Inverting GPR Distance Curves to Obtain Anomaly Geometries

Timothy J. McLarnan
Department of Mathematics, Earlham College

Summary

The distances measured by reflecting ground-penetrating radar from a buried cylindrical object of unknown size and orientation (an anomaly) unfortunately depend only weakly of the geometry of the anomaly. A series of examples will show that a single straight-line scan across an anomaly does not suffice to obtain the object's vertical or horizontal dimension, its curvature, or even its rough size. An alternative multiple-scan method is proposed in which 2 initial parallel scans determine the orientation and dip angle of the cylinder, and third scan is then made at an angle based on this data. This alternative seems to allow reasonably accurate estimates of the size of a cylindrical object, on the assumption that each scan can be analysed based on the planar geometry of the vertical section of the anomaly beneath the track of the GPR device.

Statement of the Problem

The basic problem is to image a buried cylindrical object using ground-penetrating radar. More precisely, we assume we are given a buried object of cylindrical cross-section but of unknown orientation. A GPR unit moves above this object in a straight line. The vertical plane in which the GPR unit moves therefore cuts the buried cylinder to form an ellipse. This ellipse could be a circle if the cylinder is horizontal and is sliced perpendicular to its axis. It will be an ellipse whose horizontal axis is longer than its vertical axis if the cylinder is horizontal and is sliced obliquely. It will be an ellipse whose vertical axis is longer than its horizontal axis if the cylinder is, say, inclined with the upper end pointing north and if the GPR unit moves across it in the east-west direction.

We assume that the reflection of the radar signal can be understood entirely by considering the plane of the GPR unit and this ellipse. The radar signal propagates in the plane and reflects from the point on the ellipse whose normal is oriented directly toward the GPR unit. Measuring the travel time of the signal gives us a distance from the GPR unit to this point on the ellipse, but does not tell us where this point is located in space.

By moving the GPR unit along a line over the buried cylinder, we obtain a curve of distances from the radar unit to the cylinder as a function of the position of the radar unit. The problem is to use this curve to infer the size and shape of the elliptical cross-section of the cylinder over which the GPR unit travels.

For concreteness, I'll assume that typical objects we are looking for measure a few centimeters from the center to the edge of the ellipse, and that they are buried roughly 25 cm deep. This would be typical of an 81 mm mortar round 10 inches underground. I'll assume the GPR unit returns a measurable signal from points on the surface no more than 50 cm away from the point directly above the center of the object, and that observed distances are accurate to within a few millimeters. The elliptical cross-section, the ground-surface path of the GPR unit, and the curve of measured distances are shown below.

(Throughout this paper, I've left in place the *Maple* commands needed to produce plots. I have not justified these formulas, assuming that my readers are after the results, not the details. My advice is to ignore the *Maple* commands and to read the text and the pictures, which should speak for themselves.)

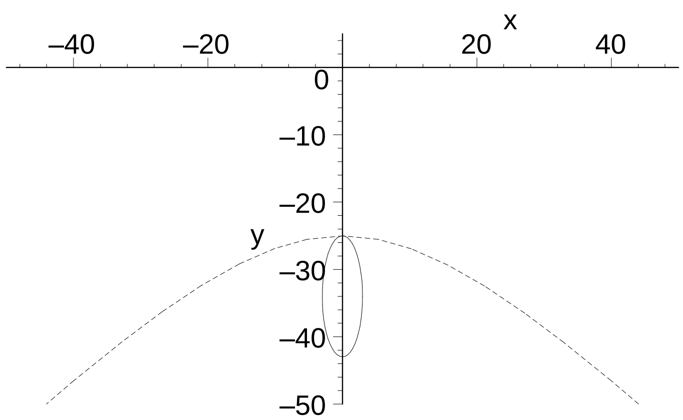
```
> xcoord := (theta,a,b,h) ->
  (a^2-b^2)/a*cos(theta)+b*h/a*cot(theta);
```

$$xcoord := (\theta, a, b, h) \rightarrow \frac{(a^2 - b^2)\cos(\theta)}{a} + \frac{b\,h\cot(\theta)}{a}$$

```
> len := (theta,a,b,h) -> sqrt((xcoord(theta,a,b,h)-a*cos(theta))^2
  + (h-b*sin(theta))^2);
```

$$len := (\theta, a, b, h) \rightarrow \sqrt{(\mathrm{xcoord}(\theta, a, b, h) - a\cos(\theta))^2 + (h - b\sin(\theta))^2}$$

```
> picture := (a,b,delta,xrange,yrange) ->
  plot([[a*cos(t), b*sin(t)-(b+delta), t=0..2*Pi],
  [xcoord(theta,a,b,b+delta), -len(theta,a,b,b+delta),
  theta=0..Pi]],
  xrange,yrange, color =[black, black], linestyle=[1,3],thickness=1,
  scaling=CONSTRAINED):
> picture(3,9,25,x=-50..50,y=-50..5);
```



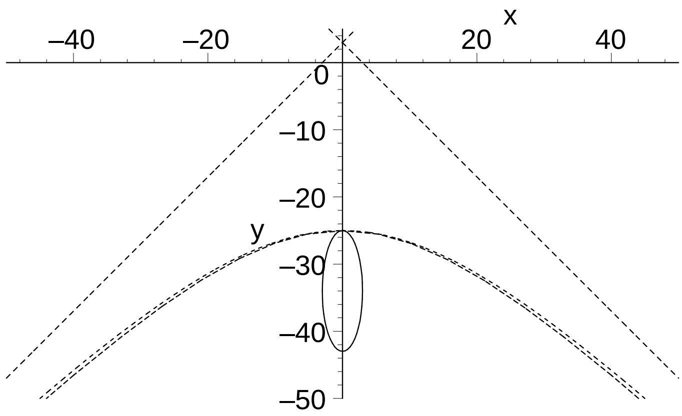
The ground here is at height 0. The cylinder’s elliptical cross-section is a solid line. The dashed curve represents the distance function. In other words, the distance of the dashed curve below the x -axis at a given value of x is the distance from which a radar unit located at x receives a reflection from the ellipse.

The problem is to use the measured dashed curve (the GPR distance curve) to infer the shape of the buried ellipse.

What We Know About the Distance Curve

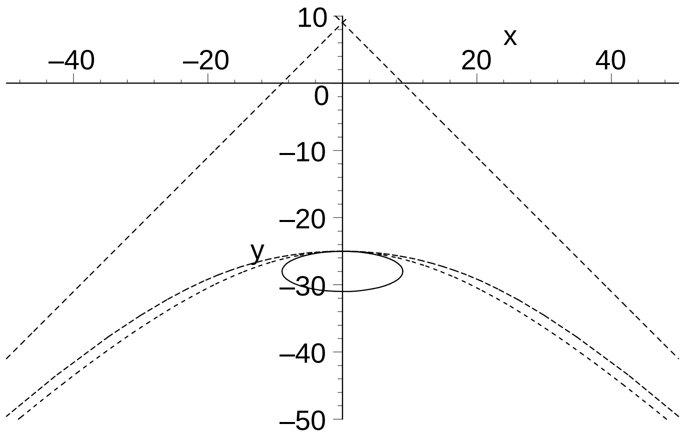
It would be natural to assume that the GPR distance curve is a hyperbola, but this turns out not to be true. In fact, its exact analytic description seems to be a complicated mess involving 4th degree equations, though it can be plotted parametrically without much trouble. On the other hand, it has lots of properties in common with hyperbolas. It comes to a maximum at the top point on the ellipse, and it drops off to approach asymptotically a pair of lines inclined 45 degrees to the horizontal. The next picture shows the ellipse and distance function above, and adds the asymptotes (dotted) and the unique hyperbola (dot-dash) having the same apex and asymptotes as the distance curve:

```
> picture := (a,b,delta,xrange,yrange) ->
  plot([[a*cos(t), b*sin(t)-(b+delta), t=0..2*Pi],
  [xcoord(theta,a,b,b+delta), -len(theta,a,b,b+delta), theta=0..Pi],
  -x+a, x+a, a-sqrt(x^2+(a+delta)^2)],
  xrange,yrange, color =[black, black,black,black,black],
  linestyle=[1,3,2,2,4], scaling=CONSTRAINED):
> picture(3,9,25,x=-50..50,y=-50..5);
```



The distance curve lies slightly below the hyperbola, which is typical for vertical ellipses; horizontal ellipses have the distance curve above the hyperbola, and for circles, the distance curve is a hyperbola. For very flat ellipses, the hyperbola and the distance curve are farther apart:

```
> picture(9,3,25,x=-50..50,y=-50..10);
```



Both the hyperbola and the distance curve are asymptotic to the dotted lines. It is easy to prove that the dotted asymptotes meet at a point on the *y*-axis the same height above ground as the half-width of the ellipse (3 in the upper figure, and 9 in the lower figure). Unfortunately, reading the asymptotes from the curves is not so easy, as we shall see further below.

