

Introduction to Robot Operating System (ROS)

Application to mobile robots

Amr Elhussein

Advisor: Dr. Suruz Miah

Department of Electrical and Computer Engineering

Bradley University

1501 W. Bradley Avenue

Peoria, IL, 61625, USA

Friday, May 31, 2019

- Introduction
 - Historical Background
 - Robot Programming Before ROS
 - ROS is ..
 - ROS Equation
 - Applications
- ROS Concepts
 - Filesystem
 - Computation Graph
 - Community level
- ROS installation
- Future of ROS

Introduction

History and Legacy

- Started in 2007 by researches from Stanford AI Robot (Stair) and the Personal Robots (PR) Program and was sponsored by Willow Garage a visionary robotics incubator.
- Used Worldwide in Research and Industry.
- Currently supported by the Open Source Robotics Foundation.



Figure: Stair

Introduction

Robot Programming Before ROS

- No common platform for developing robotics
- Build every thing from scratch
- Algorithm implementation

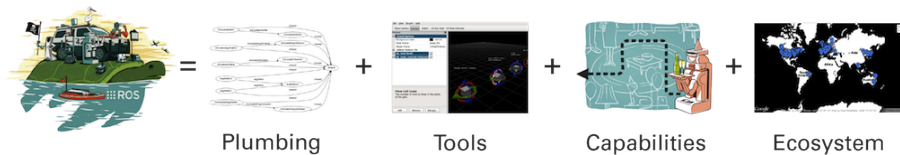
Introduction

ROS is ..

A flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.

