Sensors in Robotics

Sensors are used in robotics to calculate the condition and environment of robots, using functions similar to the human sensory organs. A variety of sensors are required by different robots to navigate their environment while performing tasks.

Sensors in Robotics refer to a mechanical function used to calculate the condition and environment of a robot. This sensor is based on the functions of the human sensory organs. Robots receive a broad range of data about their surroundings, such as position, size, orientation, velocity, distance, temperature, weight, force, etc. This information is what allows the robot to function efficiently while interacting with its environment to perform complex tasks.

The working of robot sensors derives from the principle of energy conversion, also known as transduction. Different sensors are required by different robots to attain measures of control and respond flexibly in their environment.

Using neural networks to teach robots and cars

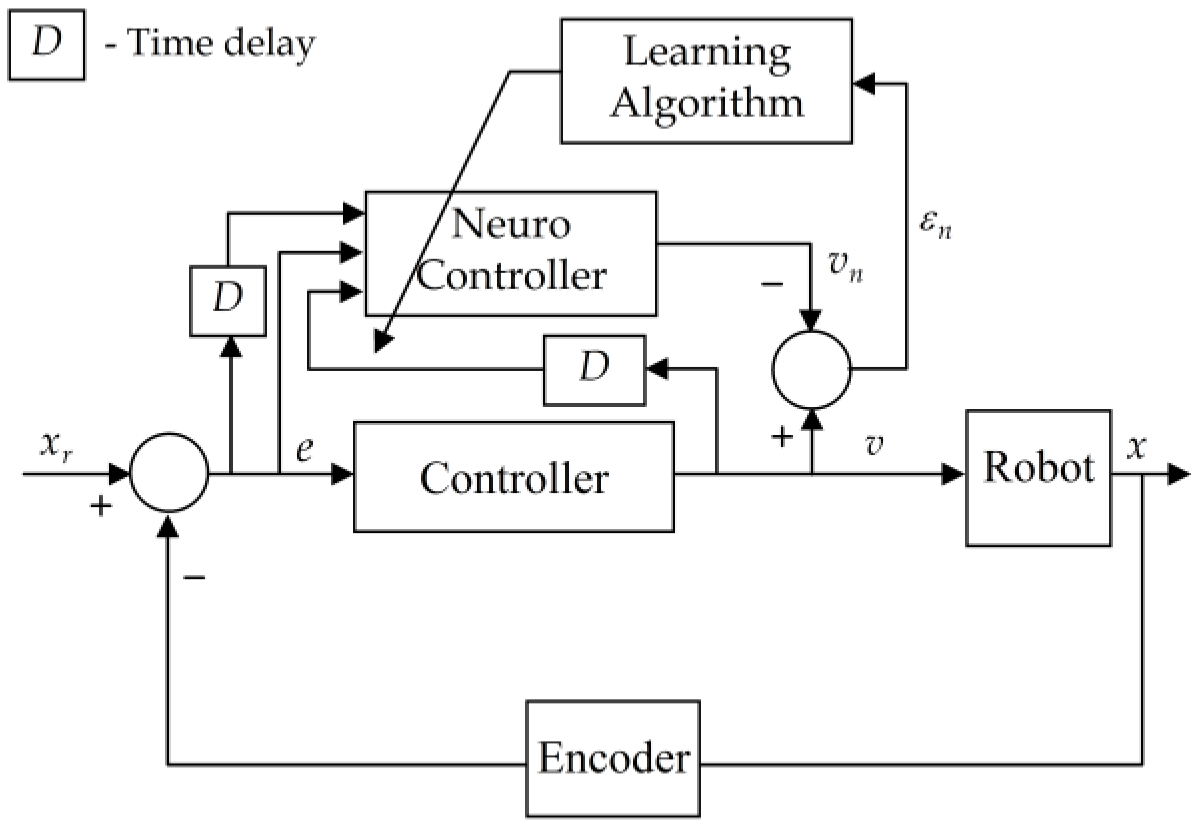
Neural networks can be used to teach robots and cars how to perform various actions, such as navigating by using data from cameras, sensors, or maps; recognizing objects by using data from images, sounds, or text; and interacting with humans by using data from speech, gestures, or emotions. For example, a neural network can learn to steer a car based on the images from a front-facing camera, recognize faces based on the features and expressions of the images, detect traffic signs based on the shape and color of the images, synthesize speech based on the text or context, and respond to gestures based on the movement or intention. This technology can help robots and cars sense their surroundings, plan their routes, identify and classify objects, communicate with humans, and cooperate with them.

