

ODEs together PDEs and Vector Fields in the SoftAge

Márcio Antonio de Faria Rosa

Campinas State University – Unicamp, Brazil

Daniela Pereira Mendes Peres

Campinas State University – Unicamp, Brazil, d113949@dac.unicamp.br

Rafael Peres

Campinas State University – Unicamp, Brazil

Abstract: At this moment when we can employ software, mathematics education has to be reviewed. In this article, we point out that ODEs should be taught from a geometric and qualitative point of view together with an introduction to PDEs and vector fields. This would increase the skills of the future mathematics user, not only to obtain explicit solutions from a straight command like *DSolve*, but also in the situations where this command does not help. The geometric interpretation and the concept of direction fields with images generated by software will give us a good understanding of possible system evolutions.

Keywords: PDE, ODE, Directions field, Vector fields, Software.

Introduction

Nowadays, when mathematics education can be applied with softwares, it needs to be reviewed. We point out in this article that ODEs should be taught from a geometric and qualitative point of view, together with an introduction to PDEs and vector fields. This approach would increase the skills of the future mathematics user. Abell and Braselton (2008, p. 49) have applied *Wolfram's Mathematica* in some ODEs problems using the command *DSolve*. See the example 2.2.2 in their book.

$$\text{In}[1] := \text{DSolve} \left[y'[x] == \frac{y[x] \text{Cos}[x]}{1 + (y[x])^2}, y[x], x \right]$$

The answer given by the software is the following:

$$\text{Out}[1] := \left\{ \left\{ y[x] \rightarrow -\sqrt{\text{ProductLog}[e^{2C_1 + 2\text{Sin}[x]}]} \right\}, \left\{ y[x] \rightarrow \sqrt{\text{ProductLog}[e^{2C_1 + 2\text{Sin}[x]}]} \right\} \right\}$$

The software tries to find an explicit solution writing y , the dependent variable, as a function of the independent one, x . The authors comment that *ProductLog*[z] returns the main value of w that satisfies the equation $z = w e^w$. Well, here the software solves a problem creating another problem.

Then the authors suggest the traditional techniques for separable equations with the help of the software.

$$\frac{1 + y^2}{y} dy = \text{Cos}[x] dx,$$

$$\text{In}[2] := \int \frac{1+y^2}{y} dy$$

$$\text{Out}[2] := \frac{y^2}{2} + \text{Log}[y]$$

$$\text{In}[3] := \int \text{Cos}[x] dx$$

$$\text{Out}[3] := \text{Sin}[x]$$

We must be careful with the answers the software provides. In this case, the modulus does not appear in the integral of $1/y$. The solution is a family of curves given by

$$\frac{y^2}{2} + \text{Log}[|y|] = \text{Sin}[x] + C$$

We point out that this implicit solution, which is much better than the explicit one above, could be obtained with the *DSolve* command itself, but applying it to the following PDE,

$$(1 + y^2) \frac{\partial u}{\partial x} + (y \text{Cos}[x]) \frac{\partial u}{\partial y} = 0,$$

In the above PDE we look for the solutions $u[x, y]$, then u is the dependent variable, x and y are the independent ones.

Such PDE could be regarded as the declaration that we look for u so that $w \bullet \nabla u = 0$, that is, the gradient of u is orthogonal to the vector field

$$w[x, y] = \{1 + y^2, y \text{Cos}[x]\}.$$

Therefore, w being orthogonal to ∇u is also tangent to level sets of u .

The quotient between $w_2 = y \text{Cos}[x]$ and $w_1 = 1 + y^2$ is the slope of the vector field w in each point.

As w is tangent to curves which are the solutions of the original ODE, working as its direction field, since

$$y'[x] = y \text{Cos}[x]/(1 + y^2) = w_2/w_1,$$

these curves agree with the level sets of any solution $u[x, y]$ of the associated PDE.

