Welcome to:

Components of an Intelligent Robotic System

Unit objectives

After completing this unit, you should be able to:

- Gain knowledge on the basic concepts of robotics and its components
- Gain an insight into the role of machine learning in modern day robotics industry
- Learn about the kinematic and dynamic control concept with a focus on intelligent gripping systems
- Gain an insight into the design and development of robotic components
- Learn about environment capturing sensors like CCD cameras
- Gain knowledge on the integration of sensors with real time robotic system
- Learn about the fuzzy classification and uncertainties in tool condition monitoring system

Introduction to robotics

- Robotics is an inter disciplinary branch of engineering and science that deals with
  - Design.
  - Construction.
  - Operation.
  - Use of robots.
  - Computer systems for their control.
  - Sensory feedback.
  - Information processing.
- Robotics is a multi-disciplinary domain it comprises of:
  - Mechanical Engineering- deals with the machinery and structure of the robots.
  - Electrical Engineering - deals with the controlling and sensing of robots.
  - Computer Engineering -deals with the movement development and observation of robots.
- The robotics technologies are used to develop automated systems. It can be a replace for humans and imitate human actions.
- Uses of Robots:
  - Manufacturing processes.
  - Where humans cannot survive (e.g. in space).
  - An attempt to replicate walking, lifting, speech, cognition, and basically anything a human can do.
  - Contribute to the field of bio-inspired robotics.

Types of robots

- The types of robots are:
  - Outer space.
  - The intelligent home.
  - Exploration.
  - Military robots.
  - Farms robots.
  - The car industry.
  - Hospitals.
  - Disaster areas.
  - Entertainment.

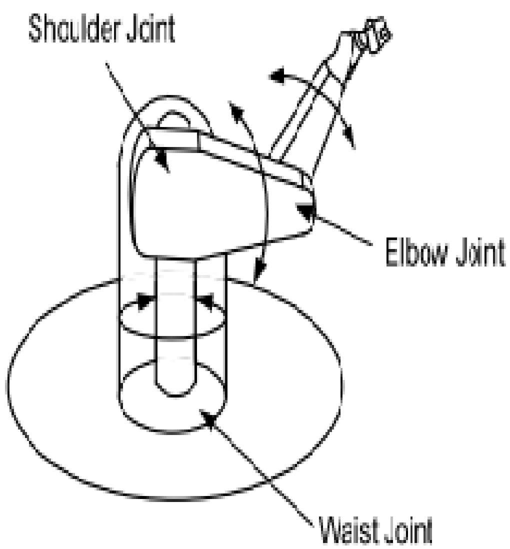
Classification of robots

- Applications:
  - Industrial.
  - Domestic or household robots.
  - Medical robots.
  - Service robots.
  - Military robots.
  - Entertainment robots.
  - Space robots.

- Robotics life:
  - Robots were invented by the humans.
  - Assist humans in various sectors.
  - Good for dreary, recurring tasks.
  - Make living more convenient.

Components of robot (1 of 4)

- Articulated arm or RRR robot - robot having 3 revolute joints and it mimics the human arm.



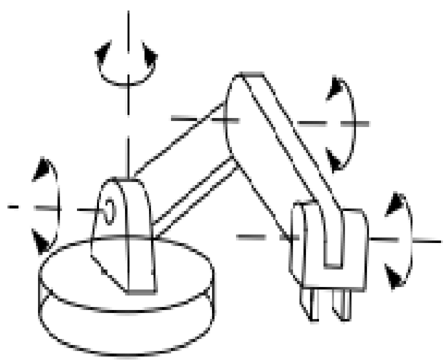


Figure: A schematic diagram of revolute robot

Figure: A revolute robot

Components of robot (2 of 4)

- Cartesian or PPP robot:
  - Simplest industrial manipulator.
  - Contains three prismatic joints.

