Intelligent Control of Robot Manipulators

Aiswarya Menon

Intelligent Systems Lab
IIT Kanpur

January 30, 2019

Introduction

What is Intelligent Control?

- Uses the concepts of conventional control strategies and couples them with additional strategies required.
- Uses artificial intelligence tools like *Neural Networks or Fuzzy Neural Networks* for the controller design.

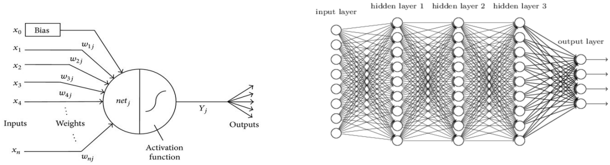


Figure: Neural Network

Why Intelligent Control?

- This control strategy benefits the most for the design of non linear control systems.
- Takes care of model uncertainities.
- Works well in cases of unknown model dynamics.

Optimal Kinematic Control of Robot Manipulator

- In kinematic control, a kinematic command is given as the control input to the robot system to achieve a desired task.
- Optimal kinematic control not only achieves the desired task but minimises a global cost function as well which helps in minimising the resources used in the task. Global cost function is,

$$J = \Phi(x(t_f), t_f, x(t_0), t_0) + \int_{t_0}^{t_f} L(x(t), u(t), t)dt$$

subject to the first order dynamic constraints

$$\dot{x}(t) = a[x(t), u(t), t]$$

- Optimal control policy is derived using Hamilton-Jacobi-Bellman (HJB) formulation. HJB equation:

$$\dot{J}(x(t), t) + \min_{u}\{\nabla J(x(t), t).\dot{x}(t) + L(x(t), u(t), t)\} = 0$$

- Since analytic solution of HJB is still a major challenge as they are obtained offline, *Neural Network and Fuzzy Network* schemes have become popular

Proposed Approach- Kinematic Control of Robot Manipulator using Single Network Adaptive Critic (SNAC)

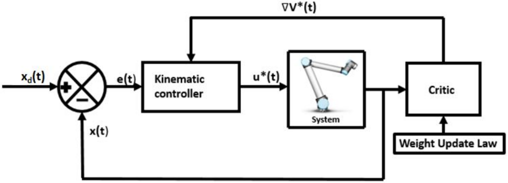


Figure: Block Diagram

Task: Online Optimal kinematic control of a robot manipulator where the robot end effector position follows a task space trajectory under SNAC framework. *S*ystem dynamics:

$$x(t) = f(\theta(t))$$

$$\dot{x}(t) = J\dot{\theta}(t)$$

$$\dot{x}(t) = Ju(t)$$

where, $x \in \mathbb{R}^n$ and $\theta \in \mathbb{R}^m$.

Proposed Approach (contd)

We solve for an optimal control input $u(t)$ for a robot manipulator such that tracking error given below is minimised.

$$e(t) = x(t) - x_d(t)$$

where, $x_d(t)$ is the desired trajectory/fixed position.
Major Contribution: A novel critic weight update law for the critic was proposed that ensures convergence to the desired optimal cost while guaranteeing the stability of the closed loop kinematic control.
Error dynamics,

$$\dot{e}(t) = \dot{x}(t)$$
$$\dot{e}(t) = Ju(t)$$

Task 1: Regulation Problem: Robot end effector reaching a desired fixed position $x_d \in \mathbb{R}^n$ is formulated as a regulation problem.
Infinite horizon HJB cost function to be minimised:

$$V(e(t)) = \int_t^\infty e(t)^T Qe(t) + u(t)^T Ru(t)dt$$

Proposed Approach (contd)

Optimal control input obtained by solving HJB:

$$u^*(e) = \frac{-1}{2} R^{-1} J^T \nabla V^*(e)$$

Task 2: Tracking Problem: Robot end effector following a desired time varying trajectory $x_d(t) \in \mathbb{R}^n$ is formulated as a tracking problem. An Augmented system is constructed as $\xi(t) = [e(t)^T, x_d(t)^T]^T \in \mathbb{R}^{2n}$

$$\dot{\xi}(t) = F(\xi(t)) + G(\xi(t))u(t)$$

Infinite horizon HJB cost function to be minimised:

$$V(\xi(t)) = \int_t^\infty \xi(t)^T \bar{Q} \xi(t) + u(t)^T R u(t) dt$$

where, $\bar{Q} = diag\{Q, 0_{n \times n}\}$. $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$ are positive definite matrices.

