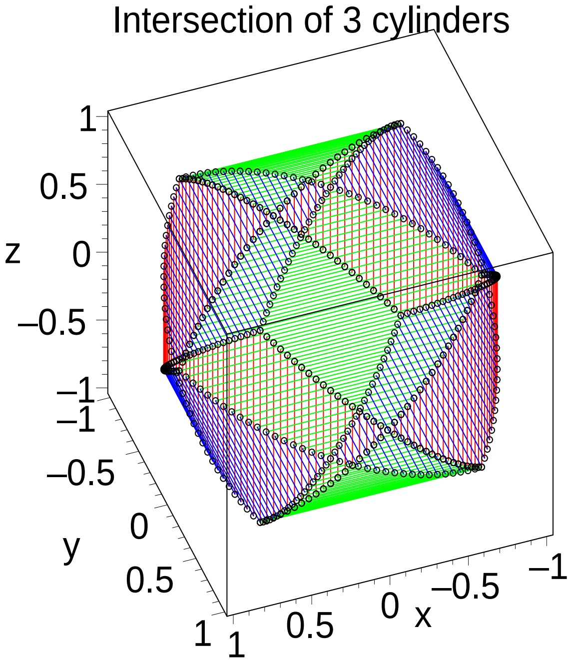
Some Calculus III Notes

Carl Eberhart



January 5, 2009

Extreme values

```
> with(plots):    #execute this line first
```

1 least squares fit

If you have 4 points in space (x_i, y_i, z_i), i = 1..4, there probably isn't a plane which contains all of them. However, you can find the plane z = ax + by + c which best fits the point in the sense that g(a, b, c) = $\sum_{i=1}^{4} (a\,x_i + b\,y_i + c - z_i)^2$ is as small as possible. There is only one critical point for this function

It is the solution of the system 3 linear equations in a,b and c $g_a = 0$, $g_b = 0$, $g_c = 0$

```
> data:=([[0,0,2],[2,0,4],[2,3,8],[0,3,5]]):
> matrix(data);
```

$$\begin{bmatrix} 0 & 0 & 2 \\ 2 & 0 & 4 \\ 2 & 3 & 8 \\ 0 & 3 & 5 \end{bmatrix}$$

make up the function g

```
> g := (c - 2)^2 + (2*a+c -4)^2 + (2*a+3*b+c-8)^2 + (3*b+c-5)^2;
```

$$g := (c-2)^2 + (2\,a + c - 4)^2 + (2\,a + 3\,b + c - 8)^2 + (3\,b + c - 5)^2$$

find its critical point. Since g is a sum of squares, it will have a minimum at the critical point.

```
> sol:=solve({diff(g,a),diff(g,b),diff(g,c)},{a,b,c});
```

$$sol := \{a = \frac{5}{4},\, b = \frac{7}{6},\, c = \frac{7}{4}\}$$

If we put these values for a, b, and c into g and then compute sqrt(g(a,b,c)/4) we get a measure of the goodness of fit.

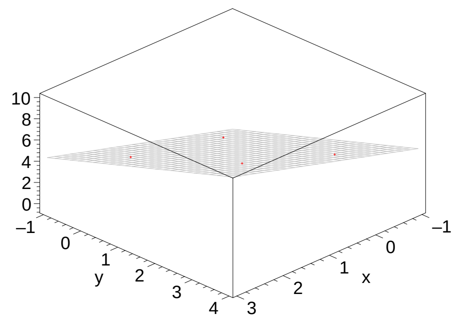
```
> goodnessoffit:=sqrt(subs(sol,g)/4);
```

$$goodnessoffit := \frac{1}{4}$$

this says on average the measured value is within 1/4 of the 'correct' value.

we can draw the points and the plane to see how well they fit.

```
> display
  (plot3d(subs(sol,a*x+b*y+c),x=-1..3,y=-1..4,color=grey,style=wireframe
  ),pointplot3d({op(data)},symbol=cross,color=red),axes=boxed);
```



2 hunting for a local extreme value of f(x,y)

This procedure takes a function f of two variables, a starting point (a,b), a step length h, a maximum number of steps n, and a stopping distance er. It returns a list [a,b,f(a,b)] ,[a2,b2,f(a2,b2)], ... where f(a,b) < f(a2,b2) < ...

The method is to compute the gradient of f at (a,b) and move along it a step of length h to a new point (a',b'). f(a',b') is compared with f(a,b) and if it is smaller than f(a,b)-ez, then we cut steplength in half and start again. This is repeated until f(a',b') > f(a,b) or the steplength is less than er it is added to the list

```
> indets(x^2+y^2,symbol);
```

$$\{x, y\}$$

```
> hunt := proc(fn,a,b,h,n,er)
  local i,p,gd,ai,bi,aj,bj,hi,f;
  if  not type(fn,function) then f:= unapply(fn,op(indets(fn)))
> else f := op(fn) fi;
  p := [a,b,evalf(f(a,b))]:
  i:= 1:
> ai:=a:  bi:=b:
  while i<n+1 do
  hi:=h;
> gd:=evalf(subs({x=ai,y=bi},[diff(f(x,y),x),diff(f(x,y),y)]));
  gd := 1/(gd[1]^2+gd[2]^2)*gd;
> aj:=ai+hi*gd[1]:  bj:=bi+hi*gd[2]:
  while f(aj,bj)<=f(ai,bi) and hi>er  do
> hi:=.5*hi;
  aj:=ai+hi*gd[1]:  bj:=bi+hi*gd[2]:
  od;
> if f(aj,bj)>f(ai,bi) then
  ai:=aj:  bi:=bj;
  p := p,[ai,bi,f(ai,bi)];
```

```
> fi:
  i:= i+1:
  od;
> matrix([p]) end:
```

Here we test hunt on the function 10 - x^2 -y^2

```
> hunt(10-x^2-y^2,1,1,1,10,.00003);
```

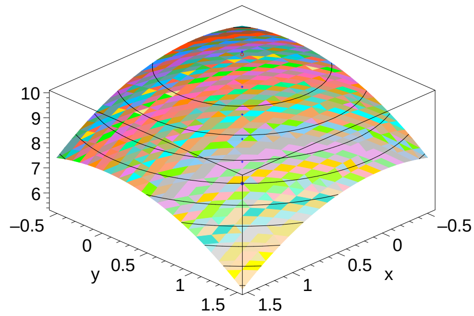
$$\begin{bmatrix} 1 & 1 & 8. \\ 0.7500000000 & 0.7500000000 & 8.875000000 \\ 0.4166666667 & 0.4166666667 & 9.652777778 \\ -0.1833333333 & -0.1833333333 & 9.932777778 \\ 0.1575757577 & 0.1575757577 & 9.950339762 \\ -0.0407415498 & -0.0407415498 & 9.996680252 \\ 0.00719783910 & 0.00719783910 & 9.999896382 \\ -0.001281810890 & -0.001281810890 & 9.999996714 \end{bmatrix}$$

Showhunt draws the graph of the track of the hunt for a maximum on the graph of f.

It has the same inputs as hunt, except for the parameter w which controls the size of the window used to show the track.

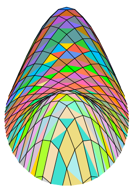
```
> showhunt:=proc(f,a,b,h,n1,er,w)
  local lst,xrng,yrng,n:
  lst:=hunt(f,a,b,h,n1,er);
> n:=linalg[rowdim](lst);
  xrng:=[a,seq(lst[i,1],i=1..n)];
  xrng:=[min(op(xrng)),max(op(xrng))]:
> xrng:=(xrng[1]-w*(xrng[2]-xrng[1]))..(xrng[2]+w*(xrng[2]-xrng[1]));
  yrng:=[b,seq(lst[i,2],i=1..n)];
  yrng:=[min(op(yrng)),max(op(yrng))]:
> yrng:=(yrng[1]-w*(yrng[2]-yrng[1]))..(yrng[2]+w*(yrng[2]-yrng[1]));
  plots[display](plot3d(f(x,y),x=xrng,y=yrng),
  plots[pointplot3d]({seq([lst[i,1],lst[i,2],lst[i,3]],i=1..1)
> },symbol=circle,color=blue),
  plots[pointplot3d]({seq([lst[i,1],lst[i,2],lst[i,3]],i=1..n-1)
  },symbol=cross,color=blue),
> plots[pointplot3d]({seq([lst[i,1],lst[i,2],lst[i,3]],i=n..n)
  },symbol=circle,color=red),
  axes=boxed,style=patchcontour);
> end:
```

```
> showhunt(10-x^2-y^2,1,1,.5,20,.0001,.5);
```

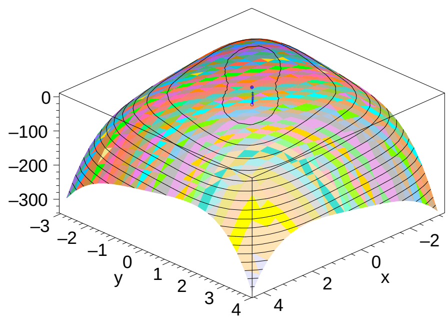


Play around with hunt and showhunt.

```
> plot3d(4*x*y+1-x^4-y^4,x=-2..2,y=-2..2,view=[-2..2,-2..2,-1..6]);
```



```
> showhunt(4*x*y+1-x^4/2-y^4,.1,.1,.2,10,.03,4);
```



3 finding critical points algebraically and applying the 2nd derivative test.