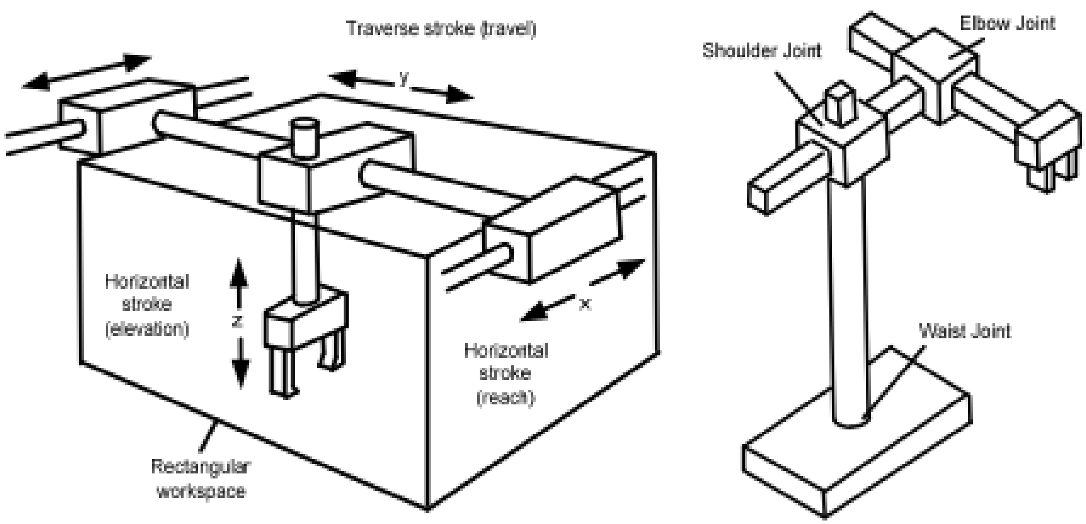


Figure: A schematic diagram of Cartesian (PPP) robot

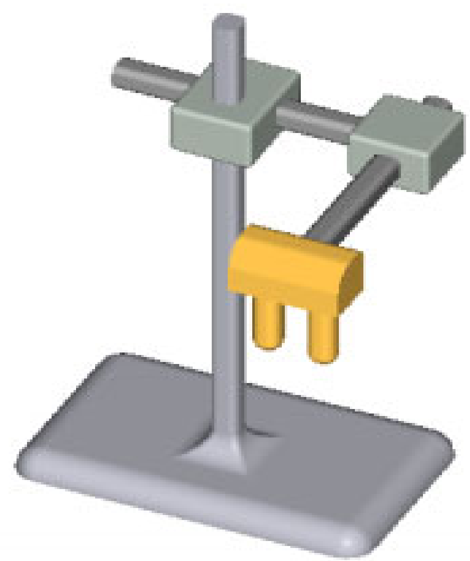


Figure: A Cartesian robot

Components of robot (3 of 4)

- Cylindrical or PRP Robot
  - Replacing a joint of a Cartesian arm by a revolute joint.
  - A cylindrical geometry arm can be formed.
  - Has a cylindrical workspace.

