algorithms developed in the Computer Analysis depend in a high degree on the possibility to combine the tried methods of numerical computing with symbolic procedures. Many algorithms have their roots in the past and are only now practicable because the fast development of hardware and software technology gives the possibility to implement and improve them.

Finding analytic approximate solutions for differential equations is one of the main tasks of the Computer Analysis. In the field of ODEs several algorithms could be developed and tried in the last two decades [9, 10, 11, 13, 15]. In accordance with the purpose of Computer Analysis preference is given to mathematical approximation methods allowing an automatic adaption to a given problem and giving a transparent short formula expression at a reasonable expense.

Good results on this way could be achieved by a two - step concept. In the first step the given problem will be adapted by an suitable chosen "neighbor" problem whose exact solution can be determined by means of Computer Algebra algorithms. The fundamental solutions form the base for an adequate approximate ansatz in the second step.

Basic concept of the adaptive approximation [10, 14]:
Let be given a (linear or nonlinear) ODE of order n

$$L(y) = 0$$

with n linearly independent additional constraints (initial or boundary conditions)

$$U_k(y) = r_k \quad (k = 1(1)n)$$

In the first step (adaption step) $L(y)$ is replaced by a differential operator $\tilde{L}(y; a_0, ..., a_m)$ of sufficiently high order $q > n$ with free parameters $a_i (i = 0(1)m)$, for which a general method for constructing the exact solutions is known. For a good efficiency should be recommended to choose $\tilde{L}$ in the form

$$\tilde{L}(y) = \sum_{i=0}^{m} a_i L_i^{n_i}(y),$$

where $L_i^{n_i} (i = 0(1)n)$ are differential operators of the order $n_i \leq q$. The parameters $a_i (i = 0(1)m)$ are computable by several "adaption criterions". Then it is possible to determine q linearly independent fundamental solutions $\{\varphi_1, \varphi_2, ..., \varphi_q\}$ of the adapted task with

$$\tilde{L}(\varphi_i) = 0 \quad (i = 1(1)q).$$

These allow a linear ansatz

$$\tilde{y} = \sum_{i=1}^{q} c_i \varphi_i.$$

The parameter $c_i (i = 1(1)k)$ are computable by several "approximation criterions" according to the given problem under consideration of the additional conditions (approximation step). $\tilde{L}$ allows to adapt the system of ansatz functions to the qualitative attributes of the solution (e. g. polynomial, rational functions, exponential functions, trigonometric functions,...). Dividing the computation of approximate solutions $\tilde{y}$ into two steps has several advantages. This concept allows a close problem adaption and offers a wide field for the development of several algorithms depending first on the approximation criterions used in both steps and second on the selected family of "neighbouring" problems. The user has a double possibility to control the accuracy of the approximate process.

2 ODEs with Polynominals Coefficients

A modification of this strategy results in an algorithm for the computation of approximate solutions for the class of linear ordinary differential equations with polynomial coefficients. Based on efficient algorithms for the finding exponential solutions or some subclasses of such solutions a relatively wide class of adapted differential equations has been made available which may serve a basis for approximations. This class consists of linear differential equations with polynomial coefficients, the relevant differential operators of which are completely factorizable by differential operators of first order with exponential solutions.
The following procedure was developed by O. Becken. He defended his PhD-thesis to the topic "On adaptive approximation and D-finite functions" [3] two years ago.
Topic of the investigation is linear homogeneous ordinary differential equations of order n (singular or regular) with polynomial coefficients

$$L(y) = \sum_{i=0}^{n} q_i y^{(i)}(x) = 0 \qquad (2.1)$$

$$\text{with} \quad q_i \in \mathbb{K}[x], \quad (i = 0(1)n)$$

$$q_n \not\equiv 0, \quad gcd(q_0, q_1, ..., q_n) = 1.$$

$\mathbb{K}$ is some constant field with characteristic 0. Becken calls the solutions differentiably finite functions (D-finite functions). The class of D-finite functions is an scientifically very interesting class and important for many practical applications. It contains the commonly used analytic functions: the algebraic, Bessel, cosine, Gaussian error, exponential, hypergeometric, logarithm, power, rational, sine and many more functions are D-finite. D-finite functions form an algebra which is closed under sum and product, substitution of algebraic functions, differentiation and integration. Each function can be uniquely defined by an ODE (2.1) and n linearly independent linear boundary constraints

