Asymptotic Methods of ODEs:
Exploring Singularities of the Second Kind

Christopher J. Winfield

We develop symbolic methods of asymptotic approximations for solutions of linear ordinary differential equations and use them to stabilize numerical calculations. Our method follows classical analysis for first-order systems and higher-order scalar equations where growth behavior is expressed in terms of elementary functions. We then recast our equations in modified form, thereby obtaining stability.

■ Introduction and Review

Following [1], we study methods to develop asymptotic estimates for a system of ordinary differential equations given in the form

$$\frac{d}{dt}\overrightarrow{w(t)} = t^r A(t) \overrightarrow{w(t)}, \tag{1}$$

where A is an $n \times n$ matrix (with elements depending on t) and $\overrightarrow{w(t)}$ is a column matrix whose elements are unknown functions. We suppose that A has analytic coefficients near infinity and can be written as a series $A = \sum_{j=0}^{\infty} t^{-j} A_j$ (convergent for large t) for constant matrices A_j, where A_0 is nontrivial. Such a singularity at finite t may be turned into a singularity at infinity by a change of variables. This type of singularity for $r > -1$ is called a *singularity of the second kind*, also known as an "irregular" singularity [2].

Differential equations of the form $\sum_{j=0}^{n} t^{r \, j} a_j(t) \frac{d^{n-j} y}{dt^{n-j}} = 0$, with the a_j bounded for large $t > 0$ and $a_0 \equiv 1$, can also be written in the above general matrix form by setting

$$A(t) = \mathcal{A}_1 + \mathcal{A}_2 + \mathcal{A}_3, \tag{2}$$

where the $\mathcal{A}_j$ have zero elements except: $\mathcal{A}_1$ has diagonal elements $0, -r \cdot 1 \, t^{-1-r}, \ldots, -r(n-1) \, t^{-1-r}$; $\mathcal{A}_2$ has ones on the diagonal above the main diagonal; the last row of $\mathcal{A}_3$

consists of the block matrix $(-a_n(t), -a_{n-1}(t), \ldots, -a_1(t))$. Here, solutions y and their first $n-1$ derivatives are given by the respective components of $\vec{w}$ as $y^{(j-1)} = t^{(j-1)r} \cdot w^{(n-j-1)}$, $j = 1, 2, \ldots, n$.

■ Asymptotic Procedures

Consider a system of the form $\vec{w}' = t^r A \vec{w}$, where the elements of A are analytic (near infinity) and $A \sim A_0 + t^{-1} A_1 + \ldots + t^{-r} A_{r+1}$ (formal series) where A_0 is a diagonal matrix of distinct complex numbers λ_j. Then, there are constant matrices R, Q_j, $j = 0, 1, \ldots, r+1$, and P_j, $j = 0, 1, \ldots$, so that the columns of

$$\Phi(t) = \left(\sum_{j=0}^{\infty} t^{-j} P_j \right) t^R \exp\left(\sum_{j=0}^{r+1} t^{-j} Q_j \right) \tag{3}$$

are each asymptotic series for some solution of (1). Here the matrices R, Q_j are diagonal, P_0 may be taken as the identity matrix, and Q_0 may be taken to equal A_0; exp denotes the matrix exponential and $t^R = \exp(\ln t R)$. This is a special case of theorems 2.1 and 4.1 of [1, Chapter 5]. In fact, this method produces exact solutions in cases not treated here; see [1, Chapter 4]. We do not repeat the entire proof of this result here, but we elaborate on the construction of the Q_j and R. By methods which amount to a matrix version of the dominant balance method, the matrices P_j, Q_{j-1}, $1 \le j \le r+1$, and R satisfy

$$P_k Q_0 - Q_0 P_k = \sum_{l=1}^{k} (A_l P_{k-l} - P_{k-l} Q_l) \text{ for } 1 \le k \le r,$$

$$P_{r+1} Q_0 - Q_0 P_{r+1} = A_{r+1} - R + \sum_{l=1}^{r} (A_l P_{r+1-l} - P_{r+1-l} Q_l).$$

