>

UPDATED TO MAPLE V, RELEASE 4, BY DOUGLAS B. MEADE (6 November 1996)

Note that the text has not been updated; this will need more work before the output will generate a reasonable LaTeX file. Some changes to the graphical output have been made, but more should be made. All calculations seem to agree with the Release 3 worksheet. The jump procedure was re-written -- it's now simpler, and it works for this problem (the Release 3 version was OK, but no longer gave reliable answers in Release 4 -- I think this had something to do with simplification of Heaviside functions).

```
% Maple Worksheet for the MapleTech paper on the Parachute Problem
%
% Created by Douglas B. Meade (meade@math.sc.edu)
%
% 17 July 1996
%
% This worksheet can be typeset by either latex2.09 or latex2e. Simply
% comment out the header information for the version that is not needed.
%
% It is also necessary to comment out the \documentstyle, ... lines added
% by Maple. (This is described in more detail below.)


% LaTeX2.09 header information
%
% To create the complete LaTeX2.09 source file:
%   - export as latex (math formatting width: 3.2 inches for two column)
%   - edit article.tex (delete all lines down through the end of the first mapleinput -- 2 lines)
%   - typeset using latex

\documentstyle[fancyheadings,twocolumn,mtn]{article}


% LaTeX2e header information
%
% To create the complete LaTeX2e source file:
%   - export as latex (math formatting width: 3.2 inches for two column)
%   - edit article.tex (delete all lines down through the end of the first mapleinput -- 2 lines)
%   - typeset using latex2e

%\documentclass[twocolumn,10pt]{mtn}
```

```latex
%\usepackage{times}
%\usepackage{mtnplots}
%\usepackage{latexsym}

%%\input epsf.def


% Generic preample information (both 2.09 and 2e)

\input psfig

\textwidth = 7in
\pagestyle{fancy}

%\setcounter{page}{5}

\newcommand{\clearemptydoublepage}{\newpage{\pagestyle{empty}\cleardoublepage}}
\newcommand{\R}{\mbox{\boldmath{R}}}
\def\Maple{Waterloo Maple}

\def\ft{$^\prime$\ }

\begin{document}


 \HEADING%
{Maple and the Parachute Problem:\\ Modelling with an Impact}
{Douglas B.\ Meade%
 \footnote{Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA;
	E-mail: {\tt meade@math.sc.edu};
	WWW URL: {\tt http://www.math.sc.edu/\mbox{$\tilde{\ }$}meade/}}
}%
{Parachute Problem}


\heading{Introduction}
```

The ‘‘parachute problem’’ is an application found in many differential equations textbooks. While the details vary, the general problem is the same: determine the motion of a skydiver when the coefficient of air resistance changes between free-fall and final descent. The author has previously shown that many of the problems posed in the textbooks are not physically realistic \cite{meade}. An improved --- but not more complicated --- model is proposed using real-life information about skydiving.

In this paper we demonstrate how the symbolic, numeric, and graphic tools in Maple can be used to facilitate the analysis of both models, including the discussion of parameter sensitivity and a control problem. While the basic problems are not too sophisticated, manual computation of the solution is prohibitive. A working knowledge of Maple makes the analysis appropriate for post-calculus students.

\heading{The Original Model}

The ‘‘parachute problem’’ will be considered as a system of two initial value problems for the position, x, and (vertical) velocity, v, of a skydiver under the forces of gravity and air resistance. Assuming the force due to air resistance is proportional to the velocity of the parachutist, an appropriate model is:
\begin{equation}
\begin{array}{rcll}   \label{EOM}
v’ &=& -g - \frac{k}{m}v, & v(0)=0, \\
x’ &=& v,   & x(0)=x_0.
\end{array}
\end{equation}
where g is the gravitational constant, k is the coefficient of air resistance, m is the mass of the skydiver, and x_0 is the altitude at which the jump begins. The coefficient of air resistance typically (\cite{ab}, \cite{drucker}, \cite{ep}, \cite{ka}, \cite{ns}) has the form
\begin{equation}   \label{k-pwconst}
k(t) =
\left\{
 \begin{array}{ll}
 k_1, & t<t_d \\
 k_2, & t\ge t_d
 \end{array}
\right.
\end{equation}
where t_d is the time when the parachute is deployed. Note that t_d may be specified implicitly, {\em e.g.}, deployment occurs when the velocity is 95\% of the terminal free-fall velocity \cite{ka}, in which case the IVP may not be linear. Parameter values found in a variety of textbooks are summarized in Table~\ref{param-table}. (Observe that the ratio k/m, with units 1/time, can be used in the place of k and m.)

\begin{table}
\begin{center}

```
\small
\begin{tabular}{|c|c|c|c|c|}
      \hline
Reference          & $k_1/m$ & $k_2/m$     & $t_d$          & $x_0$      \\
      \hline
%Abell/Braselton
p. 141, \#20 \cite{ab}
                 & 1/6     & 5/3         & 60sec          & n/a        \\
%Drucker
\cite{drucker}
                 & 4/15    & 4/3         & $x(14)=0$      & 1200ft     \\
%Edwards/Penney
p. 95, \#10 \cite{ep}
                 & 3/20    & 3/2         & 20sec          & 10,000ft   \\
%Kostelich/Armbruster
p. 109, \#20 \cite{ka}
                 & 1/5     & $\approx1.56$& $v(t_d)=v^*$  & 2000m      \\
%                                        =0.95v_\infty$ & 2000m      \\
%Nagle/Saff
p.112, Ex.\ 3 \cite{ns}
                 & 1/5     & 7/5         & 60sec          & 4000m      \\
      \hline
\end{tabular}
\end{center}
\caption{ \label{param-table}
 Coefficients of air resistance, deployment criteria, and jump height
 found in different versions of PP.
}
\end{table}
```

```
> restart;
  with( DEtools ):
  with( plots ):
  setoptions( thickness=2,
              font=[TIMES,ITALIC,18],
              axesfont=[TIMES,ROMAN,18],
              titlefont=[TIMES,ITALIC,18] );
```

The basic problem to be studied is

```
> SYS := { diff( v(t), t ) = -g - k/m*v(t),
           diff( x(t), t ) = v(t) };
```

$$SYS := \{\frac{\partial}{\partial t}v(t) = -g - \frac{k\,v(t)}{m}, \frac{\partial}{\partial t}x(t) = v(t)\}$$

with initial conditions

```
> IC := { v(0) = 0, x(0) = x0 };
```

$$IC := \{v(0) = 0, x(0) = x0\}$$

As the system of ODEs and initial conditions are both sets, it will be useful to have a means to extract a specific entry from either object.

```
> ode := (SET,y) ->
         op( select( has, SET, diff(y(t),t) ) ):
  ic := (SET,y0) ->
        op( select( has, SET, y0 ) ):
```

The position (height above ground level), velocity, acceleration, and jerk -- rate of change of acceleration -- provide a good description of the motion of the skydiver. Given an initial value problem in terms of x and v, the procedure \verb+motion+ uses \verb+dsolve,numeric+ to compute the motion of the object at given times; \verb+plot_motion+ creates a single plot containing any combination of the position, velocity, acceleration, and jerk as computed by \verb+motion+.

\newpage

```
>  motion := proc( IVP::set(equation),
                   TIMES::list(numeric) )
    local POS, VEL, ACC, JERK, pts, acc, jerk,
           soln, T;
    global t;
    options `Copyright 1996 by Douglas B. Meade`;
    acc:=solve( ode(IVP,v), diff(v(t),t) );
    jerk:=diff(acc,t);
    soln:=dsolve(IVP, {x(t),v(t)}, type=numeric,
                 output=listprocedure):
    POS:=subs( soln, x(t) );
    VEL:=subs( soln, v(t) );
    ACC:=unapply( subs( v(t)='VEL(t)', acc ), t );
    JERK:=unapply(subs( {v(t)='VEL(t)',
                        'diff(v(t),t)'='ACC(t)'},
                        jerk ), t );
    pts:=seq(map(evalf,
                 [T,POS(T),VEL(T),ACC(T),JERK(T)]),
              T=sort(TIMES) ):
    t:=evaln(t);
    RETURN( [ pts ] );
  end:
> plot_motion :=
    proc( pts::list, scale::set(equation) )
    local P, PLOTS, S, VARS;
```

```
    options `Copyright 1996 by Douglas B. Meade`;
    PLOTS := NULL;
    VARS := map(lhs,scale);
    if member(X,VARS) then
      S:=subs(scale,X);
      PLOTS:=PLOTS,
             plot([ seq( [P[1],P[2]/S], P=pts )],
             color = RED ):
    fi;
    if member(V,VARS) then
      S:=subs(scale,V);
      PLOTS:=PLOTS,
             plot([ seq( [P[1],P[3]/S], P=pts )],
             color = BLUE ):
    fi;
    if member(A,VARS) then
      S:=subs(scale,A);
      PLOTS:=PLOTS,
             plot([ seq( [P[1],P[4]/S], P=pts )],
             color = GREEN ):
    fi;
    if member(J,VARS) then
      S:=subs(scale,J);
      PLOTS:=PLOTS,
             plot([ seq( [P[1],P[5]/S], P=pts )],
             color = CYAN ):
    fi;
    RETURN( display( { PLOTS } ) );
  end:
```