Here is a function to study.

```
> f:=x^4+y^4-4*x*y + 1;
```

$$f := x^4 + y^4 - 4\,x\,y + 1$$

Calculating the first partial derivatives of f.

```
> fx := diff(f,x); fy:=diff(f,y);
```

$$fx := 4\,x^3 - 4\,y$$

$$fy := 4\,y^3 - 4\,x$$

Using solve to find the critical points. We see that there are only three. The other two are complex pairs

```
> evalf(solve({fx,fy},{x,y}));
```

$$\{x = 0.,\ y = 0.\},\ \{x = 1.,\ y = 1.\},\ \{x = -1.,\ y = -1.\},\ \{x = 1.\,I,\ y = -1.\,I\},$$
$$\{x = 0.7071067812 + 0.7071067812\,I,\ y = -0.7071067812 + 0.7071067812\,I\}$$

You can also use fsolve to find the critical points.

```
> s1:=fsolve({fx,fy},{x,y},{x=-2..2,y=-2..2});
```

$$s1 := \{x = 0.,\ y = 0.\}$$

```
> s2:=fsolve({fx,fy},{x,y},{x=-2..2,y=-2..2},avoid={s1});
```

$$s2 := \{x = -1.000000000,\ y = -1.000000000\}$$

```
> s3:=fsolve({fx,fy},{x,y},{x=-2..2,y=-2..2},avoid={s1,s2});
```

$$s3 := \{x = 1.000000000,\ y = 1.000000000\}$$

Use the second derivative test on the 3 critical points

```
> fxx:=diff(fx,x); fyy:=diff(fy,y); fxy:=diff(fx,y);
```

$$fxx := 12\,x^2$$

$$fyy := 12\,y^2$$

$$fxy := -4$$

```
> discrcim:=fxx*fyy-fxy^2;
```

$$discrcim := 144\,x^2\,y^2 - 16$$

We see that at (0,0) the discriminant is <0 so there is no local extreme. At (1,1) and (-1,-1) however, the discriminant is >0 and fxx >0 so the function has

local maxima at (1,1) and (-1,-1).

4 maximizing the volume of a box with a half top, and a quarter front

A box is to be made from 10 square feet of metal so that its top is only half there and its front is 3/4 missing. Find the maximum volume of the box.

Maximimize V = xyz subject to 10 = 3/2 xy + 2yz + 7/4 xz

first solve the constraint for one of the variables and reduce V to a function of 2 variables.

```
> constraint:=10 = 3/2* x*y + 2*y*z + 7/4 *x*z;
```

$$constraint := 10 = \frac{3}{2}\,x\,y + 2\,y\,z + \frac{7}{4}\,x\,z$$

```
> sol := solve(constraint,z);
```

$$sol := -\frac{2\,(-20 + 3\,x\,y)}{8\,y + 7\,x}$$

```
> V := x*y*sol;
```

$$V := -\frac{2\,x\,y\,(-20 + 3\,x\,y)}{8\,y + 7\,x}$$

Find critical points

```
> Vx := diff(V,x); Vy:=diff(V,y);
```

$$Vx := -\frac{2\,y\,(-20 + 3\,x\,y)}{8\,y + 7\,x} - \frac{6\,x\,y^2}{8\,y + 7\,x} + \frac{14\,x\,y\,(-20 + 3\,x\,y)}{(8\,y + 7\,x)^2}$$

$$Vy := -\frac{2\,x\,(-20 + 3\,x\,y)}{8\,y + 7\,x} - \frac{6\,x^2\,y}{8\,y + 7\,x} + \frac{16\,x\,y\,(-20 + 3\,x\,y)}{(8\,y + 7\,x)^2}$$

These are ugly. Lets simplify

```
> Vx:=simplify(Vx);
```

$$Vx := -\frac{2\,y^2\,(-160 + 48\,x\,y + 21\,x^2)}{(8\,y + 7\,x)^2}$$

```
> Vy:=simplify(Vy);
```

$$Vy := -\frac{4\,x^2\,(-70 + 12\,y^2 + 21\,x\,y)}{(8\,y + 7\,x)^2}$$

By inspection, we see that Vx and Vy will = 0 when x = 0 = y. However these values are not in the domain of V

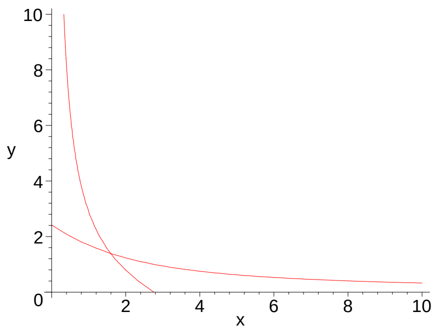
So we next look for the zero's of the other two factors of the top of Vx and Vy

```
> eqns:={-160+48*x*y+21*x^2=0,-70+12*y^2+21*x*y=0};
```

$$eqns := \{-160 + 48\,x\,y + 21\,x^2 = 0,\ -70 + 12\,y^2 + 21\,x\,y = 0\}$$

We can draw these two curves in the plane using implicitplot.

```
> implicitplot(eqns,x=0..10,y=0..10);
```



We want the point of intersection of the two curves.

```
> (solve(eqns,{x,y}));
```

$$\{x = -\frac{8}{7}\,\mathrm{RootOf}(6\,\_Z^2+35),\ y = \mathrm{RootOf}(6\,\_Z^2+35)\},$$
$$\{x = \frac{8}{21}\,\mathrm{RootOf}(2\,\_Z^2-35,\ label = \_L8),\ y = \frac{1}{3}\,\mathrm{RootOf}(2\,\_Z^2-35,\ label = \_L8)\}$$

Here again, this is ugly. Lets evalf it (convert to decimal approximations)

```
> solxy:=evalf(solve(eqns,{x,y}))[2];
```

$$solxy := \{x = 1.593638146,\ y = 1.394433377\}$$

The first solution is complex and out of domain. The second one is clearly the one we want So we declare that

```
> Max_volume:=subs(solxy,V);
```

$$Max\_volume := 2.656063576$$

Around 2.656 cubic feet is the best we can do here. What is the height z of the best box?

```
> subs(solxy,sol);
```

$$1.195228609$$

Note this box doesn't have a square base or side.

More extreme values

```
> with(plots):  #execute this line first
```

5 Discriminant = 0 examples for the second derivative test.

We know that if the discriminant D $= f_{xx}\, f_{yy} - {f_{xy}}^2 < 0$ at a critical point (x_0, y_0), then there is not a local max or local min at (x_0, y_0).

This is because the signs of f_{xx} and f_{yy} are different so f is concave at (x_0, y_0) on one of the lines (x_0, y) and (x, y_0) and concave down on the other line.

Also, we know that if D is positive, then f(x_0, y_0) is a max or min value depending on whether $f_{xx}(x_0, y_0)$ is negative or positive respectively.

This is because of the second derivative test for functions of 1 variable.

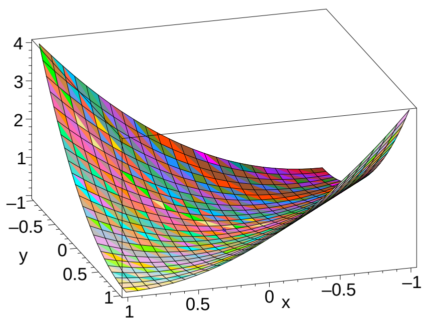
Now we need examples to show that when D = 0, then f(x_0, y_0) could be a max value or a min value or neither max nor min.

Example 1: D=0 at the critical point (0,0), and f(0,0) is a local minimum value.
f(x, y) $= x^2 - 2\,x\,y + y^2$

. Note that $f_{xx}\, f_{yy} - {f_{xy}}^2 = 2\,(2) - 2^2 = 0$, f($x$, y) $= (x - y)^2$ is a parabolic cylinder with minimum value 0 everywhere on the line
$y = -x$

.

```
> display(
  pointplot3d([0,0,0],symbol=circle,color=blue),
  spacecurve([t,t,0],t=-1..1, color=red),
> plot3d(x^2-2*x*y+y^2,x=-1..1,y=-1..1),axes=boxed,orientation
  =[73,57]);
```



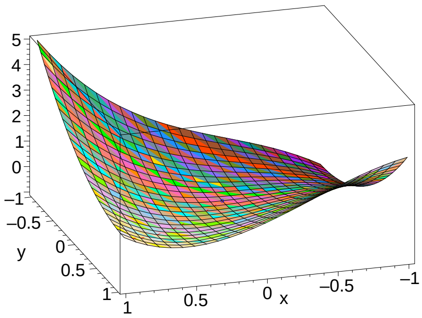
Example 2. D=0 at (0,0) but f(0,0) is a local min.