The *DSolve* command applies also to PDEs, as shown in:

$$\text{In}[4] := \text{DSolve}[(1 + y^2)\text{D}[u[x, y], x]$$

$$+ (y \text{Cos}[x])\text{D}[u[x, y], y] == 0, u[x, y], \{x, y\}]$$

$$\text{Out}[4] := \left\{ \left\{ u[x, y] \rightarrow c_1 \left[\frac{1}{2} \left(y^2 - \text{Log} \left[\frac{e^{2\text{Sin}[x]}}{y^2} \right] \right) \right] \right\} \right\}$$

We can observe that:

$$\begin{aligned} v[x, y] &= 1/2(y^2 - \text{Log}[e^{2\text{Sin}[x]}/y^2]) = \\ &= y^2/2 + \text{Log}[y] - \text{Sin}[x]. \end{aligned}$$

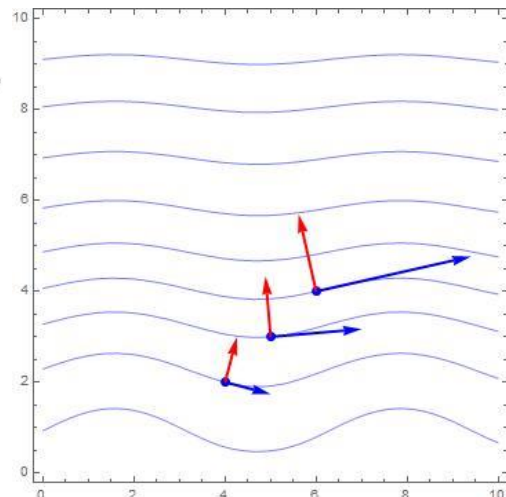
That is, including the modulus the software does not consider, we see that the PDE's solutions are written as $u[x, y] = c_j[v[x, y]]$, where $v[x, y] = (y^2/2) + \text{Log}[y] - \text{Sin}[x]$.

Any solution $u[x, y]$ has the same level sets as the presented solution $v[x, y]$, c_j is a function that changes values associated by v to those associated by u in each level set.

Well, it seems that the software has made the inverse of the process we have done above, going from the PDE to the ODE, which it has solved, finding the solutions not in the explicit, but in the implicit form $v[x, y] = C$.

```

In[5] := v[x_, y_] = y^2/2 + Log[y] - Sin[x];
In[6] := sl = ContourPlot[v[x, y], {x, 0, 10}, {y, 0, 10},
ContourShading -> False, Contours ->
Table[v[2 + k, k], {k, 1, 9}], ContourStyle -> Blue];
In[7] := w[x_, y_] = {1 + y^2, y Cos[x]};
In[8] := gradv[x_, y_] = Grad[v[x, y], {x, y}];
In[9] := vecs = Graphics[{{PointSize[Large], Blue,
Thick, Point[Table[{2 + k, k}, {k, 2, 4}]],
Arrow[Table[{{2 + k, k}, {2 + k, k} +
(0.2) w[2 + k, k]], {k, 2, 4}]}],
{PointSize[Large], Red, Thick,
Arrow[Table[{{2 + k, k}, {2 + k, k} +
(0.4) gradv[2 + k, k]], {k, 2, 4}]}]}];
In[10] := Show[sl, vecs]
    
```



We have shown in *Out[10]* some solutions as in figure 2-5, Martha and Braselton (2008, p.51), but together with the vector field w and the gradient of the solution v in three points. The *ProductLog* also appears when we solve, with the *DSolve* the ODE from exercise 5.55, Bronson and Costa (2006, p. 41).

$$\text{In[11]} := \text{DSolve}[y'[x] == (-3x^2(y[x])^2)/(2x^3y[x] + x^3(y[x])^4), y[x], x]$$

$$\begin{aligned} \text{Out[11]} := & \{ \{ y[x] \rightarrow -(-2)^{1/3} \text{ProductLog}[-(1/2)\sqrt{e^{3c_1}/x^9}]^{1/3} \}, \\ & \{ y[x] \rightarrow 2^{1/3} \text{ProductLog}[-(1/2)\sqrt{e^{3c_1}/x^9}]^{1/3} \}, \\ & \{ y[x] \rightarrow (-1)^{2/3} 2^{1/3} \text{ProductLog}[-(1/2)\sqrt{e^{3c_1}/x^9}]^{1/3} \}, \\ & \{ y[x] \rightarrow -(-2)^{1/3} \text{ProductLog}[1/2\sqrt{e^{3c_1}/x^9}]^{1/3} \}, \\ & \{ y[x] \rightarrow 2^{1/3} \text{ProductLog}[1/2\sqrt{e^{3c_1}/x^9}]^{1/3} \}, \\ & \{ y[x] \rightarrow (-1)^{2/3} 2^{1/3} \text{ProductLog}[1/2\sqrt{e^{3c_1}/x^9}]^{1/3} \} \end{aligned}$$

