Evolution of robotic simulators: Using UE4 to enable real-world quality testing of complex autonomous robots in unstructured environments

Patrick Wolf, Tobias Groll, Steffen Hemer, Karsten Berns

Introduction

- Simulation is important for the development of autonomous systems (early testing, design & sensor optimization, improved controller development).
- There exists a gap between testing in simulation and the real world. Complex, relevant features for robot perception cannot be completely modeled.
- Autonomous commercial vehicles have specialized supplements and kinematics for task fulfillment that have to be regarded by simulation.
- This contribution aims to narrow this gap and to increase the expressiveness of robot control software tests.

Simulation Framework

- **Finroc**: Framework for intelligent robot control (C++, Java). Decomposition of functionality into modules, data exchange with ports (features zero-copy, lock-free implementations).
- **Unreal Engine 4**: High performance in visualizing, performs outstandingly in representing very realistic environments, feasible real-time simulation capabilities, good representation of outdoor landscapes and physically correct vehicles.
- **Interface Plugin**: Port functionality of Finroc is loaded by UE 4 via C++ shared object file. Interface components assert data type compatibility and can be attached to any actor (C++ classes & BLUEPRINTS).

![Finroc Interface Plugin](image)

Sensors

- **IMU**: Access linear velocity & angular accelerations from physic actor. Apply standard gravity to simulate earth gravity. Add drifts/ slips over time in a preprocessing step to approximate the physical sensor’s quality.
- **Odometry**: Access skeletal mesh of robot and read angular velocities of the corresponding wheel bones to correctly represent wheel slippage.
- **GNSS**: GNSS systems are heavily affected through the environment, e.g. shadowing. The visible satellite constellation is described as the dilution of precision (DOP). GNSS simulation uses raytracing to determine the visibility of the receiver. Simulation DOP constants are approximated using real-world GNSS data. A positioning error is applied according to the visible satellite constellation.

![IMU Sensor](image)

Actuators

- **Driving Kinematics**: Realization of specialized driving kinematics. E.g. axle shrinking or pivot steering. Additional joints were built into the skeleton of the vehicle to model the additional properties.
- **Complex Body Actuation**: Realization of parallel kinematic chains on the example of an excavator arm. Arm segments move using hydraulic cylinders. A bone is created for each part and a parallel kinematic chain is added. Hydraulic drives move the appropriate body part. Spring-damper systems generate forces and simulate hydraulics actuators.

![Actuators](image)

Applications & Conclusion

- Application development benefits from simulation realism. Examples are autonomous excavation in an open-pit mine & the safe and reliable autonomous off-road driving inside cluttered environments.
- Similar robot control behavior can be achieved for simulation and real-world testing which reduces test effort tremendously.

![Applications](image)

Contact: {patrick.wolf, groll, s_hemer, berns}@cs.uni-kl.de

---

SIMULTECH 2020

Robotics Research Lab, Dep. of Computer Science, TU Kaiserslautern, Kaiserslautern, Germany

Finroc.org

rrlab.cs.uni-kl.de