A neural network (NN) performs the system model identification that will be used to design the appropriate intelligent mobile robot controller. The usage of NN for controlling a mobile robot is justified from the following reasons: the operational conditions considered raises complex nonholonomic mobile robot kinematics and NN has universal approximation and supervised learning capabilities.

The neural network controller in Fig. 4, based on the recurrent network architecture, has a time-variant feature: once a trajectory is learned, the following learning takes a shorter time.

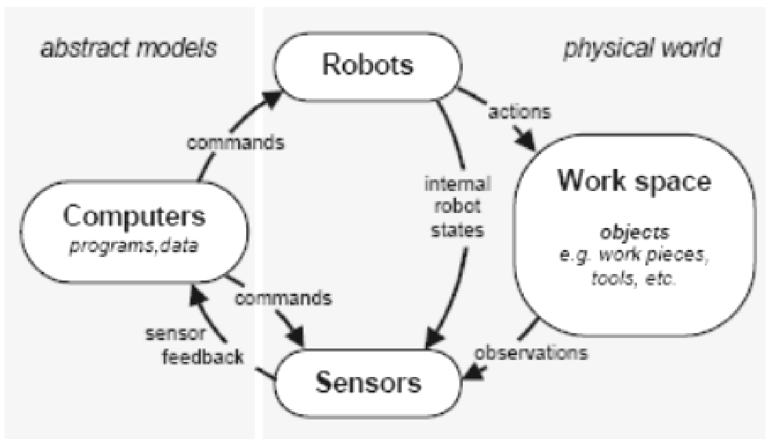
The dynamic neural network is composed of two layered static neural network with feedbacks (one hidden and one output layers) (Fig. 5). The hidden layer contains ten tansigmoidal neurons and the output layer has one neuron with a linear activation function.

It is important to note that rather then learning explicit trajectories, the neural network controller learns the relationship between linear velocities and position errors of the mobile robot. This network is trained using the backpropagation algorithm through time with an adaptive learning rate. In the training phase the network is presented with a series of input-answer pairs.