Introduction

Ros Equation



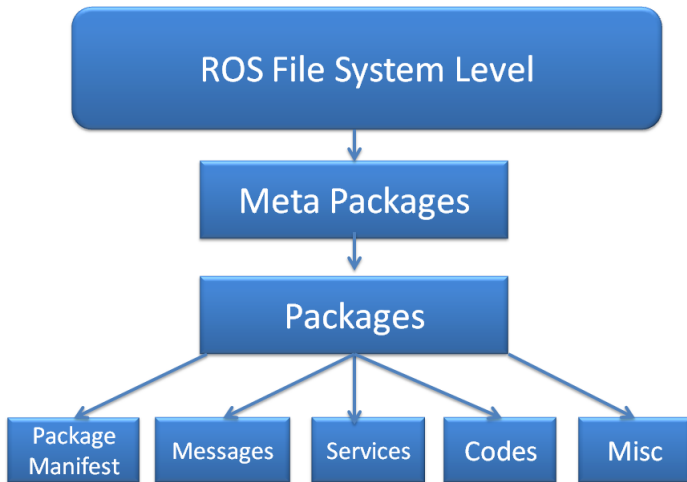
Introduction

Applications



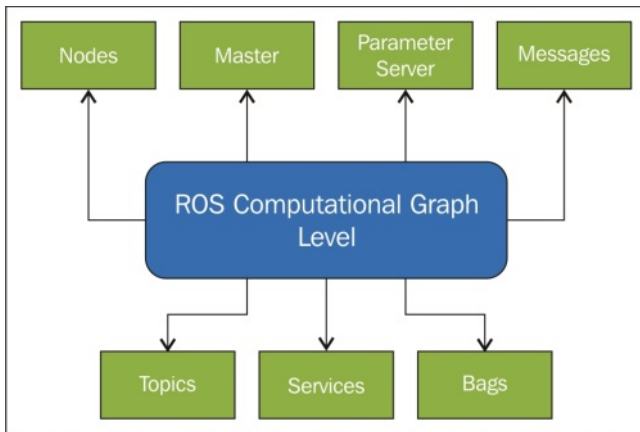
ROS Concepts

Filesystem



ROS Concepts

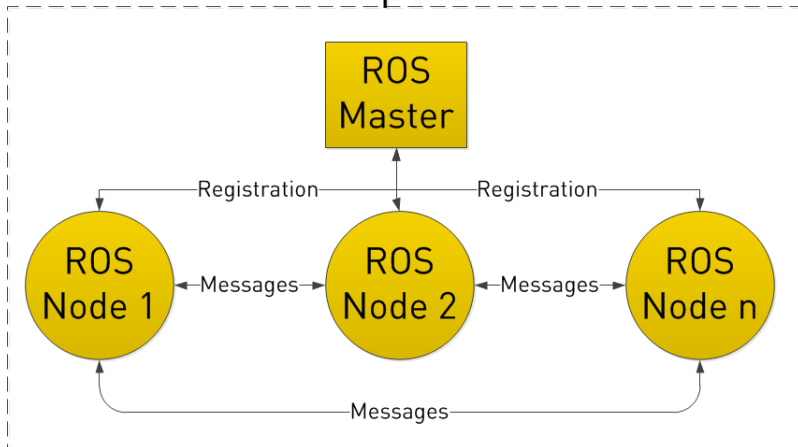
Computation Graph



ROS Concepts

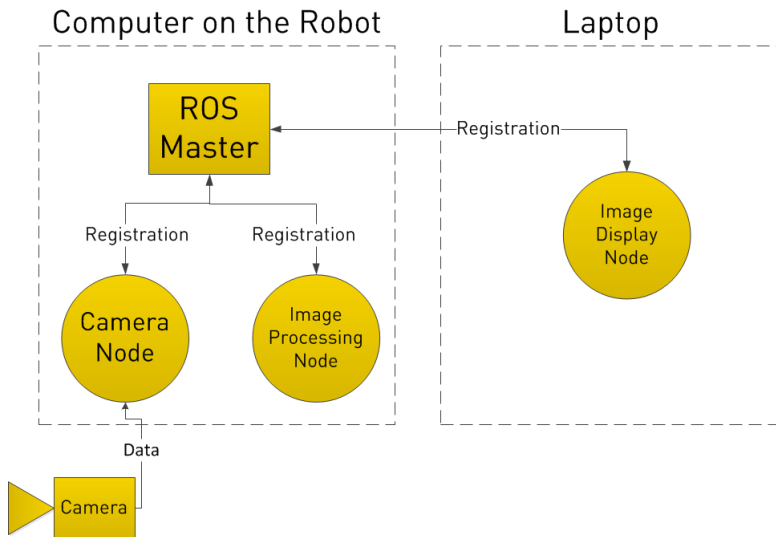
Computation Graph: Master

Computer 1



ROS Concepts

Computation Graph: Master



ROS Concepts

Community level

ROS Community Level

Distributions

Blog

ROS Wiki

Mailing list

Repositories

Bug ticket System

ROS Answers

Installation

- Debian-based distributions such as Ubuntu.
- Many robots.
- Current supported distributions
 - ROS Kinetic Kame, Released May, 2016.
 - ROS Melodic Morenia, Released May, 2018

After choosing the distribution follow the instruction on ROS Wiki which start by:

- Configure your Ubuntu repositories.
- Setup your sources.list.
- Set keys.
- Install with "sudo apt-get install ros-kinetic-desktop-full".

Future of ROS

- Security
- Critical Missions
- Distributed Processing



Thanks !

Matlab Robotics Systems Toolbox

Application to mobile robots

Amr Elhussein

Advisor: Dr. Suruz Miah

Department of Electrical and Computer Engineering
Bradley University
1501 W. Bradley Avenue
Peoria, IL, 61625, USA

Friday, June 4, 2019

- Introduction
- Workflow
 - Desktop prototyping
 - Standalone ROS Nodes
- Examples

According to mathworks.com

Robotics System Toolbox provides algorithms and hardware connectivity for developing autonomous robotics applications for aerial and ground vehicles, manipulators, and humanoid robots. Toolbox algorithms include path planning and path following for differential drive robots, scan matching, obstacle avoidance, and state estimation. For manipulator robots, the system toolbox includes algorithms for inverse kinematics, kinematic constraints, and dynamics using a rigid body tree representation.