The Bad News

The bad news is that inverting the distance curves to extract information about the ellipses is almost impossible, for reasons that in retrospect seem rather obvious. Consider all the ellipses whose topmost points are at some constant depth (say, 25 cm). All these infinitely many ellipses will give rise to GPR distance curves that have roughly the same properties: they will peak at the top point of the ellipse and will be asymptotic to lines of slope 1 and -1. Ellipses of very different sizes and shapes turn out to give rise to distance curves which are almost indistinguishable. In the next few sections, I'll look quickly at what we are unable to learn from the distance curves.

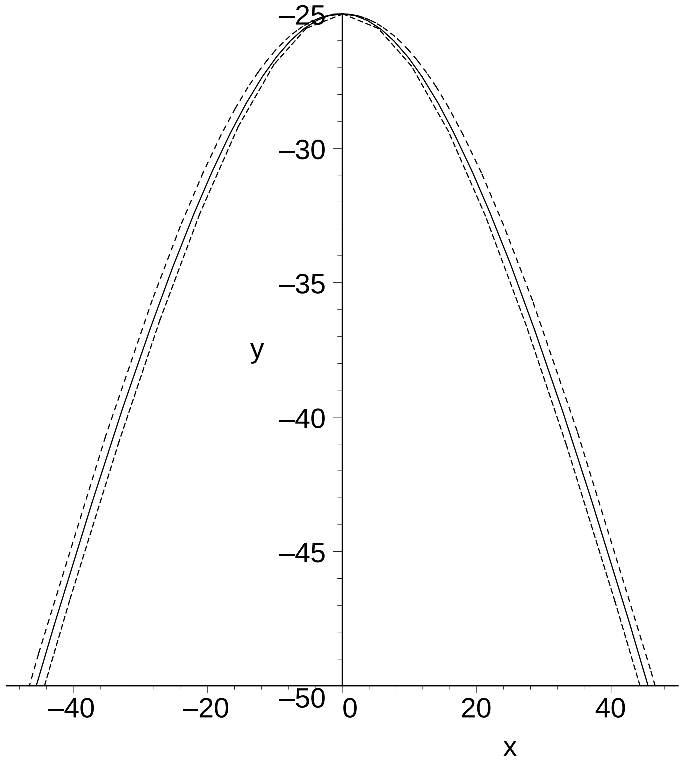
We Can't Determine the Vertical Dimension of the Ellipse

Since all the radar signals available for us to analyse are reflected off the top surface of the ellipse (and not even all that far down the sides), one might guess that getting the vertical dimension of the ellipse would be difficult. Indeed, even given the horizontal measure of the ellipse, the vertical measure is difficult to infer from the distance curve.

The figure below shows the theoretical distance curves that would be measured for 3 ellipses, all with horizontal width 8 cm, and all buried so that the top surface is 25 cm down. The dashed curve is for an ellipse of vertical height 24 cm, the solid curve is for a circle of height 8 cm, and the dotted curve is for an ellipse of height 2.67 cm.

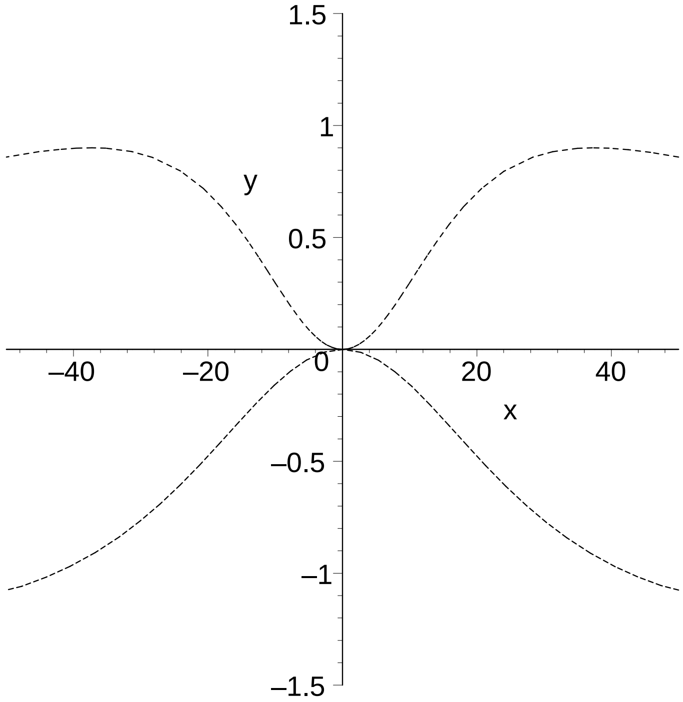
```
> plot([[xcoord(theta,4,12,12+25), -len(theta,4,12,12+25),
  theta=0.01..Pi-0.01],
```

```
[xcoord(theta,4,1.33,1.33+25), -len(theta,4,1.33,1.33+25),
theta=0.01..Pi-0.01],
[xcoord(theta,4,4,4+25), -len(theta,4,4,4+25),
theta=0.01..Pi-0.01]], x=-50..50,y=-50..-25,
color =[black,black,black],
linestyle=[3,2,1],scaling=UNCONSTRAINED);
```



It is clear that all three of these curves are quit close together. Here is a plot of their differences, dashed-solid and dotted-solid:

```
> plot([[xcoord(theta,4,12,12+25),
-len(theta,4,12,12+25)-(4-sqrt(xcoord(theta,4,12,12+25)^2+(4+25)^2
)), theta=0.01..Pi-0.01],
[xcoord(theta,4,1.33,1.33+25),
-len(theta,4,1.33,1.33+25)-(4-sqrt(xcoord(theta,4,1.33,1.33+25)^2+
(4+25)^2)), theta=0.01..Pi-0.01],0],
x=-50..50,y=-1.5..1.5,color=[black,black,black],linestyle=[3,2,1],
scaling=UNCONSTRAINED);
```



The implication is this: even if we are told the width of the ellipse, we can only tell a circle from an ellipse three times as tall as it is wide or three times as wide as it is tall if our GPR distances are accurate to a couple of millimeters when the unit is 20 cm from directly above the anomaly, or if our measurements are accurate to about 5 mm when we are 40 or 50 cm out from the anomaly. Unless accuracy on this level can be guaranteed, then even the artificial hypothesis that we know the width of the ellipse doesn't let us determine its height.

We Can't Determine the Horizontal Dimension of the Ellipse

Here we plot predicted distance curves for three families of ellipses. The dashed curves are ellipses with width 8 cm, the solid curves have width 12 cm, and the dotted curves have width 4 cm. Within each family, there is a circle, an ellipse 3 times as high as it is wide, and an ellipse 3 times as wide as it is high. All ellipses have top points 25 cm deep.

