Fundamentos de la robótica autónoma

Simulación

Facultad de Ingeniería
Instituto de Computación

Contenido

- Introducción
- Gazebo
- Gazebo + ROS
- Ejemplo: Turtlebot3

Introducción (1/2)

- ¿Que es un simulador?
- Un simulador pretende reproducir tanto los aspectos físicos (velocidad, aceleración, percepción del entorno) como el comportamiento de lo que se pretende simular

Introducción (2/2)

- Ejemplos:
  - Simuladores de vuelo
  - Simuladores de conducción
  - Simuladores de redes
  - Simuladores médicos
  - etc

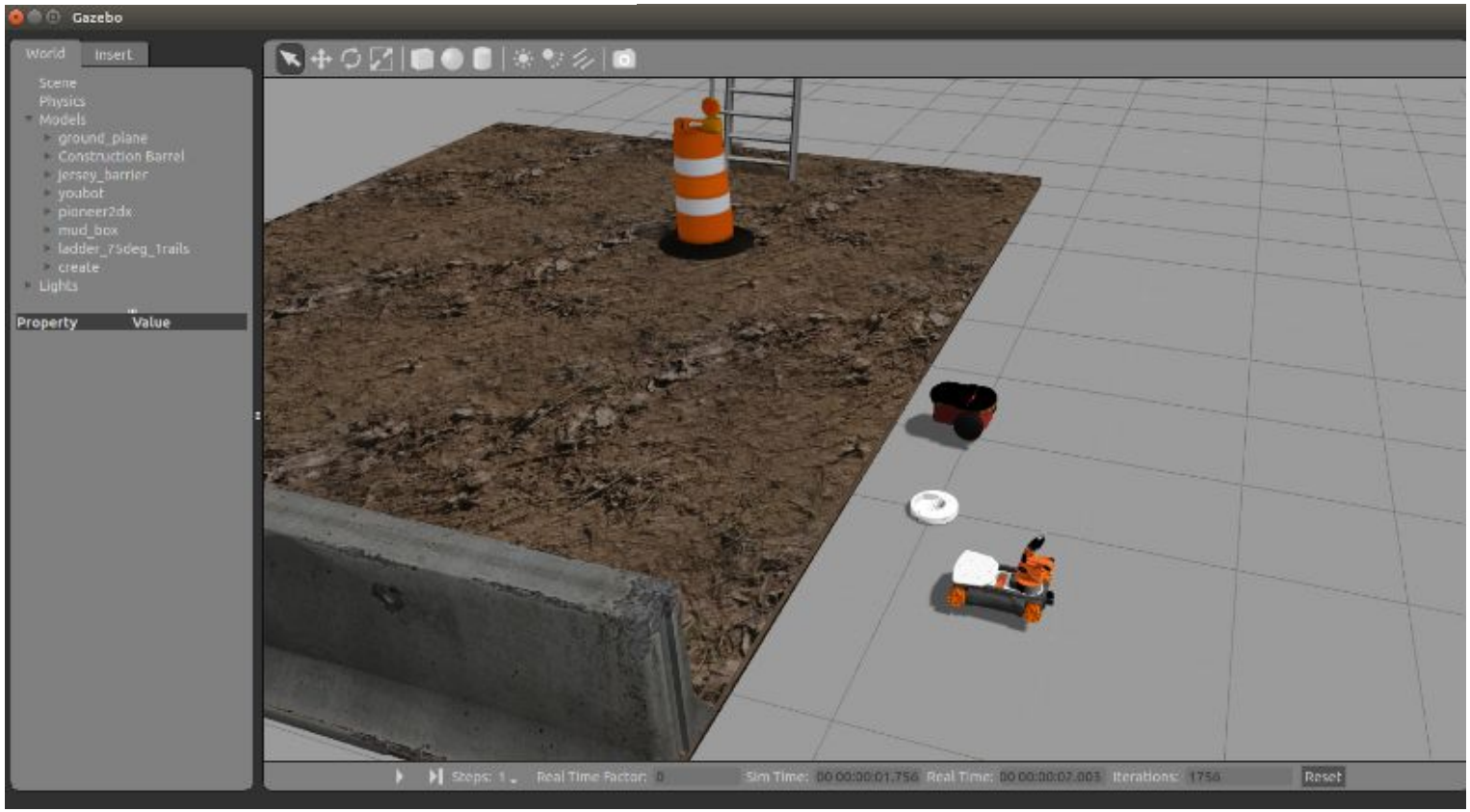
Ventajas

- Testeo rápido de algoritmos y mecánica
- Diseño rápido y económico de robots y entornos
- Cambiar características del entorno mediante parámetros
- Se evitan problemas típicos (Baterías, comunicación, componentes físicos)
- Permiten concentrarse en el comportamiento
- Pueden ser ejecutados por tiempo indefinido
- Permiten manipular el paso del tiempo