Just take the negative of example 1

Example 3. D = 0 at (0,0) but f(0,0) is neither a local max nor a local min.

there isn't a quadratic example of this. But if we take example and add x^3 to it, this turns (0,0) into a 'hanging valley' and does not alter the value of D.

```
> display(
  pointplot3d([0,0,0],symbol=circle,color=blue),
  spacecurve([t,t,t^3],t=-1..1, color=red),
> plot3d(x^2-2*x*y+y^2+x^3,x=-1..1,y=-1..1),axes=boxed,orientation
  =[73,57]);
```



6 finding extreme values of a continuous function on a closed and bounded set in the plane

Here is an 'algorithm' for finding the extreme values of a continuous function f on a closed and bounded domain D.

1. Identify the boundary of D and the interior of D. The **interior** of D is the set of points (a,b) in D for which there is a positive ε so that if (x,y) is within ε of (a,b), then (x,y) lies in D. The boundary of D is the points of D that are not in the interior of D. Those points have the property that there are points arbirarily close to them which are not in D.

We assume that f is differentiable on the interior of D.

2. Find the critical points of f in the interior of D, and tabulate the function there.

3. Find the extreme values of f on the boundary of D. The boundary of D is usually a curve of some sort. So one way to optimize f on the boundary would be
to parameterize the boundary with a function r(t) and then optimize the composition f r.

Note 1: r might be piecewise defined. If the boundary is complicated, you might have to break it into pieces and parameterize the pieces.

Note 2: Sometimes it is useful to draw a few gradient vectors for f to see how f is increasing. This might help narrow down the search for maximum and minimum values. So if all the gradient vectors are pointing into the interior away from boundary at a certain place, probably there is not a maximum value of f on the boundary there, because the gradient vectors point in the direction of (greatest) increasing f values.

Examples:

1 Optimize (ie find all extreme values of) f(x,y) = $x\,y + x^2$ on

D = the triangular region with vertices (0,2), (0,-2), and (1,-2)

The boundary of D consists of the three segments connecting the vertices and the interior is all points inside the triangle.

One checks that $f_x = 0 = f_y$ only at (0,0) which is not in the interior of D. Thus the maximum value of f on D must lie on the boundary of D.
We can find the extreme values on each edge separately:
on the edge (0,2) to (0,-2) f(x,y) = 0
on the edge (0,2) to (1,-2), $y = 2 - 4\,x$ so we want to optimize $\mathrm{f}(x,\,y) = \mathrm{f}(x,\,2 - 4\,x) = x\,(2 - 4\,x) + x^2 = 2\,x$
for $0 \le x \ \le 1$
the derivative $2 - 6\,x = 0$ when x = 1/3, so the extreme values on this interval must be among the values f(0,2) = 0, f(1/3,2/3) = 1/3 and f(1,-2) = -1.
on the edge (0,-2) to (1,-2), f(x,y) = f(x,-2) = $-2\,x + x^2$ for x between 0 and 1. the derivative -2 + 2x = 0 when x = 1, so the extreme values
on this interval are f(0,-2) = 0 and f(1,-2) = -1
Combining these efforts, we see that f(1/3,2/3) = 1/3 is the maximum value and f(1,-2) = -1 is the minimum value of f(x,y) on the given triangular region.

We can check our work visually in Maple

```
> with(plots):
```

This is a domain restriction function. It returns (x,y) if (x,y) is in the domain of f, otherwise it returns (0,0) (so that f(r(x,y)) = f(x,y) if (x,y) is in the domain of f and f(r(x,y)) = 0 otherwise.

```
> r:= proc(x,y)
  if x>= 0 and y >=-2 and y <= 2 - 4*x then op([x,y]) else op([0,0])
  fi end;
```

$r := \mathbf{proc}(x,\,y)$
$\quad \mathbf{if}\, 0 \le x \ \mathbf{and}\ -2 \le y\ \mathbf{and}\ y \le 2 - 4 * x\, \mathbf{then}\, \mathrm{op}([x,\,y])\, \mathbf{else}\, \mathrm{op}([0,\,0])\, \mathbf{end\ if}$
$\mathbf{end\ proc}$

```
> f := (x,y)->x*y + x^2;
```

$$f := (x,\,y) \to x\,y + x^2$$

@ is the composition operator in Maple.

```
> h :=f @ r;;
```

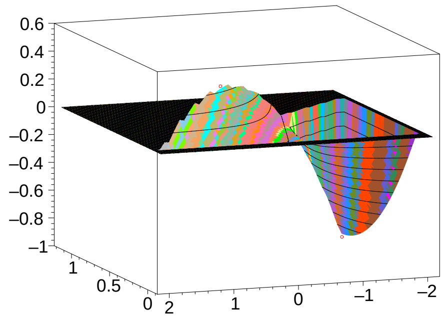
$$h := f@r$$

```
> h(1/3,2/3);
```

$$\frac{1}{3}$$

Here is the graph of f, with the absolute minimum and maximum values shown.

```
> display(pointplot3d({[1,-2,h(1,-2)],[1/3,2/3,h(1/3,2/3)]
  },color=red,symbol=circle),
  plot3d(h,
> -.1..1.2,-2.1..2.1,view=-1..3/5,style=patchcontour,numpoints=4000,orie
  ntation=[160,77],axes=boxed));
```



2. Optimize f$(x,\, y) = x\, y$ on the closed disk $x^2 + y^2 \leq 1$

you do this one. Also try to draw a picture to verify your conclusions visually.

First change the definition of r so that r(x,y) = (x,y) if x^2 + y^2 <= 1, otherwise r(x,y) = (0,0)

```
> r:= proc(x,y)
  if x>= 0 and y >=-2 and y <= 2 - 4*x then op([x,y]) else op([0,0])
  fi end;
```

$r := \mathbf{proc}(x,\, y)$
$\quad \mathbf{if}\, 0 \leq x \;\mathbf{and}\; -2 \leq y \;\mathbf{and}\; y \leq 2 - 4 * x \,\mathbf{then}\, \mathrm{op}([x,\, y]) \,\mathbf{else}\, \mathrm{op}([0,\, 0]) \;\mathbf{end\ if}$
$\mathbf{end\ proc}$

the definition of f remains unchanged.

```
> f := (x,y)->x*y + x^2;
```

$$f := (x,\, y) \to x\, y + x^2$$

Also, h will remain unchanged as the composition of f with r.

```
> h :=f @ r;;
```

$$h := f@r$$

Here is the picture you need to modify. You will need to modify the location of the extreme points on the graph.

```
> display(pointplot3d({[1,-2,h(1,-2)],[1/3,2/3,h(1/3,2/3)]
  },color=red,symbol=circle),
  plot3d(h,
  -.1..2,-2.1..2.1,view=-1..1,style=patchcontour,numpoints=4000));
```

7 The locus of the centroid of a pie slice.

Here's a problem. Find the centroid of a pie slice as a function of its radius R and its central angle T. First look at the extremes of this problem. If T = 0 then the centroid is going to be the midpoint of the radius $\frac{R}{2}$. If T = $2\,\pi$, then the centroid is going to be at the center of the circle. If T is held fixed and R is allowed to change, the centroid will move along the midline of the pie slice. The interesting question is what happens when R is held fixed and T changes from 0 to 2*Pi, what is the locus of the centroid? We can work this out if we can determine the centroid as a function of T and R.

First the area of the slice is

```
> A :=int(int(r,r=0..R),t=0..T);
```

$$A := \frac{R^2\,T}{2}$$

Next the moment of the slice about the y-axis is

```
> My := int(int(r*cos(t)*r,r=0..R),t=0..T);
```

$$My := \frac{1}{3}\sin(T)\,R^3$$

So xbar, the x-coordinate of the centroid is My/A

```
> xbar:= unapply(My/A,R,T);
```

$$xbar := (R,\,T) \to \frac{2}{3}\,\frac{\sin(T)\,R}{T}$$

```
> Mx := int(int(r*sin(t)*r,r=0..R),t=0..T);
```

$$Mx := \frac{R^3}{3} - \frac{1}{3}\cos(T)\,R^3$$

```
> ybar:= unapply(simplify(Mx/A),R,T);
```

$$ybar := (R,\,T) \to -\frac{2}{3}\,\frac{R\,(-1+\cos(T))}{T}$$

Lets check our calculation by drawing the pie slice and its centroid to see if it looks reasonable.

```
> with(plottools);
```

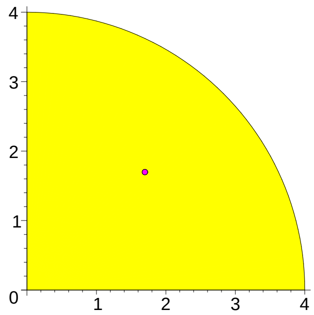
[*arc*, *arrow*, *circle*, *cone*, *cuboid*, *curve*, *cutin*, *cutout*, *cylinder*, *disk*, *dodecahedron*, *ellipse*, *ellipticArc*, *hemisphere*, *hexahedron*, *homothety*, *hyperbola*, *icosahedron*, *line*, *octahedron*, *parallelepiped*, *pieslice*, *point*, *polygon*, *project*, *rectangle*, *reflect*, *rotate*, *scale*, *semitorus*, *sphere*, *stellate*, *tetrahedron*, *torus*, *transform*, *translate*, *vrml*]

```
> pslice:= proc(R,T)
  use plots,plottools in
  display(disk(evalf([xbar(R,T),ybar(R,T)]),.01*R,color=magenta),pieslic
```

```
> e([0,0],R,0..T,color=yellow),scaling=constrained);
  end use
  end:
```

Checking the centroid of the quarter pie.

```
> pslice(4,Pi/2);
```



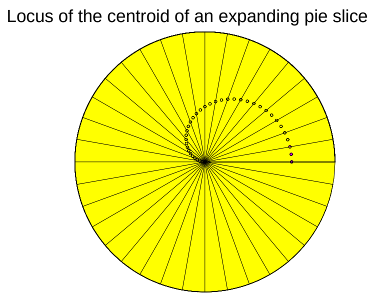

```
> xbar(R,Pi/2);
```

$$\frac{4\,R}{3\,\pi}$$

Note this agrees with our calculation in class on Wednesday.