For $j \ge 1$, let $\tilde{P}_j$ denote the same matrix as P_j above, but with zeros on its main diagonal. For our objectives, we need only to solve for the corresponding $\tilde{P}_j$ to proceed. We obtain

$$Q_1 = \text{Diag}(A_1); \text{ and } \tilde{P}_1 Q_0 - Q_0 \tilde{P}_1 = A_1 - Q_1 \text{ off the diagonal.} \tag{4}$$

By Diag( ), we mean that diagonal matrix whose elements match those of the argument along the main diagonal; this is not the same as the *Mathematica* built-in function `Diagonal`. Then for each $2 \le k \le r$ (provided $r > 1$), we may recursively compute the Q_k, $\tilde{P}_k$, and R by

$$Q_k = \text{Diag}\left(A_k + \sum_{l=1}^{k-1} \left(A_l \tilde{P}_{k-l} - \tilde{P}_{k-l} Q_l \right) \right), \tag{5}$$

$$\tilde{P}_k Q_0 - Q_0 \tilde{P}_k = \sum_{l=1}^{k} \left(A_l \tilde{P}_{k-l} - \tilde{P}_{k-l} Q_l \right) \text{ off the diagonal for } 2 \le k \le r, \text{ and} \tag{6}$$

$$R = \text{Diag}\left(A_{r+1} + \sum_{l=1}^{r} \left(A_l \tilde{P}_{r+1-l} - \tilde{P}_{r+1-l} Q_l \right) \right). \tag{7}$$

■ Example System of Equations

We begin with an example involving rational functions of t. Such examples suffice in our general setting by our hypothesis on $A(t)$. Consider the case $r = 1$, $x(0) = 0$, $y(0) = 4$, and

$$A(t) = \begin{pmatrix} -0.1 & 0 \\ 0 & -0.5 \end{pmatrix} + t^{-1} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} + t^{-2} \begin{pmatrix} 0 & -1 \\ 0 & -1 \end{pmatrix}. \tag{8}$$

We only need terms of order t and 1 for accuracy up to $O(t^{-1})$ in the first factor of (3). We follow equation (4) to solve Q_1 and $\tilde{P}_1$ and equation (7) to solve R. We denote $\tilde{P}_j$, $j = 0, 1, 2$, by P0, P1, P2 in input, and so on for other variables.

```
A0 = {{-1, 0}, {0, -0.5}}; Q0 = A0; P0 = IdentityMatrix[2];
A1 = {{1, 1}, {1, 0}};

prePl = Array[a, {2, 2}]; NN = 2;
C1 = A1 - DiagonalMatrix[Diagonal[A1]];
E1 = prePl.Q0 - Q0.prePl;

av1[j_, m_] :=
  If[j == m, 0,
   a[j, m] /.
    Flatten[
     Solve[Flatten[Table[E1[[k, l]] == C1[[k, l]],
        {k, NN}, {l, NN}]],
  Delete[Flatten[Table[a[k, l], {k, NN}, {l, NN}]],
        Array[{1 + (NN + 1) (#1 - 1)} &, NN]]]]];
P1 = Simplify[Array[av1, {NN, NN}]];
Q1 = Simplify[DiagonalMatrix[Diagonal[A1]]];
PreR = A1.P1 - P1.Q1;
R = DiagonalMatrix[Diagonal[PreR]];
```

Asymptotic solutions up to first derivatives are given by the columns of the matrix V.

```
V = Simplify[ (P0 + P1 / t).MatrixExp[Log[t] R].
    MatrixExp[ t^2 Q0 / 2 + t Q1]];
MatrixForm[V]
```

$$\begin{pmatrix} 0. + \dfrac{e^{t-\frac{t^2}{2}}}{t^{2.}} & 2.\, e^{-0.25\, t^2}\, t^{1.} \\[2mm] -\dfrac{2.\, e^{t-\frac{t^2}{2}}}{t^{3.}} & 0. + e^{-0.25\, t^2}\, t^{2.} \end{pmatrix}$$