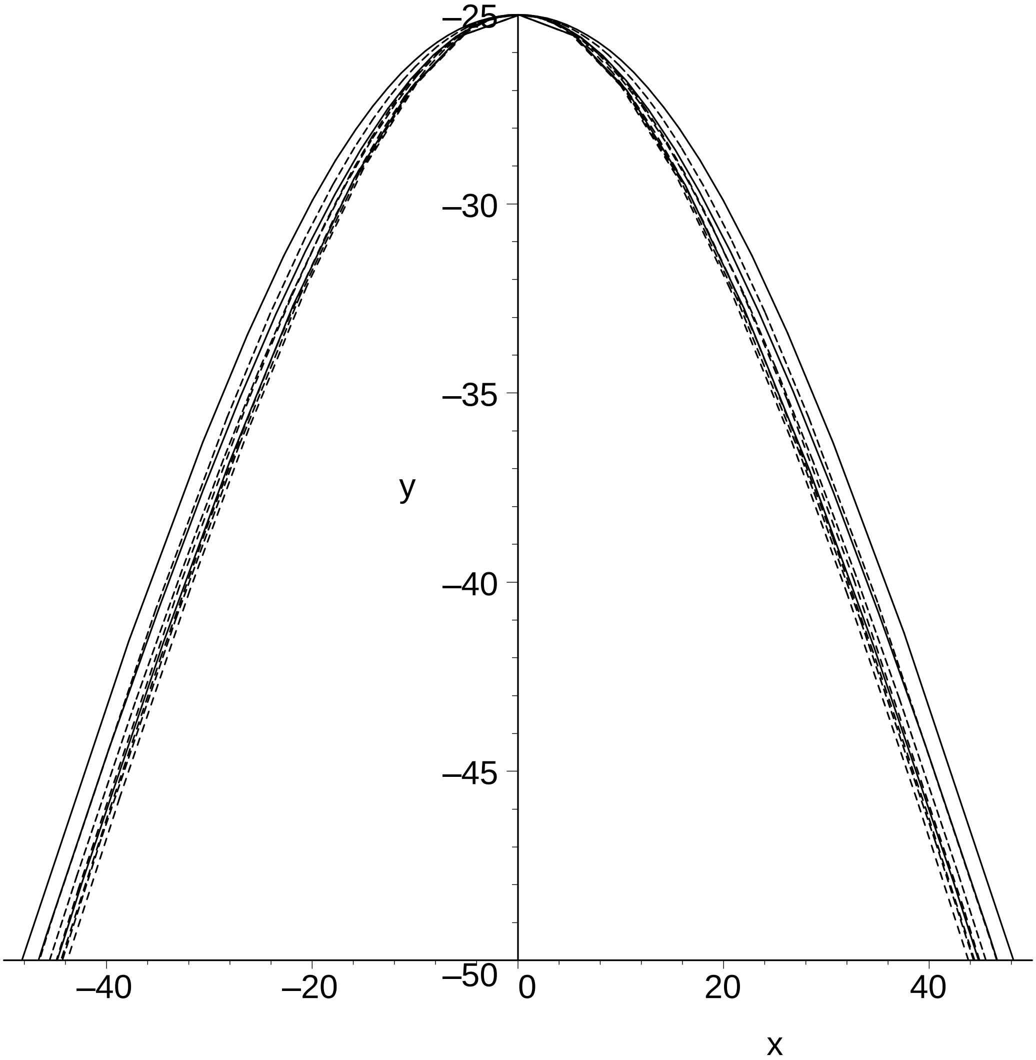
The critical observation is that there is significant overlap among these three families. There are 8 cm and 4 cm ellipses whose predicted distance curves are within 2 mm of each other over the whole field, and there are 12 cm ellipses which are almost this close to the others. It is hard to imagine that curves this close together could be distinguished in the field; if they cannot be, then we can't determine the horizontal widths of our ellipses. Here are the distance curves themselves:

```
> plot([[xcoord(theta,4,12,12+25), -len(theta,4,12,12+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,1.33,1.33+25), -len(theta,4,1.33,1.33+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,4,4+25), -len(theta,4,4,4+25),
  theta=0.01..Pi-0.01],

  [xcoord(theta,6,18,18+25), -len(theta,6,18,18+25),
```

```
theta=0.01..Pi-0.01],
[xcoord(theta,6,2,2+25), -len(theta,6,2,2+25),
theta=0.01..Pi-0.01],
[xcoord(theta,6,6,6+25), -len(theta,6,6,6+25),
theta=0.01..Pi-0.01],

[xcoord(theta,2,6,6+25), -len(theta,2,6,6+25),
theta=0.01..Pi-0.01],
[xcoord(theta,2,0.67,0.67+25), -len(theta,2,0.67,0.67+25),
theta=0.01..Pi-0.01],
[xcoord(theta,2,2,2+25), -len(theta,2,2,2+25),
theta=0.01..Pi-0.01]],
x=-50..50,y=-50..-25,
color =[black,black,black,black,black,black,black,black,black],
linestyle=[3,3,3,1,1,1,2,2,2],
scaling=UNCONSTRAINED);
```



And here are the differences between these curves and the distance curve for a sphere of radius 4 cm:

```
> ball4 := x -> 4-sqrt(x^2+(4+25)^2);
```

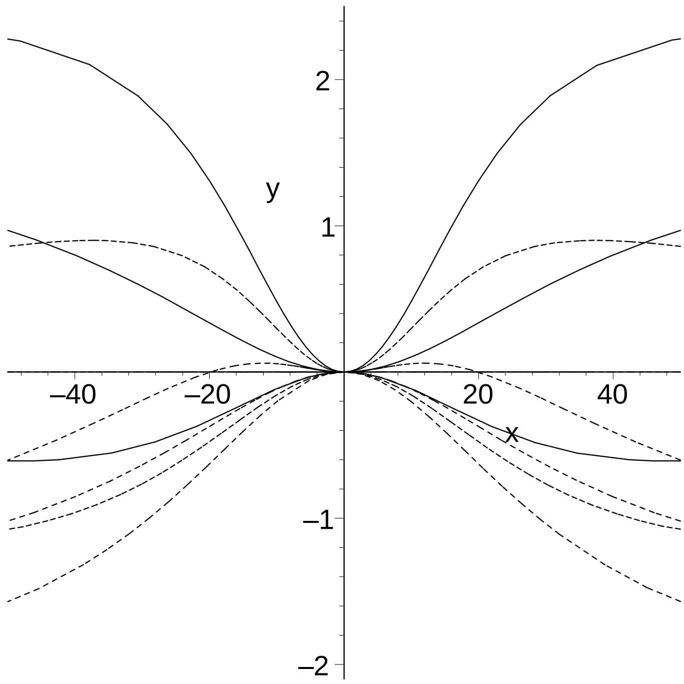
$$ball4 := x \to 4 - \sqrt{x^2 + 841}$$

```
> plot([[xcoord(theta,4,12,12+25), -len(theta,4,12,12+25)
  -ball4(xcoord(theta,4,12,12+25)), theta=0.01..Pi-0.01],
  [xcoord(theta,4,1.33,1.33+25),
  -len(theta,4,1.33,1.33+25)-ball4(xcoord(theta,4,1.33,1.33+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,4,4+25),
  -len(theta,4,4,4+25)-ball4(xcoord(theta,4,4,4+25)),
  theta=0.01..Pi-0.01],

  [xcoord(theta,6,18,18+25),
  -len(theta,6,18,18+25)-ball4(xcoord(theta,6,18,18+25)),
```

```
theta=0.01..Pi-0.01],
[xcoord(theta,6,2,2+25),
-len(theta,6,2,2+25)-ball4(xcoord(theta,6,2,2+25)),
theta=0.01..Pi-0.01],
[xcoord(theta,6,6,6+25),
-len(theta,6,6,6+25)-ball4(xcoord(theta,6,6,6+25)),
theta=0.01..Pi-0.01],

[xcoord(theta,2,6,6+25),
-len(theta,2,6,6+25)-ball4(xcoord(theta,2,6,6+25)),
theta=0.01..Pi-0.01],
[xcoord(theta,2,0.67,0.67+25),
-len(theta,2,0.67,0.67+25)-ball4(xcoord(theta,2,0.67,0.67+25)),
theta=0.01..Pi-0.01],
[xcoord(theta,2,2,2+25),
-len(theta,2,2,2+25)-ball4(xcoord(theta,2,2,2+25)),
theta=0.01..Pi-0.01]],
x=-50..50,y=-2.1..2.5,
color =[black,black,black,black,black,black,black,black,black],
linestyle=[3,3,3,1,1,1,2,2,2],
scaling=UNCONSTRAINED);
```