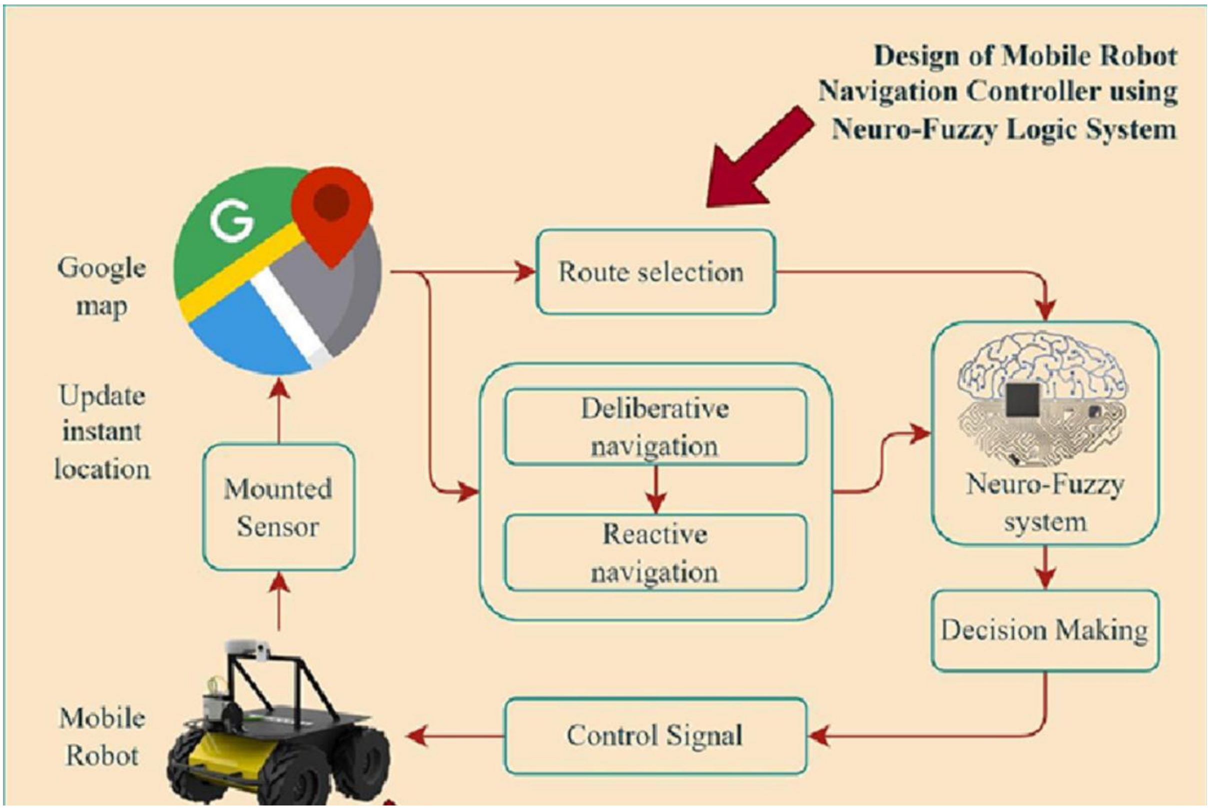
Fuzzy logic in the control of robotic manipulators

Fuzzy logic has been utilized at several hierarchical levels of a typical robotic control system. Four broad levels of application may be identified - task design, system monitoring (including self-tuning and self-organization), information filtering and pre-processing, and in-loop direct control.



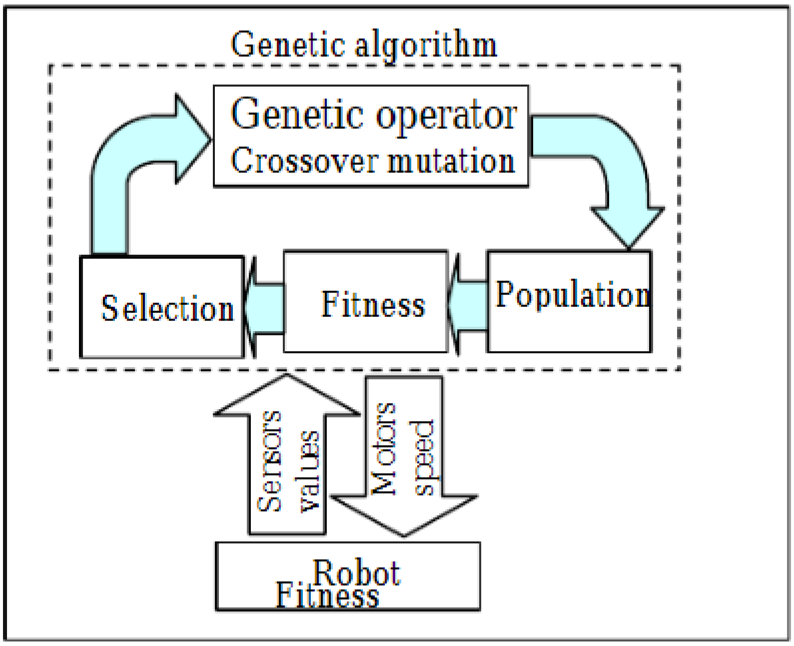
Fuzzy logic is used to keep the robot on the path, except when the danger of collision arises. In this case, a fuzzy controller for obstacle avoidance takes over. Fuzzy control can be used in conjunction with modelling and planning techniques to provide reactive guidance of their robot. Sonar is used by robot to construct a cellular map of its environment. As the robotic modelling has to deal with major uncertainty it is inevitable to keep track of fuzzy valued data in preparing programs for robot controlling. Therefore fuzzy logic plays a vital role in robotics.