Workflow

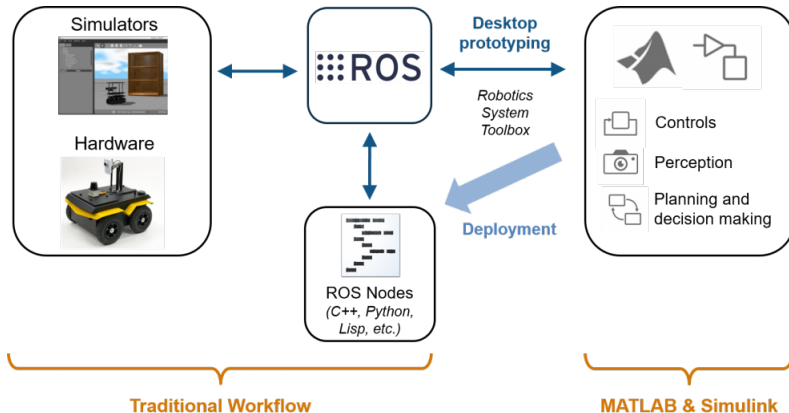


Figure: Matlab robotics tool box and ROS workflow. courtesy of mathworks.com

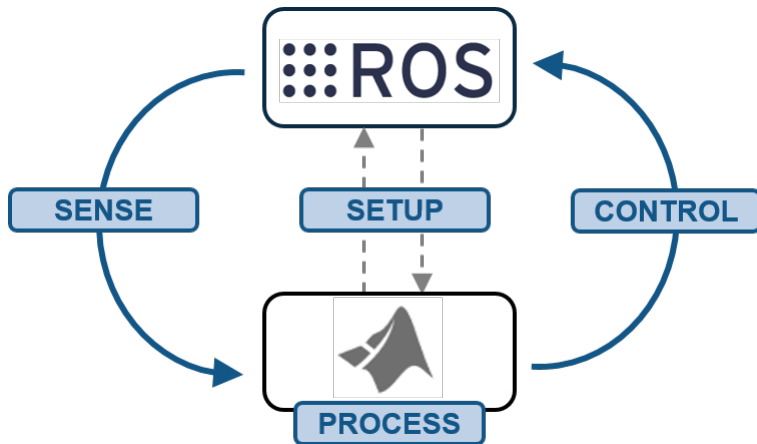


Figure: Matlab and ROS integration, courtesy of mathworks.com

Workflow

Desktop prototyping

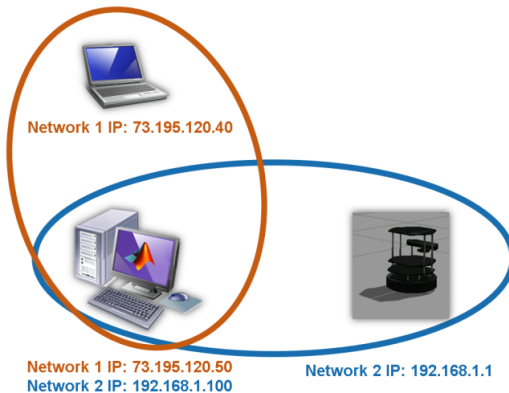
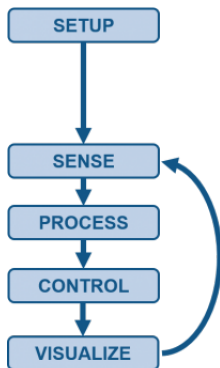


Figure: Matlab ROS desktop prototyping, mathworks.com

Workflow

Desktop prototyping



```
rosinit('ipAddress')
mySub = rossubscriber('/sub_topic');
[myPub, pubMsg] = rospublisher('/pub_topic');
currentTime = 0;

tic
while(currentTime < 10)
    recvMsg = mySub.LatestMessage;

    ctrlOut = myAlgorithm(recvMsg);

    pubMsg.FieldName = ctrlOut;
    send(myPub, pubMsg);

    currentTime = toc;
    plot(currentTime, ctrlOut)
end
```