We find that there are two linearly independent solution vectors $\vec{V}_1$ and $\vec{V}_2$ satisfying

$$\vec{V}_1 = e^{t-\frac{t^2}{2}}\begin{pmatrix} t^{-2}(1+o(t)) \\ -2\,t^{-3}(1+o(t)) \end{pmatrix};\ \vec{V}_2 = e^{-0.25\, t^2}\begin{pmatrix} 2\,t(1+o(t)) \\ t^2(1+o(t)) \end{pmatrix} \text{ as } t \to +\infty.$$

We compare our result with the exact solution of an initial value problem of the following asymptotically equivalent [3] system:

$$\frac{d}{dt}\overrightarrow{x(t)} = \begin{pmatrix} 1-t & 1 \\ 1 & -t/2 \end{pmatrix}\overrightarrow{x(t)};\ \overrightarrow{x(0)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Routine calculations, involving hypergeometric functions, show that $\overrightarrow{x(t)} = C\,e^{-0.25\, t^2}\begin{pmatrix} 2\,t(1+o(t)) \\ t^2(1+o(t)) \end{pmatrix}$ for a constant $C \approx 12.09$.

☐ Instability

A natural way to check our results is to compute numerical results via `NDSolve`. Here, we compare our dominant asymptotic calculations from $\vec{V}_2$ to numerical results of the original equation (via `Interpolation`). Below we expect our graphs to have horizontal asymptotes; instead, we find the calculations to be unstable for $t > 9$.

```
F[t_] := -1 / t^2 + 1 / t;
S = NDSolve[{x'[t] == (1 - t) x[t] + t F[t] y[t],
    y'[t] == -.5 t y[t] + t F[t] x[t], x[2] == 5, y[2] == 5},
  {x, y}, {t, 2, 11}];
Testx = x[t] / V[[1, 2]] //. S;
Testy = y[t] / V[[2, 2]] //. S;
Plot[{Testx, Testy}, {t, 2, 11}, PlotRange → Automatic,
 AxesLabel → {t}]
```
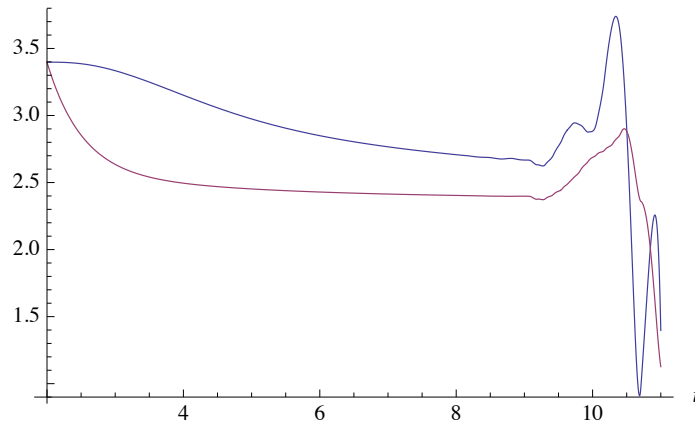
☐ **Modification**