We can make a movie showing the locus being traced out by the centroid of the pie slice as the angle of the slice goes from 0 to 360.

```
> plots[display](seq(pslice(4,(i+.0001)*(2*Pi/36)),i=0..36),scaling=con
  strained,axes=none,insequence=true,title="Locus of the centroid of an
  expanding pie slice");
```



What are some questions that occur to us as we watch this movie?

First. we notice that when T is small, the centroid is not close to the midpoint, as we might expect it to be. The reason becomes clear when look again at the centroid, [xbar(R,T),ybar(R,T)]

```
> xbar(R,T),ybar(R,T);
```

$$\frac{2}{3}\,\frac{\sin(T)\,R}{T},\ -\frac{2}{3}\,\frac{R\,(-1+\cos(T))}{T}$$

What happens to xbar as T appoaches 0? Since $\frac{\sin(T)}{T}$ goes to 1 and $\frac{1-\cos(T)}{T}$ goes to 0 we see the centroid approaches the point [$\frac{2R}{3}$, 0]! So, there is a jump discontinuity in the locus at T = 0.

Other questions (In all of these, consider R to be fixed.):
What is the maximum value of ybar?
What is the minimum value of xbar?
What is the length of the locus?
What is area under the locus?
What is the centroid of the region under the locus?

Other questions.

Take any nice 1 parameter family of regions and ask for the locus of the centroid. For example, it was asked in class

where the centroid of a trapezpoidal region is?

Making this question very specific, for the right triangle with vertices [0,0], [a,0], [0,b] and h between 0 and b, let T be

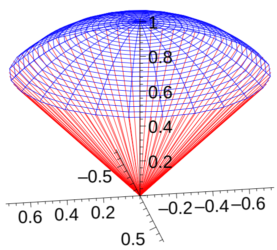
the trapezoid with base [0,0] to [a,0] and top [0,h] to $[a\,(1-\frac{h}{b}),\,h]$. Calulate the centroid of T as a function of a, b, and h. Use this to investigate the locus of the centroid with a and b fixed and h going from 0 up to b. Draw pictures to check your work.

```
> restart:with(plottools):with(plots):
```

8 The centroid of an ice cream cone (spherical box)

Fix ϕ_0 between 0 and π. We want to compute the centroid of the solid S_{ϕ_0} consisting of all points (ρ, ϕ, θ) such that ρ is between 0 and 1, ϕ is between 0 and ϕ_0, and θ is between 0 and 2π. Here is it's picture for $\phi_0 = \frac{\pi}{4}$.

```
> icc:=(phi0,n)->display(plot3d(1, t=0..2*Pi, p=0..phi0,
  coords=spherical,
  style=wireframe,color=blue),seq(line([0,0,0],[sin(phi0)*cos(2*Pi*i/n),
> sin(phi0)*sin(2*Pi*i/n),cos(phi0)],thickness=2,color=red),i=1..n),scal
  ing=constrained,axes=normal,orientation=[80,70]):
> icc(Pi/4,72);
```




```
> Int(x^3,x=0..3);
```

$$\int_0^3 x^3\,dx$$

```
> evalf(%);
```

$$20.25000000$$

By inspection, we can see that the centroid is (0,0,a) where a is somewhere between 0 and 1, depending on ϕ_0. The volume and moment about the xy-plane can best be evaluated as iterated integrals in spherical coordinates.

```
> Int(Int(Int(1,z),x=S[phi[0]]..``),y)=Int(Int(Int(1*rho^2*(sin(phi)),r
  ho=0..1),phi=0..phi0),theta=0..2*Pi);
```

$$\int\int_{S_{\phi_0}}\int 1\,dz\,dx\,dy = \int_0^{2\pi}\int_0^{\phi 0}\int_0^1 \rho^2\sin(\phi)\,d\rho\,d\phi\,d\theta$$

So the volume of the ice cream cone with cone angle $2\phi_0$ and cone slant height 1 is

```
> V:=int(int(int(1*rho^2*(sin(phi)),rho=0..1),phi=0..phi0),theta=0..2*P
  i);;
```

$$V := \frac{2\pi}{3} - \frac{2}{3}\cos(\phi 0)\,\pi$$

And the moment about the xy plane is

```
> Int(Int(Int(z,z),x=S[phi[0]]..``),y)=Int(Int(Int(rho*cos(phi)*rho^2*si
n(phi),rho=0..1),
phi=0..phi[0]),theta=0..2*Pi);
```

$$\int\int_{S_{\phi_0}}\int z\,dz\,dx\,dy = \int_0^{2\pi}\int_0^{\phi_0}\int_0^1 \rho^3\cos(\phi)\sin(\phi)\,d\rho\,d\phi\,d\theta$$

This evaluates to

```
> Mxy:=int(int(int(rho*cos(phi)*rho^2*sin(phi),rho=0..1),phi=0..phi0),t
heta=0..2*Pi);
```

$$Mxy := \frac{\pi}{4} - \frac{1}{4}\cos(\phi 0)^2\,\pi$$

So we can calculate the z-coordinate of the centroid, zbar as

```
> zbar:= unapply(Mxy/V,phi0);
```

$$zbar := \phi 0 \to \frac{\frac{\pi}{4} - \frac{1}{4}\cos(\phi 0)^2\,\pi}{\frac{2\pi}{3} - \frac{2}{3}\cos(\phi 0)\,\pi}$$

And for example when the cone angle is 90 degrees, zbar is

```
> simplify(zbar(Pi/4))=evalf(zbar(Pi/4));
```

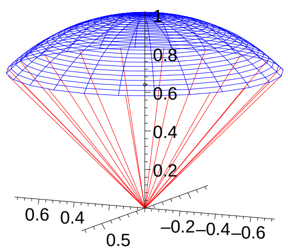
$$-\frac{3}{8\,(-2+\sqrt{2})} = 0.6401650420$$

We can add the centroid to the picture and check to see if it looks reasonable.

```
> picture:=
phi0->display(pointplot3d([0,0,evalf(zbar(phi0))],symbol=circle,color=
black,thickness=3),
icc(phi0,20),scaling=constrained,orientation=[116,80]);
```

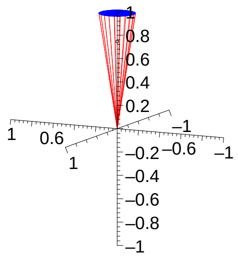
$$picture := \phi 0 \to plots:-display(plots:-pointplot3d([0,\ 0,\ \mathrm{evalf}(\mathrm{zbar}(\phi 0))],$$
$$symbol = plottools:-circle,\ color = black,\ thickness = 3),\ \mathrm{icc}(\phi 0,\ 20),$$
$$scaling = constrained,\ orientation = [116,\ 80])$$

```
> picture(Pi/4);
```



This seems eminately reasonable. Now lets make a movie of the locus of the centroid of the spherical box as ϕ_0 goes from 0 to Pi.

```
> display(seq(picture(i*Pi/20),i=1..20),scaling=constrained,insequence=
true);
```



As in the case of the centroid of the pie slice, we see that as ϕ_0 approaches π , the centroid moves to the center of the ball. At the other extreme , when ϕ_0 is getting closer to 0, the ice cream cone is approaching a segment (with centroid at the middle of the segment), but the centroid is approaching a number closer to 1 than 0. In the case of the pie slice, this number was 2/3, the limit of the centroid of the triangle approximation to the pie slice. In this case the number is

```
> Limit(zbar(t),t=0,right)=limit(zbar(t),t=0,right);
```

$$\lim_{t \to 0+} \frac{\frac{\pi}{4} - \frac{1}{4}\cos(t)^2\,\pi}{\frac{2\,\pi}{3} - \frac{2}{3}\cos(t)\,\pi} = \frac{3}{4}$$

Question: Why that number? Perhaps if you solve the problem below, you could come up with an explanation.

Problem: Calculate the centroid of the cone of slant height 1 and base radius r.