- Repetitividad
- Trabajo distribuido
- Validación económica que descarta soluciones inviables

Desventajas

- Complejo simular un agente real
- Se simplifican aspectos que pueden ser importantes
- Diferente lenguaje en el simulador que en el robot
- Modelos complejos pueden demorar mucho al simular
- Una herramienta más a utilizar

GAZEBO

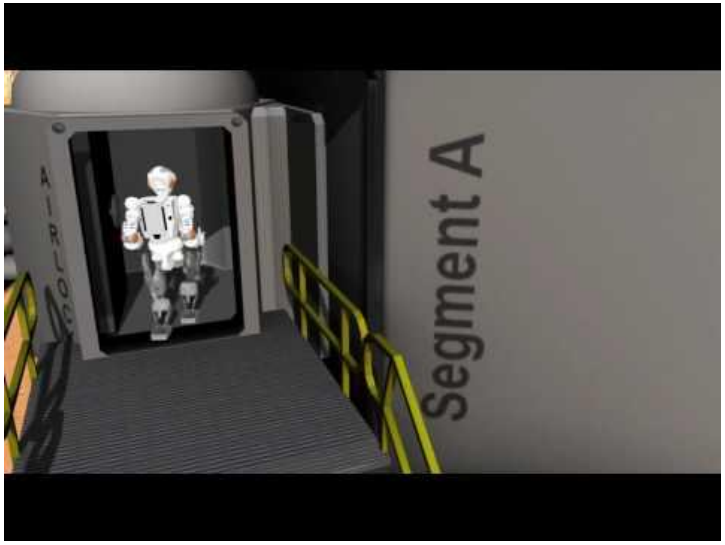

Gazebo
World
Insert
Scene
Physics
Models
ground_plane
Construction Barrel
jersey_barrier
youbot
pioneer2dx
mud_box
ladder_75deg_1rails
create
Lights
Property
Value
Steps: 1
Real Time Factor: 0
Sim Time: 00 00:00:01.756
Real Time: 00 00:00:07.003
Iterations: 1756
Reset

- Comienza en 2002 en la Universidad “University of Southern California”.
- Actualmente en desarrollo por la OSRF (Open Source Robotics Fundation)
- Plataformas soportadas: Linux, Windows e IOS
- Plugins: C++
- Integrado con ROS
- Licencia: Open source - Apache 2.0