We apply the *DSolve* to an associated DPE obtaining the ODE's implicit solution,

$$\text{In[12]} := \text{DSolve}[(2x^3y + x^3y^4) D[u[x, y], x] + (-3x^2y^2) D[u[x, y], y] == 0, u[x, y], \{x, y\}]$$

$$\text{Out[12]} := \{ \{ u[x, y] \rightarrow c_1 \left[\frac{1}{3} \left(y^3 - 2 \text{Log} \left[\frac{-1}{x^{9/2} y^3} \right] \right) \right] \} \}$$

Now we go to exercise 16, Boyce, DiPrima and Meade (2017, p. 75)

$$\text{In[13]} := \text{DSolve}[y'[x] == -y[x]/(x - y[x]\text{Sin}[y[x]]), y[x], x]$$

$$\text{Out[13]} := \text{Solve}[x == c_1/y[x] + (\text{Sin}[y[x]] - \text{Cos}[y[x]]y[x])/y[x], y[x]]$$

It seems that the software is telling us that it has to isolate y in the equation

$$x = c_1/y + (\text{Sin}[y] - \text{Cos}[y]y)/y,$$

that is, the software is giving the implicit solution and this answer, that must appear when the software assumes it cannot isolate the dependent variable, is much better than when it is able to do it, and comes to a very complicated expression.

Sometimes *DSolve* does not give an understandable answer applied to the ODE nor to the associated PDE. An example is the autonomous system below, a model for logistic growth with a threshold from Boyce, DiPrima and Meade (2017, p. 65), for which we have chosen values and $y(t)$ gives the evolution of a population along the time in the interval $[0, 5]$.

$$\frac{dy}{dt} = -(1/2) y (1 - y) (1 - (y/3))$$

The commands will not help much, but the geometric interpretation and the concept of directions field with the help of the images generated by the software, as in many other similar problems, will give us a very good understanding of the system's possible evolutions. We will see what happens with the straight commands as *DSolve* and *Integrate*, using first *DSolve* applied to the ODE,

$$\text{In[14]} := \text{DSolve}[y'[x] == -(1/2)y[x](1 - y[x])(1 - (y[x]/3)), y[x], x]$$

$$\left\{ \left\{ y[t] \rightarrow 1 - \frac{-9 e^{2t} + 9 e^{t+6 c_1}}{9 (e^t - e^{6 c_1}) \left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}} + \frac{\left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}}{e^t - e^{6 c_1}} \right\}, \right.$$

$$\left. \left\{ y[t] \rightarrow 1 + \frac{(1 + i \sqrt{3}) (-9 e^{2t} + 9 e^{t+6 c_1})}{18 (e^t - e^{6 c_1}) \left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}} - \frac{(1 - i \sqrt{3}) \left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}}{2 (e^t - e^{6 c_1})} \right\}, \right.$$

$$\left. \left\{ y[t] \rightarrow 1 + \frac{(1 - i \sqrt{3}) (-9 e^{2t} + 9 e^{t+6 c_1})}{18 (e^t - e^{6 c_1}) \left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}} - \frac{(1 + i \sqrt{3}) \left(e^{3t} - 2 e^{2t+6 c_1} + e^{t+12 c_1} + \sqrt{-e^{5t+6 c_1} + 3 e^{4t+12 c_1} - 3 e^{3t+18 c_1} + e^{2t+24 c_1}} \right)^{1/3}}{2 (e^t - e^{6 c_1})} \right\} \right\}$$

Out[14] is not understandable. So, when this command is applied to the associated PDE, a good direction field is now $\{I, -(I/2)y(I-y)(I-(y/3))\}$, and things do not become much better.

```
In[15] := DSolve[(1)D[u[t, y], t] + (-(1/2)y(1 - y)(1 - (y/3)))D[u[t, y], y] == 0, u[t, y], {t, y}]
{{u[t, y] -> c1 [Log[-(e^(t/6) (-3 + y)^(1/6) y^(1/3)) / sqrt(-1 + y)]]}, {u[t, y] -> c1 [Log[(e^(t/6) (-3 + y)^(1/6) y^(1/3)) / sqrt(-1 + y)]]},
{u[t, y] -> c1 [Log[-((-1)^(1/3) e^(t/6) (-3 + y)^(1/6) y^(1/3)) / sqrt(-1 + y)]]}, {u[t, y] -> c1 [Log[(-1)^(1/3) e^(t/6) (-3 + y)^(1/6) y^(1/3) / sqrt(-1 + y)]]},
{u[t, y] -> c1 [Log[-((-1)^(2/3) e^(t/6) (-3 + y)^(1/6) y^(1/3)) / sqrt(-1 + y)]]}, {u[t, y] -> c1 [Log[(-1)^(2/3) e^(t/6) (-3 + y)^(1/6) y^(1/3) / sqrt(-1 + y)]]}}
```