9 drawing the intersection of perpendicular cylinders

9.1 Two cylinders

The goal is to draw a picture of the intersection of the two cylinders of radius 1 centered on the x-axis and y-axis respectively, and calculate the volume.

```
> with(plots):
> with(plottools):
```

We can draw a cylinder of color clr and radius r about the line y = tan(theta)*x in the xy plane using tubeplot

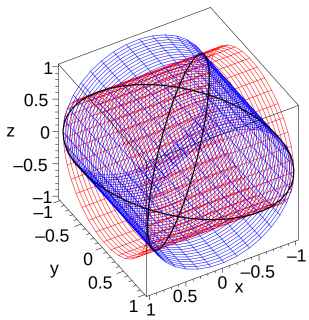
```
> ?plots[tubeplot]
> cylint:= (r,theta,clr)
  ->display(tubeplot([t*cos(theta),t*sin(theta),0],t=-1..1,
  numpoints=40,tubepoints=36,radius=r),scaling=constrained,color=clr,axe
> s=boxed):
```

The intersection of the cylinders of radius 1 about the x-axis and y-axis can be obtained by solving the equations x^2 + z^2 = 1 and y^2+z^2=1 simultaneously. What we get is two ellipses x = y, x^2+z^2 = 1 and x = -y, x^2+z^2 = 1. These can be parameterized with x=sin(t), y = sin(t), z = cos(t) t = 0..2*Pi and x=sin(t), y = -sin(t), z = cos(t), t = 0..2*Pi. We can use spacecurve to draw these.

```
> inter :=
  display(spacecurve([sin(t),sin(t),cos(t)],t=0..2*Pi,thickness=3,color=
  black)):
> inter2 :=
  display(spacecurve([sin(t),-sin(t),cos(t)],t=0..2*Pi,thickness=3,color
  =black)):
```

Now we can draw roughly the intersection of the cylinders

```
> display(inter,inter2,cylint(1,Pi/2,blue),cylint(1,0,red),labels=[x,y,
  z],style=wireframe,orientation=[60,50]);
```



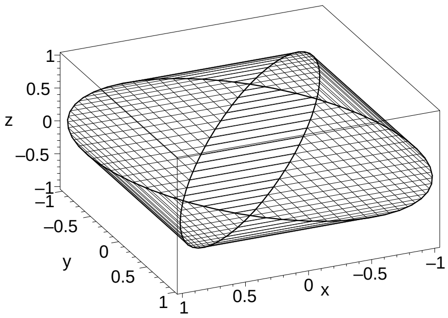
If you look at this from the top, you can see that the cross-sections by planes parallel with the xy-plane with the intersection are squares. We can draw those squares with sq below. Then frame defined underneath shows a bunch of the cross-sections.

```
> sq := (t,clr)->display(polygon([[sin(t),sin(t),cos(t)],
  [sin(t),-sin(t),cos(t)],[-sin(t),-sin(t),cos(t)],
  [-sin(t),sin(t),cos(t)]],color=clr)):
```

```
> frame:=
  n->display(seq(sq(i*Pi/36,black),i=1..n),inter,inter2,labels=[x,y,z],s
  tyle=wireframe,orientation=[66,50]);
```

$$frame := n \to plots:-display(\mathrm{seq}(\mathrm{sq}(\frac{i\,\pi}{36},\ black),\ i=1..n),\ inter,\ inter2,\ labels=[x,\ y,\ z],$$
$$style = wireframe,\ orientation = [66,\ 50])$$

```
> pic:=display(frame(36),axes = boxed):
> pic;
```



From this picture we can see that the cross-section by a plane at height h between -1 and 1 is a square of side $2\sqrt{1-h^2}$. Hence the volume of the intersection of the two cylinders is $\int_{-1}^{1} 4 - 4\,h^2\,dh = \frac{16}{3}$

```
> Int(4*(1-h^2),h=-1..1) = int(4*(1-h^2),h=-1..1);
```

$$\int_{-1}^{1} 4 - 4\,h^2\,dh = \frac{16}{3}$$

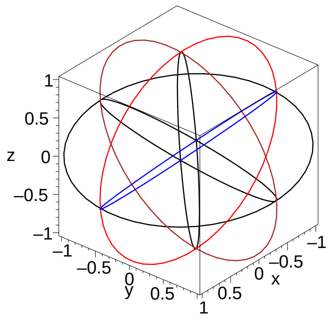
Further questions: What is the surface area of the intersection? also what is the total length of the edges of the intersection.

9.2 Three Cylinders

Now lets draw the intersection of the 3 mutually perpendicular cylinders of radius 1 about the x, y, and z axes. The intersection of the cylinder about the z-axis with the other two cylinders in succession give 4 more ellipses which are parameterized and drawn below.

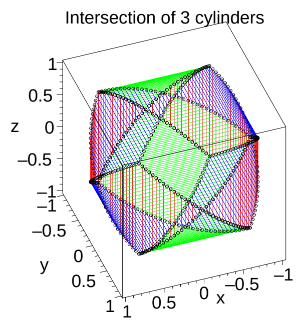
```
> inter3 :=
  display(spacecurve([sin(t),cos(t),sin(t)],t=0..2*Pi,thickness=3,color=
  black)):
> inter4 :=
  display(spacecurve([-sin(t),cos(t),sin(t)],t=0..2*Pi,thickness=3,color
  =red)):
> inter5 :=
  display(spacecurve([cos(t),sin(t),sin(t)],t=0..2*Pi,thickness=3,color=
  blue)):
> inter6 :=
  display(spacecurve([cos(t),-sin(t),sin(t)],t=0..2*Pi,thickness=3,color
  =brown)):
```

```
> display(inter5,inter6,inter4,inter3,inter,inter2,labels=[x,y,z],scali
  ng=constrained,orientation=[40,60],axes=boxed);
```



We can see that the surface of the intersection consists of 12 congruent quadrilateral pieces of a cylinder. They are grouped into 3 groups of 4 pieces of each of the intersecting cylinders. Here is a color coded picture of the surface.

```
> f := proc(t)
  options operator;
  if evalf(t)<=evalf(Pi/4) then [sin(t),sin(t),cos(t)]
> else [cos(t),sin(t),cos(t)] fi; end:
  g := proc(t)
  options operator;
> if evalf(t)<=evalf(Pi/4) then [-sin(t),sin(t),cos(t)]
  else [-cos(t),sin(t),cos(t)] fi; end:
> surf:=clr->display(seq(line(f(i*Pi/(2*36)),g(i*Pi/(2*36)),color=clr),
  i=1..36),seq(pointplot3d({f(i*Pi/(2*36)),g(i*Pi/(2*36))
  },symbol=circle,color=black),i=0..36)):
> bluesurf:=rotate(display(seq(line(f(i*Pi/(2*36)),g(i*Pi/(2*36)),color
  =blue),i=1..36)),0,0, Pi/2):
> pic:=display(seq(rotate(bluesurf,0,i*Pi/2,0),i=0..3),seq(rotate(refle
  ct(surf(green),[[0,0,0],[0,0,1],[1,0,0]]),i*Pi/2,0,0 ),i=0..3),
  seq(rotate(rotate(surf(red),0, Pi/2,0),0,0,i*Pi/2),i=0..3),
> scaling=constrained,labels=[x,y,z],axes=boxed,orientation=[70,50],titl
  e="Intersection of 3 cylinders"):
> pic;
```



Further examination shows there are 6 vertices of order 4, 2 for each pair of intersecting cylinders and 8 vertices of order 3, representing the $2^3 = 8$ points of intersection of the 3 cylinders. There are also 24 edges. It makes for ball you wouldn't want to play soccer with.

What about the volume? Well, by the symmetry it will 16*V where V is the part trapped above the region 0 <= x <= y and $x^2 + y^2 \leq 1$ in the xy-plane and the cylinder $x^2 + z^2 = 1$. This can be written as the sum of two iterated integrals: $\int_0^{\frac{1}{\sqrt{2}}} \int_0^x \sqrt{1-x^2}\, dy\, dx + \int_{\frac{1}{\sqrt{2}}}^1 \int_0^{\sqrt{1-x^2}} \sqrt{1-x^2}\, dy\, dx$

```
> V1:=int(sqrt(1-x^2)*x,x=0..1/sqrt(2));
```

$$V1 := \frac{\sqrt{2}\,\sqrt{\pi}\,(\frac{4\sqrt{2}}{3\sqrt{\pi}} - \frac{2}{3\sqrt{\pi}})}{8}$$

```
> V2:=int(sqrt(1-x^2)*sqrt(1-x^2),x=1/sqrt(2)..1);
```

$$V2 := \frac{2}{3} - \frac{5\sqrt{2}}{12}$$

So the total volume of the intersection is

```
> 16*(V1+V2)=evalf(16*(V1+V2));
```

$$2\sqrt{2}\,\sqrt{\pi}\,(\frac{4\sqrt{2}}{3\sqrt{\pi}} - \frac{2}{3\sqrt{\pi}}) + \frac{32}{3} - \frac{20\sqrt{2}}{3} = 4.686291496$$

Further questions: Calculate the surface area of the intersection. calculate the total length of the edges of the surface.

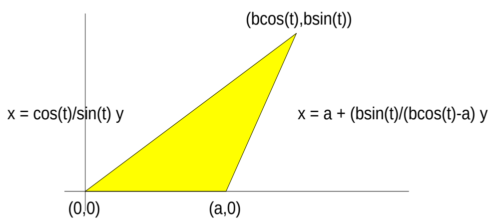
Problem: Find a formula for the volume of a truncated prism TP with base a triangle with sides a and b and included angle t, and heights h1,h2, and h3.

Solution: For starters, we might guess that the formula is the area of the triangle times the average of the heights: $V = \frac{a\,b\sin(t)}{2}\,\frac{h1+h2+h3}{3}$.