The piecewise definition of k can be implemented using the \verb+Heaviside+ command. (Another option is to use \verb+piecewise+, particularly in Release 4.)

```
> K[pw_const] :=  k1+(k2-k1)*Heaviside(t-td):
```

As a first illustration of these utilities, the motion of the skydiver that experiences air resistance as in (\ref{k-pwconst}) is governed by the initial value problem

```
> IVP := subs( k=K[pw_const], SYS ) union IC;
```

$$IVP := \left\{\frac{\partial}{\partial t}x(t)=v(t),\ v(0)=0,\ x(0)=x0,\ \frac{\partial}{\partial t}v(t)=-g-\frac{(k1+(k2-k1)\,\mathrm{Heaviside}(t-td))\,v(t)}{m}\right\}$$

The specific motion corresponding to the parameters in Nagle/Saff can be produced after specifying values for each parameter

```
> PARAM[ns] := k1=15, k2=105, td=60, x0=4000,
                 m=75, g=981/100:
```

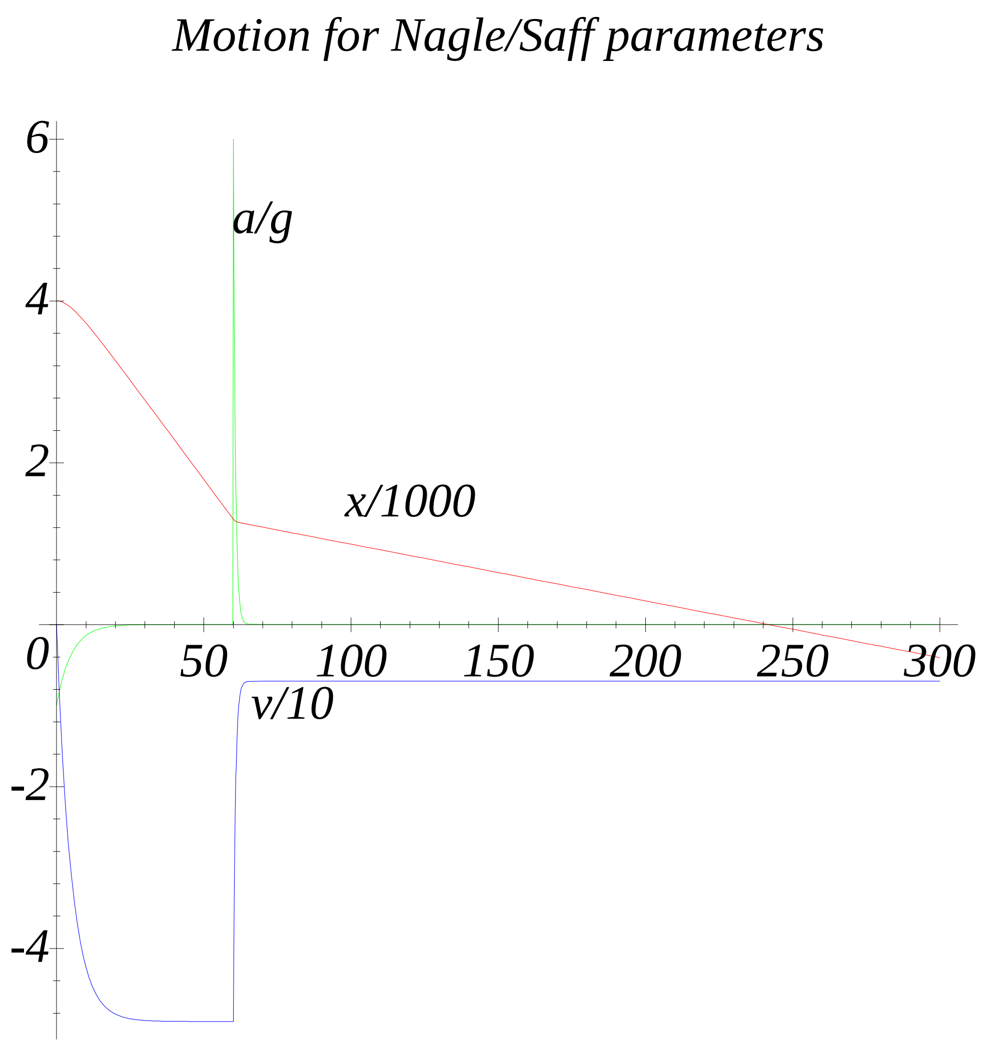
\noindent

and the times at which the solution should be plotted (note the finer resolution near the time of deployment)

```
> snapshot[ns] := [ $(0..58),
                    i/10 $ i = 590 .. 649,
                    5*i $ i = 13 .. 60 ]:
```

\noindent
A nice graph of the motion during the first five minutes of the jump is obtained as follows:

```
> PTS[ns]:=motion( subs(PARAM[ns],IVP),
                   snapshot[ns] ):
  scale[noJ] := { 'X'=1000, 'V'=10, 'A'=981/100 }:
  display(
    {
      plot_motion( PTS[ns], scale[noJ] ),
      textplot( [ [ 120, 1.5, `x/1000` ],
                  [ 80, -1, `v/10` ],
                  [ 70, 5, `a/g` ] ] )
    },
    title=`Motion for Nagle/Saff parameters` );
```



\mapleplot{motion-ns.ps}
This plot supplies good first approximations to a number of interesting characteristics of the jump. When the ripcord is pulled at the end of the first minute, the skydiver is still 1300~m above ground level (AGL) and descending at an almost constant velocity of 49~m/s. Within 5 seconds from the time the parachute is opened the motion has settled into an essentially steady descent at 7~m/s; landing occurs very close to 4 minutes after the jump began.

More precise answers can be obtained from an explicit solution to (\ref{EOM}). This solution is found by first solving the problem with the coefficient for the free-fall stage of the jump:

```
> SOL1 := dsolve( subs(k=k1,SYS) union IC,
                  { x(t), v(t) } ):
  V1 := subs( SOL1, v(t) );
  X1 := subs( SOL1, x(t) );
```

$$V1 := -\frac{e^{\left(-\frac{k1\,t}{m}\right)} g\,m\left(e^{\left(\frac{k1\,t}{m}\right)}-1\right)}{k1}$$

$$X1 := x0 - \frac{g\,m\left(k1\,t - m + m\,e^{\left(-\frac{k1\,t}{m}\right)}\right)}{k1^2}$$

The second stage of the jump is similar, except that the ''initial conditions'' are chosen to ensure continuity of both x and v at the time of deployment:

```
> IC2 := { v(td) = subs( t=td, V1 ),
           x(td) = subs( t=td, X1 ) };
```

$$IC2 := \left\{x(td) = x0 - \frac{g\,m\left(k1\,td - m + m\,e^{\left(-\frac{k1\,td}{m}\right)}\right)}{k1^2},\ v(td) = -\frac{e^{\left(-\frac{k1\,td}{m}\right)} g\,m\left(e^{\left(\frac{k1\,td}{m}\right)}-1\right)}{k1}\right\}$$

```
> SOL2 := dsolve( subs(k=k2,SYS) union IC2,
                  { x(t), v(t) } ):
  V2 := subs( SOL2, v(t) );
  X2 := subs( SOL2, x(t) );
```

$$V2 := -\frac{g\,m\left(e^{\left(\frac{k1\,td}{m}\right)}-1\right)e^{\left(-\frac{k1\,td + k2\,t - td\,k2}{m}\right)}}{k1} - \frac{e^{\left(-\frac{k2\,t}{m}\right)} g\,m\left(e^{\left(\frac{k2\,t}{m}\right)} - e^{\left(\frac{td\,k2}{m}\right)}\right)}{k2}$$

$$X2 := \left(x0\,k1^2\,k2 - g\,m\,k2\,k1\,td + g\,m^2\,k2 - g\,m^2\,k2\,e^{\left(-\frac{k1\,td}{m}\right)} + g\,m^2\,k1\,e^{\left(-\frac{k2\,(t-td)}{m}\right)} - g\,m^2\,k1\,e^{\left(-\frac{k1\,td + k2\,t - td\,k2}{m}\right)} - g\,m^2\,k1 + m^2\,e^{\left(-\frac{k1\,td}{m}\right)} g\,k1\right) \Big/ (k1^2\,k2) - \frac{g\,m\left(k2\,t - td\,k2 - m + m\,e^{\left(-\frac{k2\,(t-td)}{m}\right)}\right)}{k2^2}$$