We can apply these asymptotic estimates to produce stable solutions from `NDSolve` for large t. We replace $x(t)$ by $2\,t\exp(-.25\,t^2)\,x(t)$ and $y(t)$ by $t^2\exp(-.25\,t^2)\,y(t)$ to produce a solution (x, y) with less drastic decay rates.

```
newsys =
  {D[x[t] (t + 1) Exp[-.25 t^2], t] ==
    (1 - t) (t + 1) Exp[-.25 t^2] x[t] +
     (t + 1) ^2 Exp[-.25 t^2] y[t],
   D[Exp[-.25 t^2] (1 + t) ^2 y[t], t] ==
    (t + 1) Exp[-.25 t^2] x[t] -
     0.5 t Exp[-.25 t^2] (1 + t) ^2 y[t]};
Reformedprob =
  Flatten[NDSolve[{newsys, x[0] == 1, y[0] == 1},
    {x[t], y[t]}, {t, 0, 300}]];
Plot[{x[t] //. Reformedprob, y[t] //. Reformedprob},
 {t, 0, 600}, AxesLabel → {t}]
```
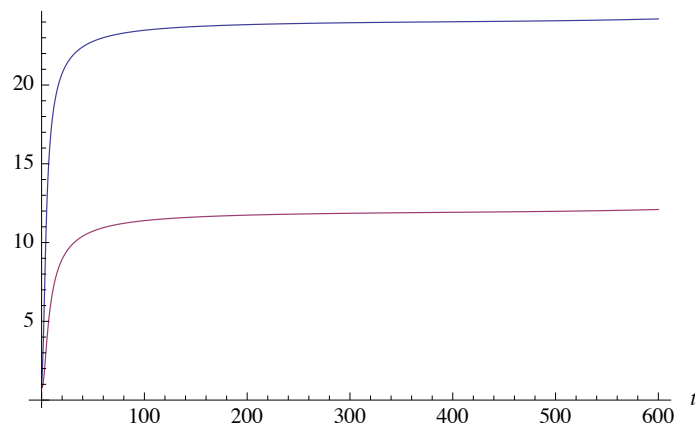
The solutions seem to be tending to limiting values for t near 600: all with no call to any special calculation options.

It is beyond the scope of this article to compare and contrast numerical schemes, since our objective is to predict theoretically the growth behavior of solutions as demonstrated using default options of `NDSolve`. Our method, we note, may serve to improve stability and reduce stiffness, yet we do not quantify such results. We do, however, note that widely varying exponential growth/decay of solutions is known to produce instability similar to what we see in the literature. References [4, 5, 6] and the `NDSolve` documentation [7], among many others, introduce rigorous methods regarding such issues.

■ Application to a Third-Order Ordinary Differential Equation

We now study asymptotic behavior of solutions to $Ly = 0$ (for $t > 0$) for the operator $L = D^3 + (t^2 - 3 t) D^2 - t^4 D$. Following (2) we study the system $\vec{w}' = M \vec{w}$, which we can write as

$$M = t^2 \left(\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 2 & -1 \end{pmatrix} + \frac{1}{t} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 3 \end{pmatrix} + \frac{1}{t^2} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \frac{1}{t^3} \begin{pmatrix} 0 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -4 \end{pmatrix} \right). \tag{9}$$

We choose this example so as to force a diagonalization step before applying the procedure to produce estimates (3), after which we identify A_j, $j = 0, 1, 2, 3$ (with $r = 2$).

```
NN = 3; rr = 2; M0 = {{0, 1, 0}, {0, 0, 1}, {0, 2, -1}};
M1 = {{0, 0, 0}, {0, 0, 0}, {0, 0, 3}};
M2 = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}};
M3 = DiagonalMatrix[Array[- (# - 1) rr &, NN]];
```

We proceed with the diagonalization step.

```
Clear[TT]; TT = Transpose[Eigenvectors[M0]];
A0 = Inverse[TT].M0.TT; A1 = Inverse[TT].M1.TT;
A2 = Inverse[TT].M2.TT;
A3 = Inverse[TT].M3.TT;
```

Here Q_0 and Q_1 are obvious and, using (4) and (5), we compute $\tilde{P}_1$ and Q_2.

```
Q0 = A0; Q1 = DiagonalMatrix[Diagonal[A1]];
preP1 = Array[a, {3, 3}];
EE10 = preP1.Q0 – Q0.preP1; EE20 = preP2.Q0 – Q0.preP2;
CC1 = A1 – Q1;
av1[j_, m_] :=
  If[j == m, 0,
   a[j, m] /.
    Flatten[
     Solve[Flatten[Table[EE10[[k, l]] == CC1[[k, l]],
        {k, NN}, {l, NN}]],
       Delete[Flatten[Table[a[k, l], {k, NN}, {l, NN}]],
        Array[{1 + 4 (# – 1)} &, NN]]]]];
P1 = Array[av1, {NN, NN}]; PreQ2 = (A1 – Q1).P1 + A2;
Q2 = DiagonalMatrix[Diagonal[PreQ2]];
```

