



**UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE**  
**CENTRO DE TECNOLOGIA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECATRÔNICA**



## **RELATÓRIO: Tutorial de Robot Operating System**

### **ROS**

Natal - RN  
2021

Iago Lucas Batista Galvão  
Jhonat Heberson Avelino de Souza  
Thiago de Araújo Brito

Relatório: Tutorial de Robot Operating System  
ROS

Relatório da disciplina EGM0007 - Sistemas Robóticos  
Autônomos do Mestrado em Engenharia Mecatrônica,  
referente a nota parcial da segunda unidade.

Professor(a): Pablo Javier Alsina.

Natal - RN  
2021

# Sumário

	<b>Páginas</b>
<b>1 INTRODUÇÃO</b>	<b>4</b>
<b>2 OBJETIVO</b>	<b>4</b>
<b>3 ROS</b>	<b>4</b>
<b>4 Interface ROS com Matlab</b>	<b>6</b>
<b>5 CONCLUSÃO</b>	<b>8</b>
<b>6 REFERÊNCIAS</b>	<b>8</b>

# 1 INTRODUÇÃO

ROS (*Robot Operational System*) é um conjunto de bibliotecas e ferramentas que ajudam na construção de aplicações para robôs em todas as áreas atuantes, seja *software* ou *hardware*. A plataforma engloba desde os drives de câmeras e motores, até algoritmos de última geração em diversas linguagens de programação e, também, poderosas ferramentas de desenvolvimento. ROS concatena praticamente todas as ferramentas de desenvolvimento necessárias para projetos de robótica, sendo este *OpenSource*.

Quando o projeto do robô possui diferentes módulos, como por exemplo, câmera, sensores, GPS, giroscópio entre outros; os quais se façam necessários a comunicação entre eles, o próprio robô e outros computadores/dispositivos, isso torna-se ser uma tarefa muito complicada para comunicarem entre si. Com o propósito de facilitar esse trabalho, o ROS facilita toda atividade, processando todos os dados em paralelo para o controle do robô e subscrevendo para cada dispositivo os comandos desejados.

Um exemplo do uso do ROS é a necessidade que sejam feitos cálculos computacionais que demandam grande capacidade da máquina, mas o computador embarcado no robô não é suficiente capaz para realizar tamanho processamento. Por meio da plataforma, é possível criar nós em um computador mestre com maior capacidade processual e esta comunica-se através de TCP/IP com o robô.

## 2 OBJETIVO

O propósito deste tutorial é descrever o passo a passo de como utilizar o ROS e apresentar esta poderosa plataforma que facilita em diversas áreas de aplicação da robótica, o pesquisador ou desenvolvedor.

## 3 ROS

Nesta seção será apresentado o tutorial de como usar o ROS e suas funcionalidades. Para isso, faz-se importante explicar algumas funcionalidades como arquitetura, infraestruturas de comunicação e recursos disponíveis.

A Arquitetura do ROS é algo muito importante, a qual faz toda diferença para criação de aplicações e facilidade no uso. Na rede em que o ROS estiver em funcionamento, existe sempre uma máquina considerada como mestre, na qual deve manter o comando *Roscore* em funcionamento. Esse comando irá inicializar a configuração necessária para que haja a comunicação entre os nós. É possível manter os nós em funcionamento em qualquer uma das máquinas da rede (um nó é basicamente um executável). Os nós se comunicam entre si por meio do protocolo TCP/IP, onde cada nó pode receber e publicar em tópicos.

A infraestrutura de comunicação no seu nível mais baixo, oferece uma interface de passagem de mensagens que fornece comunicação entre processos e é comumente referida como um *middleware*. Este oferece os recursos:

- Publicar / receber passagem de mensagem anônima;
- Gravação e reprodução de mensagens;
- Solicitação / resposta de chamadas de procedimento remoto;
- Sistema de parâmetros distribuídos.

