# Two applications of macros in `PSTricks`*

Le Phuong Quan
(Cantho University, Vietnam)

April 25, 2010

## Contents



## 1. Drawing approximations to the area under a graph by rectangles

### 1.1. Description

We recall here an application in Calculus. Let $f(x)$ be a function, defined and bounded on the interval $[a, b]$. If $f$ is integrable (in Riemann sense) on $[a, b]$, then its integration on this interval is

$$\int_a^b f(x)dx = \lim_{\|P\|\to 0} \sum_{i=1}^n f(\xi_i)\Delta x_i,$$

where $P\colon a = x_0 < x_1 < \cdots < x_n = b$, $\Delta x_i = x_i - x_{i-1}$, $\xi_i \in [x_{i-1}, x_i]$, $i = 1, 2, \ldots, n$, and $\|P\| = \max\{\Delta x_i \colon i = 1, 2, \ldots, n\}$. Hence, when $\|P\|$ is small enough, we may have an approximation

$$I = \int_a^b f(x)dx \approx \sum_{i=1}^n f(\xi_i)\Delta x_i. \tag{1}$$

Because $I$ is independent to the choice of the partition $P$ and of the $\xi_i$, we may divide $[a, b]$ into $n$ subintervals with equal length and choose $\xi_i = (x_i + x_{i-1})/2$. Then, $I$ can be approximately seen as the sum of areas of the rectangles with sides $f(\xi_i)$ and $\Delta x_i$.

We will make a drawing procedure to illustrate the approximation (1). Firstly, we establish commands to draw the *sum* of rectangles, like the area under piecewise-constant functions (called

*The author of this package is Timothy Van Zandt (email address: `tvz@econ.insead.fr`).

*step shape*, for brevity). The choice here is a combination of the macros `\pscustom` (to *join* horizontal segments, automatically) and `\multido`, of course. In particular, the horizontal segments are depicted within the loop `\multido` by

`\psplot[`*settings*`]{`$x_{i-1}$`}{`$x_i$`}{`$f(\xi_i)$`}`

The `\pscustom` will join these segments altogether with the end points $(a, 0)$ and $(b, 0)$, to make the boundary of the step shape. Then, we draw the points $(\xi_i, f(\xi_i))$, $i = 1, 2, \ldots, n$, and the dotted segments between these points and the points $(\xi_i, 0)$, $i = 1, 2, \ldots, n$, by

`\psdot[algebraic,...](*{`$\xi_i$`} {`$f(x)$`})`,
`\psline[algebraic,linestyle=dotted,...](`$\xi_i$`,0)(*{`$\xi_i$`} {`$f(x)$`})`,

where we use the structure `(*{`*value*`} {`$f(x)$`})` to obtain the point $(\xi_i, f(\xi_i))$. Finally, we draw vertical segments to split the step shape into rectangular cells by