Applying the traditional techniques to separate variables,

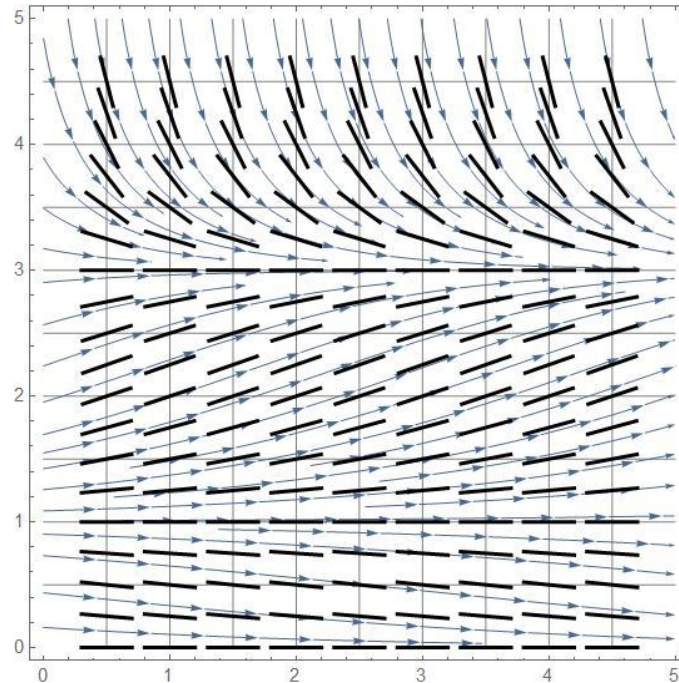
$$\boxed{-dt = dy / ((1/2)y(1 - y)(1 - (y/3)))}$$

$$\text{Int[16] := } \int \frac{1}{(1/2)y(1-y)(1-(y/3))} dy$$

$$\text{Out[16] := } -3\text{Log}[1 - y] + \text{Log}[3 - y] + 2\text{Log}[y].$$

And things get better, and will be even better using the geometric analysis of ODE employing the directions field concept. First of all, since dy/dt depends on y , the isoclines of the ODE are horizontal lines. Three of the isoclines are also solutions, $y = 0$, $y = I$ and $y = 3$. Let's draw the pictures.

```
In[17] := vec = StreamPlot[{1, -(1/2)y(1 - y)(1 - (y/3))}, {x, 0, 5}, {y, 0, 5}];
c = ContourPlot[t, {t, 0, 5}, {y, 0, 5}, ContourShading -> False];
In[18] := d = ContourPlot[y, {t, 0, 5}, {y, 0, 5}, ContourShading -> False];
In[19] := teatro = Show[c, d];
In[20] := w2[x, y] = {1, -(1/2)y(1 - y)(1 - (y/3))}
Out[20] := {1, -(1/2)(1 - y)(1 - y/3)y}
In[21] := u2[x, y] = (1/Norm[w2[x, y]])w2[x, y];
In[22] := slope[t_] = Graphics[{Thick, Black,
Line[Table[{{t, 0.25k} - 0.2 u2[t, 0.25k], {t, 0.25k} +
0.2 u2[t, 0.25k]}, {k, 0, 18}]]];
In[23] := Show[teatro, vec, Table[slope[0.5 + k0.5], k, 0, 8]]
```



Therefore, beyond the three constant solutions, for initial conditions between 0 and 1, for the long term, the population will approach $y = 0$, which is a stable solution. The solution $y = 1$ is unstable. For initial conditions between 1 and 3 or 3 and 5, the population in the long term will approach the stable solution $y = 3$, from below and above, respectively. This view is also a guide for those who intend to work with the algebraic solution given by Out[16], that we write as:

$$g[t, y] = e^t (1 - y)^{-3} (3 - y) y^2 = C.$$

Let's suppose we want to show three solutions, y_1 , y_2 and y_3 , passing through points $p_1 = \{1, 4\}$, $p_2 = \{1, 2\}$ and $p_3 = \{1, 0.5\}$, respectively. Beginning with the first.

```
In[24] := g[t_, y_] = e^t (1 - y)^-3 (3 - y) y^2 ;
```