But we can develop a formula by setting up a coordinate system, expressing the volume as the integral over the triangle of the height function, and then hoping that the integral can be evaluated as an iterated integral to reveal a 'nice' formula.

Let's choose the origin [0,0,0] to be at the vertex of the included angle t. Then label the height over the vertex h1 and give it its coordinates [0,0,h1]. The a leg we'll put along the positive x-axis with endpoint [a,0,0], the b leg then will have coordinates [b*cos(t),b*sin(t),0]. The height h2 and h3 sit over these two points at [a,0,h2] and [b*cos(t),b*sin(t),h3]. Here is what the base looks like.

```
> with(plots):with(plottools):
  display(polygon([[0,0],[2,0],[3,2]],color=yellow),textplot([4.5,1,"x =
  a + (bsin(t)/(bcos(t)-a) y"]),
> textplot([-.2,1,"x = cos(t)/sin(t) y"]),
  textplot([0,-.2,"(0,0)"]),textplot([2,-.2,"(a,0)"]),
  textplot([3.1,2.2,"(bcos(t),bsin(t))"]),scaling=constrained,xtickmarks
  =[],ytickmarks=[]);
```



The way we have set this up, it is best to think of the base triangle T as a type 2 region: As y goes from 0 to $b\sin(t)$, x goes from $\cot(t)\,y$ to $a + \frac{b\sin(t)\,y}{b\cos(t)-a}$

The height function f(x,y) can be written f(x,y) = $z = h1 + \frac{h2\,x}{a} + w\,y$, where w is obtained by substituting in the point z = h3, x = bcos(t), y = bsin(t) and solving for w.

Now we will use Maple to do the evaluation.

First, get the height as a function of x and y.

```
> z := h1+h2/a*x + w*y;
```

$$z := h1 + \frac{h2\,x}{a} + w\,y$$

```
> z:=subs(solve(subs({x=b*cos(t),y=b*sin(t)},z=h3),{w}),z);
```

$$z := h1 + \frac{h2\,x}{a} - \frac{(h1\,a + h2\,b\cos(t) - h3\,a)\,y}{b\sin(t)\,a}$$

Now perform the evaluation.

```
> V:=Int(Int(z,x=cos(t)/sin(t)*y..a+(b*cos(t)-a)/(b*sin(t))*y),y=0..b*s
  in(t));
```

$$V := \int_0^{b\sin(t)} \int_{\frac{\cos(t)\,y}{\sin(t)}}^{a+\frac{(b\cos(t)-a)\,y}{b\sin(t)}} h1 + \frac{h2\,x}{a} - \frac{(h1\,a + h2\,b\cos(t) - h3\,a)\,y}{b\sin(t)\,a}\,dx\,dy$$

```
> V:=int(int(z,x=cos(t)/sin(t)*y..a+(b*cos(t)-a)/(b*sin(t))*y),y=0..b*s
  in(t));
```

$$V := \frac{1}{3}\left(\frac{1}{2}\,\frac{h2\,(\frac{(b\cos(t)-a)^2}{b^2\sin(t)^2} - \frac{\cos(t)^2}{\sin(t)^2})}{a}\right.$$
$$\left. - \frac{(h1\,a + h2\,b\cos(t) - h3\,a)\,(\frac{b\cos(t)-a}{b\sin(t)} - \frac{\cos(t)}{\sin(t)})}{b\sin(t)\,a}\right) b^3\sin(t)^3 + \frac{1}{2}$$
$$(h1\,(\frac{b\cos(t)-a}{b\sin(t)} - \frac{\cos(t)}{\sin(t)}) + \frac{h2\,(b\cos(t)-a)}{b\sin(t)} - \frac{h1\,a + h2\,b\cos(t) - h3\,a}{b\sin(t)})\,b^2$$
$$\sin(t)^2 + h1\,a\,b\sin(t) + \frac{1}{2}\,h2\,a\,b\sin(t)$$

This is not the formula I want to remember. Let's simplify it.

```
> V:=simplify(V);
```

$$V := \frac{1}{6}\,a\,b\sin(t)\,(h2 + 2\,h1 + h3)$$

```
> V:=unapply(V,a,b,t,h1,h2,h3);
```

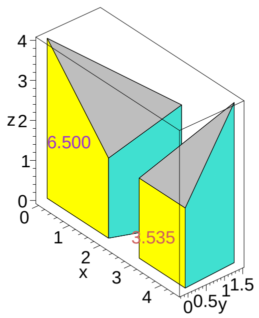
$$V := (a,\,b,\,t,\,h1,\,h2,\,h3) \to \frac{1}{6}\,a\,b\sin(t)\,(h2 + 2\,h1 + h3)$$

Much much better. We can rewrite this formula as V $= \frac{a\,b\sin(t)}{2}\;\frac{2\,h1 + h2 + h3}{3}$ = area of base times 'a weighted sum of the heights'. This is close to our guess, but we have to double the weight of the height over the vertex in the calculation.

Here is a procedure to draw the truncated prism with its volume on the front face.

```
> drawit:=proc(a,b,t,h1,h2,h3)
  local R,P,Q,A,B,C,v;
  use plots,plottools in
> R := [0,0,0]:
  P := [a,0,0]:
  Q := [b*cos(t),b*sin(t),0]:
> A:=[0,0,h1]:
  B:=[a,0,h2]:
  C:=Q+[0,0,h3]:
  v:=evalf(V(a,b,t,h1,h2,h3),4);
> display(textplot3d([a/2,-.2,h1/2,v]),polygon([R,P,Q],color=grey),poly
  gon([R,P,B,A],color=yellow),polygon([P,Q,C,B],color=turquoise),
  polygon([Q,R,A,C],color=magenta),polygon([A,B,C],color=grey),scaling=c
```

```
> onstrained);
  end use;
  end:
> display(drawit(2,3,Pi/6,4,2,3),translate(drawit(1.5,2,Pi/4,2,2,4),3,0
  ,0),orientation=[-40,57],axes=boxed,labels=[x,y,evaln(z)]);
```



Problem: Use this same idea to develop a formula for the volume of a truncated rectangular solid with base rectangle a by b, and three given heights h1, h2, and h3.

10 Line integral problems

Problem:

Let C be the curve sitting on the surface of the graph of smooth function z = f(x,y) over the arc from (0,0) to (1,2), and let g(x,y,z) be a nice smooth density function defined on C. Represent the mass and center of mass of C in terms of line integrals. Evaluate these integrals for f(x,y) = $x^2 + \frac{x\,y}{2}$ and

g(x,y,z) = z. Draw a picture showing the surface f and the curve C together with its center of mass.

A solution:

The mass of C is $M = \int_C \mathrm{g}(x,\,y,\,z)\,ds$. The moment about the xy-plane is $M_{x,\,y} = \int_C z\,\mathrm{g}(x,\,y,\,z)\,ds$. The moments about the other two planes are defined in the same way.

We can parameterize C by r(t) = (x,y,z) = (t,2*t,f(t,2*t)). So we can evaluate these 4 line integrals like so

```
> M := Int(2*t^2*sqrt(1+4+16*t^2),t=0..1);
```

$$M := \int_0^1 2\,t^2\,\sqrt{5+16\,t^2}\,dt$$

```
> M:=int(2*t^2*sqrt(1+4+16*t^2),t=0..1);;
```

$$M := -\frac{5}{8}\,\frac{\frac{5\sqrt{\pi}}{128} - \frac{5}{64}\left(\frac{1}{2} - 6\ln(2) + \ln(5)\right)\sqrt{\pi} - \frac{37\sqrt{\pi}\sqrt{21}}{40} + \frac{5}{32}\sqrt{\pi}\ln\left(\frac{1}{2} + \frac{\sqrt{21}}{8}\right)}{\sqrt{\pi}}$$

```
> M:= evalf(M);
```

$$M := 2.517952442$$

```
> Mxy:=Int((2*t^2)^2*sqrt(1+4+16*t^2),t=0..1);
```

$$Mxy := \int_0^1 4\,t^4\,\sqrt{5+16\,t^2}\,dt$$

```
> Mxy:=int((2*t^2)^2*sqrt(1+4+16*t^2),t=0..1);
```

$$Mxy := -\frac{5}{4}\,\frac{-\frac{125\sqrt{\pi}}{12288} + \frac{25}{2048}\left(\frac{5}{6} - 6\ln(2) + \ln(5)\right)\sqrt{\pi} - \frac{711\sqrt{\pi}\sqrt{21}}{1280} - \frac{25}{1024}\sqrt{\pi}\ln\left(\frac{1}{2} + \frac{\sqrt{21}}{8}\right)}{\sqrt{\pi}}$$

```
> Mxy:=evalf(Mxy);
```

$$Mxy := 3.222893594$$

```
> zbar:=Mxy/M;
```

$$zbar := 1.279966031$$

```
> Myz:=Int((2*t^2)*t*sqrt(1+4+16*t^2),t=0..1);
```

$$Myz := \int_0^1 2\,t^3\,\sqrt{5+16\,t^2}\,dt$$

```
> Myz:=int((2*t^2)*t*sqrt(1+4+16*t^2),t=0..1);
```