`\psline[algebraic,...](`$x_i$`,0)(*{`$x_i$`} {`$f(x - \Delta x_i/2)$`})`

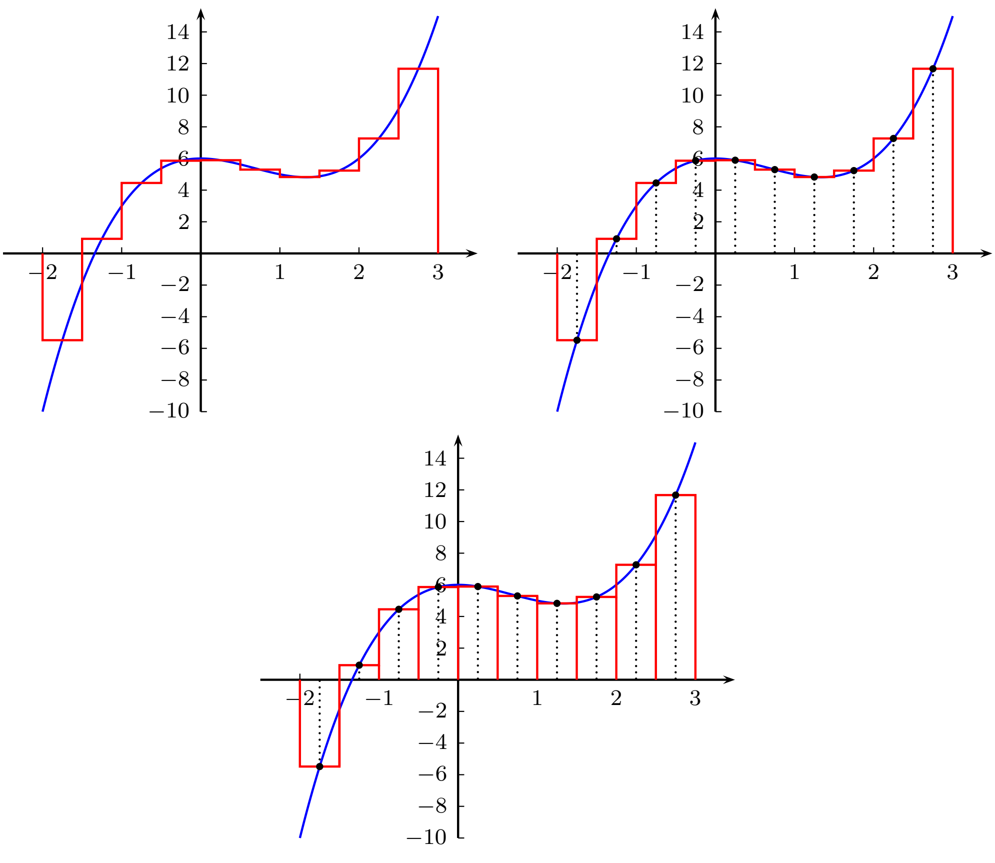
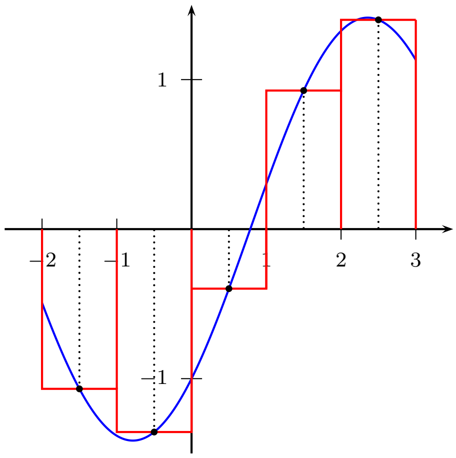


Figure 1: Steps to make the drawing procedure.

We can combine the above steps to make a procedure whose calling sequence consists of main parameters $a$, $b$, $f$ and $n$, and dependent parameters $x_{i-1}$, $x_i$, $\xi_i$, $f(\xi_i)$ and $f(x \pm \Delta x_i/2)$. For instant, let us consider the approximations to the integration of $f(x) = \sin x - \cos x$ on the interval $[-2, 3]$ in the cases of $n = 5$ and $n = 20$. Those approximations are given in Figure 2.



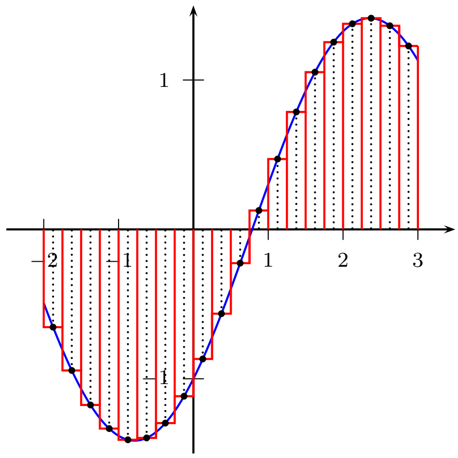




Figure 2: Approximations to the integration of $f(x) = \sin x - \cos x$ on $[-2, 3]$.