Optimal control found by solving HJB:

$$u^*(\xi) = \frac{-1}{2} R^{-1} G(\xi)^T \nabla V^*(\xi)$$

Neural network control design:

- Instantaneuous cost $V^*(\xi(t))$:
  It is approximated by a single hidden layer neural network with p hidden neurons and non-linear activation function:
  $$\hat{V}^*(\xi(t)) = \hat{W}_c^T \phi_c(\xi)$$
  $$\nabla \hat{V}^*(\xi(t)) = \nabla \phi_c^T(\xi) \hat{W}_c$$
  where, $\hat{W}_c \in \mathbb{R}^p$ is the estimated weight to be tuned and $\phi \in \mathbb{R}^p$.
- Weight update rule:
  $$\dot{\hat{W}}_c = \alpha \nabla \phi_c G(\xi) R^{-1} G(\xi)^T \nabla J_s(\xi)$$
  where, α is the learning rate and $J_s(\xi) = \frac{1}{2}\xi^T \xi$.
- Control Policy:
  $$\hat{u}^*(\xi) = \frac{-1}{2} R^{-1} G(\xi)^T \nabla \phi_c^T \hat{W}_c$$
  $$u^*(\xi) = \frac{-1}{2} R^{-1} G(\xi)^T (\nabla \phi_c^T W_c + \nabla \varepsilon_c)$$

Results and Discussion

- Numerical simulations and real time experimental validations on a real 6 DOF UR10 robot manipulator are conducted to demonstrate the effectiveness of the proposed kinematic control.
- Obervations: After a short transient time, error trajectory converges to zero and the joint velocity is always within the limits.

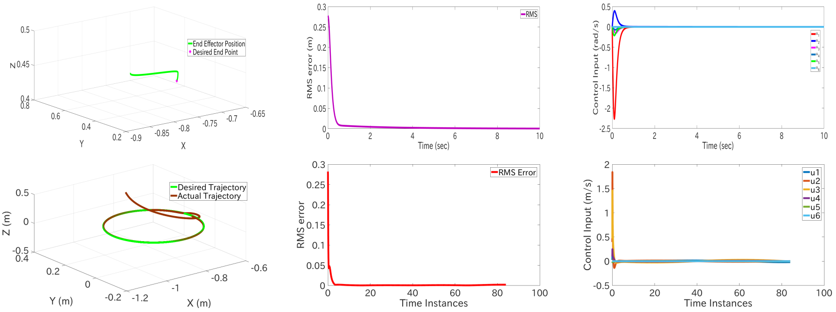


Figure: Simulation results for optimal regulation (a-c) and optimal tracking (d-f).

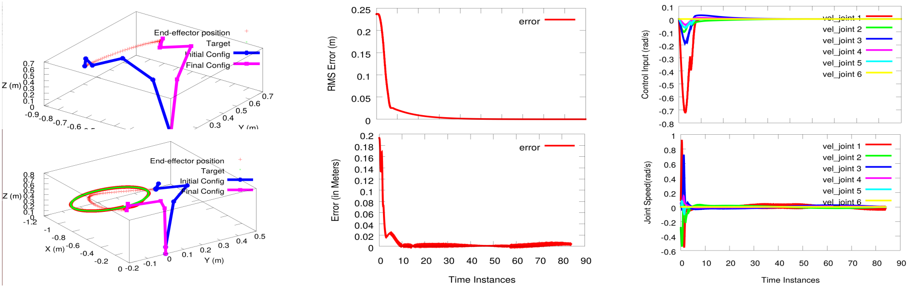
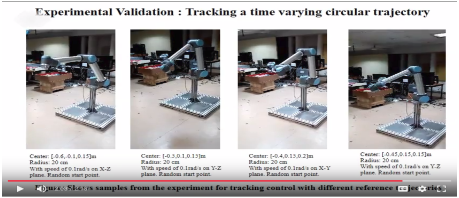


Figure: Experimental results for optimal regulation (a-c) and optimal tracking (d-f)



Click here for video showing the experimental validation

Applications and Future Works:

- Applications: Agriculture, warehouse applications, domestic applications etc.
- Learning rate of the neural network is to be made adaptive so that it learns faster.
- Collaboration with visual surveillance

THANKYOU