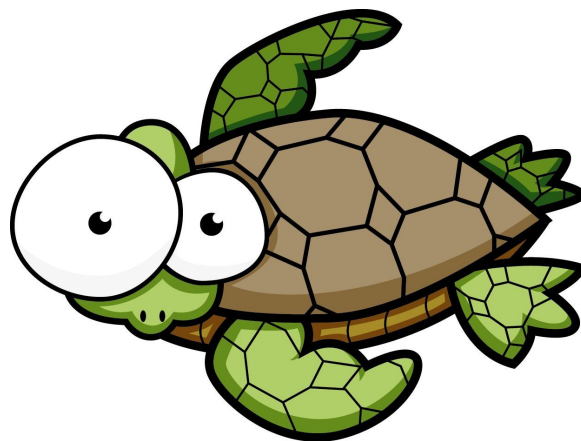


A Guide to the TurboTurtle Package for R

A. Cena, M. Kosinski, B. Zogala-Siudem



1 The TurboTurtle Package Introduction

The TurboTurtle package offers to R-users functionality of the famous "turtle graphics" from Logo educational programming language. The main idea standing behind it is to keep kids away from drugs and encourage children to learn programming and show that working with computer can be fun and creative.

The TurboTurtle package allows to create sophisticated graphics on basis of lines. The main idea is that the Turtle, described by its location and orientation, moves with commands that are relative to its own position. The line he left behind can also be controlled, by enabling it or setting its color and type.

The TurboTurtle package offers functions to move forward or backward a given distance and to turn the Turtle in a choosen direction. The graphical parameter of the plot, like for example color, type or visibility of the line, can also be easily changed.

2 The TurboTurtle moving

2.1 Let the party started!

```
require("TurboTurtle")  
  
## Loading required package: TurboTurtle  
## Loading required package: grid  
## Loading required package: png
```

```
##
## Attaching package: 'TurboTurtle'
##
## Następujący obiekt został zakryty z 'package:methods':
##
## show
```

The program is started with `turtle_init()` function, which creates a plot region and places the Turtle in its central point.

```
turtle_init()
```

Now, we can move the Turtle forward, for example for a distance of 3 units, with the use of the `move_forward()` function.

```
move_forward(dist = 3)
```

If you would like to change directions – you may use `left()` or `right()` functions, which change the Turtle direction by a given angle. For example, we may turn 45 degrees in the right direction. The results are presented in Figure 1

```
right(angle = 45)
move_forward(dist = 5)
```

2.2 Visibility settings

To enable the path from being drawn we can simply use `up()` function. Let us consider a simple example. We will turn the Turtle in right direction about 90 degrees and then `up()` function. Now when we move forward the path is not visible. If we want to draw the path we should call `down()` function.

```
right(90)
up()
move_forward(dist = 3)
right(45)
down()
move_forward(dist = 4)
```

Similarly, we may show or hide the Turtle image, using `show()` and `hide()` functions respectively.

```
hide()
right(60)
move_forward(4)
show()
```

The results are depicted in Figure 2.

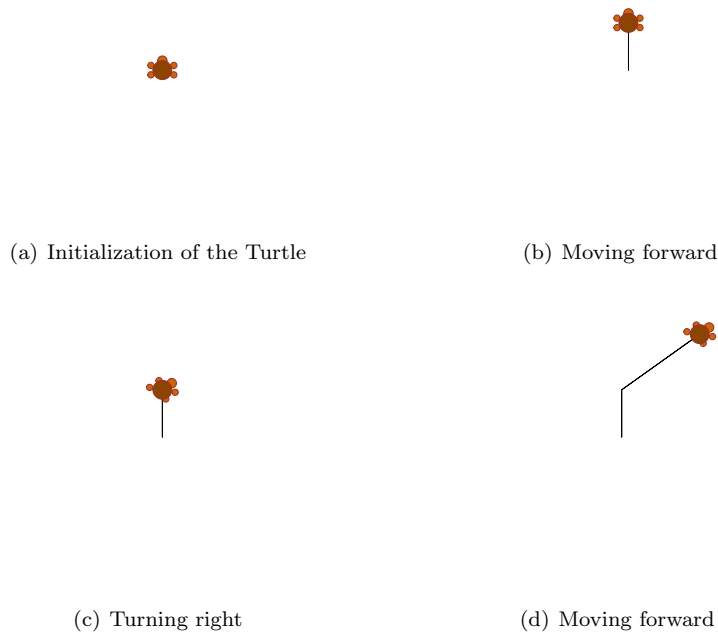


Figure 1: TurboTurtle

2.3 Graphical parameters

To set graphical parameters please use the `set_param()` function, which allows to change the color of the path, its types and thickness e.g.

```
set_param(col = "red", lwd = 2, lty = 3)
move_forward(5)
```

Results are presented in Figure 3

You may also change graphical parameters more easily by functions `set_col()`, `set_lty()`, `set_lwd()`. Each of them takes exactly one argument to be passed to `set_param()`.