GAZEBO


Wide range of assets
Or import Collada, STL, or OBJ files

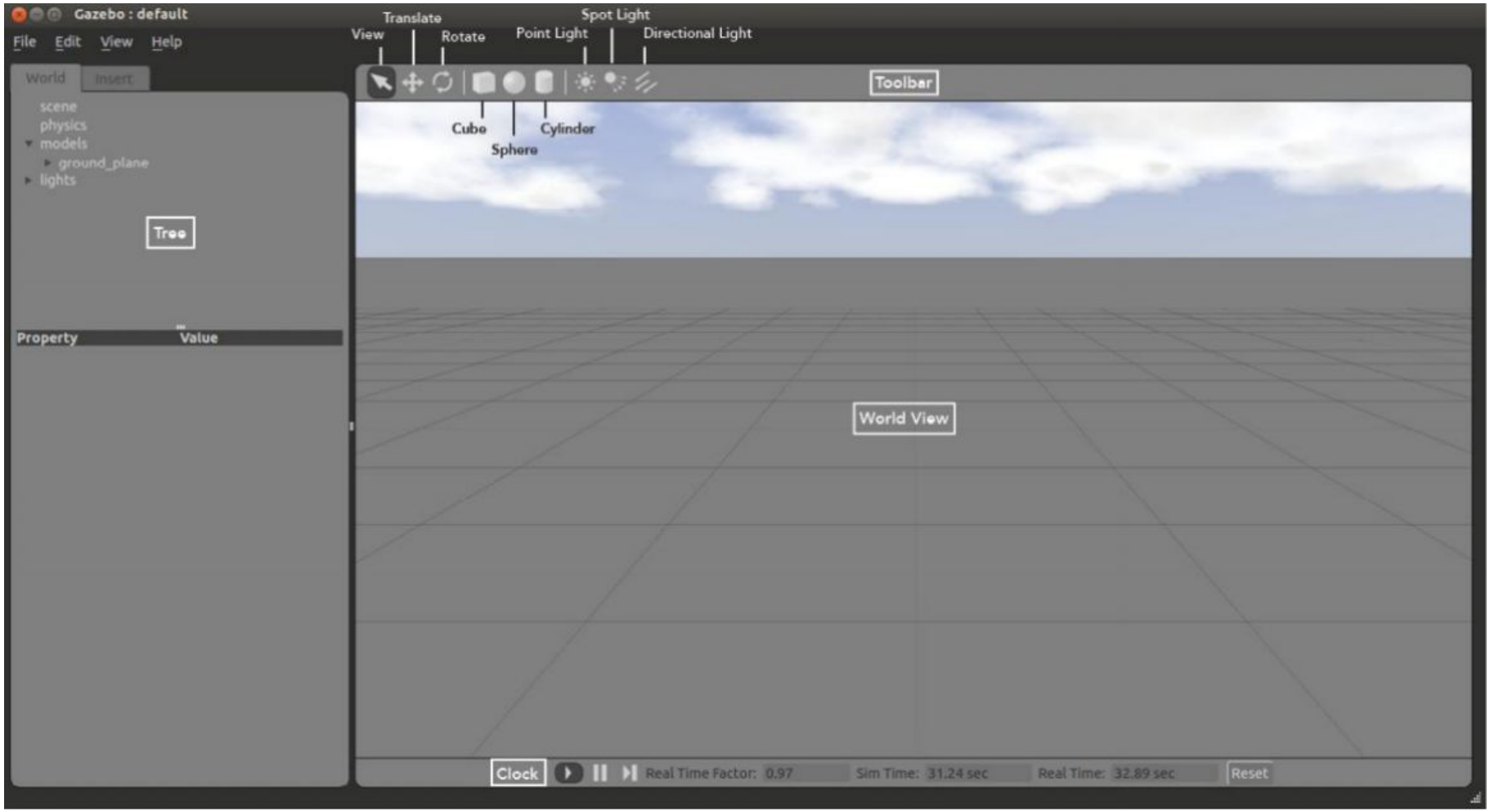

Picking Parts from Conveyor
The system performs several planning iterations to adjust for part movement before pickup


Segment A

Arquitectura

- Dos procesos:
  - Servidor: Simula la física, renders y genera los datos de los sensores
    - gzserver
  - Cliente: Provee interacción y visualización de la simulación
    - gzclient

Interfaz


Gazebo : default
File Edit View Help
World Insert
scene
physics
models
ground_plane
lights
Tree
Property Value
View
Translate
Rotate
Point Light
Spot Light
Directional Light
Toolbar
Cube
Sphere
Cylinder
World View
Clock
Real Time Factor: 0.97
Sim Time: 31.24 sec
Real Time: 32.89 sec
Reset

Gazebo + ROS

- Se encuentran integrados a través del paquete gazebo_ros
- Comunicación bidireccional entre gazebo y ros
- Información de los sensores y física es mandada desde gazebo a ROS
- ROS manda a gazebo los comandos para los actuadores