In fact, we can make a procedure, say `RiemannSum`, whose calling sequence is of the form:

$$\texttt{\textbackslash RiemannSum\{}a\texttt{\}\{}b\texttt{\}\{}f(x)\texttt{\}\{}n\texttt{\}\{}x_{\text{ini}}\texttt{\}\{}x_{\text{end}}\texttt{\}\{}x_{\text{choice}}\texttt{\}\{}f(x+\Delta x_i/2)\texttt{\}\{}f(x-\Delta x_i/2)\texttt{\}},$$

where $x_0 = a$ and for each $i = 1, 2 \ldots, n$:

$$x_i = a + \frac{b-a}{n} i, \quad \Delta x_i = x_i - x_{i-1} = \frac{b-a}{n},$$
$$x_{\text{ini}} = x_0 + \Delta x_i, \quad x_{\text{end}} = x_1 + \Delta x_i, \quad x_{\text{choice}} = \frac{x_{\text{ini}} + x_{\text{end}}}{2} = \frac{x_0 + x_1}{2} + \Delta x_i.$$

Note that $x_{\text{ini}}$, $x_{\text{end}}$ and $x_{\text{choice}}$ are given in such forms to be suitable to variable declaration in `\multido`. They are nothing but $x_{i-1}$, $x_i$ and $\xi_i$, respectively, at the step $i$-th in the loop.

Tentatively, in `PSTricks` language, the definition of `RiemannSum` is suggested to be

```
\def\RiemannSum#1#2#3#4#5#6#7#8#9{%
\psplot[linecolor=blue]{#1}{#2}{#3}
\pscustom[linecolor=red]{%
\psline{-}(#1,0)(#1,0)
\multido{\ni=#5,\ne=#6}{#4}
{\psline(*{\ni} {#8})(*{\ne} {#9})}}
\multido{\ne=#6,\nc=#7}{#4}
{\psdot(*{\nc} {#3})
\psline[linestyle=dotted,dotsep=1.5pt](\nc,0)(*{\nc} {#3})
\psline[linecolor=red](\ne,0)(*{\ne} {#9})}}
```

### 1.2. Examples

We just give here two more examples for using the drawing procedure with ease. In the first example, we approximate the area under the graph of the function $f(x) = x - (x/2)\cos x + 2$ on the interval $[0, 8]$. To draw the approximation, we try the case $n = 16$; thus $x_0 = 0$ and for each $i = 1, \ldots, 16$, we have $x_i = 0.5\, i$, $\Delta x_i = 0.5$, $x_{\text{ini}} = 0.00 + 0.50$, $x_{\text{end}} = 0.50 + 0.50$ and $x_{\text{choice}} = 0.25 + 0.50$.

To get Figure 3, we have used the following LaTeX code:

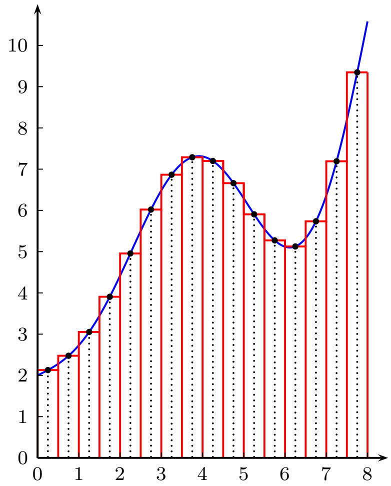



Figure 3: An approximation to the area under the graph of $f(x) = x - (x/2)\cos x + 2$ on $[0, 8]$.

```
\begin{pspicture}(0,0)(4.125,5.5)
\psset{plotpoints=500,algebraic,dotsize=2.5pt,unit=0.5}
\RiemannSum{0}{8}{x-(x/2)*cos(x)+2}{16}{0.00+0.50}{0.50+0.50}{0.25+0.50}
{x+0.25-((x+0.25)/2)*cos(x+0.25)+2}{x-0.25-((x-0.25)/2)*cos(x-0.25)+2}
\psaxes[ticksize=2.2pt,labelsep=4pt]{->}(0,0)(8.5,11)
\end{pspicture}
```