\noindent
The complete solution can be written in terms of the Heaviside function as

```
> V := V1 + (V2-V1)*Heaviside(t-td):
  X := X1 + (X2-X1)*Heaviside(t-td):
```

or, converted to piecewise-defined functions

```
> 'V' = convert( V, piecewise, t );
  'X' = convert( X, piecewise, t );
```

$$V = \begin{cases} -\frac{e^{\left(-\frac{k1\,t}{m}\right)} g\,m\,e^{\left(\frac{k1\,t}{m}\right)}}{k1} + \frac{e^{\left(-\frac{k1\,t}{m}\right)} g\,m}{k1}, & t \le td \end{cases}$$

$$-\frac{m\,g\,e^{\left(-\frac{k1\,td+k2\,t-td\,k2}{m}\right)}\,e^{\left(\frac{k1\,td}{m}\right)}}{k1}+\frac{m\,g\,e^{\left(-\frac{k1\,td+k2\,t-td\,k2}{m}\right)}}{k1}-\frac{e^{\left(-\frac{k2\,t}{m}\right)}\,g\,m\,e^{\left(\frac{k2\,t}{m}\right)}}{k2}+\frac{e^{\left(-\frac{k2\,t}{m}\right)}\,g\,m\,e^{\left(\frac{td\,k2}{m}\right)}}{k2}$$

$$, td < t$$

$$X = \begin{cases} x0-\frac{g\,m\,t}{k1}+\frac{m^2\,g}{k1^2}-\frac{g\,m^2\,e^{\left(-\frac{k1\,t}{m}\right)}}{k1^2}, & t \le td \\ -\frac{g\,m\,td}{k1}-\frac{m^2\,g\,e^{\left(-\frac{k1\,td}{m}\right)}}{k1^2}+\frac{g\,m^2\,e^{\left(-\frac{k2\,(t-td)}{m}\right)}}{k1\,k2}-\frac{g\,m^2\,e^{\left(-\frac{k1\,td+k2\,t-td\,k2}{m}\right)}}{k1\,k2}-\frac{g\,m^2}{k1\,k2}+\frac{m^2\,e^{\left(-\frac{k1\,td}{m}\right)}\,g}{k1\,k2} \\ \quad -\frac{g\,m\,t}{k2}+\frac{g\,m\,td}{k2}+\frac{g\,m^2}{k2^2}-\frac{g\,m^2\,e^{\left(-\frac{k2\,(t-td)}{m}\right)}}{k2^2}+x0+\frac{m^2\,g}{k1^2}, & td < t \end{cases}$$

\noindent
At the time of deployment the position and velocity are

```
> subs( t=60, PARAM[ns], [ X, V ] );
  evalf( " );
```

$$\left[\frac{5209}{4}-\frac{981}{4}\,e^{(-12)}+\left(-\frac{2943}{98}+\frac{2943}{98}\,e^{0}\right)Heaviside(0),-\frac{981}{20}\,e^{(-12)}\,(e^{12}-1)\right]$$

$$[1302.248493, -49.04969862]$$

\noindent
The time of impact is found to be

```
> t_impact[ns]:=solve( subs(PARAM[ns],X2)=0, t);
  't_impact[ns]' = evalf(t_impact[ns]);
```

$$t\_impact_{ns} := \frac{5}{6867}\,\frac{981\,e^{12}\,LambertW\left((6\,e^{12}-7)\,e^{\left(-1/981\,\frac{261127\,e^{12}-41202}{e^{12}}\right)}\right)+331759\,e^{12}-41202}{e^{12}}$$

$$t\_impact_{ns} = 241.5601769$$

\noindent
(Note that W is the Lambert Ω function, {\em i.e.}, the branch of the function Ω defined by $\Omega(x) e^{\Omega(x)}=x$ that is analytic at 0). The corresponding position and velocity are estimated to be

```
> subs( t=t_impact[ns], PARAM[ns], [ X, V ] ):
  fnormal( " );
```

$$\left[\frac{1}{196}\,\frac{41202-41202\,e^{(-12)}\,e^{12}}{e^{12}}, 0\right]$$

These results show good agreement with the graphical solution; how does this motion compare with real-life? Is the free-fall terminal velocity of 49~m/s (110~mph) reasonable? Is a landing velocity of 7~m/s (15.6~mph) survivable? According to \cite{af490}, terminal free-fall velocity is approximately 53.6~m/s (120~mph) and the impact velocity should be comparable to a free-fall from a five foot (1.52~m) high wall. To convert this information into a specific velocity, we use the free-fall solution with a much lower initial height, {\em i.e.},

```
> PARAM[five]:=x0=152/100, m=75, k1=15, g=981/100:
```

\noindent
The corresponding motion is

```
> X5 := subs( PARAM[five], X1 );
  V5 := subs( PARAM[five], V1 );
```

$$X5 := \frac{24677}{100} - \frac{981}{20}\,t - \frac{981}{4}\,e^{(-1/5\,t)}$$

$$V5 := -\frac{981}{20}\,e^{(-1/5\,t)}\,(e^{(1/5\,t)} - 1)$$

\noindent
The duration of this jump, and the associated velocity are

```
> T5 := fsolve( X5=0, t, 0..infinity );
  V_max := evalf( subs( t=T5, V5 ) );
```

$$T5 := .5671998658$$

$$V\_max := -5.260230670$$

Thus, the 7~m/s landing velocity exceeds this threshold by 33\%.

Other problems with this solution are signalled by the discontinuities in the acceleration and jerk ($j=\frac{da}{dt}$). In practice, the ‘‘opening shock’’ (jerk) is a ‘‘heavy but smooth tug’’ \cite[p.~22]{af490}. Thus, the acceleration should be (at least) C^1 for $t>0$. In the textbook case, when k is piecewise constant, the acceleration and jerk are

```
> A := subs( k=K[pw_const], rhs(ode(SYS,v)) );
  J := subs( 'diff(v(t),t)'='a(t)', diff(A,t) );
```

$$A := -g - \frac{(k1 + (k2 - k1)\,\mathrm{Heaviside}(t - td))\,v(t)}{m}$$

$$J := -\frac{(k2 - k1)\,\mathrm{Dirac}(t - td)\,v(t)}{m} - \frac{(k1 + (k2 - k1)\,\mathrm{Heaviside}(t - td))\,a(t)}{m}$$

The jumps in the acceleration and jerk are the coefficients of \verb+Heaviside(t-td)+. Thus, the acceleration and jerk do not possess the necessary smoothness (unless $k_1=k_2$, {\em i.e.}, the parachute does not open). Explicit formulae for the jump in the jerk is not immediately available due to the fact that the jerk is expressed in terms of the acceleration, which is itself discontinuous.. The following procedure is useful for this calculation.

```
> jump := proc( f, p::equation )
```

```
    RETURN( factor( limit( f, p, left ) - limit( f, p, right ) ) );
  end:
```

The two jumps are now found to be

```
> Ajump := jump(A, t=td );
  Jjump := jump(subs(a(t)=A,J), t=td );
```

$$Ajump := -\frac{v(td)\,(-k2+k1)}{m}$$

$$Jjump := \frac{(-k2+k1)\,(v(td)\,k1+g\,m+v(td)\,k2)}{m^2}$$

\noindent
For the motion resulting from the use of the Nagle/Saff parameter values, the numerical values for these jumps are consistent with the graphical solution.

```
> subs( v(td)=subs(t=td,V), PARAM[ns],
        Ajump/g ):
  'Ajump' = "*G, `` =evalf(",6)*G;
```

$$Ajump = -6\,e^{(-12)}\,(e^{12}-1)\,G,\; = -5.99997\,G$$

```
> subs( v(td)=subs(t=td,V), PARAM[ns],
        Jjump/g ):
  'Jjump' = " * `G/s`, `` = evalf(",6) * `G/s`;
```

$$Jjump = \left(\frac{48}{5}\,e^{(-12)}\,(e^{12}-1)-\frac{6}{5}\right) G/s,\; = 8.39995\,G/s$$

```
>
```

\noindent
Even if a discontinuous acceleration or jerk were physically realistic, a snatch force and opening shock of this magnitude would be quite uncomfortable.

