How Can You Visualize Green's Theorem?

Introduction

OBJECTIVE: Visualize a force field with a path superimposed on it to enhance geometrical insight into Green's Theorem and compute line integrals over vector fields using parametrizations.

If you can visualize a force field and a curve through it, how can that help you understand Green's Theorem? In this module, you will explore integration over vector fields and use parametrizations to compute line integrals. You will also explore how you can determine the closed curve around which your work integral is a maximum and whether or not it makes any difference if a force is conservative.

▪ Technology Guidelines

NOTE: If you have just finished a module, restart *Mathematica* or close the *Kernel* before executing a new module.
TO OPEN CELLS, put your cursor on the right cell bracket and double click.
TO STOP AN EXECUTION
Select the *Kernel* pull-down menu and click on *Abort Evaluation.*
ORDER OF EXECUTION
Execute cells in the order given. Do not skip any Input cells within a given notebook.
SAVING NOTEBOOKS
You can save anytime to any directory you choose, and it is wise to save often.
However, before you do your final save, it is a good idea to delete all your output by selecting the
Delete All Output selection under the *Kernel* pull-down menu.
EXPERIENCING MAJOR PROBLEMS
Save if appropriate, then shut down *Mathematica* and start it up again.

Part I: Green's Theorem (Circulation-Curl Form)

▪ Visualizing the Problem

Start by loading a package that will allow you to display the vector field. Be certain to load that package before you attempt to plot a vector field, and recall that a package should be loaded only once.

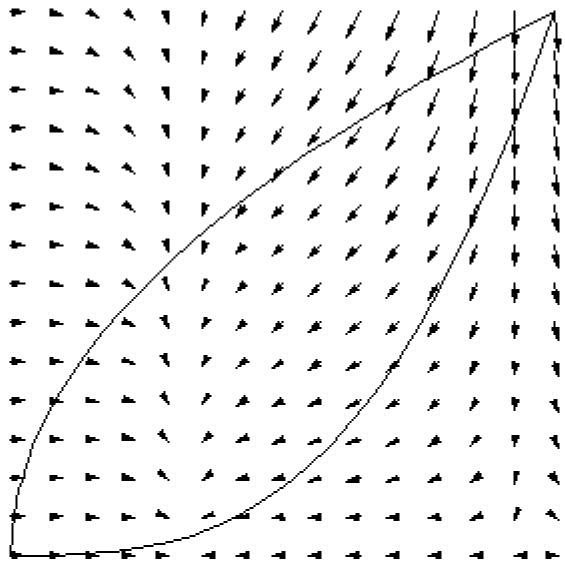
In[1]:=

```
<< Graphics`PlotField`
```

The force field is given as a vector with component *Cos(5x)* in the horizontal direction and (-3x y) in the vertical direction. We will plot two paths between (0, 0) and (1, 1) on this force field: $y = \sqrt{x}$ and $y = x^3$.

In[2]:=

```
Off[General::spell]

Off[General::spell1]

Clear[x, y, z, force]

force = {Cos[5 x], -3 x y};
pv = PlotVectorField[force, {x, 0, 1}, {y, 0, 1
  DisplayFunction -> Identity];
pc = Plot[{√x , x^3}, {x, 0, 1}, DisplayFunctio
Show[pv, pc, DisplayFunction :> $DisplayFunc
```



If you travel in a counterclockwise direction (first along $y = x^3$ and then back along $y = \sqrt{x}$) from (0,0) to (1,1) and then back again, can you predict if the work done by the force will be positive or negative or zero? Recall that the integrand of the work integral is the dot product of the force and a vector in the direction of motion.

Computing the Line Integrals

Find the work done in traveling along the lower part of the closed curve (x^3). Note the following parametrization for that curve.

In[6]:=

```
Clear[x, y, t]
x = t;
y = t^3;
r1 = {x, y};
dr1 = D[r1, t];
Print["The first portion of work done is ",
  w1 = Integrate[force.dr1, {t, 0, 1}] // N]
```

```
The first portion of work done is -1.4775
```

Does the fact that this answer is negative fit with what you might have predicted?