$$U_{x_0} Y(x_0) + U_{x_1} Y(x_1) = v \qquad (2.2)$$

$$\text{with} \quad x_0 \in \mathbb{R} \bigcap \mathbb{K}, \quad x_0 < x_1, \quad Y = (y, y', ..., y^{(n-1)})^T$$

$$U_{x_0}, U_{x_1} \in (\mathbb{C} \bigcap \mathbb{K})^{n \times n}$$

$$v = (v_0, v_1, ..., v_{n-1})^T, \quad v_i \in \mathbb{C} \bigcap \mathbb{K}$$

or by a homogenous linear recurrence equation (RE) with polynomial coefficients

$$Ra_m = \sum_{j=0}^{r} p_j F^j a_m = 0 \qquad (2.3)$$

$$\text{with} \quad Fa_m = a_{m-1}, \quad p_j \in \mathbb{K}[m], \quad (j = 0(1)r)$$

$$p_0 \not\equiv 0, p_r \not\equiv 0.$$

It can be shown, that the generating function $y(x) = \sum_{i \in Z} a_i x^i$ of such a RE is D-finite and that reciprocally the sequence of Taylor coefficients of a D-finite function is solution of a RE [18].

3 A Class of Adaptive Differential Operators

In preparation the adaption step it is necessary to choose the class of adaptive differential operators. These operators should have the following properties:

- There should be a practicable algorithm to compute the fundamental system for each differential operator.
- The fundamental system of each differential operator should consist of functions which are elementary over $\mathbb{K}(x)$, i. e. they belong to an elementary extension of $\mathbb{K}(x)$ in the sense of differential algebra. It is clear, non-elementary Liouvillian solutions are not practicable enough.
- The adaptive differential operators should be a subset of the linear differential operators with polynomial coefficients.

These demands can be realized by the set of differential operators $\mathbb{L}$ which

- are linear differential operators of order n with polynomial coefficients
- are completely first order decomposable over $\mathbb{K}(x)$, i. e. $\tilde{L}$ can be written as $\tilde{L} = L_n ... L_2 L_1$ where $L_i (i = 1(1)n)$ are linear differential operators of first order with coefficients in $\mathbb{K}'(x)$ ($\mathbb{K}'$ is algebraic closure of $\mathbb{K}$)
- have for $\tilde{L}(y) = 0$ only elementary solutions over $\mathbb{K}(x)$.

The question, whether a given first order decomposable differential operator has only elementary solutions over $\mathbb{K}(x)$, can algorithmically be decided by Risch integration. The class of differential operators defined in this way is the biggest class of adaptive differential operators. In the past, some other classes were already used [10, 14]:

$$POL = \{\mathbb{L} \mid V(\tilde{L}) \subset \mathbb{K}'[x]\}$$
$$RAT = \{\mathbb{L} \mid V(\tilde{L}) \subset \mathbb{K}'[x]\}$$
$$CON = \{\tilde{L} \mid \exists c_1, ..., c_n \in: \mathbb{K}' : \tilde{L} = \sum_{i=0}^{n} c_i D^i\}$$
$$EUL = \{\tilde{L} \mid \exists c_1, ..., c_n, x_0 \in \mathbb{K}' : \tilde{L} = \sum_{i=0}^{n} c_i (x - x_0)^i D^i\}$$

$$(V(\tilde{L}) \quad \text{set of all solutions of} \quad \tilde{L}(y) = 0, Dy(x) = y'(x)).$$

POL, RAT, CON and EUL are proper subclasses of the class $\mathbb{L}$ of adaptive differential operators fixed above. For instance, each Eulerian differential operator E $\in$ EUL at the point x_0 factors into

$$E = ((x - x_0)D + e_n ... ((x - x_0)D + e_2)((x - x_0)D + e_1)$$

where the $e_i \in \mathbb{K}' \quad (i = 1(1)n)$.