$$Myz := -\frac{5\,(-\frac{133\sqrt{\pi}\sqrt{21}}{200} - \frac{\sqrt{\pi}\sqrt{5}}{24})}{8\sqrt{\pi}}$$

```
> Myz:=evalf(Myz);
```

$$Myz := 1.962863960$$

```
> xbar:=Myz/M;
```

$$xbar := 0.7795476703$$

Note that since y = 2*x, Mxz = 2Myz. and so ybar = 2*xbar

```
> Mxz:=2*Myz;
```

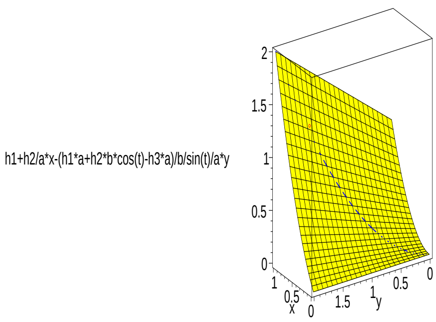
$$Mxz := 3.925727920$$

```
> ybar:=Mxz/M;
```

$$ybar := 1.559095341$$

Now to draw the picture:

```
> use plots in
  display(pointplot3d([xbar,ybar,zbar],symbol=circle,color=red), #put
  this in last
> spacecurve([t,2*t,2*t^2],t=0..1,color=blue,thickness=3), #then added
  this
  plot3d(x^2+x*y/2,x=0..1,y=0..2,color=yellow),  #drew this first
> scaling=constrained,  #these plot options were added to make the
  picture more legible.
  axes=boxed,
> labels=[x,y,z],
  orientation=[147,72]);
> end use;
```



If this seems a little high, recall that the density function is the z-coordinate and so the center of mass is pulled up significantly toward the upper end of the wire.

Problem for you: Recalculate the center of mass if the density of the wire is constant (say 1). Redraw the wire.

11 Drawing vector fields

Given a vector field F = <P,Q> we want to plot enough of its values to be able to sketch in (or at least to visualize) the streamlines of F (the curves which are tangent to the field value at each point where they are). Maple has a nice word (fieldplot) in the plots package to draw these values for you.

```
> with(plots);
```

[*animate*, *animate3d*, *animatecurve*, *arrow*, *changecoords*, *complexplot*, *complexplot3d*, *conformal*, *conformal3d*, *contourplot*, *contourplot3d*, *coordplot*, *coordplot3d*, *densityplot*, *display*, *dualaxisplot*, *fieldplot*, *fieldplot3d*, *gradplot*, *gradplot3d*, *graphplot3d*, *implicitplot*, *implicitplot3d*, *inequal*, *interactive*, *interactiveparams*, *intersectplot*, *listcontplot*, *listcontplot3d*, *listdensityplot*, *listplot*, *listplot3d*, *loglogplot*, *logplot*, *matrixplot*, *multiple*, *odeplot*, *pareto*, *plotcompare*, *pointplot*, *pointplot3d*, *polarplot*, *polygonplot*, *polygonplot3d*, *polyhedra_supported*, *polyhedraplot*, *rootlocus*, *semilogplot*, *setcolors*, *setoptions*, *setoptions3d*, *spacecurve*, *sparsematrixplot*, *surfdata*, *textplot*, *textplot3d*, *tubeplot*]

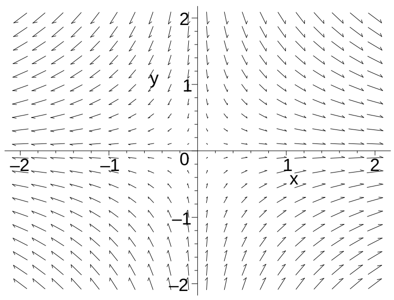
you can get examples of how to use the word (and the 3d version fieldplot3d) by looking at the bottom of the helpsheet that pops when you execute

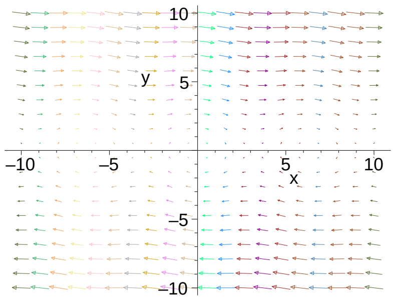
```
> ?fieldplot
```

I just copied the examples below from that helpsheet and pasted them into this worksheet.

11.1 Examples

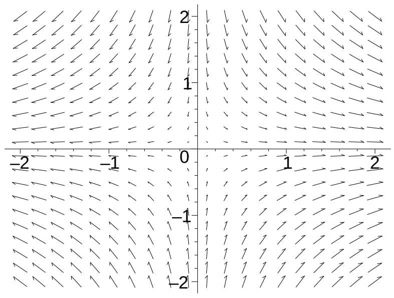
```
> with(plots):
  fieldplot([x/(x^2+y^2+4)^(1/2),-y/(x^2+y^2+4)^(1/2)],x=-2..2,y=-2..2);
  fieldplot([y,-sin(x)-y/10],x=-10..10,y=-10..10,arrows=SLIM,
  color=x);
```

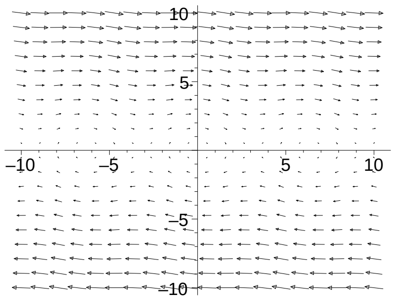




The following two examples gives the same results as above

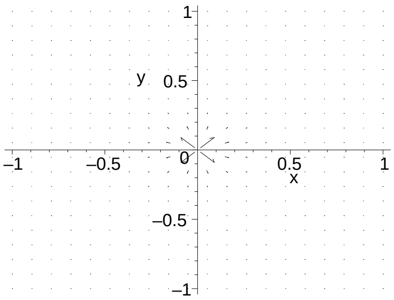
```
> f := (x,y)-> x/(x^2+y^2+4)^(1/2):  g := (x,y)-> -y/(x^2+y^2+4)^(1/2):
  fieldplot( [f,g],-2..2,-2..2);
  f := (x,y)-> y:  g := (x,y)-> -sin(x) - y/10:
  fieldplot( [f,g],-10..10,-10..10,arrows=SLIM);
```





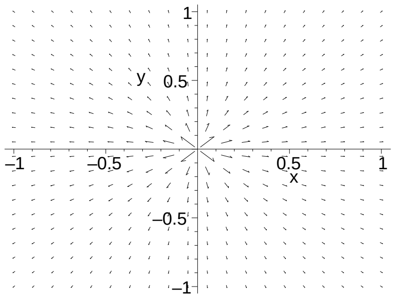
An inverse square law without fieldstrength adjustment

```
> fieldplot([x/(x^2+y^2)^(3/2),y/(x^2+y^2)^(3/2)],
  x=-1..1,y=-1..1);
```



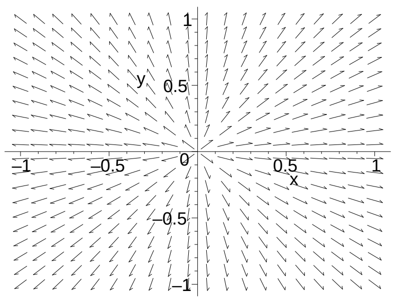
where we note that only the arrows very close to (0,0) are visible. Now using fieldstrength=log

```
> fieldplot([x/(x^2+y^2)^(3/2),y/(x^2+y^2)^(3/2)],
  x=-1..1,y=-1..1,fieldstrength=log);
```



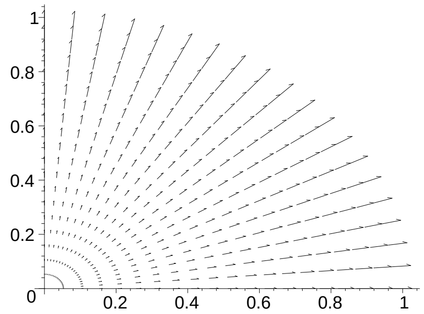
which makes the direction of the arrows much more visible. Alternatively a pure direction plot can be produced:

```
> fieldplot([x/(x^2+y^2)^(3/2),y/(x^2+y^2)^(3/2)],
  x=-1..1,y=-1..1,fieldstrength=fixed);
```



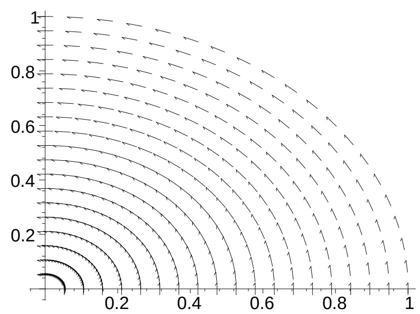
A radial field in polar coordinates

```
> fieldplot([r,0],r=0..1,t=0..Pi/2,coords=polar);
```



A field in polar coordinates that will not draw the arrows at the origin (the field direction is undefined there).

```
> fieldplot([0,1],r=0..1,t=0..Pi/2,coords=polar);
```



12 Greens theorem.

Green's theorem states that if F = <P,Q> has continuous partial derivatives on an open set containing the positively oriented simple closed curve C and the region D it bounds, then the line integral of the the tangential component of F with repect to arc length around C is the double integral over D of the difference $(\frac{\partial}{\partial x} Q) - (\frac{\partial}{\partial y} P)$.