We Can't Determine Curvatures

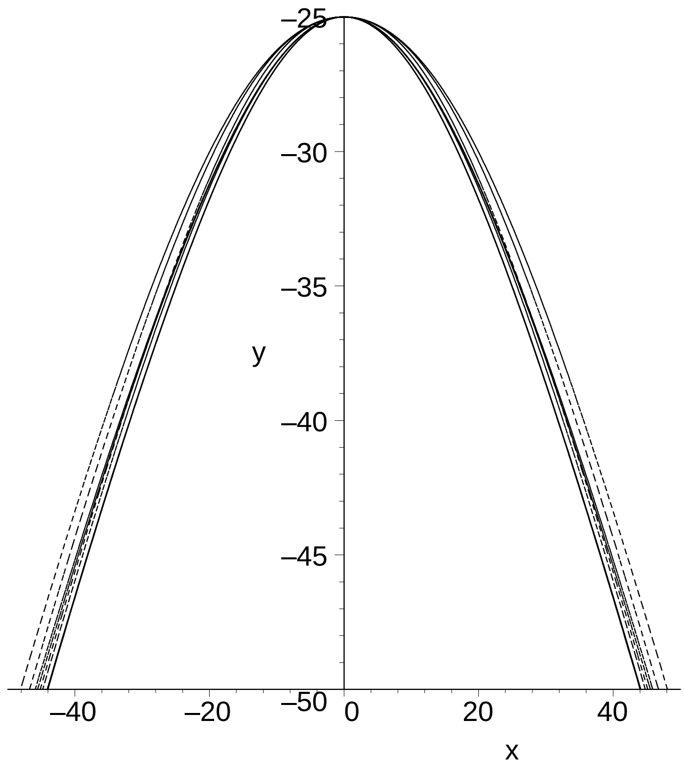
One might now reason like this: Our radar data is basically giving us data on the top surface of the buried cylinder. It might therefore be that what we can determine is the curvature of the top center of the ellipse. Mathematically, this curvature is measured by the second derivative of the ellipse at its apex.

We therefore plot families of ellipses varying in their curvature at the top of the ellipse. The solid curves all have second derivatives of -1 at the top; the dashed curves all have second derivatives of -1/4, and the dotted curves all have second derivatives of -1/16. These curves are more readily distinguishable than the earlier curves, but still there are dotted and dashed curves which lie within 4 mm of one another everywhere. Thee curves probably could not be distinguished in the field. Here are the distance functions themselves:

```
> plot([[xcoord(theta,2,4,4+25), -len(theta,2,4,4+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,16,16+25), -len(theta,4,16,16+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,36,36+25), -len(theta,6,36,36+25),
  theta=0.01..Pi-0.01],

  [xcoord(theta,2,1,1+25), -len(theta,2,1,1+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,4,4+25), -len(theta,4,4,4+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,9,9+25), -len(theta,6,9,9+25),
  theta=0.01..Pi-0.01],

  [xcoord(theta,2,0.25,0.25+25), -len(theta,2,0.25,0.25+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,1,1+25), -len(theta,4,1,1+25),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,2.25,2.25+25), -len(theta,6,2.25,2.25+25),
  theta=0.01..Pi-0.01]],
  x=-50..50,y=-50..-25,numpoints=1000,
  color =[black,black,black,black,black,black,black,black,black],
  linestyle=[1,1,1,3,3,3,2,2,2],
  scaling=UNCONSTRAINED);
```



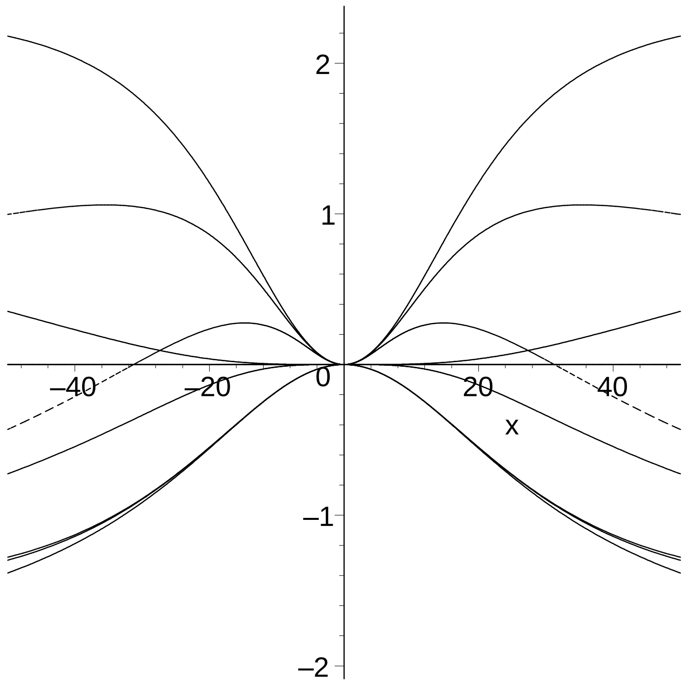
And here are the differences between these functions and the distance function for a circle of radius 4. Again, we need distance measurements accurate on the level of a couple of millimeters to a fraction of a millimeter before we can reliably measure curvature.

```
> plot([[xcoord(theta,2,4,4+25),
  -len(theta,2,4,4+25)-ball4(xcoord(theta,2,4,4+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,16,16+25),
  -len(theta,4,16,16+25)-ball4(xcoord(theta,4,16,16+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,36,36+25),
  -len(theta,6,36,36+25)-ball4(xcoord(theta,6,36,36+25)),
  theta=0.01..Pi-0.01],

  [xcoord(theta,2,1,1+25),
  -len(theta,2,1,1+25)-ball4(xcoord(theta,2,1,1+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,4,4+25),
  -len(theta,4,4,4+25)-ball4(xcoord(theta,4,4,4+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,9,9+25),
  -len(theta,6,9,9+25)-ball4(xcoord(theta,6,9,9+25)),
  theta=0.01..Pi-0.01],

  [xcoord(theta,2,0.25,0.25+25),
  -len(theta,2,0.25,0.25+25)-ball4(xcoord(theta,2,0.25,0.25+25)),
```

```
  theta=0.01..Pi-0.01],
  [xcoord(theta,4,1,1+25),
  -len(theta,4,1,1+25)-ball4(xcoord(theta,4,1,1+25)),
  theta=0.01..Pi-0.01],
  [xcoord(theta,6,2.25,2.25+25),
  -len(theta,6,2.25,2.25+25)-ball4(xcoord(theta,6,2.25,2.25+25)),
  theta=0.01..Pi-0.01]],
  x=-50..50,numpoints=1000,
  color =[black,black,black,black,black,black,black,black,black],
  linestyle=[1,1,1,3,3,3,2,2,2],
  scaling=UNCONSTRAINED);
```



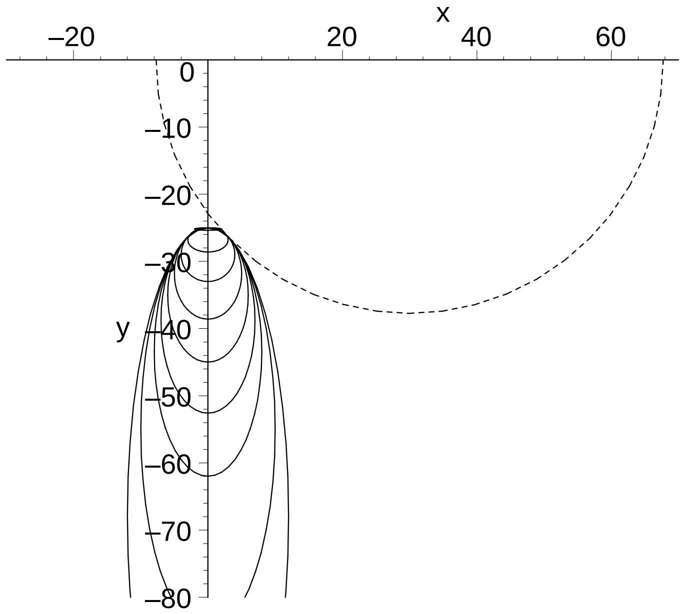
We Can't Even Tell Roughly How Big the Ellipse Is.