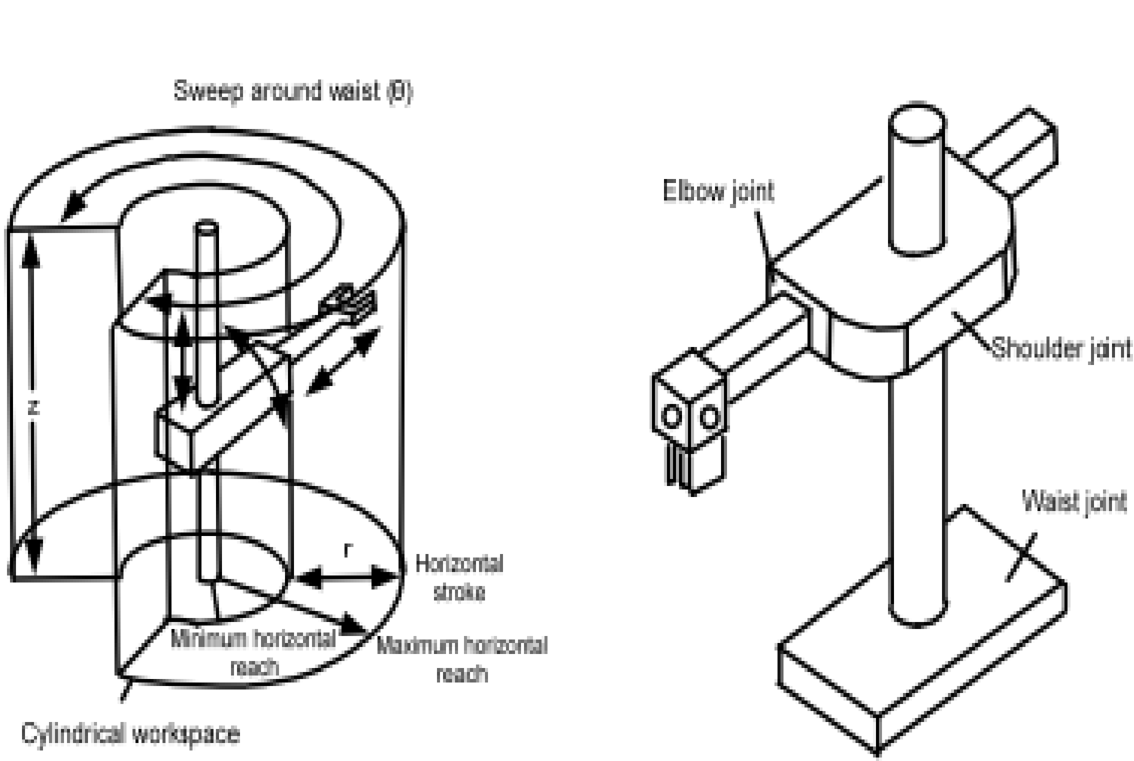


Figure: A schematic diagram of cylindrical robot

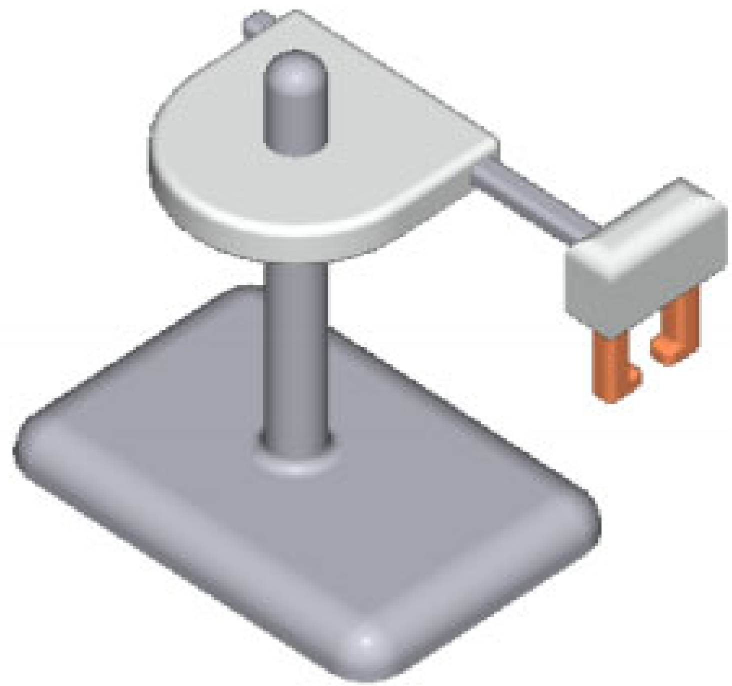
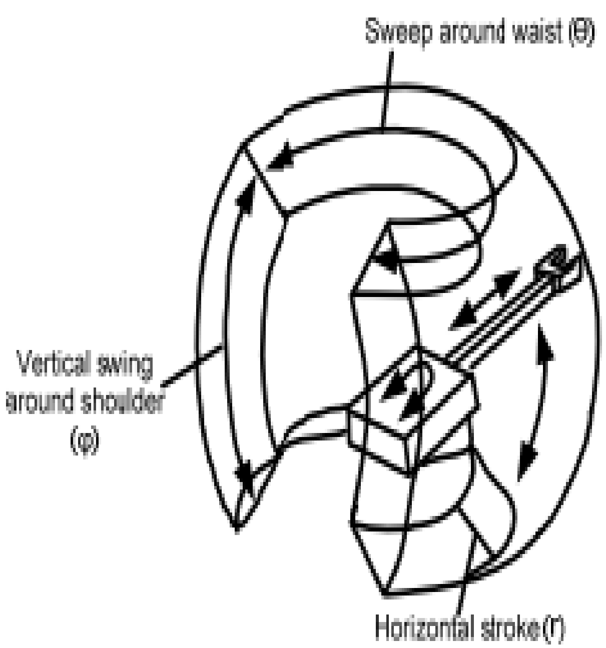
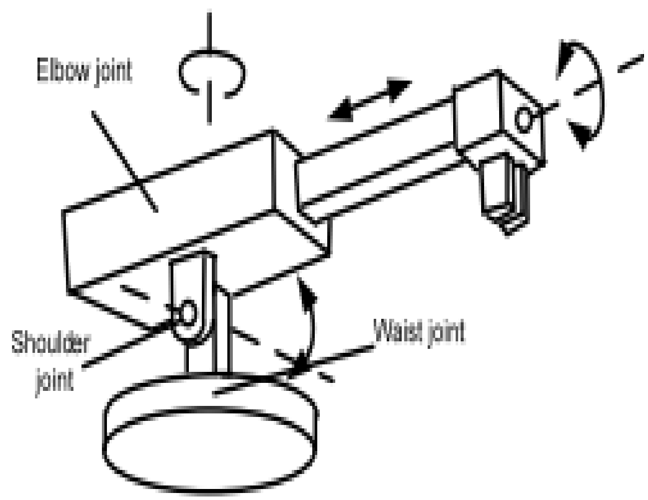


Figure: A cylindrical robot

Components of robot (4 of 4)

- Polar or RRP Robot
  - A revolute joint substitutes another prismatic joint then a polar geometry is formed.