One way to overcome the deficiencies in the original model is to generalize the form of the coefficient of air resistance. The model proposed in \cite{meade} (see also \cite{mf}) adds a transition layer to the coefficient of air resistance during which the air resistance will increase from the free-fall value to the final descent value. That is,
\begin{equation} \label{k-cont}
k(t) =
\left\{
 \begin{array}{ll}
 k_1, & t<t_d \\
 k_d, & t_d \le t<t_d+\tau \\
 k_2, & t \ge t_d+\tau
 \end{array}
\right.
\end{equation}
where k_d is the coefficient of air resistance during deployment and τ is the length of

time required for the chute to be fully deployed. In Maple this could be represented by

```
> Kgen := k1 + (KD-k1)*Heaviside(t-td)
              + (k2-KD)*Heaviside(t-td-tau):
```

Note that choosing \$k_d\$ to be the linear interpolant forces the acceleration to be only continuous. One choice of \$k_d\$ that makes the acceleration continuously differentiable (on \$t>0\$) is the interpolating cubic polynomial with vanishing derivative at \$t=t_d\$ and \$t=t_d+\tau\$. To analyze the motions corresponding to these two new coefficients of air resistance, it is necessary to have general expressions for \$k\$.

```
> Kd[linear] := alpha*t+beta:
  Kd[cubic] := int( alpha*(t-td)*(td+tau-t), t )
                 + beta:
  TYPE := map( op, [ indices(Kd) ] ):
  for nam in TYPE do
    BC := Kd[nam]=kk;
    solve( { subs(t=td, kk=k1, BC),
             subs(t=td+tau, kk=k2, BC) },
             {alpha,beta} );
    DEPLOY := subs( ", Kd[nam] );
    K[nam] := subs( KD=DEPLOY, Kgen );
  od:
```

\noindent
The full set of coefficients available for analysis, including the original piecewise constant coefficient, can be displayed using

```
> print(K);  # output suppressed
```

table([

$$linear = k1 + \left(-\frac{(-k2+k1)\,t}{\tau} + \frac{-td\;k2 + k1\;td + k1\;\tau}{\tau} - k1 \right) Heaviside(t-td)$$

$$+ \left(k2 + \frac{(-k2+k1)\,t}{\tau} - \frac{-td\;k2 + k1\;td + k1\;\tau}{\tau} \right) Heaviside(t-td-\tau)$$

$$pw\_const = k1 + (k2-k1)\,Heaviside(t-td)$$

$$cubic = k1 + \left(-6\,\frac{(-k2+k1)\left(-\frac{1}{3}t^3 + \frac{1}{2}(2\,td+\tau)\,t^2 - td\,(td+\tau)\,t\right)}{\tau^3}\right.$$

$$\left. - \frac{-2\,td^3\,k2 + 2\,td^3\,k1 - 3\,td^2\,\tau\,k2 + 3\,td^2\,\tau\,k1 - k1\,\tau^3}{\tau^3} - k1 \right) Heaviside(t-td) + \left(k2 \right.$$

$$+ 6\,\frac{(-k2+k1)\left(-\frac{1}{3}t^3 + \frac{1}{2}(2\,td+\tau)\,t^2 - td\,(td+\tau)\,t\right)}{\tau^3}$$

$$+\frac{-2\,td^3\,k2+2\,td^3\,k1-3\,td^2\,\tau\,k2+3\,td^2\,\tau\,k1-k1\,\tau^3}{\tau^3}\Bigg)\mathit{Heaviside}(t-td-\tau)$$

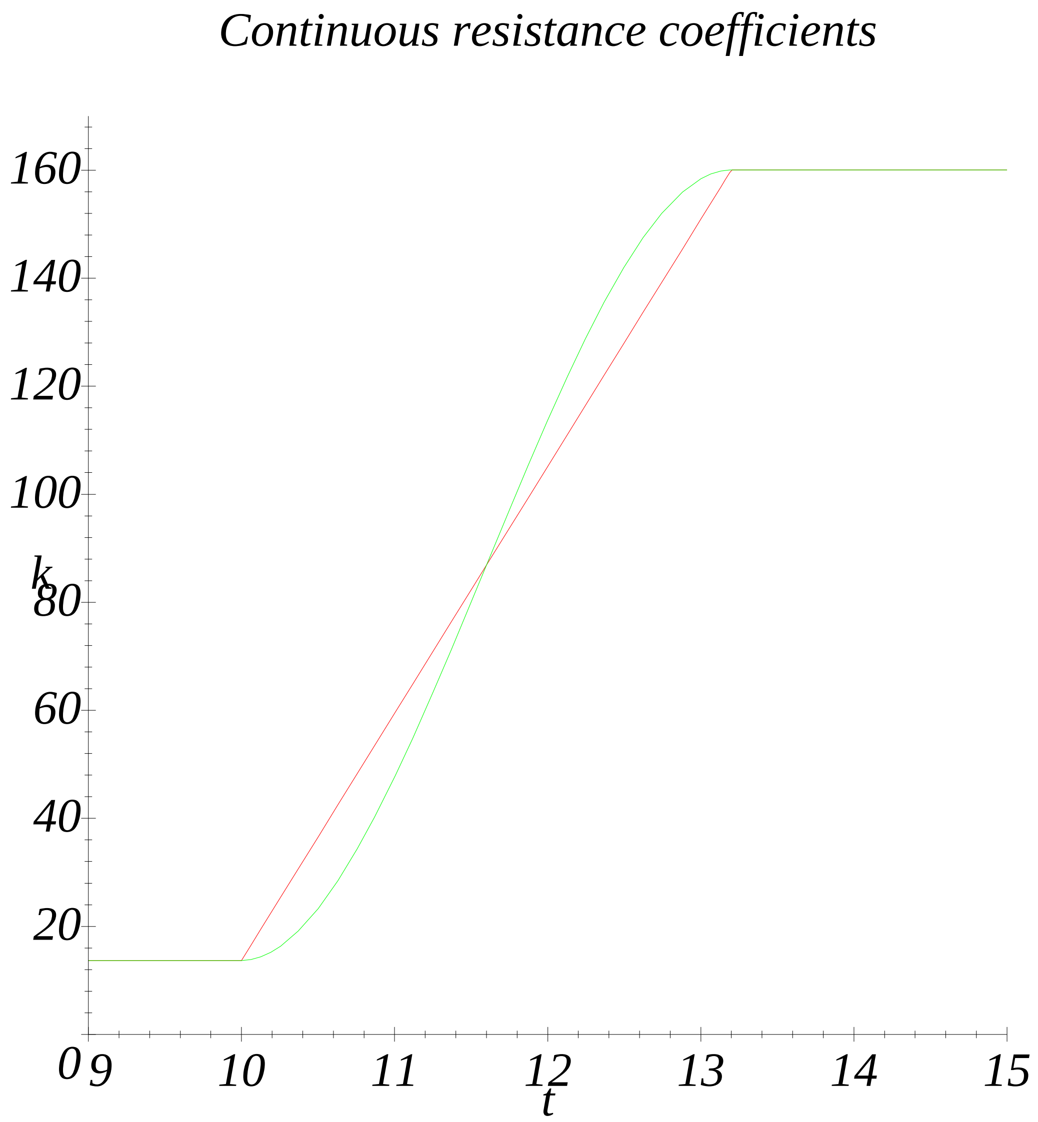
$])$

Explicit solutions for the linear and cubic coefficients can be obtained by a three-step process similar to the one used earlier for the piecewise constant coefficient. This process, and the accompanying results, is omitted due to the length and complexity of the solutions. Instead, each motion is computed using \verb+motion+ and displayed using \verb+plot_motion+. Since the Nagle/Saff parameters have already been shown to be non-physical, different values must be selected. The full set of parameters consistent with the Air Force Academy Training Guide \cite{af490} is

```
> PARAM[af] := k1=2/11*75, k2=32/15*75, td=10,
               tau=3.2, x0=4000*0.3048, m=75,
               g=981/100:
```

Prior to displaying the motion, it is a good idea to check that the new coefficients of air resistance have been properly defined. A plot focusing on the times during which deployment occurs is most appropriate.

```
> plot(subs( PARAM[af], { seq( K[i], i=TYPE ) } ),
       t=9..15, k=0..170,
       title=‘Continuous resistance coefficients‘);
```



\mapleplot{k-cont.ps}
\noindent
Students should be encouraged to identify anticipated qualitative differences (and similarities) between the two models before generating a graphical solution with \verb+plot_motion+. For example, observe that the jump height is significantly lower; thus, deployment and landing will occur earlier.