In the second example below, we will draw an approximation to the integration of $f(x) = x \sin x$ on $[1, 9]$. Choosing $n = 10$ and computing parameters needed, we get Figure 4, mainly by the command

$$\texttt{\textbackslash RiemannSum}\{1\}\{9\}\{x \sin x\}\{10\}\{1.00 + 0.80\}\{1.80 + 0.80\}\{1.40 + 0.80\}$$
$$\{(x + 0.4)\sin(x + 0.4)\}\{(x - 0.4)\sin(x - 0.4)\}$$

in the drawing procedure.

## 2. Drawing the vector field of an ordinary differential equation of order one

### 2.1. Description

Let us consider the differential equation

$$\frac{dy}{dx} = f(x, y). \tag{2}$$

At each point $(x_0, y_0)$ in the domain $D$ of $f$, we will put a vector $\mathbf{v}$ with slope $k = f(x_0, y_0)$. If $y(x_0) = y_0$, then $k$ is the slope of the tangent to the solution curve $y = y(x)$ of (2) at $(x_0, y_0)$. The $\mathbf{v}$'s make a *vector field* and the picture of this field would give us information about the shape of solution curves of (2), even we have not found yet any solution of (2).

The vector field of (2) will be depicted on a finite grid of points in $D$. This grid is made of lines, paralell to the axes $Ox$ and $Oy$. The intersectional points of those lines are called *grid points* and

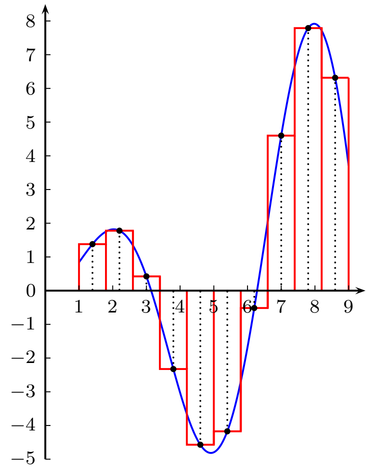



Figure 4: An approximation to the integration of $f(x) = x \sin x$ on $[1, 9]$.

often indexed by $(x_i, y_j)$, $i = 1, \ldots, m$, $j = 1, \ldots, n$. For convenience, we will use polar coordinate to locate the terminal point $(x, y)$ of a field vector, with the initial point at the grid point $(x_i, y_j)$. Then, we can write

$$\begin{aligned} x &= x_i + r\cos\varphi, \\ y &= y_j + r\sin\varphi. \end{aligned}$$

Because $k = f(x_i, y_j) = \tan\varphi$ is finite, we may take $-\pi/2 < \varphi < \pi/2$. From $\sin^2\varphi + \cos^2\varphi = 1$ and $\sin\varphi = k\cos\varphi$, we derive

$$\cos\varphi = \frac{1}{\sqrt{1+k^2}}, \quad \sin\varphi = \frac{k}{\sqrt{1+k^2}}.$$

The field vectors should all have the same magnitude and we choose here that length to be 1/2, that means $r = 1/2$. Thus, vectors on the grid have their initial points and terminal ones as

$$(x_i, y_j), \quad \Big(x_i + \frac{1}{2}\cos\varphi, y_j + \frac{1}{2}\sin\varphi\Big).$$

Of macros in `PSTricks` to draw lines, we select `\parametricplot`[1] for its fitness. We immetiately have the simple parameterization of the vector at the grid point $(x_i, y_j)$ as

$$\begin{aligned} x &= x_i + \frac{t}{2}\cos\varphi = x_i + \frac{t}{2\sqrt{1+k^2}}, \\ y &= y_j + \frac{t}{2}\sin\varphi = y_j + \frac{tk}{2\sqrt{1+k^2}}, \end{aligned}$$

where $t$ goes from $t = 0$ to $t = 1$, along the direction of the vector. The macro `\parametricplot` has the syntax as

`\parametricplot[`*settings*`]{`$t_{\min}$`}{`$t_{\max}$`}{`$x(t)$`|`$y(t)$`}`,

where we should use the option `algebraic` to make the declaration of $x(t)$ and $y(t)$ simpler with `ASCII` code.