In order to see just how difficult it is to invert the distance curves to get the ellipses, consider the following question. Suppose we have a buried cylinder and we know that the top center of its elliptical cross-section is at depth 25 cm. Suppose also we know that when the GPR unit is offset 30 cm horizontally from the top center, it detects a reflection from the ellipse at a distance $\sqrt{30^2+29^2}-4$, i.e., at the distance from which a 4 cm radius circle would reflect the signal. What ellipses satisfy these two constraints?

The figure below shows a number of these ellipses. (Of course, there are really infinitely many of them.) The ellipses shown vary in size from one of width 4 cm and height 0.36 cm to one of width 24 cm and height 86 cm.

```
> plot([[30+(sqrt(30^2+29^2)-4)*cos(theta),
          (sqrt(30^2+29^2)-4)*sin(theta), theta=0..2*Pi],
         [4*cos(theta), -29+4*sin(theta), theta=0..2*Pi],
         [2*cos(theta), -25+0.18*(-1+sin(theta)), theta=0..2*Pi],
```

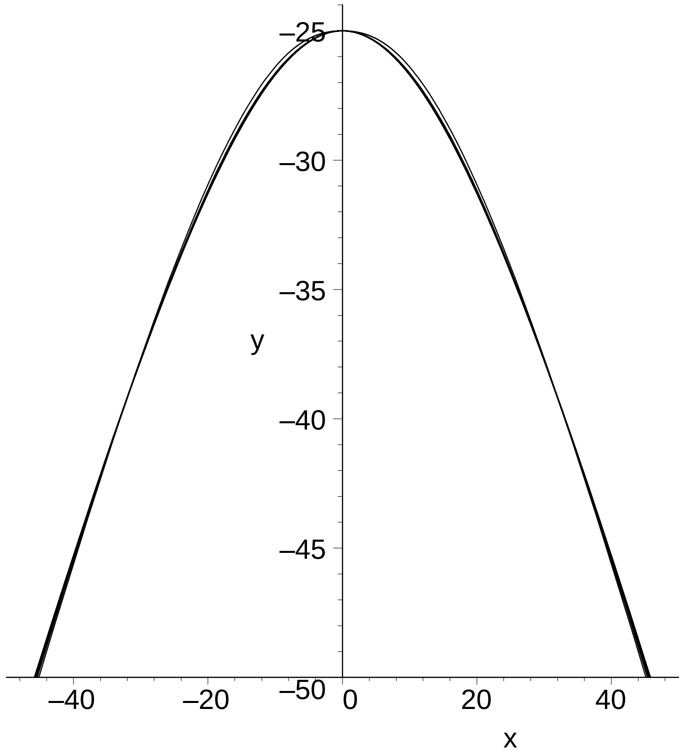
```
        [3*cos(theta), -25+1.8*(-1+sin(theta)), theta=0..2*Pi],
        [5*cos(theta), -25+6.8*(-1+sin(theta)), theta=0..2*Pi],
        [6*cos(theta), -25+10*(-1+sin(theta)), theta=0..2*Pi],
        [7*cos(theta), -25+13.8*(-1+sin(theta)), theta=0..2*Pi],
        [8*cos(theta), -25+18.5*(-1+sin(theta)), theta=0..2*Pi],
        [10*cos(theta), -25+30*(-1+sin(theta)), theta=0..2*Pi],
        [12*cos(theta), -25+43*(-1+sin(theta)), theta=0..2*Pi]],
       x=-30..70, y=-80..0,
  color=[black,black,black,black,black,black,black,black,black,black
  ],linestyle=[2,1,1,1,1,1,1,1,1,1],
  scaling=CONSTRAINED);
```



What's special about all these ellipses is that they have identical values of the distance curve at *x*=0 cm and at *x*=30 cm. The complete distance curves for all these ellipses are shown here:

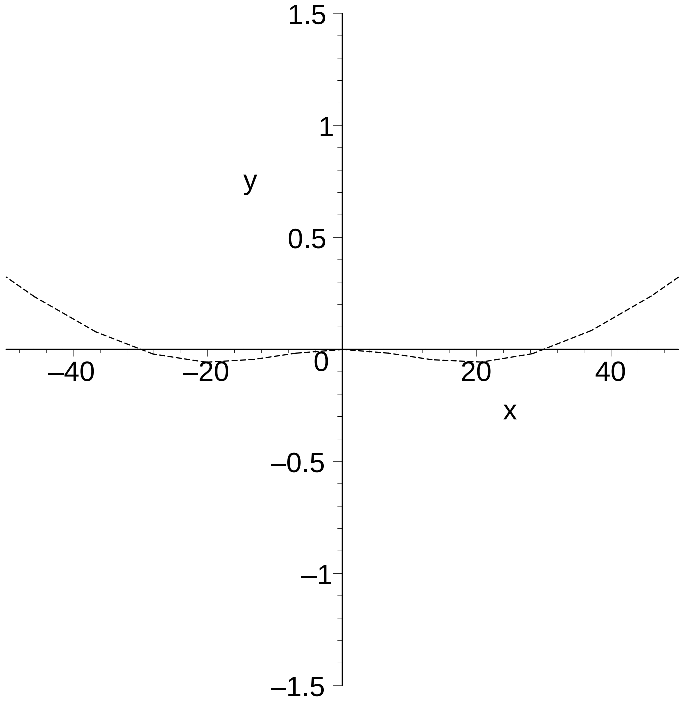
```
> picture := (a,b,delta,xrange,yrange) ->
  plot([xcoord(theta,a,b,b+delta), -len(theta,a,b,b+delta),
  theta=0..Pi],
  xrange,yrange,numpoints=400,color=black,
  scaling=UNCONSTRAINED):
  plots[display]({picture(2,0.18,25,x=-50..50,y=-50..-24),
           picture(3,1.8,25,x=-50..50,y=-50..-24),
           picture(4,4,25,x=-50..50,y=-50..-24),
           picture(5,6.8,25,x=-50..50,y=-50..-24),
           picture(6,10,25,x=-50..50,y=-50..-24),
           picture(7,13.8,25,x=-50..50,y=-50..-24),
           picture(8,18.5,25,x=-50..50,y=-50..-24),
```

```
picture(10,30,25,x=-50..50,y=-50..-24),
picture(12,43,25,x=-50..50,y=-50..-24)});
```



In order to see just how close together all these distance curves are, let's look at the difference between the distance curves for the 8 cm circle and for the 24-by-86 cm ellipse:

```
> plot([[xcoord(theta,12,43,43+25),
  -len(theta,12,43,43+25)-(4-sqrt(xcoord(theta,12,43,43+25)^2+(4+25)
  ^2)), theta=0.01..Pi-0.01],0],
  x=-50..50,y=-1.5..1.5,color=[black,black],linestyle=[3,1],
  scaling=UNCONSTRAINED);
```