Figure: Desktop prototyping code template, courtesy of mathworks.com

Workflow

Standalone Node

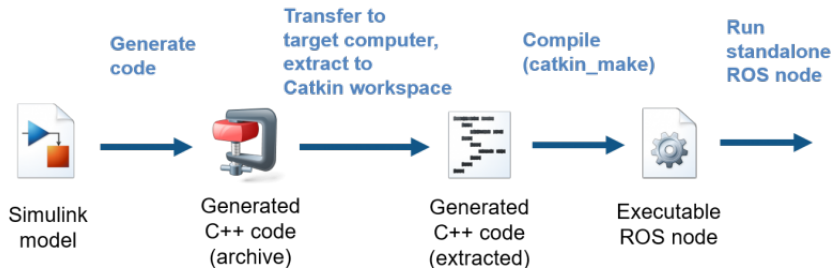


Figure: Generation of ROS standalone node, courtesy of mathworks.com

Workflow

Standalone Node

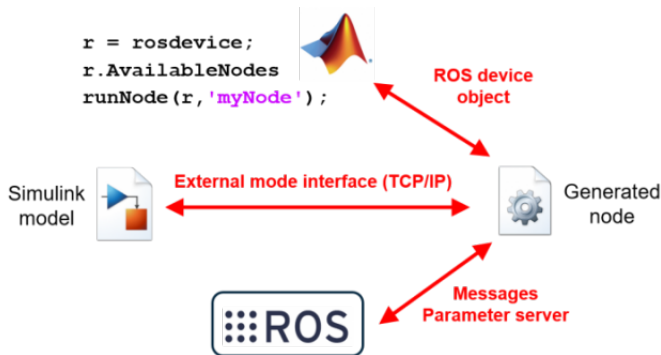


Figure: Access to ROS standalone node, courtesy of mathworks.com

Examples

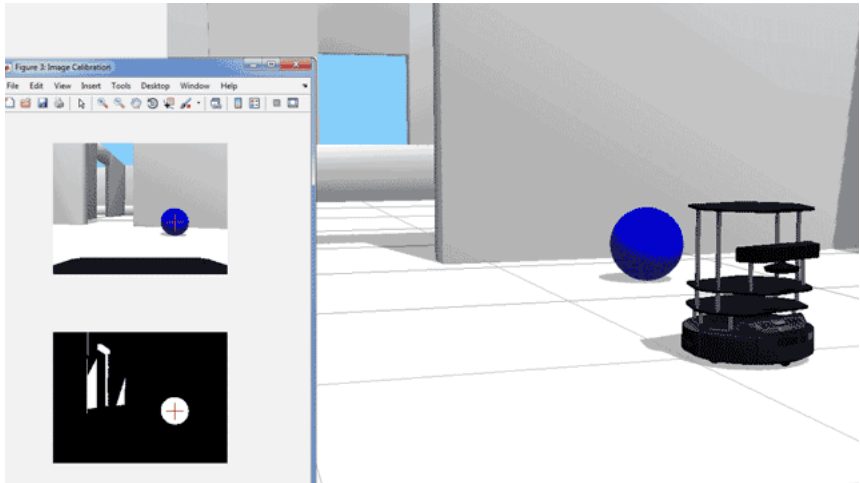


Figure: Turtle bot example, courtesy of mathworks.com

Area Coverage Optimization

Progress Report

Amr Elhussein
Advisor: Dr. Suruz Miah

Department of Electrical and Computer Engineering
Bradley University
1501 W. Bradley Avenue
Peoria, IL, 61625, USA

Friday, June 21, 2019

- Introduction to V-REP
- Interfacing Matlab and ROS on the same Machine
- Line following simulation
- Leader follower simulation
- Area Coverage simulation

General purpose robot simulator with
integrated development environment
"coppeliarobotics.com".