[1]This macro is of ones, often added and updated in the package `pstricks-add`, the authors: Dominique Rodriguez (`dominique.rodriguez@waika9.com`), Herbert Voß (`voss@pstricks.de`).

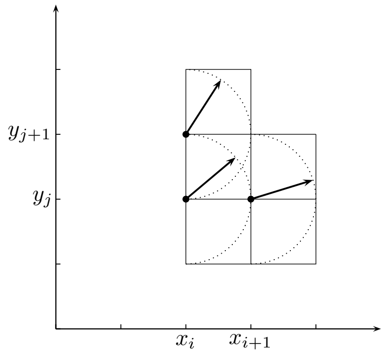



Figure 5: Field vectors on a grid.

From the above description of one field vector, we go to the one of the whole vector field on the grid in the domain $R = \{(x, y)\colon a \le x \le b,\ c \le y \le d\}$. To determine the grid belonging to the interior of $R$, we confine grid points to the range

$$a + 0.25 \le x_i \le b - 0.25, \quad c + 0.25 \le y_j \le d - 0.25. \tag{3}$$

With respect to the indices $i$ and $j$, we choose initial values as $x_1 = a + 0.25$ and $y_1 = c + 0.25$, with increments $\Delta x = \Delta y = 0.5$, as corresponding to the length of vectors and the distance between grid points as indicated in Figure 5. Thus, to draw vectors at grid points $(x_i, y_j)$, we need two loops for $i$ and $j$, with $0 \le i \le [2m]$, $0 \le j \le [2n]$, where $m = b - a$, $n = d - c$. Apparently, these two loops are nested `\multido`'s, with variable declaration for each loop as follows

$$\begin{aligned}\texttt{\textbackslash nx} &= \text{initial value} + \text{increment} = x_1 + \Delta x,\\ \texttt{\textbackslash ny} &= \text{initial value} + \text{increment} = y_1 + \Delta y.\end{aligned}$$

Finally, we will replace `\nx`, `\ny` by $x_i$, $y_j$ in the below calling sequence for simplicity.

Thus, the main procedure to draw the vector field of the equation (2) on the grid (3) is

$$\texttt{\textbackslash multido}\{y_j = y_1 + \Delta y\}\{[2n]\}\Big\{\texttt{\textbackslash multido}\{x_i = x_1 + \Delta x\}\{[2m]\}$$
$$\Big\{\texttt{\textbackslash parametricplot[}\textit{settings}\texttt{]\{0\}\{1\}}\Big\{x_i + \frac{t}{2\sqrt{1 + [f(x_i, y_j)]^2}}\Big|\ y_j + \frac{tf(x_i, y_j)}{2\sqrt{1 + [f(x_i, y_j)]^2}}\Big\}\Big\}$$

where we at least use `arrows=->` and `algebraic` for *settings*.

We can combine the steps mentioned above to define a drawing procedure, say `\vecfld`, that consists of main parameters in the order as `\nx`=$x_1 + \Delta x$, `\ny`=$y_1 + \Delta y$, $[2m]$, $[2n]$, $r$ and $f($`\nx`, `\ny`$)$. We may change these values to modify the vector field or to avoid the vector intersection. However, we often take $\Delta x = \Delta y = r$. Such a definition is suggested to be

```
\def\vecfld#1#2#3#4#5#6{%
\multido{#2}{#4}{\multido{#1}{#3}
{\parametricplot[algebraic,arrows=->,linecolor=red]{0}{1}
{\nx+((#5)*t)*(1/sqrt(1+(#6)^2))|\ny+((#5)*t)*(1/sqrt(1+(#6)^2))*(#6)}}}}
```

### 2.2. Examples

Firstly, we consider the equation that describes an object falling in a resistive medium:

$$\frac{dv}{dt} = 9.8 - \frac{v}{5}, \tag{4}$$

where $v = v(t)$ is the speed of the object in time $t$. In Figure 6, the vector field of (4) is given on the grid $R = \{(t, y)\colon 0 \le t \le 9,\ 46 \le v \le 52\}$, together with the graph of the equilibrium solution $v = 49$.