An appropriate set of times at which to record the motion is

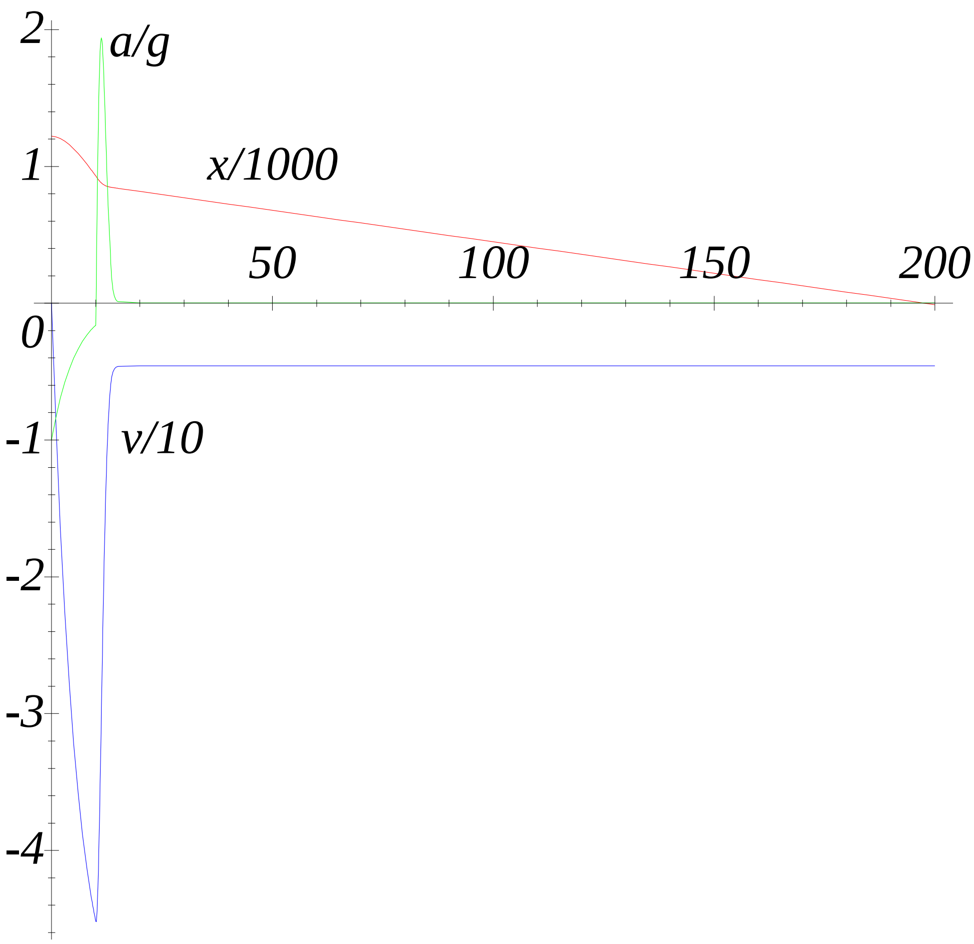
```
> snapshot[af] := [ $(0..8), i/10 $ i = 90 .. 149,
                   5*i $ i = 3 .. 40 ]:
```

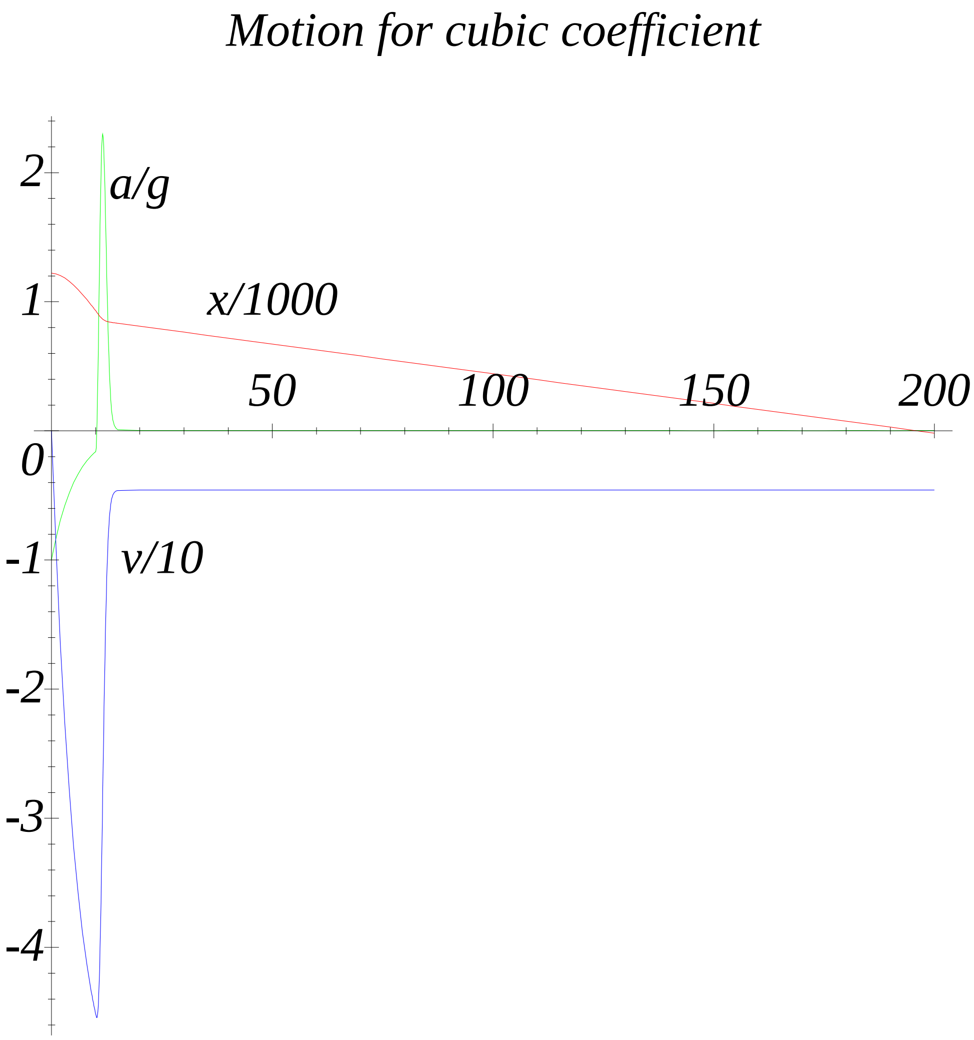
\noindent
To see the effect of the smoothness of k on the jumps, the two motions are displayed using \verb+plot_motion+.

```
> TYPE := [ linear, cubic ]:
```

```
for nam in TYPE do
  IVP := subs( k=K[nam], SYS ) union IC;
  PTS[nam] := motion( subs(PARAM[af],IVP),
                      snapshot[af] );
  print( display( { plot_motion( PTS[nam], scale[noJ] ),
                    textplot( [ [ 50, 1, `x/1000` ],
                                [ 25, -1, `v/10` ],
                                [ 20, 1.9, `a/g` ] ] )
                  },
                  title=`Motion for `.nam.` coefficient` ) );
od:
```