Next, find the work done in traveling along the upper part of the closed curve ($\sqrt{x}$). As before, you need an appropriate parametrization. Note the direction in which you were traveling. You are now at (1, 1) and need to return to (0, 0).

In[12]:=

```
Clear[x, y, t]

x = t;

y = √t ;

r2 = {x, y};

dr2 = D[r2, t];

Print["The second portion of work done is "
w2 = Integrate[force.dr2, {t, 1, 0}] // N]
```

```
The second portion of work done is 0.941785
```

Now, add the two together to get the total work.

In[18]:=

```
Print[
 "The work done in traveling around the clos
 workaroundclosedpath = w1 + w2 ]
```

```
The work done in traveling
   around the closed path is -0.535714
```

Applying Green's Theorem

Apply Green's Theorem, and verify that your answers agree. We allow for a tolerance level of 0.001 in case numerical integration has to be used at any point. You **must** clear *x* and *y* before performing this next integration.

In[19]:=

```
Clear[x, y]
```

```
inside =
 Integrate[-D[force[[1]], y] + D[force[[2]], x],
   {y, x^3, √x}] // N

If[workaroundclosedpath <= inside + .001 &&
   workaroundclosedpath >= inside - .001,
 Print["Green's Theorem is validated."],
 Print["There's a problem."]]
```

Out[20]=

```
-0.535714
```

```
Green's Theorem is validated.
```

"> About *Mathematica*

The integrand for the double integral in Green's Theorem is the curl of the force function dotted into a unit vector perpendicular to the element of area; in this case, this is in the positive *z* direction. You can explore the curl vector function in the Java applet, "Concept of the Curl."

You Try It: Part I

Demo of Selection of Closed Path to Maximize Work Done

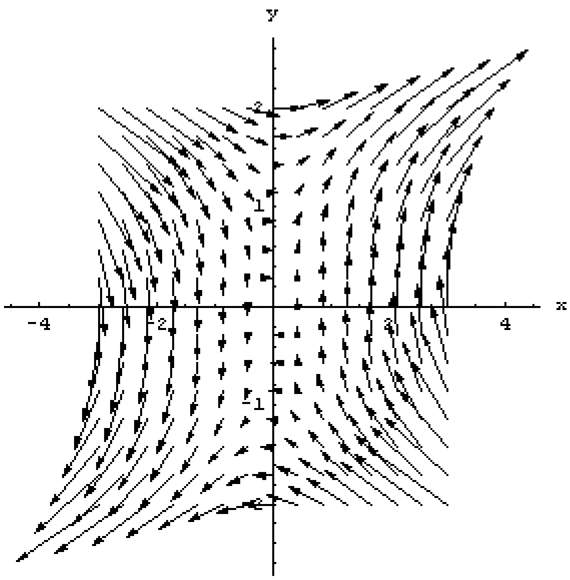
Section 16.4 Exercise 34

In this problem, you are asked to find the closed path over which the work integral is a maximum. The force function given is $\{\frac{1}{4} x^2 y + \frac{1}{3} y^3, x\}$. Begin by plotting the vector field.

In[22]:=

```
Clear[x, y, force, pv]

force = {1/4 x^2 y + 1/3 y^3, x};
```

```
pv = PlotVectorField[force, {x, -3, 3}, {y, -2
  AxesLabel → {"x", "y"}, ScaleFactor → 1.5, A:
```

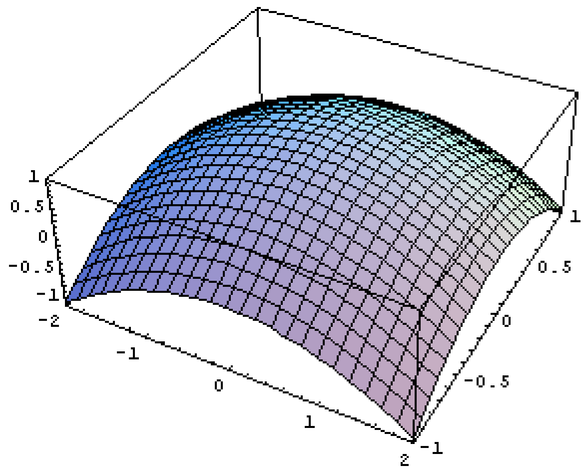


Next, examine the integrand for the double integral that is equivalent, per Green's Theorem, to the work integral.