  - Has a spherical workspace.





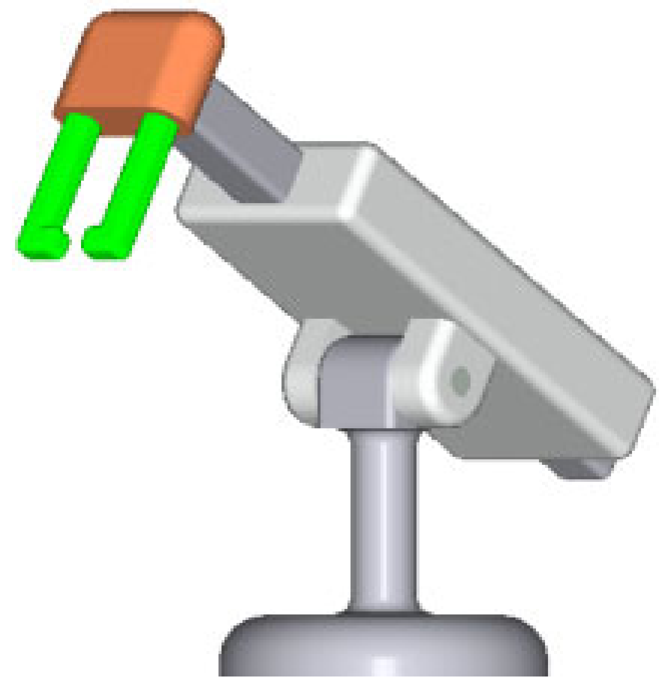


Figure: A schematic diagram of spherical robot

Figure: A spherical robot

Manipulation arms

- Robot arm:
  - Consists of joints and links.
  - Joints connect different parts/links of the robot and they can slide/rotate.