Motion for linear coefficient

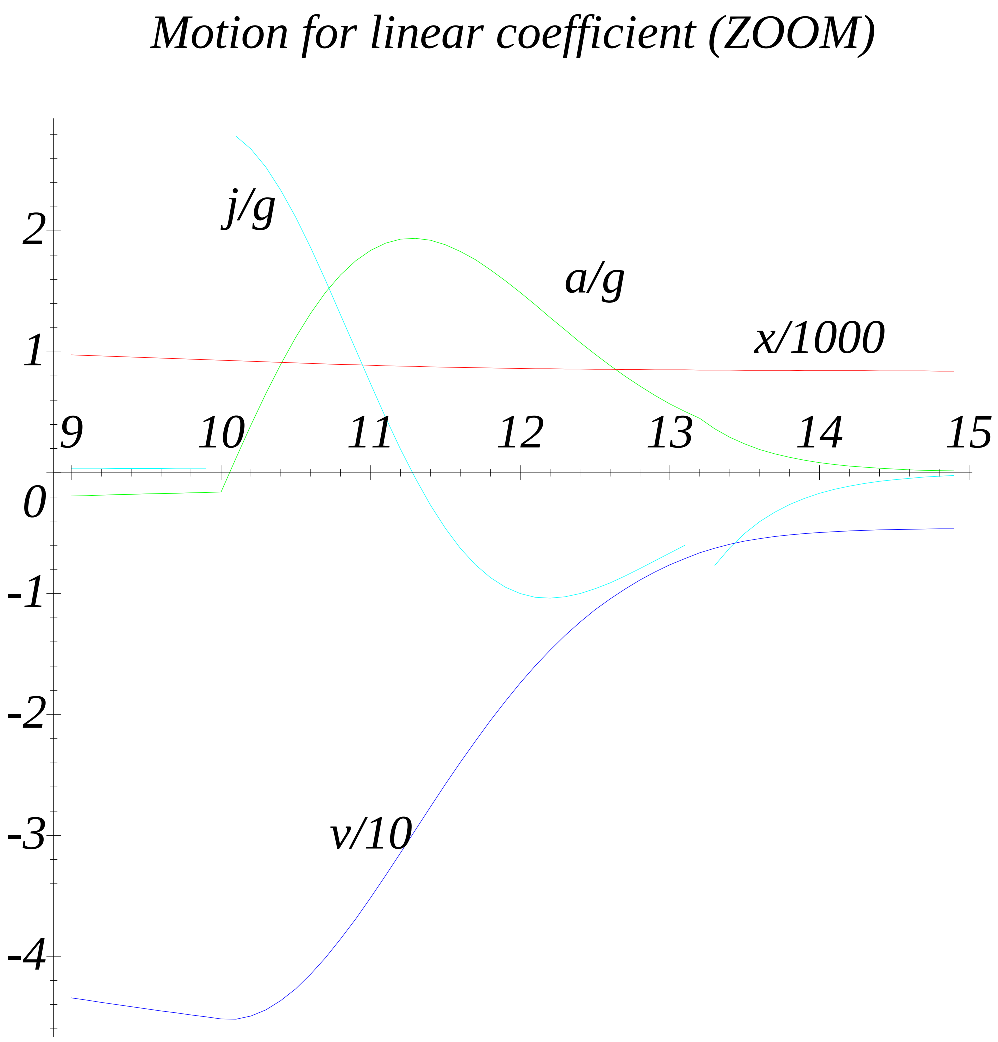


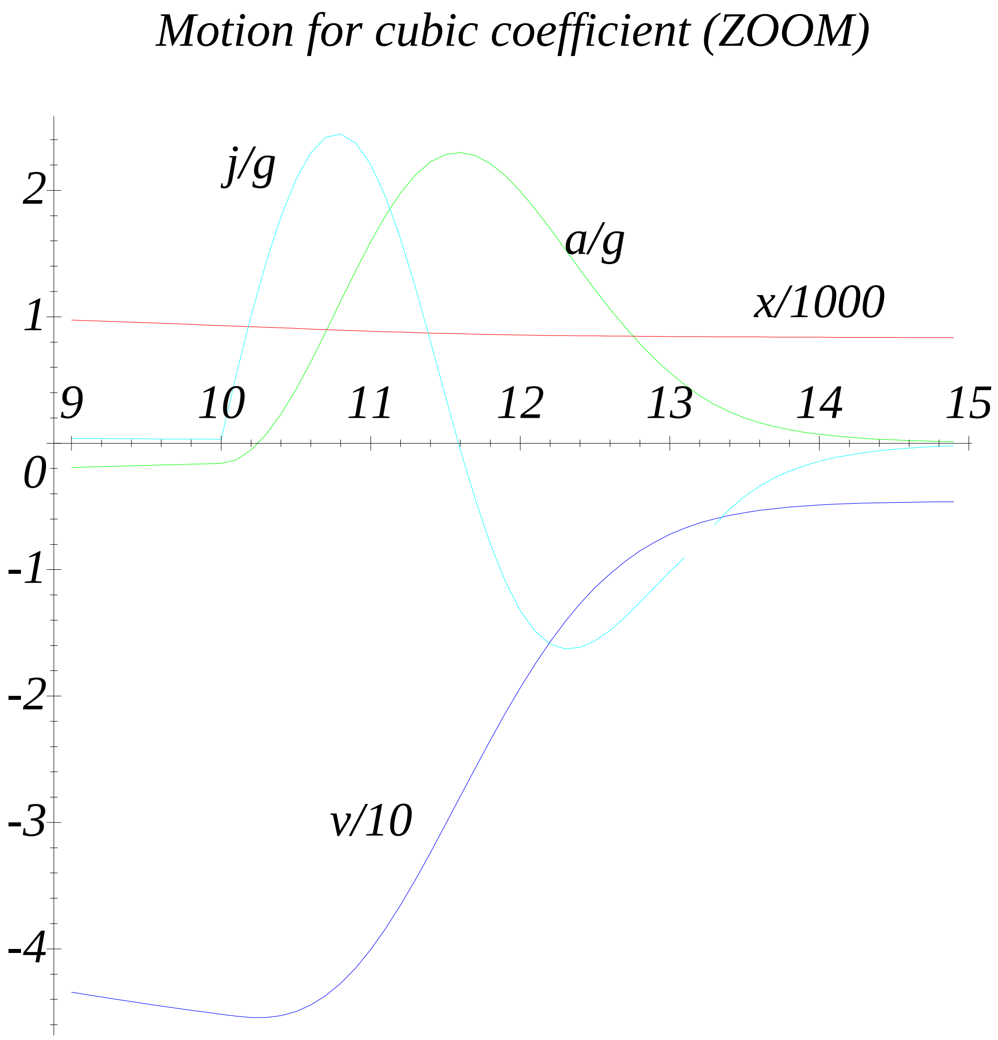


\mapleplot{motion-L.ps}
\mapleplot{motion-C.ps}
\noindent
The motions appear to be quite similar: impact occurs at $\approx$4.6~m/s after $\approx$196~s. This similarity is to be expected during the free-fall, and even in the final descent. For a closer look at the motion during deployment, zoom in on the previous plots and include the jerk in the pictures.

```
> scale[all] := { 'X'=1000, 'V'=10, 'A'=981/100,
                  'J'=981/100 }:
  for nam in TYPE do
    pts:=select( P->evalb(P[1]>=9 and P[1]<15),
                 PTS[nam] );
    print( display(
    {
```

```
    plot_motion( pts, scale[all], heading ),
    textplot( [ [ 14, 1.1, ‘x/1000‘ ],
                [ 11, -3, ‘v/10‘ ],
                [ 12.5, 1.6, ‘a/g‘ ],
                [ 10.2, 2.2, ‘j/g‘ ] ] )
  },
  title=‘Motion for ‘.nam.
        ‘ coefficient (ZOOM)‘ ) );
od:
```