The left hand side of Green's theorem has a nice meaning: If the integral is positive, then on average the tangential component of F is positive, so if we take the view that F is a velocity flow for a fluid, that says that the net flow of the fluid on the boundary curve C is counterclockwise. If we divide by the length of the curve, we would get average signed speed that a point on the curve is rolled around the curve by the velocity field. If the curve is a small circle of radius r then Green's theorem says that this average angular velocity is approximately one half the above difference of partial derivatives. For this reason the the difference is a measure of the rotational tendency of the velocity field at each point, and as such is called the curl of F.

Note: The curl of a 3 dimensional velocity field F = <P, Q, R> is the vector Del X F = < $(\frac{\partial}{\partial y} R) - (\frac{\partial}{\partial z} Q)$, $(\frac{\partial}{\partial z} P) - (\frac{\partial}{\partial x} R)$, $(\frac{\partial}{\partial y} Q) - (\frac{\partial}{\partial x} P)$> . If F is 2 dimensional (R = 0) we can call the 3rd component the curl of F. **Stokes theorem** is a generalization of Greens theorem to the situation in space where C is a space curve and D is a surface bounded by C. It says that the line integral of the tangential component of F around C is equal to the surface integral of the normal component of curl F over D.

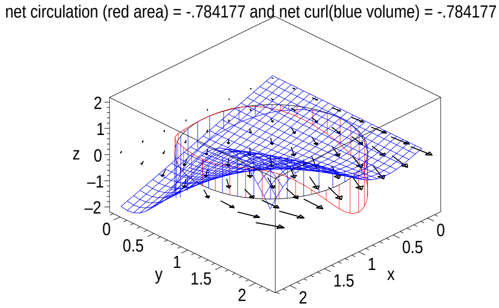
If instead of integrating the tangential component of the velocity field F around the curve C, we integrate the normal component of F, we get another line integral which can also be equated to a double integral by Green's theorem. The outward normal n to the curve C is obtained by rotating the tangent vector <dx/dt, dy/dt>/—r'(t)— 90 degrees clockwise to get n = <dy/dt, -dx/dt>/—r'(t)—

The line integral of the normal component of F = <P,Q> then becomes $\int_C P\,dy - \int_C Q\,dx$. By Greens theorem this is equal to the the double integral $\int\int (\frac{\partial}{\partial x} P) + (\frac{\partial}{\partial y} Q)\,d\ dA$, This is called the **normal form** of Greens theorem. The right hand side of this equation measures the transport of fliud across the curve C (positive means fluid is flowing out on average, negative means fluid is flowing in). The integrand of the right hand integral is called the divergence of the velocity field F. More generally, the divergence of a vector field F = <P,Q,R> is defined as div F = del dot F = $(\frac{\partial}{\partial x} P) + (\frac{\partial}{\partial y} Q) + (\frac{\partial}{\partial z} R)$. The **Divergence theorem** is a generalization of the normal form of Green's theorem to the situation in space where you have a surface R bounding a soliid S in space where there is a nice velocity field F defined on an open set containing S and R. It says that the surface integral of the (outward) normal component of F over R is equal to the triple integral of the divergence of F over the solid S.

12.1 procedure 1 to investigate velocity fields: Tangential form of Greens theorem.

This procedure takes a vector field F1 (a list of two expressions in x and y), a radius R, and a center P and draws the circle C of radius R centered at P. Then it draws the the red graph of the tangential component of F over the curve and shades vertically the red fence trapped between the graph and the circle C. This enables you to estimate the net circulation of fluid around C. Then it also draws the graph of the curl of F over a square containg D and draw the blue projection of C up onto that graph together with a blue fence. This enables you to visually estimate the net curl of F over D. Green's theorem says that these two quantities are equal numerically.

```
> restart:with(plots):
  greenpic1:=proc(R,P,F1)
  #R=radius of D, P=center of D, F1=[y,-x] velocity field
> local xrng,yrng,trng,rp,F,Fdr,QP,r,wrk,Rd,crl;
  use  plottools in
  F:=unapply([F1[1],F1[2],0],x,y):
> xrng:=(P[1]-1.1*R)..(P[1]+1.1*R):
  yrng:=(P[2]-1.1*R)..(P[2]+1.1*R):trng:=0..2*Pi:
  r:=[P[1]+R*cos(t),P[2]+R*sin(t),0]:
> rp:= [diff(r[1],t),diff(r[2],t),0]:
  Fdr := F(r[1],r[2])[1]*rp[1]+F(r[1],r[2])[2]*rp[2]:
> QP := diff(F(x,y)[2],x)-diff(F(x,y)[1],y):
  wrk:=evalf(Int(Fdr,t=trng),6):
  crl:=evalf(Int(Int(subs({x=P[1]+Rd*cos(theta),y=P[2]+Rd*sin(theta)
  },QP*Rd),Rd=0..R),theta=0..2*Pi),6);
> display(
  plot3d(QP,x=xrng,y=yrng,style=wireframe,color=blue),
  fieldplot3d(F(x,y),x=xrng,y=yrng,z=0..(.01),arrows=SLIM),
> seq(line([P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]/36),0]
  ,[P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]/36),subs(t=i*o
  p(trng)[2]/36,Fdr)],color=red),i=0..36),
> rotate(display(seq(line([P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(
  trng)[2]/36),0],[P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]
  /36),subs(t=i*op(trng)[2]/36,subs({x=r[1],y=r[2]
> },QP))],color=blue),i=0..36)),Pi/36,[[P[1],P[2],0],[P[1],P[2],1]]),
  spacecurve([r[1],r[2],Fdr],t=trng,color=red),
  spacecurve([r[1],r[2],subs({x=r[1],y=r[2]
> },QP)],t=trng,thickness=1,color=blue),
  spacecurve(r,t=trng),color=black,axes=boxed,labels=[x,y,z],title=cat("
  net circulation (red area) = ",convert(wrk,string)," and net curl(blue
> volume) = ",convert(crl,string)));
  end use;
> end:
> greenpic1(1,[1,1],[sin(x*y),y]);
```



12.2 procedure 2 to investigate velocity fields Normal form of Greens theorem

This procedure takes a vector field F1 (a list of two expressions in x and y), a radius R, and a center P and draws the circle C of radius R centered at P. Then it draws the the magenta graph of the normal component of F over the curve and shades vertically the magenta fence trapped between the graph and the circle C. This enables you to visually estimate the net outflow of fluid thru C. Then it also draws the graph of the div of F over a square containg D and draw the navy projection of C up onto that graph together with a navy fence of the solid. This enables you to visually estimate the net divergencel of F over D. Green's theorem says that these two quantities are equal numerically.

```
> with(plots):
> greenpic2 := proc(R,P,F1)
  #R=radius of D, P=center of D, F1=[y,-x] velocity field
  local xrng,yrng,trng,rp,F,Fdn,PQ,r,prpwk,Rd,dv;
> use plottools in
  F:=unapply([F1[1],F1[2],0],x,y):
  xrng:=(P[1]-1.1*R)..(P[1]+1.1*R):
> yrng:=(P[2]-1.1*R)..(P[2]+1.1*R):trng:=0..2*Pi:
  r:=[P[1]+R*cos(t),P[2]+R*sin(t),0]:
> rp:= [diff(r[1],t),diff(r[2],t),0]:
  Fdn := F(r[1],r[2])[1]*rp[2]-F(r[1],r[2])[2]*rp[1]:
  PQ := diff(F(x,y)[1],x)+diff(F(x,y)[2],y):
> prpwk:=evalf(Int(Fdn,t=trng),6):
  dv:=evalf(Int(Int(subs({x=P[1]+Rd*cos(theta),y=P[2]+Rd*sin(theta)
  },PQ*Rd),Rd=0..R),theta=0..2*Pi),6);
> display(
  plot3d(PQ,x=xrng,y=yrng,style=wireframe,color=navy),
  fieldplot3d(F(x,y),x=xrng,y=yrng,z=0..(.01),arrows=SLIM),
> seq(line([P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]/36),0]
  ,[P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]/36),subs(t=i*o
  p(trng)[2]/36,Fdn)],color=magenta),i=0..36),
> rotate(display(seq(line([P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(
  trng)[2]/36),0],[P[1]+R*cos(i*op(trng)[2]/36),P[2]+R*sin(i*op(trng)[2]
  /36),subs(t=i*op(trng)[2]/36,subs({x=r[1],y=r[2]
```

```
> },PQ))],color=maroon),i=0..36)),Pi/36,[[P[1],P[2],0],[P[1],P[2],1]]),
  spacecurve([r[1],r[2],Fdn],t=trng,color=magenta),
  spacecurve([r[1],r[2],subs({x=r[1],y=r[2]
> },PQ)],t=trng,thickness=1,color=navy),
  spacecurve(r,t=trng),color=black,axes=boxed,labels=[x,y,z],title=cat("
  net outflow (magenta area) = ",convert(prpwk,string)," and net
> divergence(navy volume) = ",convert(dv,string)));
  end use;
> end:
```

```
> greenpic2(1,[1,1],[sin(x*y),y]);
```