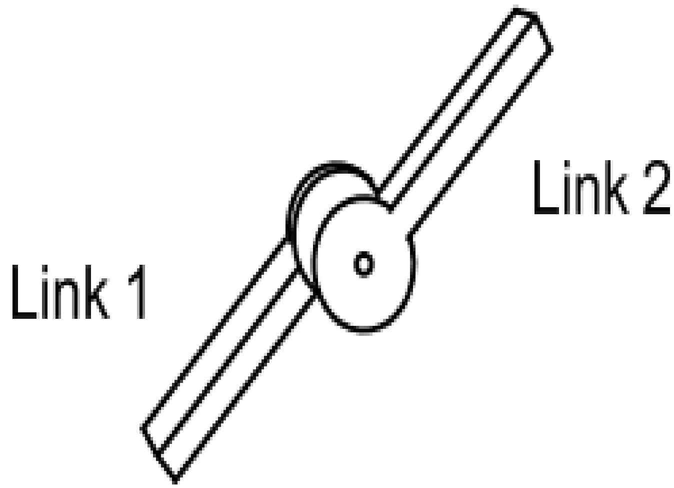


Figure: A revolute joint

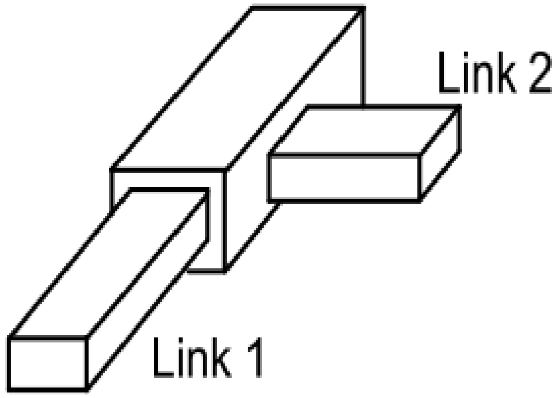


Figure: A prismatic join

Merits and demerits of robot types with different geometries

Robot	Coordinates	Merits and Demerits
PPP or Cartesian	x y z	Linear Motion in 3D Simple kinematics model Rigid structure Easy to visualize Can use inexpensive pneumatic drives for pick and place operation Requires large volume to operate in Workspace is smaller than robot volume Unable to reach areas under objects Guiding surfaces of prismatic joints must be covered to prevent ingress of dust.
PRP or Cylindrical	z θ r	Simple kinematic model Easy to visualize Good access into cavities and machine openings Very powerful when hydraulic drives are used Restricted workspace Prismatic guides difficult to seal from dust and liquids Back of robot can overlap work volume
RRP or Spherical	θ φ r	Covers a large area from a central support Can bend down to pick objects off the floor Complex kinematic model Difficult to visualize
RRR or Articulated	θ_1 θ_2 θ_3	Maximum flexibility Covers large area of work relative to volume of robots Revolute joints are easy to seal Suits electrical motors Can reach over and under objects Complex kinematics Difficult to visualize Control of linear motions is difficult Structure not very rigid at full reach

Wrists

- A robot arm has a wrist to orient the end effectors.

- Wrist has three degrees of freedom:
  - roll (z axis rotation),
  - pitch (y axis rotation).
  - yaw (axis rotation.