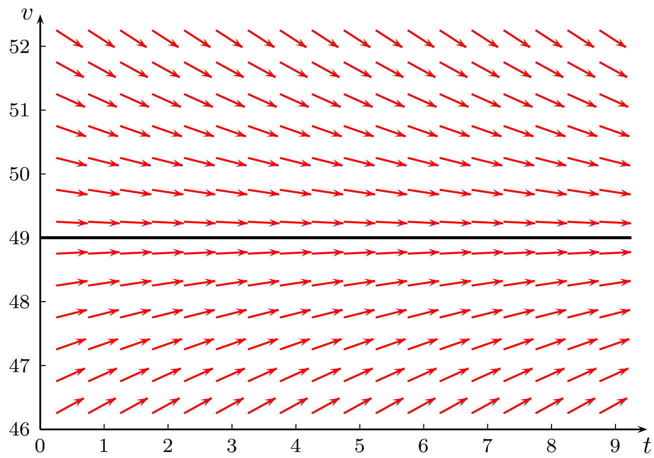



Figure 6: The vector field of (4).

Figure 6 is made of the following LaTeX code:

```
\begin{pspicture}(0,46)(9.5,52.5)
\vecfld{\nx=0.25+0.50}{\ny=46.25+0.50}{18}{12}{0.5}{9.8-0.2*\ny}
\psplot[algebraic,linewidth=1.2pt]{0}{9}{49}
\psaxes[Dy=1,Dx=1,Oy=46]{->}(0,46)(0,46)(9.5,52.5)
\rput(9.5,45.8){$t$}\rput(-0.2,52.5){$y$}
\end{pspicture}
```

Let us next consider the problem

$$\frac{dy}{dx} = x + y, \quad y(0) = 0. \tag{5}$$

It is easy to check that $y = e^x - x - 1$ is the unique solution to the problem (5). We now draw the vector field of (5) and the solution curve[2] on the grid $R = \{(x, y)\colon 0 \le x \le 3,\ 0 \le y \le 5\}$ in Figure 7.

We then go to the logistic equation, which is chosen to be a model for the dependence of the population size $P$ on time $t$ in Biology:

$$\frac{dP}{dt} = kP\Big(1 - \frac{P}{M}\Big), \tag{6}$$

[2]We have used $\mathrm{ch}(1) + \mathrm{sh}(1)$ for the declaration of $e$, natural base of logarithmic function.

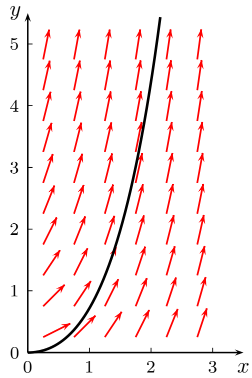



Figure 7: The vector field of (5).

where $k$ and $M$ are constants, respectively various to selected species and environment. For specification, we take, for instant, $k = 0.5$ and $M = 100$. The right hand side of (6) then becomes $f(t, P) = 0.5\,P(1 - 0.01\,P)$. In Figure 8, we draw the vector field of (6) on the grid $R = \{(t, P)\colon 0 \le t \le 10,\ 95 \le P \le 100\}$ and the equilibrium solution curve $P = 100$. Furthermore, with the initial condition $P(0) = 95$, the equation (6) has the unique solution $P = 1900(e^{-0.5t}+19)^{-1}$. This solution curve is also given in Figure 8.