Ejecutar Gazebo + ROS

- Comando para ejecutar gazebo utilizando ros:

```
$ rosrun gazebo_ros gazebo
```

Mundo

- Describe una colección de robots, objetos y parámetros globales incluyendo el cielo, luz ambiente, y propiedades físicas
- Utiliza el formato SDF y tiene una extensión .world
- El servidor de gazebo lee el archivo y genera el mundo

Ejemplo: willowgarage_world.world

```
<?xml version="1.0" ?>
<sdf version="1.5">
  <world name="default">
    <include>
      <uri>model://ground_plane</uri>
    </include>
    <include>
      <uri>model://sun</uri>
    </include>
    <include>
      <uri>model://willowgarage</uri>
    </include>
  </world>
</sdf>
```

- Tres modelos referenciados
- Primero se buscan en la base de datos de modelos de gazebo, si no se encuentran son descargados de la base de datos online

Modelo

- Representa desde una simple forma a un robot complejo. Incluso el suelo es un modelo.
- Define una entidad física, su cinemática y las propiedades de visión.
- Puede contener uno o más plugins, lo que afecta al comportamiento del mismo.
- Utiliza el mismo formato SDF que el mundo, pero contiene un único tag <model>.
- Una vez creado, se puede agregar a un archivo del mundo:

```
<include filename="model_file_name"/>
```

Ejemplo de uso: TurtleBot3

Instalación

- Instalar el paquete de simulación de Turtlebot3:

```
$ sudo apt-get install ros-melodic-turtlebot3-description
ros-melodic-turtlebot3-msgs ros-melodic-turtlebot3-teleop

$ cd ~/catkin_ws/src/

$ git clone
https://github.com/ROBOTISGIT/turtlebot3_simulations.git

$ cd ~/catkin_ws && catkin_make
```