Using (6) and (7) we compute $\tilde{P}_2$ and R.

```
preP2 = Array[aa, {3, 3}];
CC21 = A2 – Q2 – DiagonalMatrix[Diagonal[A2 – Q2]];
CC22 = A1.P1 – P1.Q1 – DiagonalMatrix[Diagonal[A1.P1 – P1.Q1]];
av2[j_, m_] :=
  If[j == m, 0,
   aa[j, m] /.
    Flatten[
     Solve[
      Flatten[
       Table[EE20[[k, l]] == CC21[[k, l]] + CC22[[k, l]],
        {k, NN}, {l, NN}]],
       Delete[Flatten[Table[aa[k, l], {k, NN}, {l, NN}]],
        Array[{1 + 4 (# – 1)} &, NN]]]]];
P2 = Array[av2, {NN, NN}]; PreR = (A1 – Q1).P2 + (A2 – Q2).P1 + A3;
R = DiagonalMatrix[Diagonal[PreR]];
```

We are ready to compute asymptotic solutions after a change of basis.

```
preV =
  Simplify[MatrixExp[Log[t] R].
    MatrixExp[t^3 Q0 / 3 + t^2 Q1 / 2 + t Q2]];
V = Simplify[TT.preV];
```

We multiply rows of the solution matrix by various powers of t to produce the various asymptotic estimates of solutions to $Ly = 0$, which we can read off from the columns of `Soln`.

```
S = DiagonalMatrix[Array[t^(rr (# - 1)) &, NN]]; Soln = S.V;
MatrixForm[Soln]
```

$$\begin{pmatrix} \dfrac{e^{-\frac{1}{3}t\,(2-3\,t+2\,t^2)}}{t^{32/9}} & \dfrac{e^{\frac{1}{6}t\,(4+3\,t+2\,t^2)}}{t^{22/9}} & 1 \\[2ex] -\dfrac{2\,e^{-\frac{1}{3}t\,(2-3\,t+2\,t^2)}}{t^{14/9}} & \dfrac{e^{\frac{1}{6}t\,(4+3\,t+2\,t^2)}}{t^{4/9}} & 0 \\[2ex] 4\,e^{-\frac{1}{3}t\,(2-3\,t+2\,t^2)}\,t^{4/9} & e^{\frac{1}{6}t\,(4+3\,t+2\,t^2)}\,t^{14/9} & 0 \end{pmatrix}$$

We find three different types of behavior for asymptotic solutions: rapid decay, rapid growth, and a constant solution. We can check that any constant function is indeed a solution, so the constant behavior is precise and not just asymptotic in this case. Moreover, any solution y is majorized by `Soln[[1, 2]]` $= t^{-22/9}\,e^{\frac{1}{6}t\,(4+3\,t+2\,t^2)}$.