\>

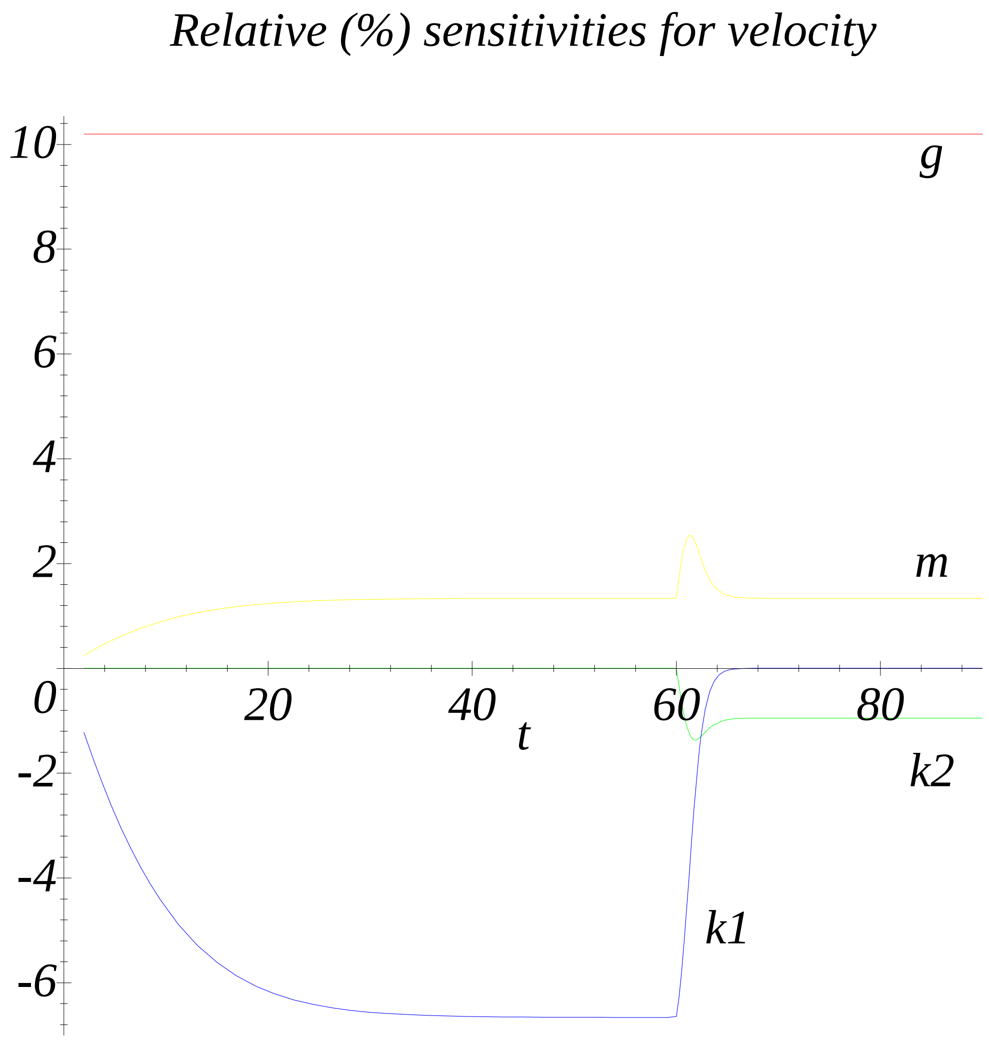
\mapleplot{motion-L-Z.ps}
\mapleplot{motion-C-Z.ps}
These plots illustrate several important differences between the two motions. Moreover, note that while the acceleration is continuous for all $t>0$ for both coefficients, the jerk is continuous only for the cubic interpolant. These issues are not solely of mathematical interest; these forces have a lot to say about the enjoyment -- even survivability -- of the jump. Additional analysis of these solutions can be found in~\cite{meade}.

\heading{Parameter Sensitivity}

Specific parameter values must be selected for the preceding numerical and graphical analysis. However, these values are known with only a limited amount of certainty. For example, landing velocity should be between 15 and 17~ft/s (4.6 and 5.2~m/s) and the gravitational constant changes with altitude --- particularly at altitudes above 25,000\ft \cite{zim}. How dependent is the solution on each of the parameters?

One means of addressing this is to examine the relative rate of change of the position and/or velocity with respect to each of the parameters of the problem. (Since the only explicit solution available at this time is the solution with piecewise constant air resistance, this solution will be used to illustrate these ideas.) The relationship between velocity and position means that the corresponding absolute sensitivities will be similarly related. This property can be difficult to verify in a plot of relative sensitivity; the fact that the position decreases to zero does not help the situation. For these reasons the remainder of this discussion will focus solely on the velocity.

```
> Vp := { seq( diff(V,p)/V*100,
          p=[ k1, k2, g, m ] ) }:
  Vplot := plot( subs( PARAM[ns], Vp ),
                 t=0..90 ):
  Vtext := textplot( [ [ 65, -5, 'k1' ],
                       [ 85, -2, 'k2' ],
                       [ 85, 9.8, 'g' ],
                       [ 85, 2, 'm' ] ] ):
  display( { Vplot, Vtext },
           title=`Relative (%) sensitivities`
                 .` for velocity` );
```



\mapleplot{sens-V.ps}
\noindent
The general shape of these curves seem quite reasonable: the solution is a multiple of g, so the corresponding relative sensitivity is simply $1/g\approx10.2\%$. The remaining curves show more temporal dependence, particularly before reaching terminal velocity. The curves for both m and k_2 are asymptotic to the reciprocal of the parameter value ($1/m\approx1.33\%$ and $1/k_2\approx0.95\%$). The k_1 curve approaches the corresponding asymptotic value, $1/k_1\approx6.7\%$, until the chute is deployed; after deployment, the value of this parameter quickly decays, as expected, to zero.

\heading{A Control Problem}

Consider the question: What is the latest time the parachute can be deployed while keeping the landing velocity below 17~ft/s? As this question can be answered using a numerical solution, the improved model will be used with parameter values (except deployment time) taken from \cite{af490}:

```
> PARAM[af2]:=k1 = 150/11, k2 = 160, tau = 32/10,
              x0 = 6096/5, m = 75, g = 981/100;
  IVP:=subs(k=K[cubic], PARAM[af2], SYS union IC):
```

$$PARAM_{af2} := k1 = \frac{150}{11}, k2 = 160, \tau = \frac{16}{5}, x0 = \frac{6096}{5}, m = 75, g = \frac{981}{100}$$

The key to the efficient solution of this problem is an intelligent search algorithm. Mathematically, the problem can be posed as a boundary value problem that can be solved using, {\em e.g.}, a shooting method (see \cite{meade2} for a Maple implementation of the shooting method). This is beyond the scope of the current discussion, so a hand search will be used.

To begin, note that $t_d=10$~s provides a lower bound; a crude upper bound is the length of a jump without opening the chute ({\em i.e.}, $k\equiv k_1$):

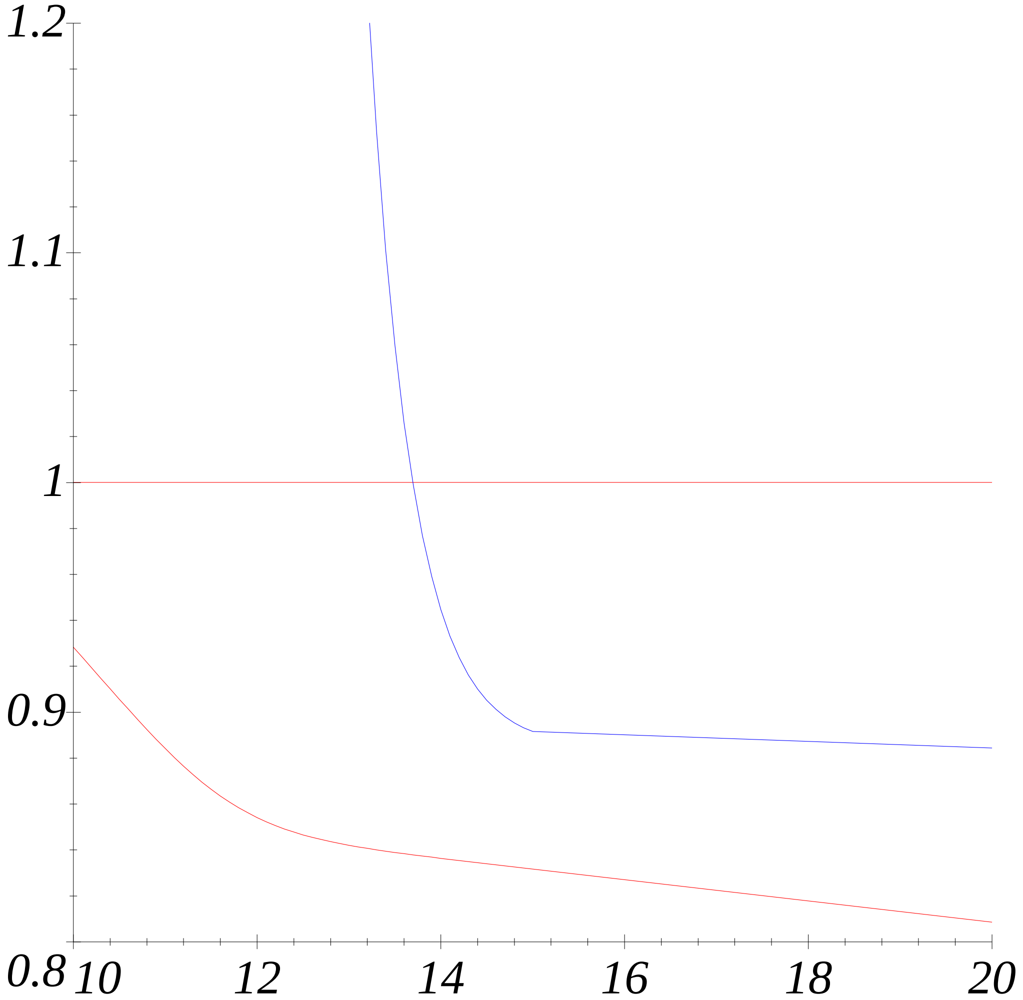
```
> XX := subs( PARAM[af2], X1 ):
  VV := subs( PARAM[af2], V1 ):
  TT := fsolve( XX=0, t, 0..infinity );
```

$$TT := 28.06315629$$

\noindent
A binary search on $[10,28]$ using \verb+motion+ could find the optimal value of t_d to within 0.1~s in 8 iterations.