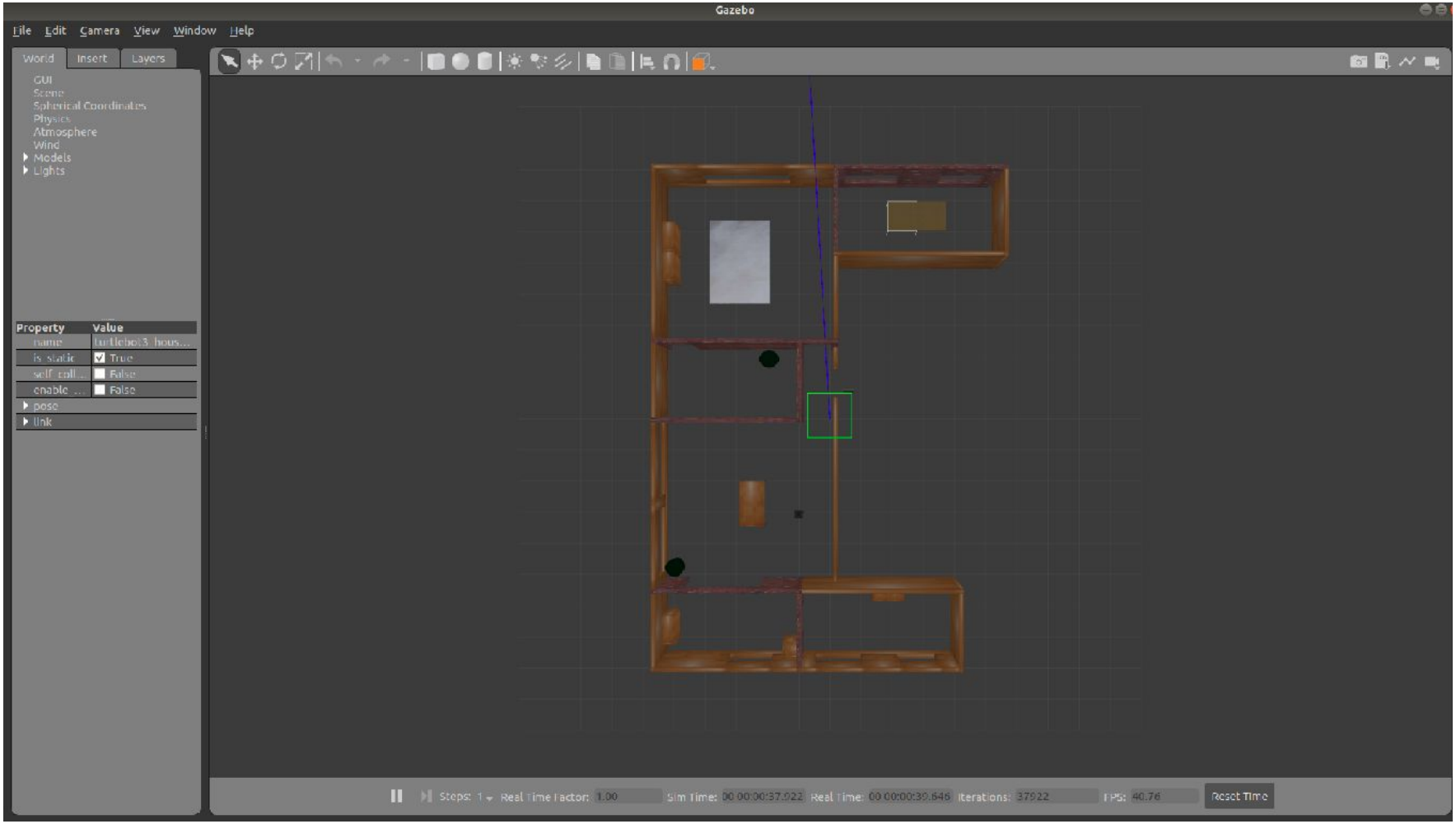
Instalación

- Especificar el modelo de Turtlebot3:

```
$ export TURTLEBOT3_MODEL=${TB3_MODEL}
TB3_MODEL: burger, waffle, waffle_pi
```

- Simular Turtlebot3 en un mundo ya diseñado:

```
$roslaunch turtlebot3_gazebo turtlebot3_house.launch
```


Gazebo
File Edit Camera View Window Help
World Insert Layers
GUI
Scene
Spherical Coordinates
Physics
Atmosphere
Wind
Models
Lights
Property Value
name turtlebot3_hous...
is_static True
self_coll... False
enable ... False
pose
link
Steps: 1 Real Time Factor: 1.00 Sim Time: 00 00:00:37.022 Real Time: 00 00:00:39.646 Iterations: 37922 FPS: 40.76
Reset Time

Moviendo a TurtleBot3: Teleoperado

- Correr el nodo para mover el robot con el teclado:

```
$ rosrun turtlebot3_teleop turtlebot3_teleop_key
```

```
Control Your TurtleBot3!
---------------------------
Moving around:
        w
   a    s    d
        x

w/x : increase/decrease linear velocity (Burger : ~ 0.22, Waffle and Waffle Pi : ~ 0.26)
a/d : increase/decrease angular velocity (Burger : ~ 2.84, Waffle and Waffle Pi : ~ 1.82)

space key, s : force stop

CTRL-C to quit
```

Moviendo a TurtleBot3: Mediante un tópico

- Publicar velocidad

Moviendo a TurtleBot3: Mediante un tópico

- Publicar velocidad:

```
$ rostopic pub /cmd_vel geometry_msgs/Twist "linear:
  x: 0.3
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0"
```

Moviendo a TurtleBot3: Mediante un nodo

- Crear un paquete para trabajar en el laboratorio:

```
$ catkin_create_pkg fra_laboratorio4 std_msgs rospy
roscpp
```

- Crear un nodo para que el robot se mueva

- Crear un archivo *.launch* en el directorio launch que levante el mundo y el nodo

Rviz

- Permite visualizar la información sensada:

```
$ export TURTLEBOT3_MODEL=waffle

$ roslaunch turtlebot3_gazebo turtlebot3_gazebo_rviz.launch
```

Preguntas?