In[25]:=

```
Print["The integrand is ",
 circintegrand = D[force[[2]], x] - D[force[[1

Plot3D[circintegrand, {x, -2, 2}, {y, -1, 1}];
```

The integrand is $1 - \frac{x^2}{4} - y^2$



Note that the circulation integrand is nonnegative only when $\frac{x^2}{4} + y^2 \leq 1$,that is,only when you are inside the ellipse represented by the equality.

Represent that ellipse parametrically by entering the correct functions in for *x* and *y* (those in red).

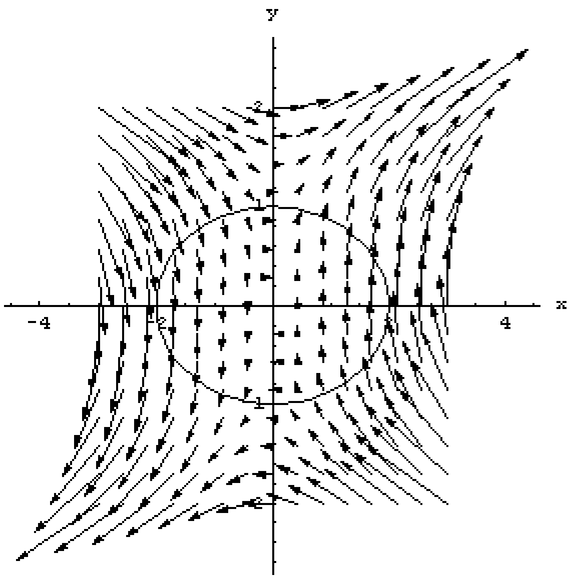
In[27]:=

```
x = 2 Cos[t];

y = Sin[t];
```

Consider a plot of your force field together with this ellipse. If your parametrization is correct and your bounds on *t* close the path, you should see your ellipse plotted with the force field by executing the next set of commands. Adjust the domain in red if necessary. If your plot looks wrong, go back and double check your parametrization.

In[29]:=

```
r = {x, y};
pp = ParametricPlot[Evaluate[r], {t, 0, 2 π},
  DisplayFunction -> Identity];

Show[pv, pp, DisplayFunction :> $DisplayFunc
```



Compute the work done in traveling around the ellipse. Adjust the domain if necessary.

In[31]:=

```
dr = D[r, t];
workaroundclosedpath = Integrate[force.dr, {t
```

Out[31]=

```
3.14159
```

Verify this using Green's Theorem.

In[32]:=

```
Clear[x, y]

inside =
 Integrate[circintegrand, {x, -2, 2},
   {y, -√(1 - x^2 / 4), √(1 - x^2 / 4)}] // N

If[workaroundclosedpath <= inside + .001 &&
  workaroundclosedpath >= inside - .001,
 Print["Green's theorem is validated"],
 Print["there's a problem"]]
```

Out[33]=

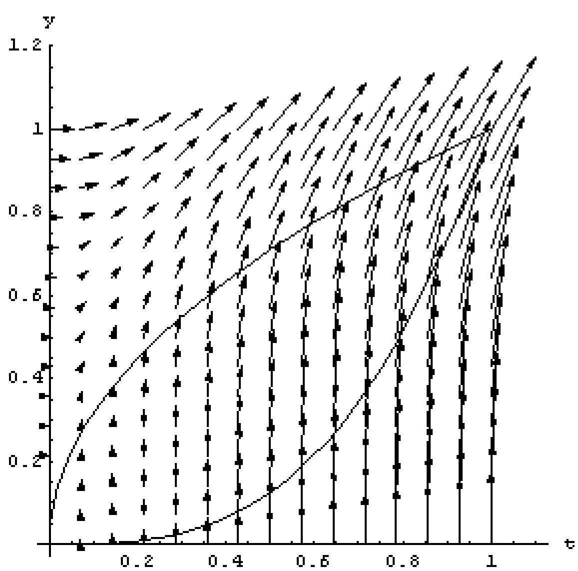
```
3.14159
```

```
Green's theorem is validated
```

Just for comparison, determine the work done by this force in traveling around the curve in the first part of this lab. Frst look at the picture.