Interfacing

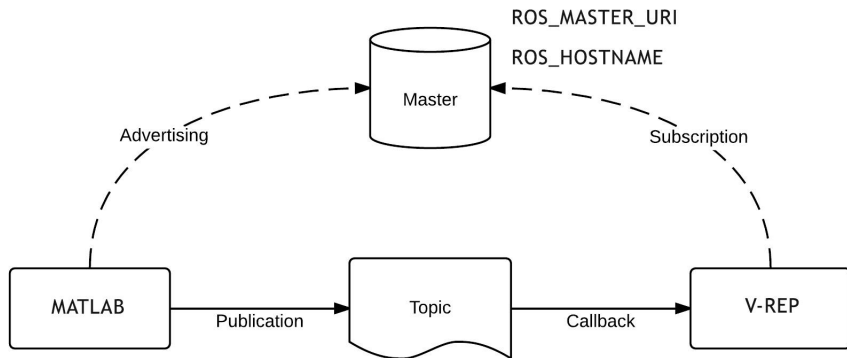


Figure: ROS, Matlab and V-REP interface

Line Following

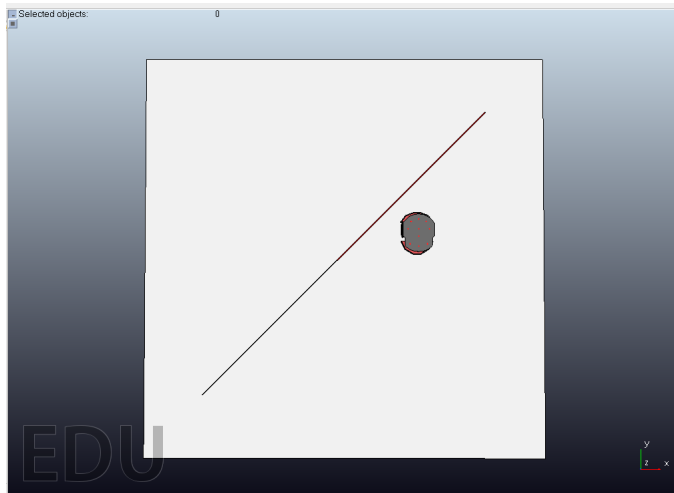


Figure: Line Following Scene

Leader Following

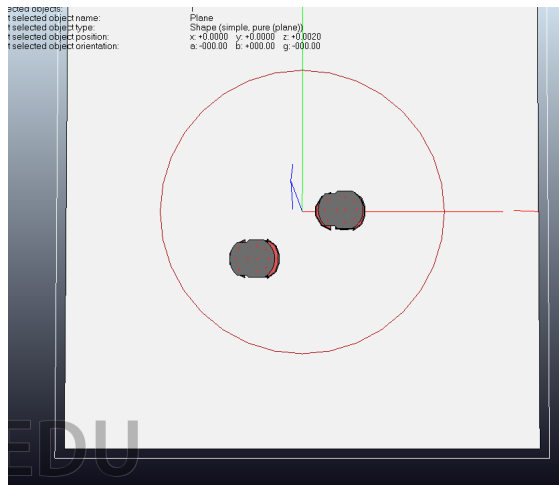


Figure: Leader Follower Scene

Area Coverage

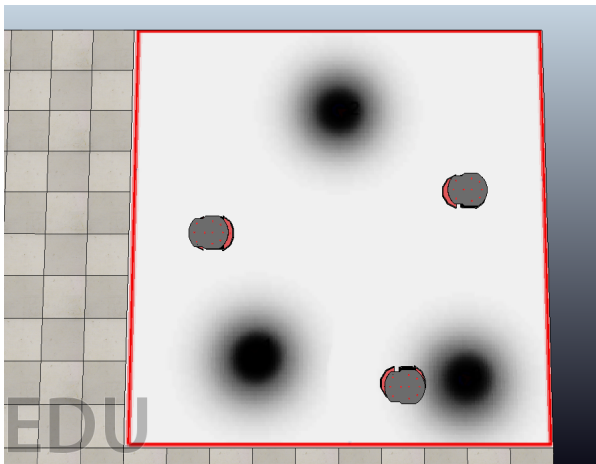


Figure: Area Coverage Scence