ROS - Practical Session 2

ROS Workshop at the School of Engineering, UNAM

Viktor Seib

vseib@uni-koblenz.de

Institute for Computational Visualistics
University of Koblenz-Landau

August 11th, 2015

COMPUTERVISUALISTIK

UNIVERSITÄT KOBLENZ · LANDAU

AGAS

Task 1: Triple Turtle Mimic

Task 2: Publish Odd Numbers

Task 3: Two Floats Math

Task 4: Turtle Random Swimming

Task 5: {Robot|Turtle} Drive Square ++

Task 1: Triple Turtle Mimic

Task 2: Publish Odd Numbers

Task 3: Two Floats Math

Task 4: Turtle Random Swimming

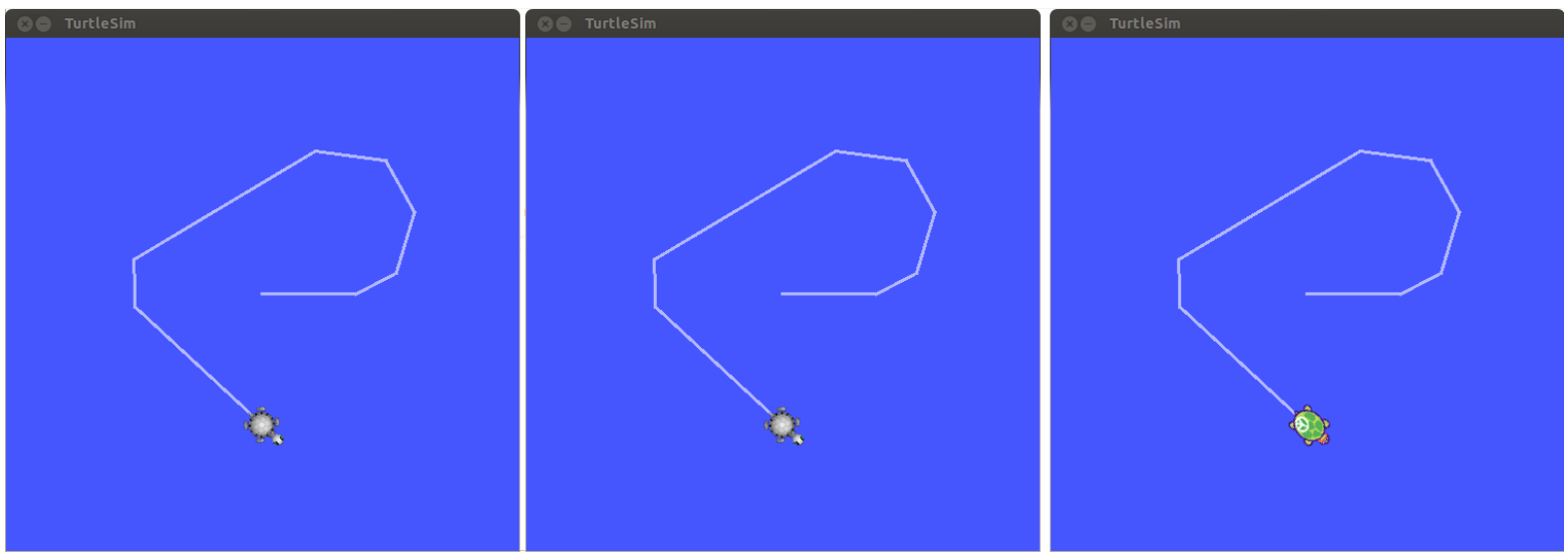
Task 5: {Robot|Turtle} Drive Square ++

Triple Turtle Mimic

Task:

- extend the `turtlemimic.launch` file from the `beginner_tutorials` package and save the result as `triple_turtlemimic.lauch`
- when starting the new launchfile 3 TurtleSim windows open
- when pressing the arrow keys all 3 turtles move identically

Goal:



Triple Turtle Mimic

Hints:

- you need to add a third `turtlesim_node` and a second `mimic` node to the launchfile
- you also need to add the `turtle_teleop_key` node
- remember to remap the topic for velocity commands in the `turtle_teleop_key` node

Publish Odd Numbers

Task:

- Modify the `talker` node in the package `beginner_tutorials`
  - additionally publish the message `Num` (`beginner_tutorials::Num`, created earlier) on topic `oddNums`
  - the message `Num` should be send whenever the variable `count` is odd
  - `Num` should contain the value of `count`

- Modify the `listener` node in the package `beginner_tutorials`
  - additionally subscribe to topic `oddNums`
  - create a callback function `oddNumsCallback` to print the content of the received message

Two Floats Math

Hint:

- for this task it is helpful to copy and modify `add_two_ints_client.cpp` and `add_two_ints_server.cpp` in the `beginner_tutorials` package

Task:

- Create a new `srv` called `two_floats_math` in the `beginner_tutorials` package:
  - the `srv`'s request consists of two float32 and one string
  - the two floats are the operands, the string the operator of a mathematical term
  - the `srv`'s response is one float32

Two Floats Math

Task:

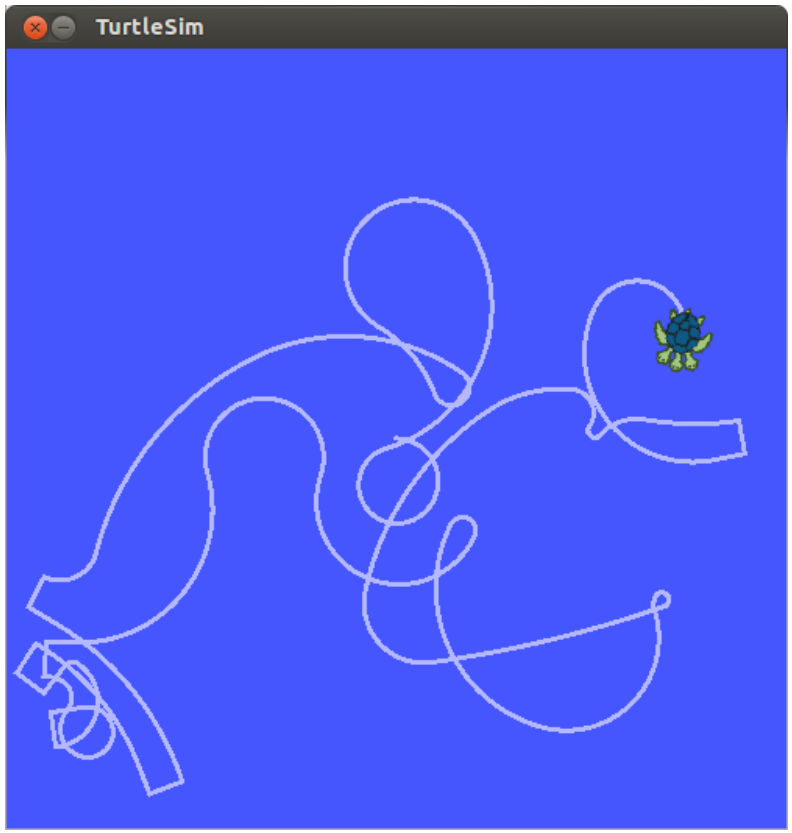
- Create a node `two_floats_math_server` in the `beginner_tutorials` package:
  - this node advertises the service `two_floats_math`
  - the returned value should be the result of *first float* *operator* *second float*
  - supported operators should be: +, -, *, /
- Create a node `two_floats_math_client` in the `beginner_tutorials` package:
  - this node takes 3 command line arguments and calls the service `two_floats_math`
  - the result should be printed on the screen

Turtle Random Swimming

Task:

- create a new package and node `turtle_random_swimming` that makes the turtle swim randomly
- create a launch file that starts the nodes `turtlesim_node`, as well as your new node `turtle_random_swimming`

Goal:



Turtle Random Swimming

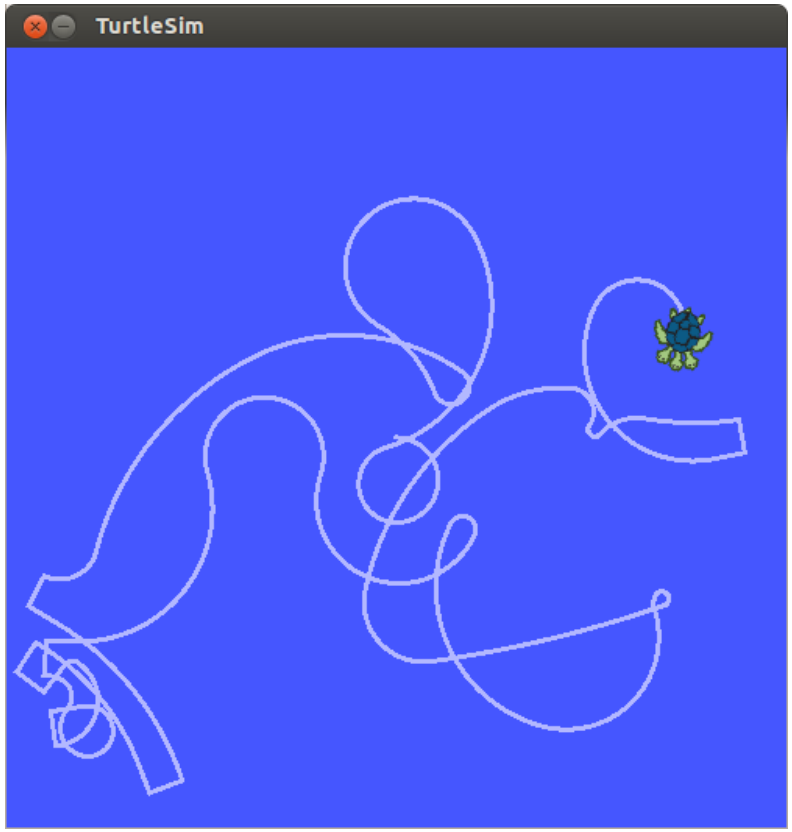
Task:

- the random motion should be smooth, the turtle should move forwards, left and right, but not backwards
- if the turtle comes close to a wall a special strategy should be implemented to avoid a collision (backwards motions is allowed)

Hint:

- you can use the provided file `turtle_random_swimming.cpp` and fill in all lines marked as “TODO”

Goal:



Robot Drive Square

Task:

- create a package `robot_drive_square` with the dependencies `roscpp`, `nav_msgs`, `geometry_msgs` and `turtlesim`
- write a node `robot_drive_square` that receives odometry data from the robot or the position data from the turtlesim
- use the odometry data to make the robot drive a square with a side length of 1 m ...
- ... or use the position data to make the turtle drive a square with a side length of 1 m
- create a launchfile to start all necessary nodes

Hint:

- you can use the provided file `robot_drive_square.cpp` and fill in all lines marked as “TODO”