In[35]:=

```
Clear[x, y, t]

pv2 = PlotVectorField[force, {x, 0, 1}, {y, 0,
  AxesLabel → {t, y}, ScaleFactor → .2, Aspectl
  DisplayFunction -> Identity];

Show[pv2, pc, DisplayFunction -> $DisplayFunc
```



Now compute the work done in traveling around this closed path with the current force field.

In[38]:=

```
Clear[x, y, t]
x = t;
y = t^3;
r1 = {x, y};
dr1 = D[r1, t];
w1 = Integrate[force.dr1, {t, 0, 1}] // N ;
Clear[x, y, t]
x = t;
y = √t ;
r2 = {x, y};
dr2 = D[r2, t];
w2 = Integrate[force.dr2, {t, 1, 0}] // N;
Print["The work around the closed path is "
  workaroundclosedpath = w1 + w2]
```

```
The work around the closed path is 0.286905
```

Is this answer smaller or larger than the work done in traveling around the ellipse?

▪ Choose a Conservative Force and See What Happens

What if you had chosen a conservative force? You can check this out by merely going back to the force function and replacing it with a different function. Nothing else has to be changed. Following is an example, {2x y, x^2 } (in red), of a conservative force. Try out any others you want. Remember to leave spaces between variables when defining functions.

First we will look at a plot.

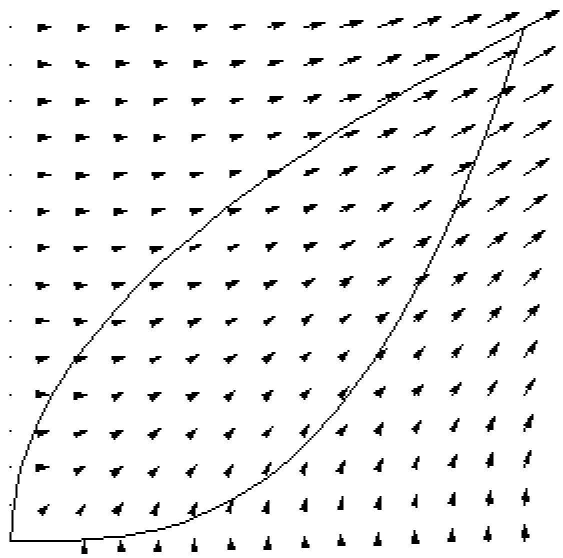
In[51]:=

```
Off[General::spell]
```

```
Off[General::spell1]
```

```
Clear[x, y, z, force]
```

```
force = {2 x y, x^2};
pv = PlotVectorField[force, {x, 0, 1}, {y, 0, 1
  DisplayFunction -> Identity];
pc = Plot[{√x , x^3}, {x, 0, 1}, DisplayFunctio
Show[pv, pc, DisplayFunction :> $DisplayFunc
```



Next, we will find the work done on each segment.

In[55]:=

```
Clear[x, y, t]

x = t;

y = t^3;

r1 = {x, y};

dr1 = D[r1, t];

Print["The first portion of work done is ",
 w1 = Integrate[force.dr1, {t, 0, 1}] // N]

Clear[x, y, t]

x = t;
```

```
y = √t ;

r2 = {x, y};

dr2 = D[r2, t];

Print["The second portion of work done is "
w2 = Integrate[force.dr2, {t, 1, 0}] // N]
```

```
The first portion of work done is 1.

The second portion of work done is -1.
```

As long as the force you use is conservative, how should the work done on the first portion be related to the work done on the second portion?

In[67]:=

```
Print[
 "The work done in traveling around the clos
 workaroundclosedpath = w1 + w2 ]
```

```
The work done in traveling
   around the closed path is 0.
```

Part II: Green's Theorem (Flux-Divergence Form)

▪ Visualizing the Problem

Start by loading a package that will allow you to display the vector field. Be certain to load that package before you attempt to plot a vector field, but if you have already loaded the package during this worksession, there is no need to read it in again.