Fuzzy control has been applied in robotics to create more adaptive and intelligent robots. The use of fuzzy logic algorithms has enabled robots to make decisions based on *varying conditions*, leading to more efficient and precise operations. For example, fuzzy control can be applied in the control system of a robotic arm to enable it to grasp objects of *varying shapes* and sizes.



To be successful, autonomous navigation must overcome challenges associated with the functions such as detecting perception, current location, planning, and regulating movements. It is possible to have a purposeful and reactive navigational approach. Robots and humans working in the same area must comply with safety laws regarding navigation. As a result, mobile robots in organized environments now have a navigation controller that can operate in reactive and deliberative approaches. Therefore, a neuro-fuzzy system is introduced to explore the benefits of both deliberative and reactive navigation control. It combines neural networks with a fuzzy logic controller and is used to investigate various strategies and give the robot corrective decisional commands. An amigo Bot equipped with a Kinect sensor is used to test the proposed approach in conjunction with the Virtual Robot Experimentation Platform (V-REP) and ROS Groovy Galapagos.

Genetic Algorithm

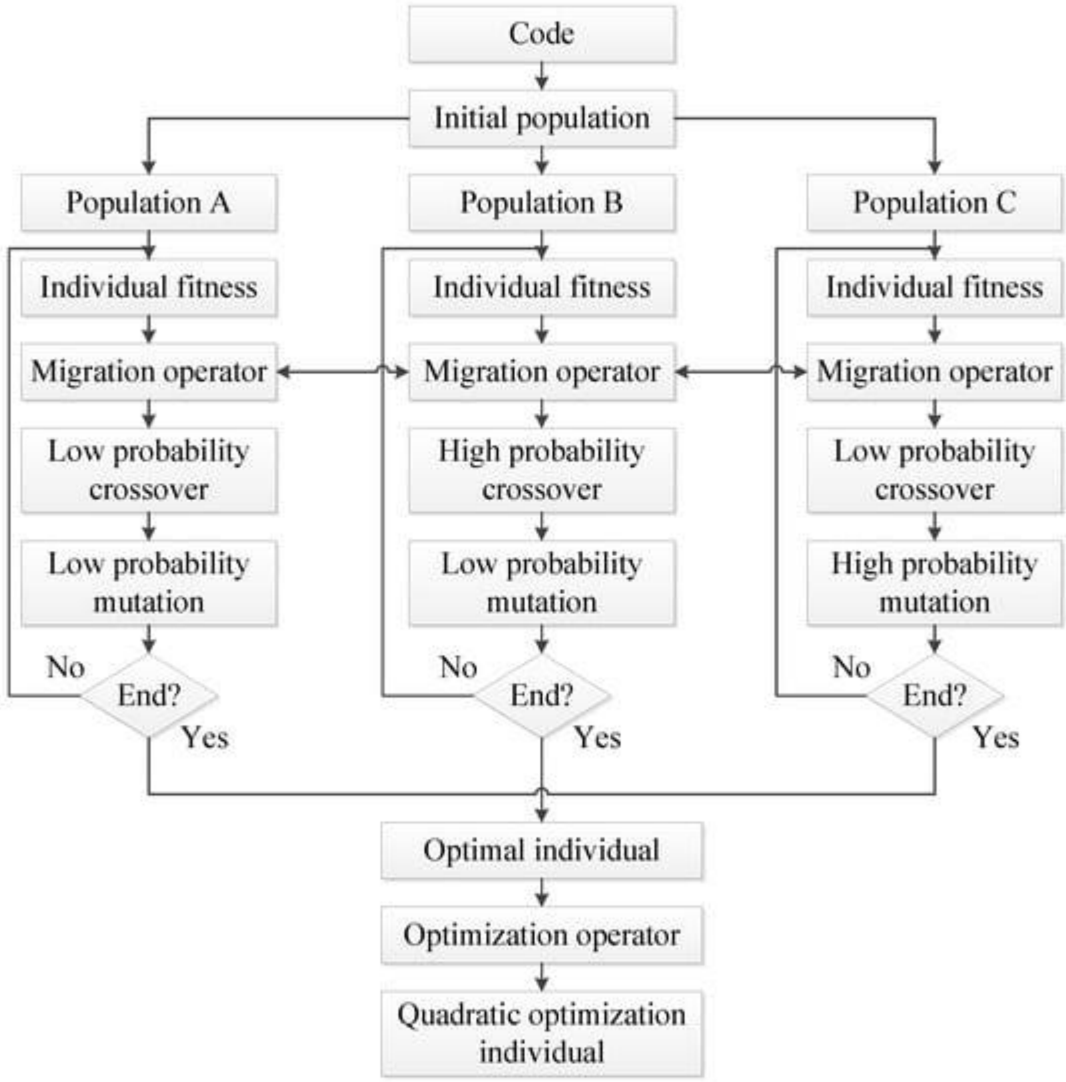


Mobile robot path planning problem is a significant research area in industrial automation, which is to determine an optimal path for a robot to reach the destination by avoiding obstacles. Path planning (PP) is one of the most researched topics in mobile robotics. Deriving an optimal path from a huge number of feasible paths for a given environment is called a PP problem. The existing optimization techniques are used to consider path safety, path length, and path smoothness. The conventional optimization techniques implemented for the mobile robot path planning problem incur a lot of cost due to the high complexity to solve.