4 Adaption Step and Approximation Step

4.1 Getting Candidates for Approximate Solutions

After defining the class of adaptive differential operators $\mathbb{L}$ it is necessary to determine candidates for approximate solutions in $\mathbb{L}$ or subclasses of $\mathbb{L}$. This is possible by means of structure theorems of Computer Algebra [3]:

Lemma 1: Each ODE (2.1) at any point $x_0 \in \mathbb{K}'$ can be written in a normalized standard form

$$L(y(x)) = \sum_{i=0}^{n} \psi_i (x - x_0)^i y^{(i)}(x) = 0$$

$$\text{with} \quad \psi_i = \sum_{j=0}^{b} c_{ij} (x - x_0)^j$$

$$\gcd(\psi_0, \psi_1 ..., \psi_n) = 1, x_0, c_{ij} \in \mathbb{K}', \exists i : c_{ib} \neq 0.$$

This standard form is the base for classifying the singular points at x_0 : regular, singular, regular singular, irregular singular.

Lemma 2: If the ODE (2.1) is regular or regular singular at the point x_0, then there exist n linearly independent solutions of the form

$$y(x) = (x - x_0)^\lambda (t_0 + t_1 \ln(x - x_0) + ... + t_{n-1} \ln(x - x_0)^{n-1}), \quad (\lambda \in \mathbb{K}),$$

where the $t_i \in \mathbb{K}'[[x - x_0]], (1 = 0(1)n - 1)$ are formal Taylor series at the point x_0.

If the ODE (2.1) is irregular singular at the point x_0, then there exist n linearly independent solutions of the form

$$y(x) = (x - x_0)^\lambda (t_0 + t_1 \ln(x - x_0) + ... + t_{n-1} \ln(x - x_0)^{n-1}) \exp(q)$$
$$\lambda \in \mathbb{K}', ri \in \mathbb{N} \setminus \{0\}, w : \mathbb{C} \to \mathbb{C}, w(x)^{ri} = x - x_0, q \in \mathbb{K}'[1/w(x)]$$

where the $t_i \in \mathbb{K}'[[w(x)]] \quad (i = 0(1)n - 1)$ are formal Puiseux series

$$t_i(x) = \sum_{j=0}^{\infty} a_j w(x)^j \quad (a_j \in \mathbb{K}')$$

at the point x_0.

Corollary: For each ODE of the form (2.1) at any regular or regular singular point $x_0 \in \mathbb{K}'$ there exists at least one extended formal Laurent series solution of the form

$$y(x) = (x - x_0)^\lambda \sum_{i=-\infty}^{\infty} a_i (x - x_0)^i \quad (a_i, \lambda \in \mathbb{K}') \tag{4.1}$$

It is helpful and efficient to use corresponding recurrence equations for the

development of algorithms aimed at finding exact solutions or candidates for approximate solutions of ODEs (2.1).

Lemma 3: In each regular or regular singular point a ODE of the form (2.1) corresponds to a RE (2.3) at the point x_0 of the form

$$R_{x0} a_m = \sum_{j=0}^{b} p_j F^j a_m = \sum_{j=0}^{b} [\sum_{i=0}^{n} c_{ij} i! \binom{r-j}{i}] a_{m-j}$$

with

$$F a_m = a_{m-1}, m \in \mathbb{Z}, a_m, \lambda \in \mathbb{K}', r = n + \lambda$$
$$p_j \in \mathbb{K}[r], \quad p_0 \not\equiv 0, p_b \not\equiv 0$$

Lemma 4: If there exists a solution of the form

$$y(x) = (x - x_0)^\lambda \sum_{i=\alpha}^{\beta} a_i (x - x_0)^i, \quad \alpha, \beta \in \mathbb{Z}, x_0, a_i, \lambda \in \mathbb{K}', a_\alpha \neq 0, \alpha_\beta \neq 0$$

for a ODE of the form (2.1), then the following condition holds for the corresponding RE at the point x_0:

$$p_0(\lambda + \alpha) = 0 \quad \text{and} \quad p_b(\lambda + \beta + b) = 0.$$

O. Becken has analysed several algorithms with reference to usability for finding exponential solutions of differential equations with polynomial coefficients [2, 3, 4]. He suggested an algorithm based on the original algorithm of Beke with including results from Abramov, Bronstein, Schwarz and Singer [1, 6, 7, 16, 17].
The algorithm of Beke [5] finds all exact solutions of the form

$$y(x) = exp(\int u(x)dx) \quad \text{with} \quad u \in \mathbb{K}'(x) \tag{4.2}$$