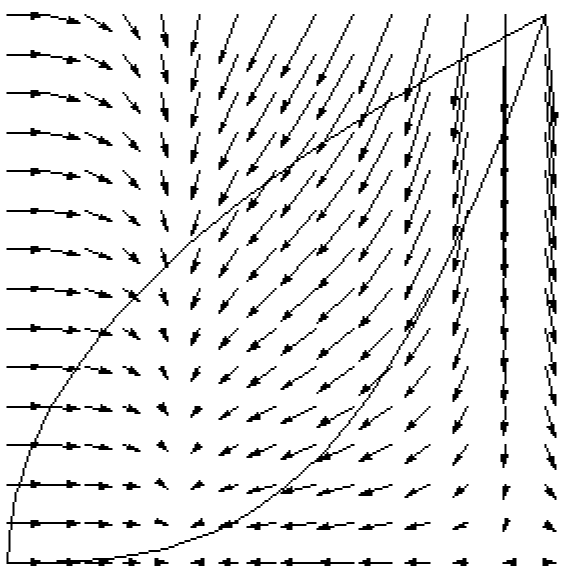
In[68]:=

```
<< Graphics`PlotField`
```

As in Part I, the force field is given as a vector with component *Cos(5x)* in the horizontal direction and (-3x y) in the vertical direction. We will plot two paths between (0, 0) and (1, 1) on this force field: $\sqrt{x}$ and x^3.

In[69]:=

```
Off[General::spell]

Off[General::spell1]

Clear[x, y, z, force]

force = {Cos[5 x], -3 x y};
pv = PlotVectorField[force, {x, 0, 1}, {y, 0, 1
  AxesLabel → {t, y}, ScaleFactor → .2, Aspectl
  DisplayFunction -> Identity];

pc = Plot[{√x , x^3}, {x, 0, 1}, Axes → True,
  DisplayFunction -> Identity];

Show[pv, pc, DisplayFunction :> $DisplayFunc
```



If you travel in a counterclockwise direction (first along y =x^3 and then back along y =$\sqrt{x}$) from (0,0) to (1,1) and then back again, can you predict whether the outward flux will be positive, negative or zero? Recall that the integrand of the flux integral is the dot product of the force and a vector **perpendicular to** the direction of motion.

Computing the Line Integrals

Find the flux in traveling along the lower part of the closed curve. Choose an appropriate parametrization.

In[75]:=

```
Clear[x, y, t]
x = t;
y = t^3;
r1 = {x, y};
dr1 = D[r1, t];
perp1 = {dr1[[2]], -dr1[[1]]};
Print["The flux over the first portion is ",
  flux1 = Integrate[force.perp1, {t, 0, 1}] //
```

```
The flux over the first portion is 0.138753
```

Does the fact that this answer is small fit with what you might have predicted?

Now find the flux along the upper part of the closed curve. As before, choose an appropriate parametrization. Note the direction in which you are traveling. Continue traveling in the same direction as above.

In[82]:=

```
Clear[x, y, t]

x = t;

y = √t ;

r2 = {x, y};

dr2 = D[r2, t];

perp2 = {dr2[[2]], -dr2[[1]]};

Print["The flux over the second portion is '
 flux2 = Integrate[force.perp2, {t, 1, 0}] // F
```

```
The flux over the second portion is -1.3841
```

Now, add the two together to get the total outward flux along the closed path.

In[89]:=

```
Print["The flux around the closed path is ",
 fluxaroundclosedpath = flux1 + flux2]
```

```
The flux around the closed path is -1.24535
```

Applying Green's Theorem

Apply Green's Theorem, and verify that your answers agree. We allow for a tolerance level of 0.001 in case numerical integration has to be used at any point. You **must** clear *x* and *y* before performing this next integration.