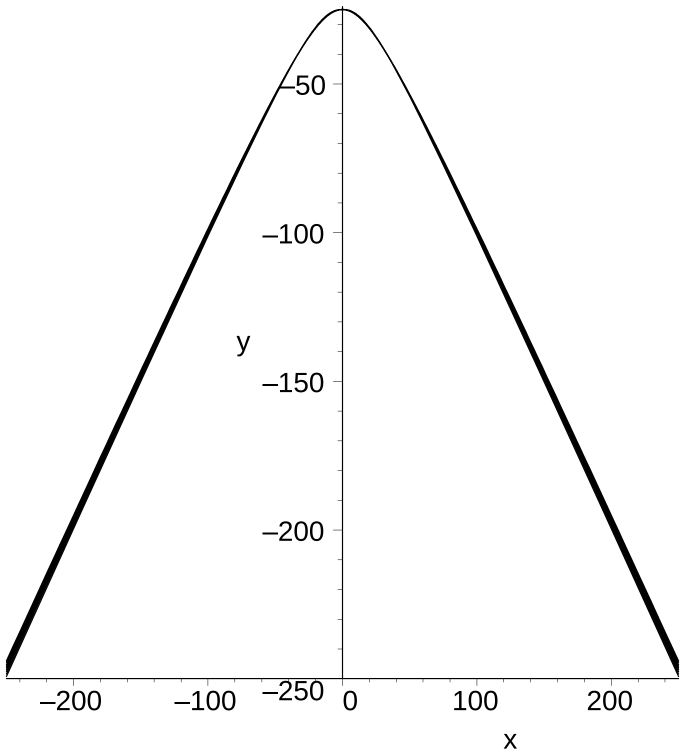


Despite the fact that these ellipses vary enormously in size and shape, they produce distance curves that are within 2 mm of each other at every point from a horizontal offset of 0 to a horizontal offset of 40 cm from the top of the ellipses.

Unless you can accurately measure distances to a fraction of a millimeter, you cannot by measuring reflection distances alone in a single pass with a GPR unit distinguish between an 81 mm mortar round and a steeply inclined cylinder nearly a foot across.

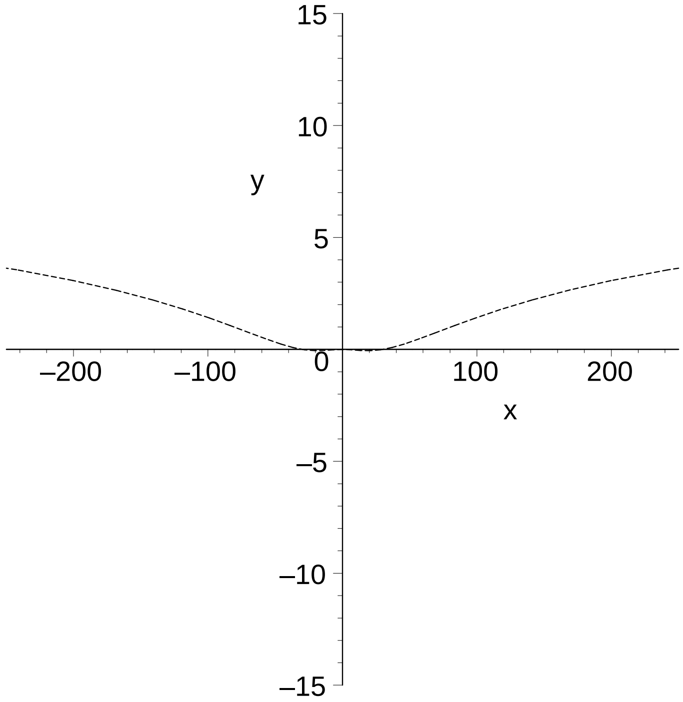
Even if we drop the requirement that we cannot receive reflections from farther away than a horizontal offset of half a meter, thing don't get much better. Here is the family of distance curves for the ellipses above measured for horizontal offsets up to 2.5 m:

```
> plots[display]({picture(2,0.18,25,x=-250..250,y=-250..-24),
           picture(3,1.8,25,x=-250..250,y=-250..-24),
           picture(4,4,25,x=-250..250,y=-250..-24),
           picture(5,6.8,25,x=-250..250,y=-250..-24),
           picture(6,10,25,x=-250..250,y=-250..-24),
           picture(7,13.8,25,x=-250..250,y=-250..-24),
           picture(8,18.5,25,x=-250..250,y=-250..-24),
           picture(10,30,25,x=-250..250,y=-250..-24),
           picture(12,43,25,x=-250..250,y=-250..-24)});
```



And here is the difference between the predicted data for the 8 cm circle and the 24-by-86 cm ellipse:

```
> plot([[xcoord(theta,12,43,43+25),
  -len(theta,12,43,43+25)-(4-sqrt(xcoord(theta,12,43,43+25)^2+(4+25)
  ^2)), theta=0.01..Pi-0.01],0],
  x=-250..250,y=-15..15,color=[black,black],linestyle=[3,1],
  scaling=UNCONSTRAINED);
```



It would not be hard to find pairs of widely different sized ellipses whose distance curves were closer together than these when extended over this wider range. One would pin the curves together not at 0 and 30 cm, as we did above, but at, say, 0 and 120 cm.

The Good News: A Proposed Alternative

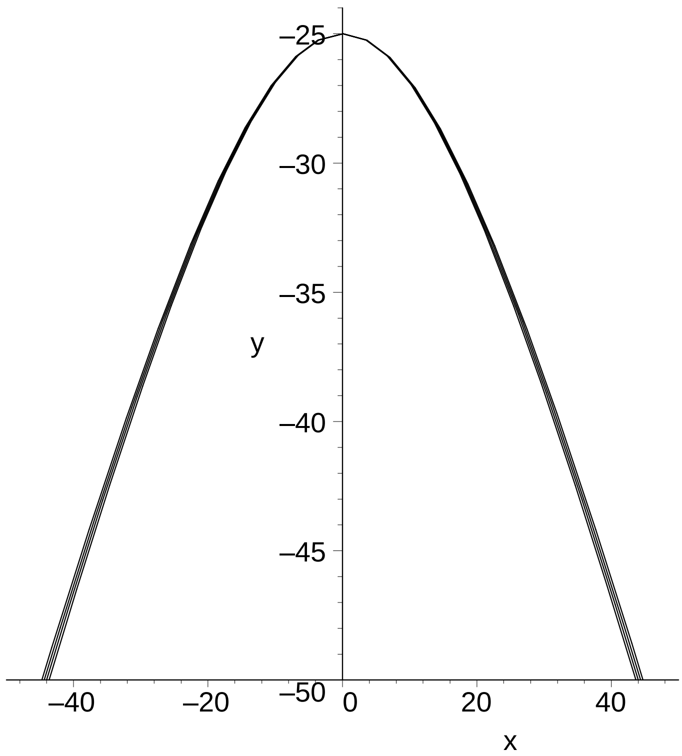
The examples above should make it clear that the information provided by the distance curve for a single track across a buried cylinder with a GPR unit does not provide enough information to make meaningful estimates of the size of the cylinder. Assuming we aren't prepared just to give up now with the project, we are left with two alternatives. First, we could try to extract more information from a single GPR track than just the distance to the cylinder. Can anything be learned from the intensity of the reflection, for instance, or are there other features that could be added to our simple model?

The other alternative would be to scan across each anomaly more than once.

For instance, if we made not one scan across the cylinder at a random angle, but two scans across the cylinder as nearly as possible perpendicular to it, and if we measured the depth and position of the top center point in each scan, then we would know the orientation of the cylinder in three dimensions: we would have the direction of its projection onto the surface of the ground, and we would know its dip angle (its angle from the horizontal). From these facts, it is easy to compute the shape (though not the size) of the elliptical cross-sections over which we have scanned. For instance, if we know the cylinder is in a north-south plane and is horizontal, then an east-west scan will encounter a circular cross section, and a scan in a direction 30 degrees away from north will encounter an elliptical cross-section which is twice as wide as it is high. If the cylinder is in a north-south plane and is inclined 60 degrees from the horizontal, then an east-west scan will encounter an ellipse twice as high as it is wide.

