Snowboarding Down A Ski Trail
2011 Skadron Prize In Computational Physics

A snowboard going down a ski trail with moguls can be modeled as a point object moving on the trail surface while experiencing the gravitational force due to its mass as well as the frictional and the normal force exerted by the trail. For simplicity, we also assume that the snowboard always remains on the trail surface. By simulating the motion of the snowboard on a computer, we can follow its motion on the trail (for more technical detail, see a pdf file, "2011SkadronTechNotes," posted at www.phy.ilstu.edu/~hmb/hmb.html, where you also find a Mathematica notebook, with which you can visualize the trajectory of your snowboard).

For this year's Skadron prize, we ask you again to create a subroutine with which you steer the snowboard first at the initial time $t = 0$ and thereafter after every second by rotating its velocity vector in the tangent plane to the trail surface at the location of the snowboard by an angle between $-\pi/6$ radians $(= -30°)$ and $+\pi/6$ radians $(= +30°)$. The goal is to complete a ski trail as fast as you can.

For this year's competition, we choose a trail similar to but slightly different from the one used for the last year's competition. This year's trail is a parallelogram defined by

$$0 \le x \le 500 \text{ m}, \qquad (-20 - 0.08x) \text{ m} \le y \le (20 - 0.08x) \text{ m},$$

and

$$z = h(x,y) = -ax - dy - b\cos(px)\cos(qy),$$

where $a = 0.25$, $d = 0.01$, $b = 0.5$ m, $p = \pi/5 \text{ m}^{-1}$, and $q = \pi/2 \text{ m}^{-1}$. The trail is mostly sloped along the x-direction with a negative slope of –0.25 and slightly sloped along the y-direction with a negative slope of –0.01. The last term, $-b\cos(px)\cos(qy)$, models moguls, each of which has a height of $2b = 1$ m while their lengths along the down-slope or x direction are $\pi/p = 5$ m each and their widths along the cross-slope or y direction are $2\pi/q = 4$ m each.

Rules of the competition are:

(1) Your snowboard must stay inside the trail.

(2) You must complete the trail by crossing the finish line at $x = 500$ m within 200 seconds.

(3) The snowboard starts at (x_0, y_0, z_0), where $x_0 = 0$, y_0 is randomly selected from an interval [1.5 m, 2.5 m], and $z_0 = h(x_0, y_0)$. The initial velocity of the snowboard is (u_0, v_0, w_0), where $u_0 = 0$, $v_0 = 0.5$ m/s, and w_0 is calculated with the assumption that the velocity vector stays in the tangent plane to the trail surface at (x_0, y_0, z_0).

(4) Each contestant will be given the same three sets of initial conditions for the snowboard. The total computation time for the three runs starting with these initial conditions must be within 5 minutes.

(5) The first prize goes to the contestant whose subroutine gets the snowboard to finish the trail in the fastest time.

Challenge

Write a FORTRAN subroutine "angle.f" that must start with the following line:

subroutine angle (theta, xa, ya, za, ua, va, wa, xi, yi, ui, vi, it)

theta, xa, ya, za, ua, va, wa, xi, yi, ui, and vi are all declared as double precision variables in the main program, "skmain11.f," provided by the committee whereas "it" is declared as an integer variable.

theta is the angle by which you want to rotate the current velocity vector, (ua, va, wa), of the snowboard in the tangent plane to the slope at the current location, (xa, ya, za), of the snowboard.

xi, yi, ui, and vi are array variables of size "it". (xi(j), yi(j)) and (ui(j), vi(j)), where $1 \le j \le it$, provide the recent history of the horizontal component of the position vector and the horizontal component of the velocity vector of the snowboard during one second since the last time you have steered the snowboard. They represent the numerical solution of the equations of motion for (x, y, u, v) and are generated by a 4-th order Runge-Kutta subroutine with a time step size of $dt = 0.01$ sec called from the main program.

The Skadron Prize committee will run your subroutine "angle.f" with the main program "skmain11.f" on our computer "meitner." The main program checks if the angle, theta, selected by your subroutine is inside the allowed range of $[-\pi/6 \text{ rad}, \pi/6 \text{ rad}]$, numerically solves the equations of motion for the snowboard using the 4-th order Runge-Kutta subroutine, and checks if the snowboard remains in the trail as well as if the snowboard has crossed the finish line.

To test your subroutine "angle.f" on our computer "meitner":

1. Copy the main program "skmain11.f" on "meitner" to your account on "meitner":

   copy ~hmb/hmb/skmain11.f skmain11.f

2. Compile and run your subroutine with the main program:

   g95 skmain11.f angle.f
   a.out

Prizes: $ 200 for the first place, $100 for the second place. The winners will be announced at the annual physics department award ceremony on Tuesday, April 19, 2011.

Who can participate: Physics majors at ISU.

Deadline: Submit your subroutine by attaching it to an email addressed to

hmb@phy.ilstu.edu

by 10 a.m. on Monday, April 18, 2011.

Appendix: Plotting a trajectory of the snowboard using Mathematica or Kaleidagraph

To visualize the trajectory of the snowboard, first copy the data file "snowboard.dat" generated by the main program to an iMac you are using. You can then use "Mathematica" or "Kaleidagraph" to plot the trajectory.

Mathematica

1. To use "Mathematica," you need to download a "Mathematica" notebook "Skadron11.nb" posted on the following web page:

   www.phy.ilstu.edu/~hmb/hmb.html

2. By clicking on the icon for the notebook, you can launch "Mathematica."
3. To have "Mathematica" execute a command or a set of commands (not the header describing the corresponding task), move the cursor to a bracket to the right of the command or commands near the right edge of the page and click on it and then press the "enter" key (not the "return" key) located at the lower right corner of the keyboard.

Kaleidagraph

You can also use "Kaleidagraph" to plot the trajectory as follows:

1. Launch "Kaleidagraph."
2. Use the "Open" command in the "File" pull-down menu in "Kaleidagraph" to open "snowboard.dat." Before locating "snowboard.dat," enable "All Documents" in the dialog box.
3. In the "Text File Input Format" dialog box, select "Space" for "Delimiter:" and ">=1" for "Number:". Type in "1" in the box below "Lines Skipped" and deselect "Read Titles" under "Options:".
4. You find four columns filled with numbers: the column 1 lists the elapsed time in seconds for the snowboard run; the column 2 lists the *x*-coordinate of the snowboard; the column 3 lists the *y*-coordinate of snowboard; the column 4 lists the *x*-component of the velocity vector of the snowboard; the column 5 lists the *y*-component of the velocity vector of the snowboard.
5. You can also create two new columns for the *y*-coordinates of the points along the two boundaries of the ski trail defined by $(-20-0.08x)\text{ m} \le y \le (20-0.08x)\text{ m}$. First, you need to append a column by using the "Append Columns…" command in the "Data" pull-down menu. You can then use the "Formula Entry" command in the "Windows" pull-down menu to calculate the *y*-coordinates for the points along one of the boundaries by typing in the following command in the "Formula Entry" window and clicking on the "Run" button: c6 = – 20 – 0.08 * c2. For the other boundary, use c7 = 20 – 0.08 * c2.
6. You can plot the trajectory and the boundaries each as a function of the *x*-coordinates of the snowboard in the column 2 using a linear line plot with the data in the column 2 as those for the *x*-axis and the data in the columns 3, 6, and 7 as those for the *y*-axis .