3 Fast Turtle plotting a.k.a. TurboTurtle

To be able to plot bigger plots as in figure 4 and bigger, you should hide your Turtle before plotting and show just after the figure is plotted. It will take much less time to do it. For example you can try plotting drunken turtle:

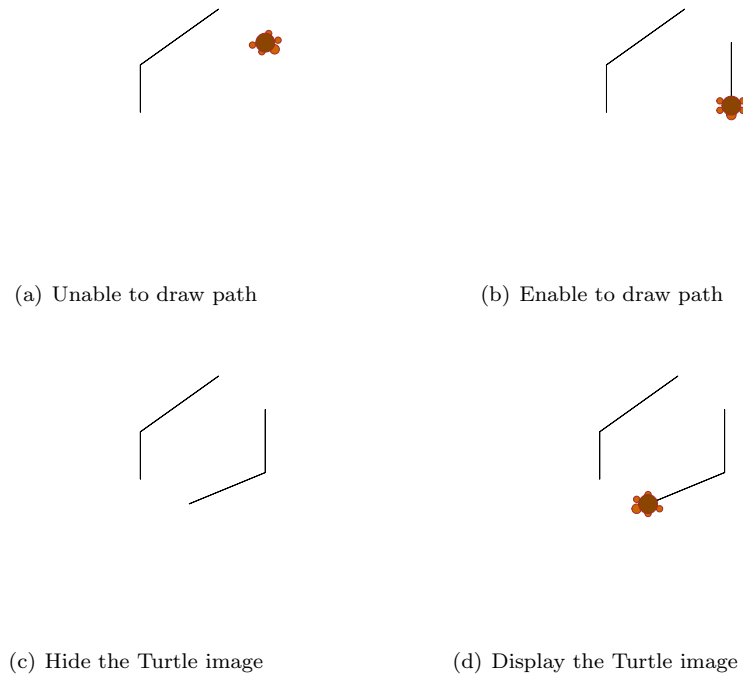


Figure 2: TurboTurtle

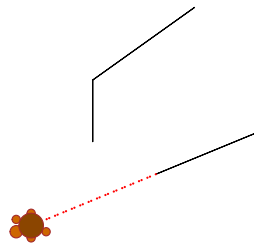


Figure 3: Changing graphical parameters

```
turtle_init()
hide()
for (i in 1:1000) {
```

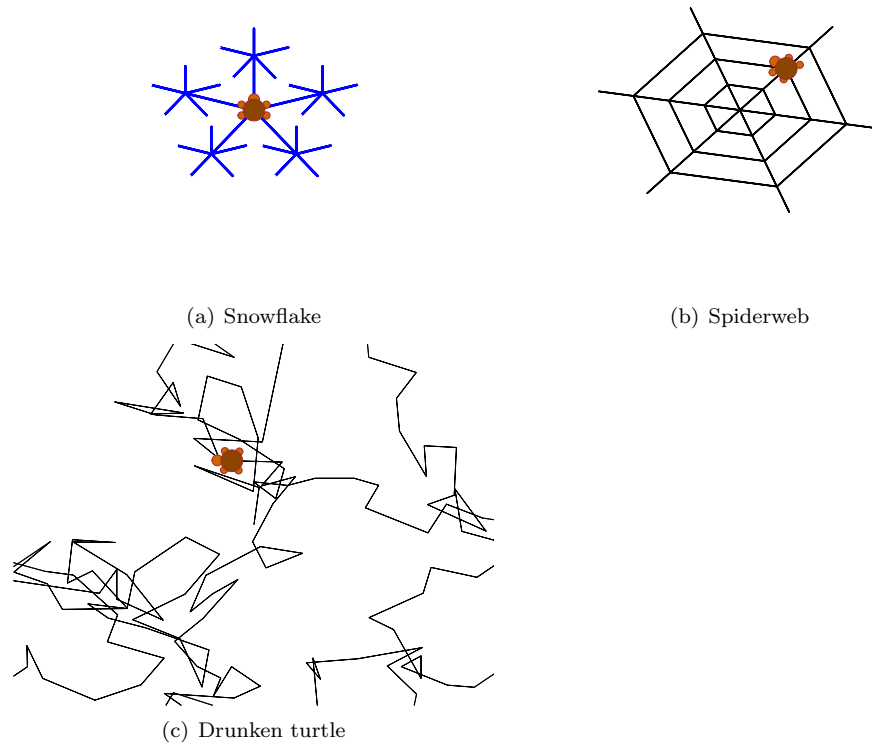


Figure 4: What turtle does in the free time

```

left(sample(1:360, 1))
move_forward(runif(1, 1, 3))
}
show()

```

4 Examples

With TurboTurtle you can draw everything and more! Examples of drawings are in the figure 4