Is it easier to determine the size of an ellipse if we know its shape? The answer turns out to depend on what the shape is. For tall, thin ellipses, even knowing the shape leaves us with a difficult problem to find the size. Here, for instance, are the theoretical distance curves for a family of ellipses, all of which are twice as high as they are wide, and whose widths are 4, 8, 12, and 16 cm:

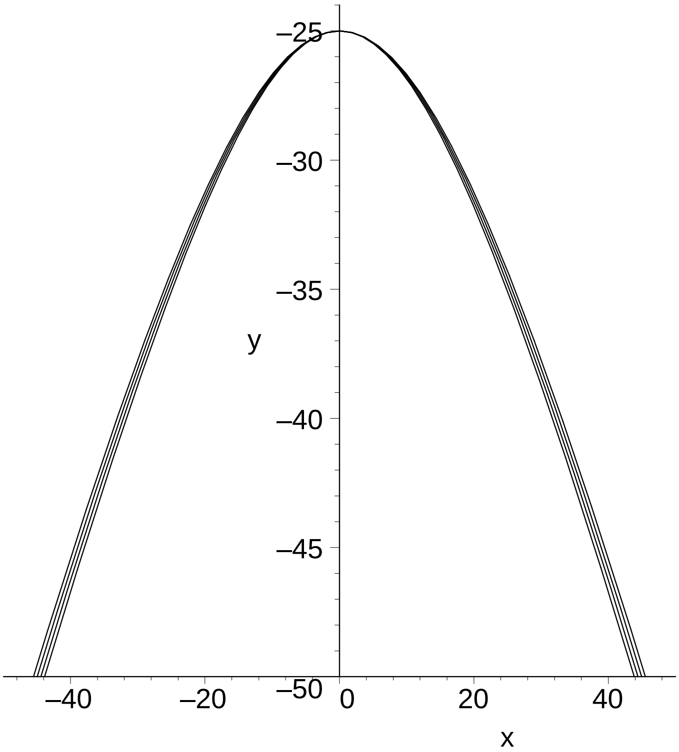
```
> picture := (a,b,delta,xrange,yrange) ->
  plot([xcoord(theta,a,b,b+delta), -len(theta,a,b,b+delta),
  theta=0..Pi],
  xrange,yrange,color=black,
  scaling=UNCONSTRAINED):
> plots[display]({picture(1,2,25,x=-50..50,y=-50..-24),
           picture(2,4,25,x=-50..50,y=-50..-24),
           picture(3,6,25,x=-50..50,y=-50..-24),
           picture(4,8,25,x=-50..50,y=-50..-24)});
```



(As always, the top center points of all these ellipses are 25 cm below the surface.) It is clear the curves are close together; inferring sizes of these ellipses, which requires distinguishing these curves, will not be easy.

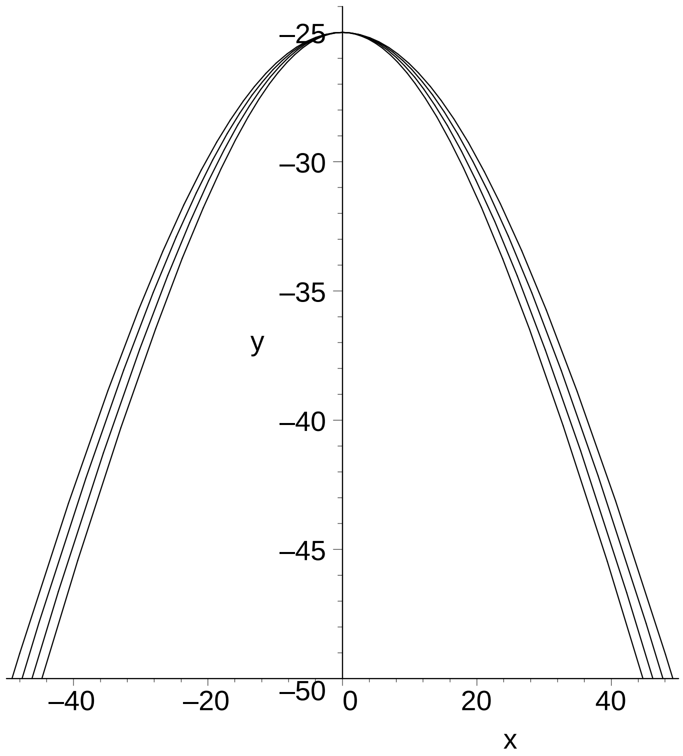
The situation is slightly but not much better if the ellipses are circles:

```
> plots[display]({picture(1,1,25,x=-50..50,y=-50..-24),
           picture(2,2,25,x=-50..50,y=-50..-24),
           picture(3,3,25,x=-50..50,y=-50..-24),
           picture(4,4,25,x=-50..50,y=-50..-24)});
```



On the other hand, if the ellipses are twice as wide as they are high, then the distance curves are much easier to distinguish. When the tall, thin ellipses above are turned on their sides, the distance curves look like this:

```
> plots[display]({picture(2,1,25,x=-50..50,y=-50..-24),
           picture(4,2,25,x=-50..50,y=-50..-24),
           picture(6,3,25,x=-50..50,y=-50..-24),
           picture(8,4,25,x=-50..50,y=-50..-24)});
```



It should not be surprising that flat ellipses are easier to measure with GPR distance curves than are tall ellipses. After all, the detectable reflections all come from the top part of the ellipse. The top pieces of every tall, skinny ellipse look similar, but the top pieces of flat, "squashed" ellipses look more distinctive. For flat ellipses, we can get reflections from points farther from the center of the ellipse than we can for tall ellipses, and this simplifies telling small flat ellipses from large flat ellipses.

A New Experimental Protocol

Here, then, is a multiple-scan method which seems to give us enough accuracy to enable us to gather at least some information about the size of an anomaly. We begin by making two scans across the anomaly, as nearly as possible perpendicular to its axis. All we’re looking for at this stage is the location of the top points on each scan, which let us determine the orientation and dip of the cylinder.

Next, we compute an angle at which we could cut the cylinder to obtain an elliptic cross-section twice as wide as it is high. There is nothing magical about saying "twice" here. There is a trade-off: picking cross-sections more nearly circular makes the size harder to infer from the distance curve. On the other hand, picking even flatter cross-sections forces us to cut the cylinder at an increasingly precise angle closer and closer to the axis of the cylinder. Small changes in this angle change the shape of the ellipse dramatically, and so change the inferred size. A width/height ratio of 2 seems like a sensible compromise between these two sources of error.

Finally, we make a third scan of the cylinder at this angle, and compare the observed distance curve with theoretical distance curves for different sized ellipses of this shape to infer the size.

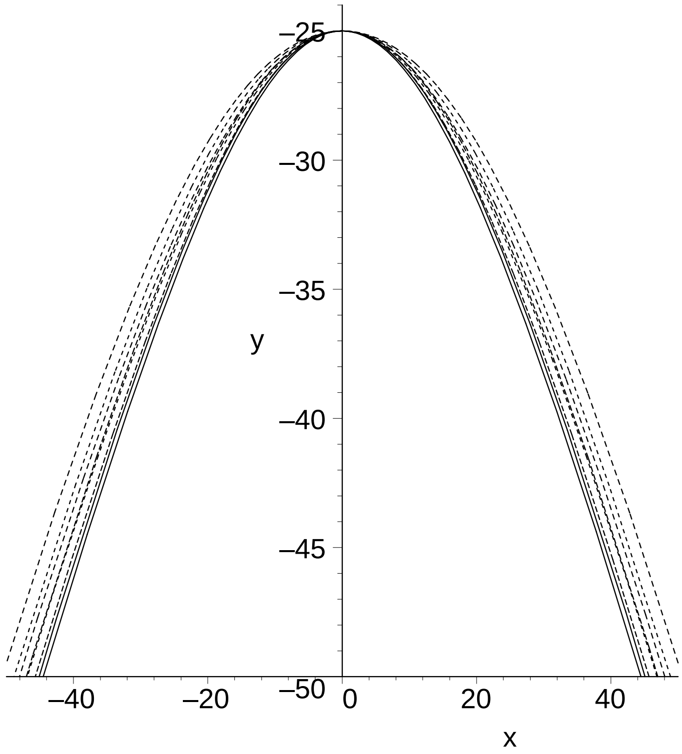
An Example of This Protocol

Let’s do a quick and dirty analysis of the error involved in using this protocol in one specific example. Suppose we have a cylinder inclined at 60 degrees to the horizontal (unknown to us) which we believe to be oriented roughly north-south. We might begin by taking two east-west scans offset by 5 cm from one another. If we can determine the top point in each scan to within 1 mm along the scan line and 1 mm vertically, then we can orient the cylinder in the horizontal plane with a directional error of .3 degrees. Similarly, the uncertainty in the heights of the two top points translates into an uncertainty of about .6 degrees in the dip angle. We therefore will obtain a measured ratio of height/width for the ellipse obtained from cutting the cylinder perpendicular to its projection on the surface of 2.035. To produce a slice which is twice as wide as it is high, we’ll try to slice the cylinder at an angle of 75.5.23 degrees to its projection on the surface. The 2.3 degree uncertainty about the precise orientation of the cylinder, though, means we will in fact end up making a third scan at an angle of 75.5.5 degrees. The ellipse beneath our scan line will therefore have a width/length ratio somewhere between 1.71 and 2.41, instead of the 2 we were striving for.