O ROS, também, nos fornece vários recursos disponíveis que são comumente utilizados no desenvolvimento de robôs. Além dos componentes centrais de *middleware*, a plataforma fornece as próprias bibliotecas e ferramentas comuns, específicas para robôs que facilitarão o seu funcionamento, como:

- Definições de mensagens padrão para robôs;
- Biblioteca de geometria do robô;
- Linguagem de descrição do robô;
- Chamadas de procedimento remoto preemptivas;
- Diagnóstico;
- Estimativa de posição;
- Localização;
- Mapeamento;
- Navegação.

Um dos recursos mais notável do ROS é o poderoso conjunto de ferramentas de desenvolvimento. Estas oferecem suporte à introspecção, depuração, plotagem e visualização do estado do sistema que está sendo desenvolvido. O mecanismo de publicação/recebimento subjacente permite que seja feita uma introspecção espontânea dos dados que fluem pelo sistema, facilitando a compreensão e a depuração de problemas à medida que estes ocorrem.

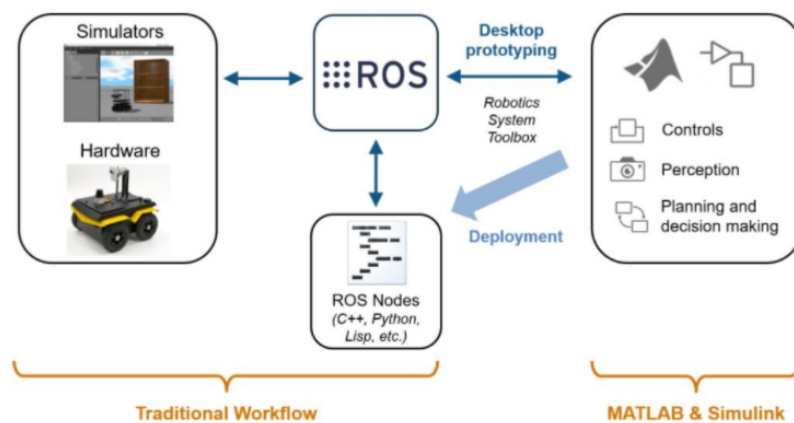
## 4 Interface ROS com Matlab

Com o uso do *Matlab* é possível conectar ao ROS a partir de qualquer sistema operacional que suporte *Matlab* e *Simulink*, para aproveitamento das funcionalidades integradas no *MathWorks toolboxes*, como por exemplo: sistemas de controle, *machine learning*, processamento de sinais e visão computacional; também, pode-se gerar nós independentes de forma automática, baseados em C++ a partir de algoritmos projetados em *Matlab* e *Simulink*.

Algumas funcionalidades inclusas no *Robotics System ToolBox*:

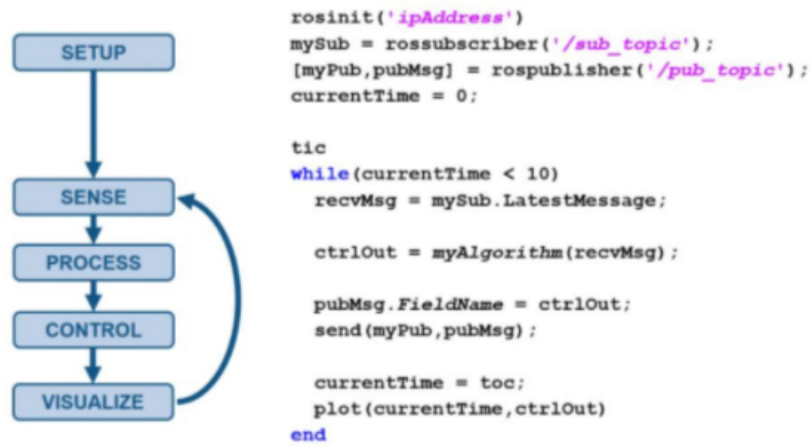
- Interface do fluxo de trabalho Matlab com ROS mostrado na Figura 1;
- Inicialização ou conexão a um ROS principal mostrado na Figura 2;
- Trabalhar com tópicos(envio/recebimento), serviços, ações e servidor de parâmetros.
- Realizar leitura de mensagens especializadas (por exemplo, imagens, varreduras lidar, nuvens de pontos e grades de ocupação);
- Aplicativo interativo de comunicação entre o robô e simulador que esteja em funcionamento o Sistema Operacional de Robôs, mostrado na Figura 3.