The fact that execution times for \verb+motion+ are approximately linear in the number of elements in the second argument make this search prohibitively slow. The search will be more efficient when the information in the solution is used in a more intelligent manner to guide the iteration towards the optimal deployment time. For example, assuming the ripcord is pulled near terminal free-fall velocity, the skydiver falls approximately the same distance during the 3.2~seconds that it takes for the parachute to deploy; likewise, the rate of descent at the end of this period (≈6.5~m/s) is essentially independent of t_d. The same logic applies to the final descent. How long does it take the speed to decrease from 6.5~m/s to the maximum safe value, 5.2~m/s? The resolution of the plots given earlier is not suitable for the accurate determination of this information. More accurate estimates can be obtained from \verb+plot_motion+ with intelligent scale factors

```
> display( { plot_motion( PTS[cubic],
                { X=1000, V=-5.2 } ),
                plot( 1, 10..20, 0.8..1.2 ) } );
```



\mapleplot{zoom.ps}
Since $t_d=10$ in this plot, it is seen that the velocity reaches the threshold in just over 3.7~s after the ripcord is pulled; the distance fallen in this period is approximately 90~m. This means that the largest value of t_d which ensures the landing velocity is below 5.2~m/s can be estimated by finding the free-fall time that puts the skydiver 90~m AGL

```
> fsolve( XX=90, t, 0..infinity );
```

26.38314907

Note that the optimal value of t_d is expected to be less than this estimate. This information is useful in the determination of the points at which the motion should be computed.

```
> snapshot[af2] := [ 5*i$i=0..5,
                     i/10 $ i = 260 .. 264,
                     27, i/10 $ i=290..302 ]:
```

The motion for $t_d=26.383$~s can now be computed:

```
> pts := motion( subs( td=26.383, IVP ),
                 snapshot[af2] ):
```

The analysis requires more detail than can be obtained from the plot (besides, plots consume too much space!). Instead, it suffices to extract the information for the times closest to $t=t_d$ and the times when the velocity is closest to the threshold. Each data point contains five quantities: time, position, velocity, acceleration, jerk.

```
> select( P->evalb(abs(P[1]-26.383)<0.1),
          fnormal(pts,5) );
  select( P->evalb(abs(P[3]+5.2)<0.2),
          fnormal(pts,5) );
```

[[26.300, 94.449, -53.503, -.082215, .014948], [26.400, 89.098, -53.511, -.0719, 1.0477]]

[[30., -12.318, -5.3923, 1.6936, -3.6130], [30.100, -12.850, -5.2398, 1.3682, -2.9189],
[30.200, -13.367, -5.1166, 1.1054, -2.3581]]

The velocity threshold is not met until the skydiver is below ground level; thus, this value for t_d is too large. To estimate an appropriate adjustment to t_d, the acceleration value for $t=30.1$~s can be used to extrapolate the velocity to find that the threshold velocity was met about 0.0291~s later and travelled another 0.1523~m. Thus, the ripcord should have been pulled when the parachute was 13.0015~m higher; the free-fall velocity at 26.3~s suggests that deployment should have begun 0.243~s earlier. This gives a second estimate of $t_d=26.140$~s:

```
> pts:=motion(subs(PARAM[af2], td=26.14, IVP),
                  snapshot[af2]):
  select( P->evalb(abs(P[1]-26.14)<0.1),
          fnormal(pts,5) );
  select( P->evalb(abs(P[3]+5.2)<0.2),
          fnormal(pts,5) );
```

[[26.100, 105.15, -53.486, -.085260, .015502], [26.200, 99.799, -53.492, .0247, 3.5966]]

[[29.800, .48127, -5.3225, 1.5447, -3.2954], [29.900, -.043776, -5.1834, 1.2480, -2.6624],
[30., -.55630, -5.0710, 1.0082, -2.1509]]

```
>
```

\noindent
Extrapolating from the information at $t=29.9$~s, the velocity threshold is crossed at $t=29.887$~s with an extrapolated position of $x=0.022$~m = 2.2~cm $< 1^{\prime\prime}$.

Thus, a jump with $t_d=26.14$~s lands after 29.887~s with a landing velocity just below the 5.2~m/s threshold.\footnote{Of course, it must be remembered that these results are obtained by numerical integration of the equations of motion. Additional tests should be conducted to ascertain the reliability and accuracy of these results.} While it might be difficult to time the pulling of the ripcord this precisely and there is no time to consider using the reserve chute in an emergency, this does explain why some skydivers survive jumps when their chutes deploy only seconds before impact. In other words, it is almost never too late!

\heading{Conclusion}

In this paper we have conducted a graphical and symbolic analysis of a common textbook version of the parachute problem. The analysis identified several non-physical features of the original model and, subsequently, suggested the formulation of an improved model. While explicit solutions can be found for the new model, all analysis of the motions resulting from the improved model was either numerical or graphical. Many of the questions addressed in this paper --- including parameter sensitivity and solution of a control problem --- are not reasonable to attempt to solve using manual techniques. Maple, with its combination of symbolic, numeric, and graphic features, is the ideal software tool to use on these problems.

\heading{Acknowledgment}

Support for this project has been received from NSF grants
DMS-9404488 and OSR-9108722.

%The Maple worksheet (Release 3) from which this paper was produced is available on the WWW; the URL is:
%\centerline{\verb+http://www.math.sc.edu/~meade/maple/worksheets/parachute.ms+}
\thebibliography{10}

\bibitem{ab}
{\sc M.~L.~Abell and J.~P.~Braselton},
{\em Modern Differential Equations: Theory, Applications, Technology},
Harcourt Brace College Publishers, 1995.

\bibitem{drucker}
{\sc J.~Drucker},
{\em Minimal time of descent},
The College Mathematics Journal, 26 (1995), pp.~232--235.

\bibitem{ep}
{\sc C.~H.~Edwards,~Jr. and D.~E.~Penney},
{\em Differential Equations and Boundary Value Problems: Computing and Modeling
},
Prentice Hall, 1996.

\bibitem{ka}
{\sc E.~J.~Kostelich and D.~Armbruster},
{\em Introductory Differential Equations: From Linearity to Chaos},
Addison--Wesley, 1996.

\bibitem{meade}
 {\sc D.~B.~Meade},
 {\em ODE models for the parachute problem},
 SIAM Review, (submitted).

\bibitem{meade2}
 {\sc D.~B.~Meade, B.~S.~Haran, and R.~E.~White},
 {\em The shooting technique for the solution of two-point boundary value problems},
 MapleTech, vol.~3, no.~1, 1996, pp.~85--93.

\bibitem{mf}
 {\sc R.~Melka and D.~Farrior},
 {\em Exploration of the parachute problem with STELLA},
 Newsletter for the Consortium for Ordinary Differential Equations Experiments,
 Summer--Fall 1995, pp.~5--6.

\bibitem{ns}
 {\sc R.~K.~Nagle and E.~B.~Saff},
 {\em Fundamentals of Differential Equations}, Fourth Edition,
 Addison--Wesley, 1996.

\bibitem{af490}
 {\em Student Handbook for Airmanship 490: Basic Free Fall Parachuting},
 USAF Academy, May 1990.

\bibitem{zim}
 {\sc H.~S.~Zim},
 {\em Parachutes},
 Harcourt, Brace and Company, 1942.

\heading{Biography}

\subheading{Douglas B.\ Meade}
Douglas B.\ Meade received a Ph.D.\ in Mathematics from Carnegie Mellon University in 1989, then spent two years as a Research Assistant Professor in the Center of Applied Mathematics at Purdue University. Since 1991 he has been an Assistant Professor in the Department of Mathematics at the University of South Carolina. Research interests are presently directed towards numerical methods for PDEs, in particular, wave propagation on unbounded domains and nonlinear reaction--diffusion equations. Dr.\ Meade has been active in the development of Maple resources for use in a variety of undergraduate and graduate courses. A Maple supplement to the Nagle/Saff

ODE text was completed in 1996; a Maple module for the Engineer’s Toolkit series, co-authored with Etan Bourkoff, is nearing completion.

```
>
>
```