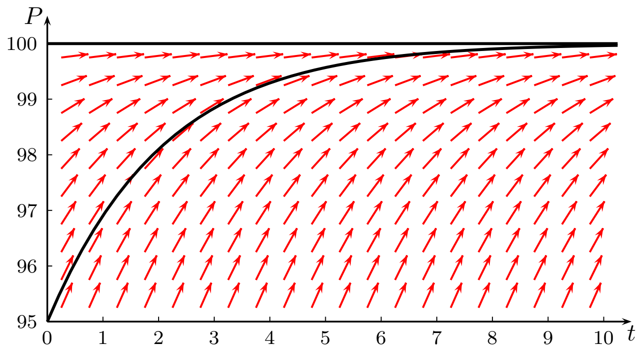



Figure 8: The vector field of (6) with $k = 0.5$ and $M = 100$.

The previous differential equations are all of seperated variable or linear cases that can be solved for closed-form solutions by some simple integration formulas. We will consider one more equation of the non-linear case whose solution can only be approximated by numerical methods. The vector field of such an equation is so useful and we will use the Runge-Kutta curves (of order 4) to add more information about the behaviour of solution curve. Here, those Runge-Kutta curves are depicted by the procedure `\psplotDiffEqn`, also updated from the package `pstricks-add`.

The vector field of the non-linear differential equation

$$\frac{dy}{dx} = y^2 - xy + 1 \tag{7}$$

will be depicted on the grid $R = \{(x, y)\colon -3 \le x \le 3,\ -3 \le y \le 3\}$ and the solutions of Cauchy problems for (7), corresponding to initial conditions

(i) $y(-3) = -1$,

(ii) $y(-2) = -3$,

(iii) $y(-3) = -0.4$,

will be approximated by the method of Runge-Kutta, with the grid size $h = 0.2$. It is very easy to recognize approximation curves, respective to (i), (ii) and (iii) in Figure 9 below.

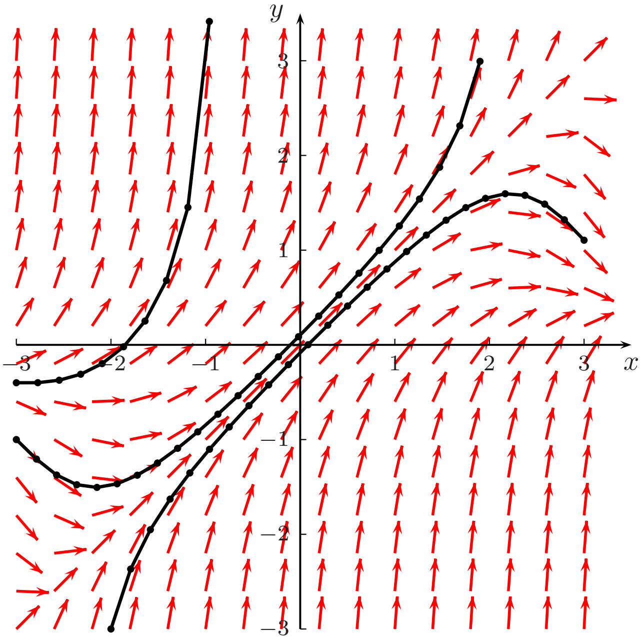



Figure 9: The vector field of (7) and the Runge-Kutta curves.

## Acknowledgment

I am very grateful to

- Timothy Van Zandt, Herbert Voß and Dominique Rodriguez for helping me with their great works on `PSTricks`.
- Hàn Thế Thành for helping me with his pdfLaTeX program.

## References

[1] Dominique Rodriguez & Herbert Voß. *PSTricks-add, additional macros for PSTricks*. Version 3.05, http://ctan.org/tex-archive/graphics/pstricks/contrib, 2008

[2] Helmut Kopka & Patrick W. Daly. *Guide to LaTeX*. Addison-Wesley, Fourth Edition, 2004, ISBN 0321173856

[3] Timothy Van Zandt. *User's Guide*. Version 1.5, http://ctan.org/tex-archive/graphics/pstricks/base, 2007