In[90]:=

```
Clear[x, y]

inside =
 Integrate[D[force[[1]], x] + D[force[[2]], y]
   {y, x^3, √x}] // N

If[fluxaroundclosedpath <= inside + .001 &&
   fluxaroundclosedpath >= inside - .001,
 Print["Green's theorem is validated"],
 Print["there's a problem"]]
```

Out[91]=

```
-1.24535
```

```
Green's theorem is validated
```

You Try It: Part II

Choose a Conservative Force and See What Happens.

Do conservative forces mean anything when computing flux integrals? What type of force would give the flux around a closed curve to be zero? You can check these issues out by merely going back to the force function and replacing it with a different function - paste over the red with either one of the functions suggested or another one of your own. Nothing else has to be changed.

Suggested functions:

In[93]:=

```
force = {2 x y, x^2}
```

Out[93]=

```
{2 x y, x^2}
```

In[94]:=

```
force = {x Sin[y], Cos[y]}
```

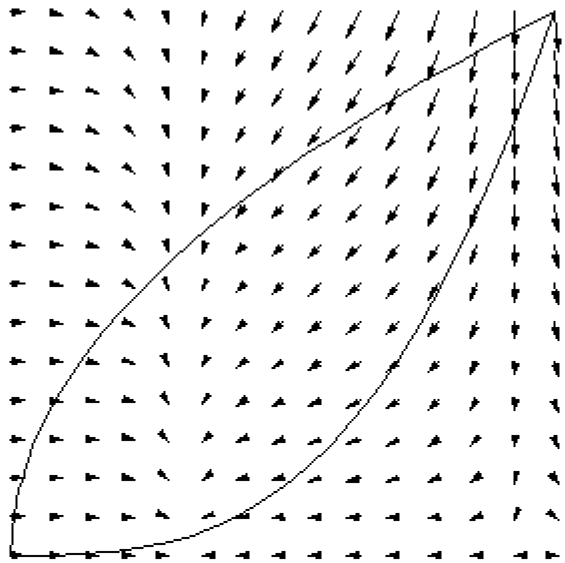
Out[94]=

```
{x Sin[y], Cos[y]}
```

This first set of commands defines and plots your functions.

In[95]:=

```
Off[General::spell]

Off[General::spell1]

Clear[x, y, z, force]

force = {Cos[5 x], -3 x y};
pv = PlotVectorField[force, {x, 0, 1}, {y, 0, 1
  DisplayFunction -> Identity];
pc = Plot[{√x, x^3}, {x, 0, 1}, DisplayFunctio
Show[pv, pc, DisplayFunction :> $DisplayFunc
```



The next set of commands finds the associated flux integrals.

In[99]:=

```
Clear[x, y, t]

x = t;

y = t^3;

r1 = {x, y};

dr1 = D[r1, t];

perp1 = {dr1[[2]], -dr1[[1]]};

Print["The flux over the first portion is ",
 flux1 = Integrate[force.perp1, {t, 0, 1}] // N

Clear[x, y, t]

x = t;

y = √t ;

r2 = {x, y};
```

```
dr2 = D[r2, t];

perp2 = {dr2[[2]], -dr2[[1]]};

Print["The flux over the second portion is '
 flux2 = Integrate[force.perp2, {t, 1, 0}] // N
```

```
The flux over the first portion is 0.138753
```

```
The flux over the second portion is -1.3841
```

Put these results together.:

In[113]:=

```
Print["The flux around the closed path is ",
 fluxaroundclosedpath = flux1 + flux2]
```

```
The flux around the closed path is -1.24535
```

Now we verify Green's Theorem for this case.

In[114]:=

```
Clear[x, y]

inside =
 Integrate[D[force[[1]], x] + D[force[[2]], y]
   {y, x^3, √x}] // N

If[fluxaroundclosedpath <= inside + .001 &&
  fluxaroundclosedpath >= inside - .001,
 Print["Green's Theorem is validated."],
 Print["There's a problem."]]
```

Out[115]=

```
-1.24535
```

```
Green's Theorem is validated.
```

Does the fact that the force field was conservative make any difference here?

About *Mathematica*

The **If** statement used here is in standard programming format. If both of the statements (&&) in the first input are true, the second input is executed; if false, the third input is executed.
Go back.