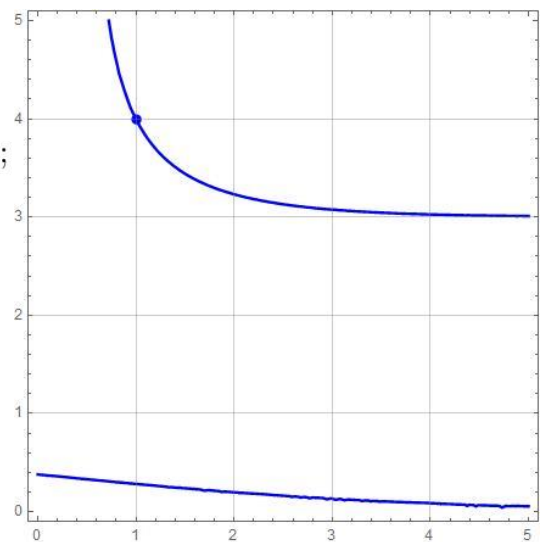
```
In[25] := Show[ContourPlot[g[t, y] == g[1, 4],
```

```
{t, 0, 5}, {y, 0, 5}, GridLines -> Automatic,
```

```
ContourStyle -> {Blue, Thick}],
```

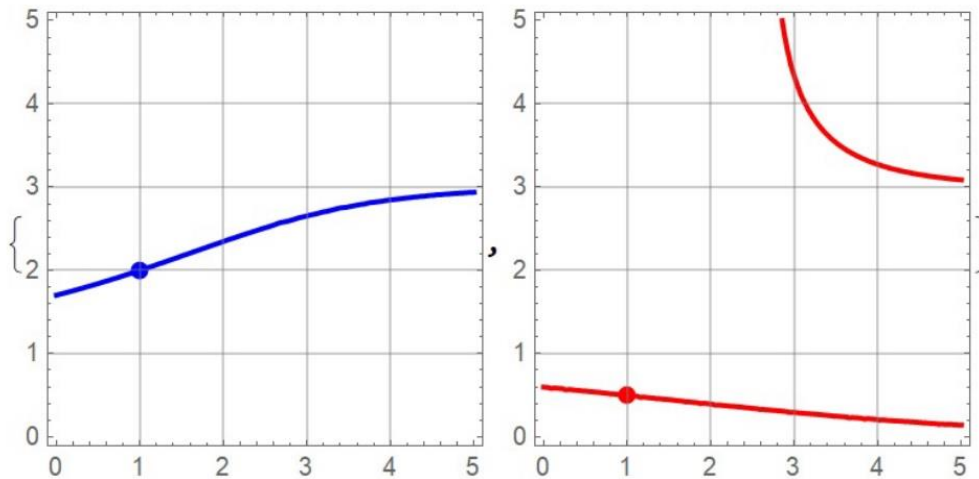
```
Graphics[{PointSize[Large], Blue,
```

```
Thick, Point[{1, 4}]}]]]
```