(exponential function over $\mathbb{K}'$). Let $\mathbb{K}_b(x) \subset \mathbb{K}'(x)$ be a differential field with the minimal number of algebraic extensions $\Theta_1, \Theta_2, ..., \Theta_s \in \mathbb{K}'$ such that $u \in \mathbb{K}_b(x)$. Using Hermite reduction and the Rothstein/Trager method y can be transformed into

$$y(x) = s \quad \exp(p + \frac{c}{d}) \prod_i r_i^{c_i}$$

$$\text{with } p, c, d \in \mathbb{K}_b[x],\ c_i \in \mathbb{K}',\ s, r_i \in \mathbb{K}_b(c_j)[x].$$

p is called the polynomial part, the partial fractions of $\frac{c}{d}$ are the irregular singular parts, because each root of d is an irregular singular point of (2.1). With

$$t = s \prod_{i:c_i \in N} r_i^{c_i}$$

$y(x)$ can be written as

$$y(x) = t \quad \exp(p + \frac{c}{d}) \prod_i r_i^{c_i}$$

$$\text{with} \quad p, c, d \in \mathbb{K}_b[x],\ c_i \in \mathbb{K}' \setminus \mathbb{N},\ t, r_i \in \mathbb{K}_b(c_i)[x].$$

There $r_i^{c_i}$ are said to be the singular parts because each root of any r_i ist a singular point of (2.1). The algorithm has the following basic strategy:

step 1 - Bound the degree of the polynomial part.
step 2 - Determine the denominators of the irregular singular parts.
step 3 - Determine the singular parts which will be needed for bounding the degree of t.
step 4 - For all members of the set of possible combinations of singular parts determine the corresponding t and the coefficients in p and c simultaneously.

It is not possible to describe the details of this algorithm in this paper. Many details base on handling corresponding recurrence equations for the given ODE. This was done, because the main interest in Beke's algorithm came from finding approximate exponential solutions and instead of the search of an exact polynomial t in step 4 can be applied Frobenius method [8], which needs the recurrence equation. Mainly handling REs for the given ODEs allows to extend the algorithm in a natural way to find formal exact solutions of the form

$$y(x) = (t_0 + t_1 1n(x - x_0) + ...t_{n-1} 1n(x - x_0)^{n-1}) \exp(p + \frac{c}{d}) \prod_i r_i^{c_i} \tag{4.3}$$

$$\text{with} \quad c_i \in \mathbb{K}',\ p, c, d, r_i \in \mathbb{K}'[x]$$

where the t_i $(i = 0(1)n - 1)$ are formal Laurent series of the from (4.1). These solutions determined by means of the Frobenius method are exact in the sense that the REs describe exactly the coefficients of the series. The solutions are only formal solutions because ad hoc nothing is known about convergence. Truncating the Laurent series t_i results in good candidates for approximate solutions in closed form.

But it is more important that Beke's method can also be used for finding approximate exponential solutions with free parameters. After determining the set P of candidates for polynomial parts p and the set S of candidates for the singular parts $r_i^{c_i}$ with their corresponding irregular singular parts each function $f \in P \times S$ has to be multiplied with a monic polynomial t with free coefficients. In this way the functions ft are candidates for approximate exponential solutions (4.2) with free parameters.

4.2 Adaption

After finding candidates for approximate exponential solutions for ODEs of the form (2.1) the adaptive differential operator $\tilde{L} \in \mathbb{L}$ to a given differential operator L has to be determined. One important property of exponential solutions is that they are solutions of ODEs (2.1) of first order. This property of the candidates can be used in a recursive procedure to split up approximate right first-order factors of L. The rest terms appearing in the right remainders are minimized by applying adaption criterions.

Adaption - Algorithm (L, R)

(L, R are ODEs of the form (2.1). The algorithm computes linearly independent approximate solutions $\Phi = \{\varphi_1, \varphi_2, ..., \varphi_q\}$ for L,R. The fundamental system of R is known).