☐ Graphical Results

We check our results using two approaches. First we simply graph ratios $\vec{w}\,/\,\text{So}$. `ln[[2, 1]]` with numerical solutions $\vec{w}$. Let us consider this using a nonconstant solution arising from the initial values $(w(0), w'(0), w''(0)) = (0, 1, 0)$. The analysis for initial values $(0, 0, 1)$ is similar, and we invite the reader to check.

```
diffop = y'''[t] - 2 t^4 y'[t] - (-t^2 + 3 t) y''[t];
initvprob =
 NDSolve[{diffop == 0, y[0] == 0, y'[0] == 1, y''[0] == 0},
  y[t], {t, 0, 15}];
Plot[(y[t] / Soln[[1, 2]]) /. initvprob, {t, 0, 15},
 PlotRange → All, AxesLabel → {t, ratio}]
```
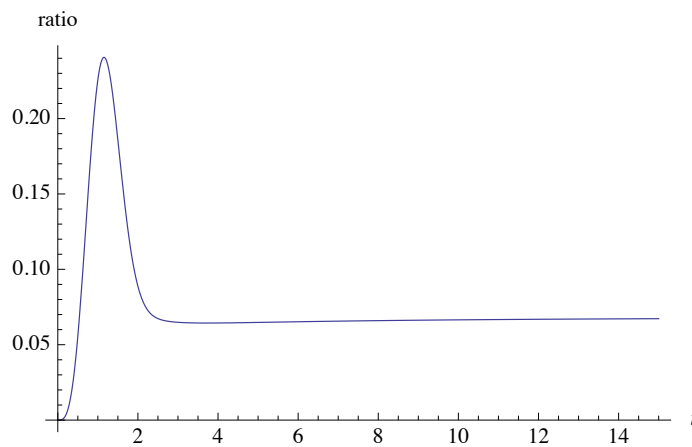
We are able to produce graphs using default options of `NDSolve` only for moderate domains of t. For our second approach, we modify the operator by substituting $y(t) \longrightarrow y(t) t^{-22/9} \exp\left(\frac{1}{6} t \left(4 + 3t + 2t^2\right)\right)$, thereby theoretically guaranteeing bounded solutions y on intervals away from the origin $t = 0$.

```
newdiffop =
  Simplify[
    (D[y[t] Soln[[1, 2]], {t, 3}] -
       2 t^4 D[y[t] Soln[[1, 2]], {t, 1}] +
       (t^2 - 3 t) D[y[t] Soln[[1, 2]], {t, 2}]) / Soln[[1, 2]]];
t0 = 30;
Modified =
  NDSolve[{newdiffop == 0, y[t0] == 1, y'[t0] == 0, y''[t0] == 0},
    y[t], {t, t0, t0 + 120}];
Plot[y[t] /. Modified, {t, t0, t0 + 120}, AxesLabel → {t, y}]
```
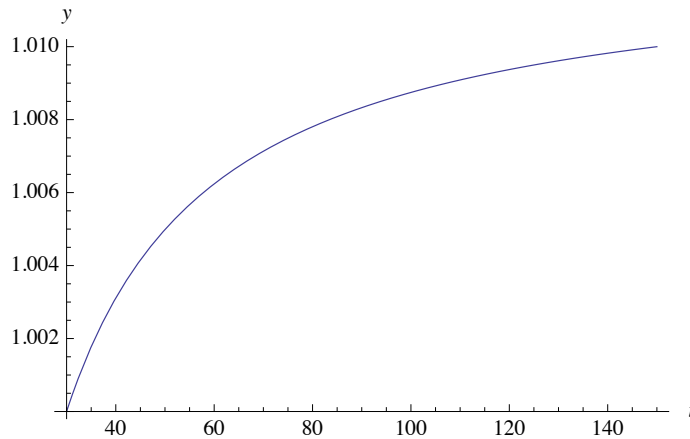

