Mathematical Models: Designing a Roller Coaster

Douglas Meade, Ronda Sanders, and Xian Wu
Department of Mathematics

Overview

There are three objectives in this lab:

- understand the mathematical reasoning associated with a real-world example,
- learn to define a piecewise-defined function in Maple, and
- learn to set up and solve a system of equations in Maple.

Maple Essentials

Important Maple commands introduced in this lab are:

Command	Description
`piecewise`	define a piecewise-defined function The general syntax to represent $\begin{cases} f_1, & cond_1 \\ f_2, & cond_2 \\ \vdots & \vdots \\ f_n, & cond_n \end{cases}$ is: `piecewise(` $cond_1$ `,` f_1 `,` $cond_2$ `,` f_2 `, ... ,` $cond_n$ `,` f_n `);` where each $cond_i$ is an inequality and each f_i is an expression.
`diff`	`diff(f(x), x);` finds the derivative of $f(x)$ with respect to x.
`solve`	solve an equation or system of equations `solve(` *`eqn, var`* `);` solves an equation, *eqn*, for one variable, *var*. `solve(` {*`eqn1, eqn2`*}, {*`var1, var2`*} `);` solves a system of two equations for two variables.
`assign`	`assign(values); assigns a set of values`

Maple does not recognize double inequalities, so if your condition is $a \leq x < b$ you would write `x>=a and x<b`.

Preparation

§2.8 and §4.1 of Stewart. Review properties of the graph of the first derivative.

Assignment

This week's assignment is to design a larger roller coaster that meets given specifications and prepare a neat and complete project report. **Project 1 will be due at the *beginning* of the next lab.**

The Problem: Design a Roller Coaster

Suppose we are asked to design a simple ascent and drop roller coaster with an overall horizontal displacement of 200 feet. By studying pictures of our favorite roller coasters, we decide to create our roller coaster using a line, a parabola and a cubic. We begin the ascent along a line $y = f1(x)$ of slope $\frac{4}{3}$ for the first 20ft horizontally. We continue the ascent and begin the drop along a parabola $y = f2(x) = ax^2 + bx + c$ for the next 100ft horizontally. Finally, we begin a soft landing at 40ft above the ground along a cubic $y = f3(x) = dx^3 + ex^2 + fx + g$ for the last 80ft.

Here are our tasks:

1. Find a system of 7 equations with the 7 unknowns ({a,b,c,d,e,f,g}) that will ensure that the track is smooth at transition points.
2. Solve the equations in (1) to find our functions. (We should get a unique solution as we have the same number of equations and unknowns.)
3. Plot the graph to see the design.
4. Find the maximum height of the roller coaster.

Solving the Problem

1. Start your Maple session with
   > `restart;`
   This clears the internal memory so that Maple acts (almost) as if just started and is very helpful if you make a mistake and want to start over.
2. We begin by defining our functions in Maple. If we choose the origin as our starting point, our first function $y = f1(x)$ is a line of slope $\frac{4}{3}$ that passes through (0,0), and we have:
   > `f1:=x-> 4/3*x;`
   > `f2:=x-> a*x^2+b*x+c;`
   > `f3:=x-> d*x^3+e*x^2+f*x+g;`
3. We will also need the first derivatives of our functions. (If you do not see why, you will soon.) To find and assign the derivatives, right-click over the function and choose **differentiate**. Then right-click over the derivative function and choose **assign to a name**. Name the derivatives $df1, df2$, and $df3$, respectively.
4. Since our roller coaster consists of three curves, it can be set up mathematically as a piecewise-defined function:
   $$F(x) = \begin{cases} f1(x), & 0 \le x \le 20 \\ f2(x), & 20 < x < 120 \\ f3(x), & 120 \le x \le 200 \end{cases}$$
   We assign F as a function in Maple as follows:
   > `F:= x -> piecewise(x<=20, f1(x), x>20 and x<120, f2(x), x>=120 and x<=200, f3(x));`
   Note: *You can verify your piecewise-defined function by typing* `F(x);`
5. Obviously, we want $F(x)$ to be continuous (so our passengers do not perish). This means that our functions should be equal at transition points. So we get the following equations:
   > `eq1:=f1(20)=f2(20);`
   > `eq2:=f2(120)=f3(120);`
6. If we are to have a smooth track, we cannot have abrupt changes in direction, so the first derivative $F'(x)$ should also be continuous. That is, the first derivatives of our functions should also be equal at transition points. So we get:
   > `eq3:=df1(20)=df2(20);`
   > `eq4:=df2(120)=df3(120);`
7. To start our landing at 40ft above the ground for the last 80ft, we would have:
   > `eq5:=f3(120)=40;`
8. Finally, in order to have a soft landing, the track should be tangent to the ground at the end:
   > `eq6:=f3(200)=0;`
   > `eq7:=df3(200)=0;`
9. We now have a system of 7 equations and 7 unknowns. We solve using the `solve` command and assign the solutions as follows:
   > `values:=solve({eq1,eq2,eq3,eq4,eq5,eq6,eq7} , {a,b,c,d,e,f,g});`
   > `assign(values);`
10. You can view your completed piecewise-defined function by typing
    > `F(x);`
    Note: *If I were preparing a project report about this roller coaster, I would definitely include this function.*
11. We can see what our coaster looks like with the following `plot` command:
    > `plot(F(x), x=0..200, scaling=constrained);`
    Note: *The option* `scaling=constrained` *scales the x and y dimensions equally.*
12. To find the maximum height, we need to find where the graph has a horizontal tangent line (where $F'(x) = 0$) and evaluate $F(x)$ at each point. The largest is the maximum height of the coaster.
    > `diff(F(x),x);` **Note:** *You can get this by right-clicking over $F(x)$ above also.*
    > `dF:= x -> ` *`label`*`;`
    > `solve(dF(x)=0, x);`
    Note: *We know that $F'(x) = 0$ when we have a horizontal tangent line (a slope of 0). This occurs at both local maximums and minimums.*