1. Use structure theorems of Computer Algebra for determining candidates $\tilde{y}_j$ for possible approximate exponential solutions of L. Each $\tilde{y}_j$ contains free coefficients. Assign Φ the empty sequence.
2. for each $\tilde{y}_j \in \{\tilde{y}_j\}$
   (a) Compute a ODE L^1 (2.1) of first order with free parameters
   $$c_1, c_2, ..., c_m \text{ and } L^1(\tilde{y}_j) = 0$$
   (b) Compute the right quotient Q of the form (2.1) and the right remainder such that
   $$pL = QL_1 + r \text{ and } p, r \in \mathbb{K}'[c_1, c_2, ..., c_m][x].$$
   (c) Determine the coefficients $c_1, ..., c_m$ in L^1 and $\tilde{y}_j$ by applying adaption criterions to r.
   (d) Determine a fundamental system of L^1R by d' Alembert reduction. Denote by φ the function, which is in the fundamental system of L^1R but not in the fundamental system of R.
   (e) if φ is elementary and $det(W_{\Phi,\varphi}) \neq 0$ then $\Phi := \Phi, \varphi$.
   (f) ψ: = Adaption - Algorithm (Q, L^1R)
   (g) for each $\varphi \in \psi$
   if $\det(W_{\Phi,\psi}) \neq 0$ then $\Phi := \Phi, \psi$.
3. return Φ

The algorithm should be started with Adaption - Algorithm (L, I), where I is the dentity differential operator.
$W_Y(x)$ is the notation of the Wronskian matrix for a sequence of functions $Y = \{y_1(x), y_2(x), ..., y_k(x)\}$. The result of the whole adaption step is a fundamental system of $\tilde{L}$ in form of a sequence of q linearly independent functions $\Phi = \{\varphi_1, \varphi_2, ..., \varphi_q\}$ The adaptive differential operator $\tilde{L}$ is not necessary in an explicit form, but $\tilde{L}$ can easily be computed by

$$\tilde{L}(y) = \frac{det(W_{\varphi_1, \varphi_2, ..., \varphi_q, y)}}{det(W_{\varphi_1, \varphi_2, ..., \varphi_q})}.$$

The fundamental system of $\tilde{L}$ was computed using d' Alembert reduction method. The integration routine decides whether the integrals are elementary functions or not.

It is usual to determine the free parameters by minimizing the defect function $L(\tilde{y})$. But minimizing the right remainder r is more advantageous, because $pL(\tilde{y}) = QL^1(\tilde{y}) + r\tilde{y}$ shows minimizing the defect function would mean minimizing $\tilde{y}$, too. Additionally, this adaption strategy has the great advantage that the adaption criterions should be applied to polynomials.
There are many possibilities to choose suitable criterions for the adaption $\tilde{L}$ to L. It depends on the kind of the task. If a boundary value problem is given, then a good approximation in a whole segment is of interest. If an initial value problem is given, then a good approximation near the initial point will be important. Possible adaption criterions are least square method with integral norms, collocation, complete Taylor approximation.

4.3 Approximate Solutions

The approximation step aims at combining the "best" functions from the q linearly independent functions $\varphi_1, \varphi_2, ..., \varphi_q$ with $\tilde{L}(\varphi_i) = 0$ (i = 1 (1) q) to get an approximate solution $\tilde{y}$ for the ODE (2.1). There are two possibilities. The usually applied procedure consists of making a linear ansatz

$$\tilde{y}(x) = \sum_{i=1}^{q} c_i \varphi_i.$$

Substitution $\tilde{y}$ into the constraints determines n of the q coefficients. The other $q - n$ coefficients are determined by applying approximation criterions to $\tilde{y}$. But this is only useful as long as q is not much greater than n and each function $\varphi_i (i = 1(1)q)$ has a simple structure. A linear combination of all functions with a lot of nonzero coefficients c_i will give a very complex expression. This is in contradiction to the demand for short transparent expressions in the Computer Analysis.

The other way is based on the property, that each function in Φ is the result of some optimization process in the adaption step and therefore is an optimal approximation in the sense of adaption criterions to a fundamental solution of the given differential operator L. Therefore the approximation step is reduced to the task to sort the functions $\varphi_1, \varphi_2, ..., \varphi_q$ by applying an approximation criterion to each of them. Possible criterions are the same once used in the adaption step.

For each function $\varphi_i \in \Phi$ (i = 1 (1) q) has to be computed the linear monic differential operator L^1 of first order with $L^1(\varphi_i) = 0$. Then it is possible to compute the right remainder r with $L = QL^1 + r, r \in \mathbb{K}'[\ln(x)](x)$. Finally, approximate criterions are applied to r and its norm is computed. After sorting the functions of the fundamental system Φ by increasing norm of the corresponding r the best n functions are selected for a linear ansatz $\tilde{y} = \sum_{i=1}^{n} c_i \varphi_i$. Substituting it into the constraints determines the coefficients c_i (i = 1 (1) n). In this procedure the norms of r are not computed for minimizing them, but for comparing one with another. Because r may by more complicated as in the adaption step and because the norms are not necessary in high precision, it is advisable to compute the norms numerically.