In order to find the optimal path for handling the mobile robot path planning problem, the mobile robot path search based on multi-objective genetic algorithm (MRPS-MOGA) is proposed. The MRPS-MOGA is designed with the novelty of genetic algorithm with multiple objective functions to solve mobile robot path planning problems.

Hence the proposed MRPS-MOGA handles five different objectives such as safety, distance, smoothness, travelling time, and collision-free path to obtain optimal path. The MOGA is applied to select an optimal path among multiple as well as feasible paths. The population with feasible paths is initialized with randomly generated paths. The fitness value is evaluated for the number of available candidate paths by applying objective functions for different objectives. Then the fitness criterion determines the paths which are to be passed to participate in the next generation. MRPS-MOGA is developed with the novelty of genetic algorithms such as tournament selection, ring crossover, and adaptive bit string mutation for discovering the optimal path. For the successive generations, the population is selected using the tournament. The genetic operator, crossover operator, is applied for swapping the input string to obtain offspring which is called ring crossover. Consequently, another GA operator mutation is carried out randomly on the sequence to achieve diversity in the population. Again the individual fitness criterion is verified to obtain an optimal path from the population.

MRPS-MOGA is a more efficient mobile robot path with higher safety, reduced energy consumption, lesser traveling time than the existing methods.


Code
Initial population
Population A
Population B
Population C
Individual fitness
Individual fitness
Individual fitness
Migration operator
Migration operator
Migration operator
Low probability crossover
High probability crossover
Low probability crossover
Low probability mutation
Low probability mutation
High probability mutation
No
End?
Yes
No
End?
Yes
No
End?
Yes
Optimal individual
Optimization operator
Quadratic optimization individual