**Figura 1.** Fluxo de trabalho



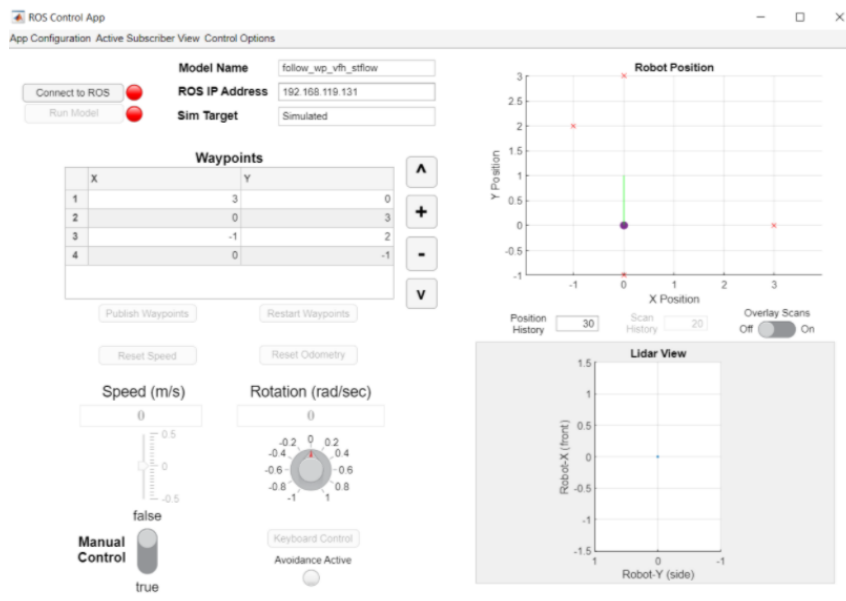
Fonte: Autor.

**Figura 2.** Como conectar de ROS com Matlab



Fonte: Autor.

**Figura 3.** ROS Control App.



Fonte: Autor.

## 5 CONCLUSÃO

Com a inovação de simuladores como este, pesquisadores de todos os níveis podem realizar suas pesquisas sem a necessidade física do dispositivo robótico. Assim, reduzindo ao máximo os possíveis erros de programação, antes de aplicar no robô físico.

A partir desta, também facilitou a compreensão da teoria mostrada em aula, principalmente, nos casos de escassez ou ausência dos materiais necessários para tal finalidade. Outro sim, destaca-se a importância dessas ferramentas simulatórias, para o caso atual de pandemia, onde as aulas são remotas.

Com a concatenação de diversos dispositivos com protocolos, linguagens, comunicação, atuação e sistemas distintos; todo o trabalho, antes demandado, para unir tais áreas da robótica é integrado e facilitado por esta incrível plataforma, amplamente utilizada em todo o mundo.

## 6 REFERÊNCIAS

- [1] CASTRO SEBASTIAN; STABILE, C. *MATLAB Apps with ROS*. <https://www.mathworks.com/videos/matlab-and-simulink-robotics-arena-matlab-apps-with-ros-1526379787323.html>. Acesso em: 14 jul. 2021.: [s.n.], 2021.
- [2] CASTRO, S. *Getting Started with MATLAB, Simulink, and ROS*. Disponível em: <https://blogs.mathworks.com/student-lounge/2017/11/08/matlab-simulink-ros/>. Acesso em: 14 jul. 2021.: [s.n.], 2017.
- [3] JUNIOR, F. E. F. *Uma introdução ao Robot Operating System (ROS)*. Disponível em: <https://www.embarcados.com.br/uma-introducao-ao-robot-operating-system-ros/>. Acesso em: 14 jul. 2021.: [s.n.], 2016.
- [4] L., S. *WHAT is ROS?*. Disponível em: <https://www.ros.org/>. Acesso em: 14 jul. 2021.: [s.n.], 2021.