5 Error Estimation

The development of practicable error estimations is a central problem of the Computer Analysis. Approximate solutions without adequate error statements are worthless for the practical application. It is important that error estimations have to be completely computable by means of the hard- and software available at present.

The following procedure is based on a proposal by N. J. Lehmann [12]. He developed a general concept to compute upper bounds for the error function of an approximate solution computed by any approximate method. The error equation uses the Green function to the given differential operator and the defect of the approximate solution with reference to the given ODE as only information about the quality of the approximate solution. But the Green function is unknown and has to be estimated.

Applying this concept to our problem is relatively easy because the ODE (2.1) is linear and the Green function to the adaption operator $\tilde{L}$ is computable.

Let $\tilde{y} \in C^n[x_0, x_1]$ be an approximate solution of the linear ODE (2.1) which fulfils the constraints (2.2). Then the error function $f(x) = y(x) - \tilde{y}(x)$ is determined by

$$L(f) = L(y) - L(\tilde{y}) = -L(\tilde{y})$$

and with the Green function $G(x, s)$ to L and (2.2) in the form

$$f(x) = -\int_{x_0}^{x_1} G(x, s) L(\tilde{y}(s)) ds.$$

Applying the inequality of Schwarz gives with the integral norm the estimation

$$| f(x) | \leq \| G(x, \bullet) \| \| L(\tilde{y}(s)) \| .$$[1]

On the other hand, let $g \in C^{(0)}[x_0, x_1]$ be an aribitrary function. With $\tilde{G}$ (x, s) as Green function to $\tilde{L}$

$$u(x) = \int_{x_0}^{x_1} \tilde{G}(x, s) g(s) ds \tag{5.1}$$

is the unique solution of $\tilde{L}(u) = q$ and the homogeneous constraints to (2.2). From $L(u) = L_{\triangle}(u) + g$ with $L_{\triangle} = L - \tilde{L}$ is concluded

$$u(x) = \int_{x_0}^{x_1} G(x, s)[L_{\triangle}(u(s)) + g(s)] ds$$

[1] The point in $G(x, \bullet)$ points at the variable concerned by the norm computation.

$$= \int_{x_0}^{x_1} G(x,s)g(s)ds + \int_{x_0}^{x_1} G(x,s)L \overset{(s)}{\triangle} (\int_{x_0}^{x_1} \tilde{G}(s,t)g(t)dt)ds. \qquad (5.2)$$

If the dermination of $\tilde{L}$ secures that $L_\triangle$ is of an order less then L it is possible to interchange n - 1 times differentiation of $L_\triangle$ with the integration.

Since g was arbitrarily chosen and regarding (5.1) and (5.2) after interchanging $L_\triangle$ and the integration sign one concludes

$$G(x,s) - \tilde{G}(x,s) = -\int_{x_0}^{x_1} G(x,t)L \overset{(t)}{\triangle} (\tilde{G}(t,s)dt.$$

The usual norm estimation gives together with the triangle inequality the following error estimation:

$$\text{Provided } K = \sqrt{\int_{x_0}^{x_1}\int_{x_0}^{x_1} \mid L \overset{(x)}{\triangle} (\tilde{G}(x,s)) \mid^2 dxds} < 1$$

(i. e. the approximation should be reasonably good), then

$$\mid y(x) - \tilde{y}(x) \mid \leq \frac{\parallel \tilde{G}(x,\bullet) \parallel}{1-K} \parallel L(\tilde{y}(x)) \parallel \forall x \in [x_0, x_1]. \qquad (5.3)$$

If the constraints (2.2) are in particular initial conditions

$$y^{(\nu)}(x_0) = v_\nu \text{ or } y^{(\nu)}(x_1) = v_\nu, \quad (\nu = 0(1)n-1)$$

then $\tilde{G}(x,s)$ can be substituted by the right Green function or the left Green function to $\tilde{G}(x,s)$.