Given this degree of error, how accurately can we determine the size of the cylinder? To answer this question, let’s just plot predicted distance curves for a series of ellipses of different sizes whose shapes vary within this range.

Below are the distance functions for two ellipses each coming from cylinders of diameter 1 cm (solid), 2 cm (dashed), 3 cm (dash-dot) and 4 cm (dotted). The two curves of each color correspond to the extreme width/height ratios of 1.71 and 2.41.

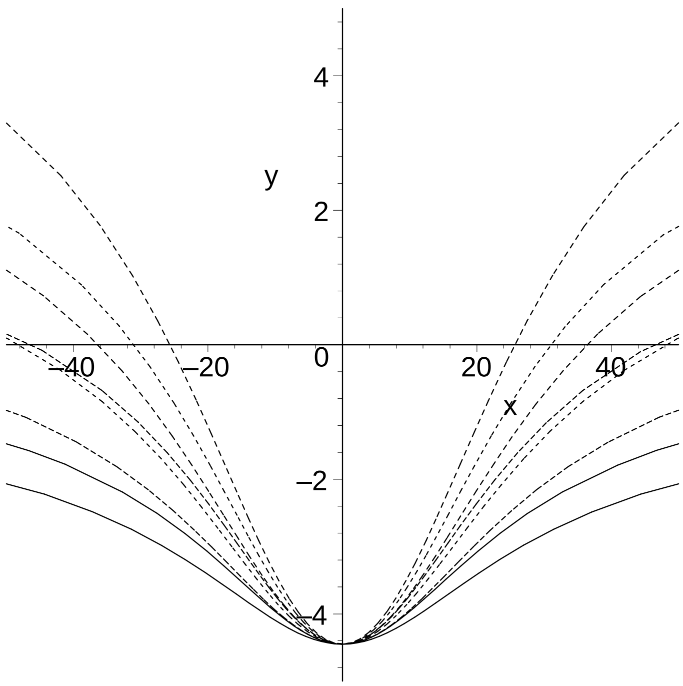
```
> picture := (a,b,delta,xrange,yrange,c) ->
  plot([xcoord(theta,a,b,b+delta), -len(theta,a,b,b+delta),
  theta=0..Pi],
  xrange,yrange, color=black, linestyle=c,
  scaling=UNCONSTRAINED):
> plots[display]({picture(1.71,1,25,x=-50..50,y=-50..-24,1),
           picture(2.4,1,25,x=-50..50,y=-50..-24,1),
           picture(3.42,2,25,x=-50..50,y=-50..-24,3),
           picture(4.8,2,25,x=-50..50,y=-50..-24,3),
           picture(5.13,3,25,x=-50..50,y=-50..-24,4),
           picture(7.2,3,25,x=-50..50,y=-50..-24,4),
           picture(6.84,4,25,x=-50..50,y=-50..-24,2),
           picture(9.6,4,25,x=-50..50,y=-50..-24,2)});
```



The first thing to notice about this picture is that although there is overlap between the curves of adjacent colors (solid and dashed, dashed and dash-dot, dash-dot and dotted), there is no overlap between colors farther apart. Distance curves coming from cylinders differing in diameter by 2 cm seem to be distinguishable, though cylinders differing in diameter by 1 cm may not be.

A clearer estimate of where we stand can be obtained by looking at the differences among all these curves. This isn't trivial to plot, since all the curves have only complicated parametric forms, but here is a plot of the difference in heights between each of these distance functions and a random hyperbola going more or less through the middle of the pack:

```
> picture := (a,b,delta,xrange,yrange,c) ->
  plot([xcoord(theta,a,b,b+delta),
  -len(theta,a,b,b+delta)-(2.22-22.77*sqrt(1+xcoord(theta,a,b,b+delt
  a)^2/22.77^2)), theta=0.01..Pi-0.01],
  xrange,yrange, color=black, linestyle=c,
  scaling=UNCONSTRAINED):
> plots[display]({picture(1.71,1,25,x=-50..50,y=-5..5,1),
           picture(2.4,1,25,x=-50..50,y=-5..5,1),
           picture(3.42,2,25,x=-50..50,y=-5..5,3),
           picture(4.8,2,25,x=-50..50,y=-5..5,3),
           picture(5.13,3,25,x=-50..50,y=-5..5,4),
           picture(7.2,3,25,x=-50..50,y=-5..5,4),
           picture(6.84,4,25,x=-50..50,y=-5..5,2),
           picture(9.6,4,25,x=-50..50,y=-5..5,2)});
```



Notice the good news here: The envelopes of curves corresponding to different sized cylinders are largely separate from one another, and the vertical separations between curves are no longer in the 1 mm range, but in the range of 1 cm. This suggests that GPR might well have sufficient resolution to distinguish cylinders differing in diameter by 2 cm, or even less.

Future Directions and Caveats

The calculation above is only a proof of concept, not a finished method. I need to get back to work preparing my classes, and this is as much work as I feel like doing just to be nice to Mic. If the experimental approach described here seems worth pursuing, then there are a number of additional steps that might be taken.

1. A more careful error analysis, varying the dip angle and depth of the cylinder, is needed.
2. An algorithmic method for fitting experimental data to theoretical distance curves is needed. This is harder than we were expecting, since the theoretical curves are not actually hyperbolas, but have complicated parametric descriptions, but it should just require a bit of time.
3. The assumptions about the accuracy of the experimental tools needs to be reviewed by somebody who understands those tools.
4. The arbitrary choice of seeking scan lines that yield ellipses twice as wide as they are high should be revisited.
5. The biggest source of error in the calculation above was the uncertainty in the horizontal orientation of the cylinder. If this error can be reduced, then the differently colored envelopes of curves above can be tightened, and our size estimates improved. One way to start on this is to realize that the third scan over the cylinder gives us a third high point, which can be combined with the first two to get a more accurate orientation. Perhaps people with better understanding of the capabilities of the hardware than I can make suggestions here as well.

6. The fact that not every plausible bit of UXO is really a cylinder ought also to get a bit of thought. This was not so crucial when we thought we could get away with a single radar scan of each anomaly, but the more complicated protocol above relies rather heavily on cylindrical geometry.

One other major concern remains, and that is the assumption underlying this whole analysis that we can treat the radar reflection as a 2-dimensional process. In reality, when we scan along a line, the signal is not restricted to a plane containing an ellipse, but reflects from points on the cylinder to the left or right of the ellipse. In a perfectly polished cylinder, there will normally be no point in the vertical plane of the scan that will be reflecting radiation back to the GPR unit. All reflections will come from off the plane. It seems possible to me that surface roughness or beam directionality or some sort of averaging of errors makes the assumption of 2-dimensionality of the reflection process reasonable. It also seems to me entirely possible that this is not so. A careful mathematical analysis of the 3-dimensional reflections is worth considering, but might be complex. In any case, significant experimentation with buried rods is needed in order to test this assumption.

Let me close by thanking you for an interesting problem, by wishing you well, and by hoping my thoughts may be of some use.