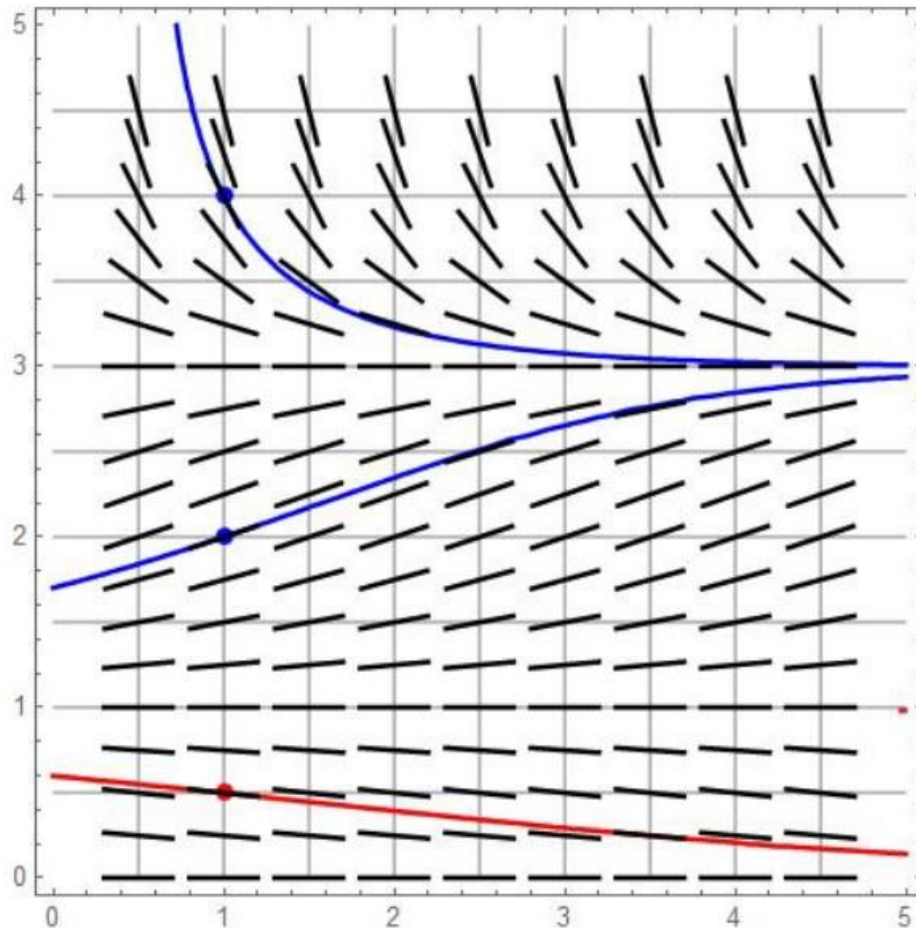
We can see that the solutions are level sets of $g[t, y]$, but, in light of the existence and the uniqueness solutions theorem for EDOs, the above level set of g corresponds to two solutions of the autonomous system, y_1 , and another solution. We will take a look to the other two solutions, y_2 and y_3 , which we intend to present,


```
In[26] := s2 = Show[ContourPlot[g[t, y] == g[1, 2], {t, 0, 5}, {y, 0, 5},
GridLines -> Automatic, ContourStyle -> {Blue, Thick}],
Graphics[{PointSize[Large], Blue, Thick, Point[{1, 2}]}]];
In[27] := Show[ContourPlot[g[t, y] == g[1, 0.5], {t, 0, 5}, {y, 0, 5},
GridLines -> Automatic, ContourStyle -> {Red, Thick}],
Graphics[{PointSize[Large], Red, Thick, Point[{1, 0.5}]}]];
In[28] := {%%, %}
```



We can see that in the interval we are working for y only one level set of g corresponds to y_2 , but the level set corresponding to y_3 also includes another solution. Therefore, we redefine the drawings to y_1 and y_3 in such a way that we have only the solutions we want.

```
In[29] := s1 = Show[ContourPlot[g[t, y] == g[1, 4], {t, 0, 5}, {y, 3, 5},
GridLines -> Automatic, ContourStyle -> {Blue, Thick}],
Graphics[{PointSize[Large], Blue, Thick, Point[{1, 4}]}]];
In[30] := s3 = Show[ContourPlot[g[t, y] == g[1, 0.5], {t, 0, 5}, {y, 0, 3},
GridLines -> Automatic, ContourStyle -> {Red, Thick}],
Graphics[{PointSize[Large], Red, Thick, Point[{1, 0.5}]}]];
In[31] := Show[teatro, s1, s2, s3, Table[slope[0.5 + k0.5], {k, 0, 8}]]
```



And we have presented the three solutions together with the directions field which includes the constant solutions that are dashed.

We suggest the teachers provide exercises in which the students can imitate, with the help of the software, classical book illustrations. The teacher can create new exercises – or enhance old ones - what is excellent work that will increase the students' understanding of the theory they are studying.

Recommendations

We recommend this type of work to engineering students, who need to learn calculus with applications and to analyze actual situations. Calculus activities when done using the software and geometric analysis can only tend to bring learning gains.

Acknowledgements

We are grateful for the fine and kind review done by Teresa Christina Waeny Pessoa de Mello Dias.

References

- Abel, L. M., & Braselton, J. P. (2008). *Differential Equations with Mathematica* (4th ed.). San Diego, California: Elsevier Academic Press.
- Bronson, R., & Costa G. B. (2006). *Differential Equations* (3th ed.). Pennsylvania, NY: McGraw-Hill Companies.
- Boyce, W. E., DiPrima, R. C., & Meade, D. B. (2017). *Elementary Differential Equations and Boundary Value Problems* (11th ed.). Danvers, MA: John Wiley & Sons.