6 Conclusion

The algorithms were implemented in Maple V Release 5. The package includes not only procedures for finding approximate solutions of ODEs (2.1) but also exact solutions in the form of finite Laurent series, polynomials , rational functions and exponential functions. The implementation is restricted to the case $\mathbb{K} = \mathbb{Q}[\sqrt{-1}]$.

Many test computations have shown the usability of the adaptive approximation concept. A comparison with other approximation algorithms was not possible because it could not be found other packages, which compute approximate solutions of ”simple” differential equations in closed form.

The construction of approximate analytic solutions for differential equations will be of a great practical importance in future too. Many differential equations generally cannot be exactly integrated. Moreover, even in those cases when a closed form solution can be obtained it is often too cumbersome and inefficient to be used for practical purposes.

References

1. S. A. Abramov, K. Y. Kvasheko. Fast algorithms to search for the rational solutions of linear differential equations with polynomial coefficients. In S. Watt, editor, ISSAC 91, International Symposium on Symbolic and Algebraic Computation, Bonn, Germany 1991, p. 267 - 270.
2. O. Becken. Algorithmen zum Lösen einfacher Differentialgleichungen. Rostocker Informatik-Berichte 17, Universität Rostock, 1995
3. O. Becken. On adaptive approximation and D-finite functions. PhD thesis. Universität Rostock, 1999
4. O. Becken. Exact and approximate solutions of simple DEs. In Proceedings of the Workshop on Symbolic and Numerical Algorithms for Differential Equations (SNADE), Prague, 1997
5. E. Beke. Die Irreducibilität der homogenen linearen Differentialgleichungen. Mathematische Annalen, 45, S. 278 - 294, 1894
6. M. Bronstein. Linear ordinary differential equations: breaking trough the order 2 barrier. In P. Wang, editor. ISSAC 92, International Symposium on Symbolic and Algebraic Computation, Berkeley, California, 1992. ACM
7. M. Bronstein. On solutions of linear ordinary differential equations in their coefficient field. Journal of Symbolic Computation, 13, p. 413 - 439, 1992
8. G. Frobenius. Über die Integration der linearen Differentialgleichungen durch Reihen. Journal fr die reine und angewandte Mathematik, 76, S. 214 - 235, 1873
9. K. Hantzschmann. Implementierbare Fehlerabschätzungen für Näherungslsungen von Randwertaufgaben bei Systemen gewöhnlicher Differentialgleichungen. Studientexte Weiterbildungszentrum für Mathematische Kybernetik und Rechentechnik 73, Technische Universität Dresden, 1984

10. K. Hantzschmann, A. Jung. Adaptive approximation zur näherungsweisen Lösung von Differentialgleichungen. Rostocker Informatik-Berichte 18, Universität Rostock, 1995
11. K. Hantzschmann, N. X. Thinh. Analytical approximation solution of singular ordinary differential equations. In D. Shirkov, V. Rostovtsev, V. Gerdt, editors, IV. International Conference on Computer Algebra in Physical Research, Dubna, USSR, 1990, p. 169 -174
12. N. J. Lehmann. Fehlerschranken für Näherungslösungen bei Differentialgleichungen. Numerische Mathematik, 10, S. 261 - 288, 1967
13. N. J. Lehmann. Die Analytische Maschine - Grundlagen einer Computer- Analytik. In Sitzungsberichte der Sächsischen Akademie der Wissenschaften zu Leipzig, Math.-Naturwissenschaftliche Klasse, V. 118.4. Akademieverlag, Berlin 1985
14. N. J. Lehmann. Adaptive Approximation und Anwendungen. Studientexte Weiterbildungszentrum Computermathematik 105, Technische Universität Dresden, 1989
15. P. Richter. Einsatz eines analytischen Iterationsverfahrens zur Lösung von Randwertaufgaben. PhD thesis, Technische Universität Dresden, 1982
16. F. Schwarz. A factorization algorithm for linear ordinary differential equations. In G. Gonnet, editor, Proceedings of the ACM-SIGSAM 1989 International Symposium on Symbolic and Algebraic Computation, ISSAC 89, Portland, Oregon, p. 17 - 25, 1989, ACM
17. M. F. Singer. Liouvillian solutions on nth order homogeneous linear differential equations. American Journal of Mathematics, 103 (4), p. 661 - 682, 1981
18. R. P. Stanley. Differentiably finite power series. European Journal of Combinatorics, 1, p. 175 - 188, 1980