We find instability as we calculate relative errors over large intervals (see [8]), as in the previous section. Alternatively, we use a modified operator along with a linear change of variable to calculate solutions on intervals for large t. We apply the transformation $t \longrightarrow \tau + q$ after applying the modification procedure above; then the initial value problem applied near $t = 0$ for the new problem is equivalent to solving the original near $t = q$ for large q. We find that in this way we can accurately compute solutions on large intervals in t.

```
Manipulate[newfactor = Soln[[1, 2]] //. t → τ + q;
 newdiffop =
  Simplify[
    (D[y[τ] newfactor, {τ, 3}] -
      2 (τ + q)^4 D[y[τ] newfactor, {τ, 1}] +
      ((τ + q)^2 - 3 (τ + q)) D[y[τ] newfactor, {τ, 2}]) /
    newfactor];
 newsol =
  NDSolve[{newdiffop ⩵ 0, y[0] ⩵ 10, y'[0] ⩵ 0, y''[0] ⩵ 0},
    y[τ], {τ, 2500}];
 Plot[y[τ] /. newsol, {τ, 0, 200}, AxesLabel → {τ, y}],
 {q, 400, 500, Appearance → "Labeled"}, SaveDefinitions → True]
```
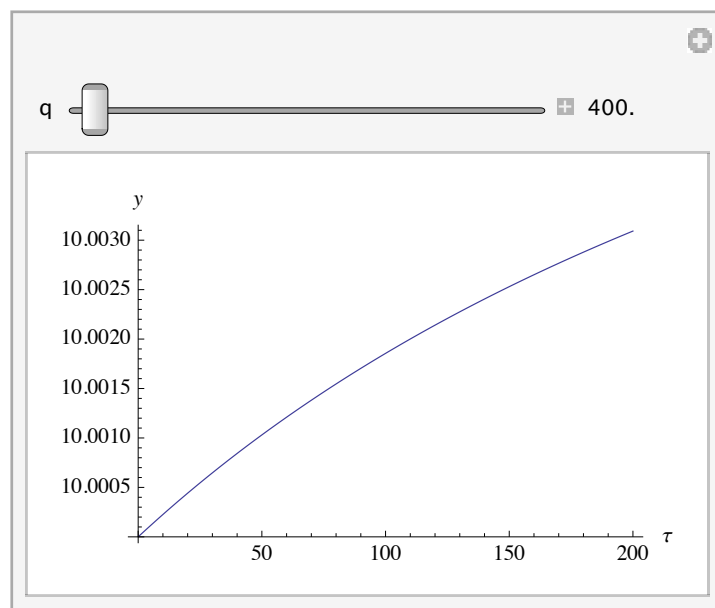

The advantage here is that the end result is of the form $y(t)\, t^{-22/9}\, \exp\!\left(\frac{1}{6}\, t\left(4 + 3\,t + 2\,t^2\right)\right)$ for large t where y is computed with considerable accuracy and the other terms are known exactly.

■ Conclusion

The classical techniques of asymptotics can be calculated symbolically and in an automated way via simple matrix manipulations. These estimates can serve to check accuracy of numerical solutions both for scalar equations and for systems of equations of the types studied. Moreover, these estimates may serve to transform differential equations to certain modified versions that admit solutions bounded as $t \to +\infty$; as solutions of modified problems can be computed with accuracy and stability, with the transformation in elementary form, accuracy in our calculations is improved overall.

■ Acknowledgments

■ References

[1] E. Coddington and N. Levinson, *Theory of Ordinary Differential Equations*, New York: Mc-Graw-Hill, 1955.

[2] A. H. Nayfeh, *Perturbation Methods*, New York: Wiley, 1973.

[3] F. Brauer and J. Nohel, *The Qualitative Theory of Ordinary Differential Equations: An Introduction*, New York: W. A. Benjamin Inc., 1969.

[4] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*, 2nd ed., New York: Cambridge University Press, 2009.

[5] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, Hoboken, NJ: Wiley, 2003.

[6] M. Sofroniou and R. Knapp, *Wolfram Mathematica Tutorial Collection: Advanced Numerical Differential Equation Solving in Mathematica*, Champaign, IL: Wolfram Research, Inc., 2008. www.wolfram.com/learningcenter/tutorialcollection/ AdvancedNumericalDifferentialEquationSolvingInMathematica.

[7] Wolfram Research, Inc. "Wolfram *Mathematica* Documentation Center: NDSolve." (Sep 29, 2011) reference.wolfram.com/mathematica/ref/NDSolve.html.

[8] C. Winfield. "MAST Research Log: Application of *Mathematica* to a Third-Order Ordinary Differential Equation." (May 18, 2011) www.madscitech.org/rlog/webref1.cdf.

About the Author

Christopher Winfield holds a Ph.D. in mathematics/real analysis from UCLA (1996) and an M.S. in physics from UW-Madison (1989). He has publications in partial differential equations, quantum scattering theory, and cosmology/Einstein's equation. He currently works as a consultant with the Madison Area Science and Technology group based in Madison, Wisconsin.

Christopher Winfield
3783 US Hwy. 45
Conover, WI 54519
cjwinfield@madscitech.org