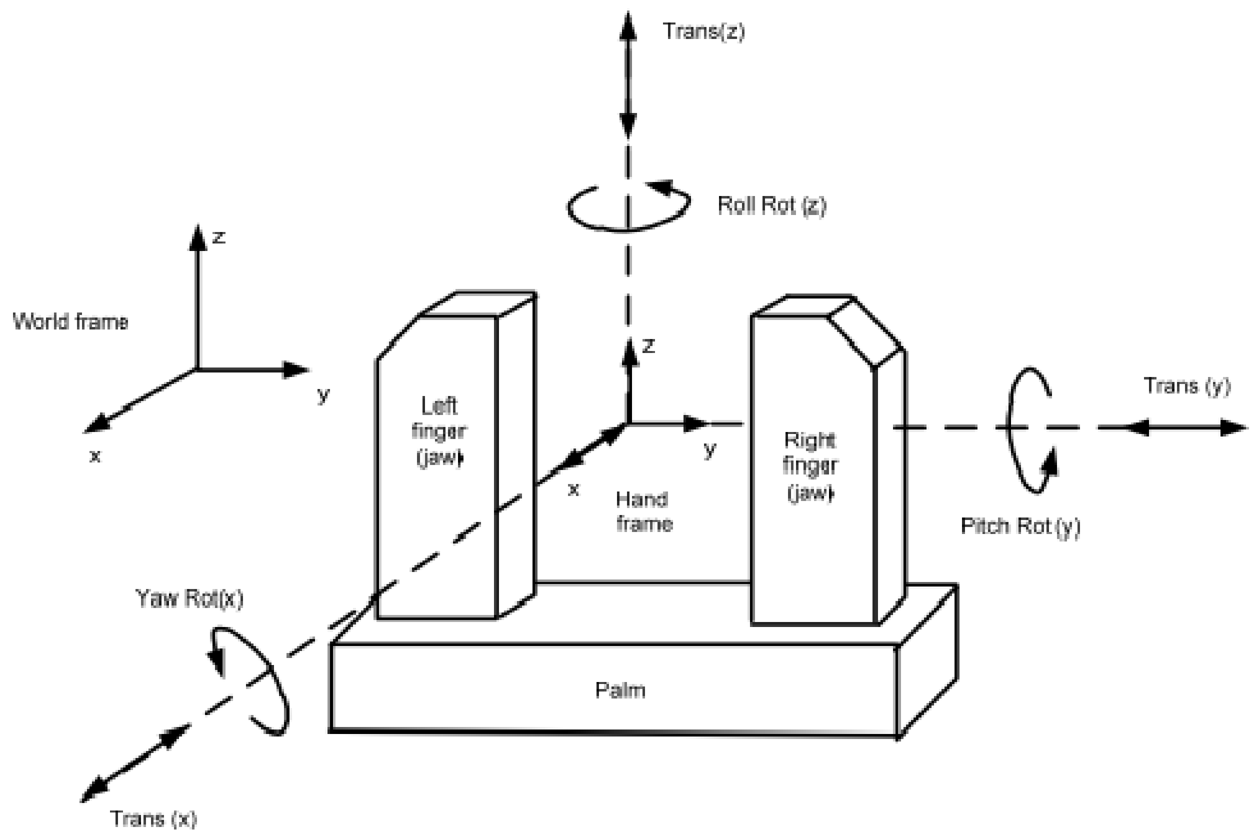


Figure: Wrists six motions for gripping.

Robot kinematics (1 of 3)

- Robot kinematics refers the analytical study of the motion of a robot manipulator.

- Two different spaces used in kinematics modelling of manipulators are:
  - Cartesian space and quaternion space.

- The transformation between two Cartesian coordinate systems can be decomposed into a rotation and a translation.

- Some of the ways to represent rotation:
  - Euler angles.
  - Gibbs vector.
  - Pauli spin matrices.
  - Hamilton's quaternions.

- Homogenous transformations based on 4x4 real matrices (orthonormal matrices) have been used most often in robotics.

Robot kinematics (2 of 3)

- It is proved that a general transformation between two joints requires four parameters. These parameters known as the Denavit-Hartenberg (DH) parameters have become the standard for describing robot kinematics.

- While the orientation of a body is represented nine elements in homogenous transformations, the dual quaternions reduce the number of elements to four. It offers considerable advantage in terms of computational robustness and storage efficiency for dealing with the kinematics of robot chains.

- The robot kinematics is studied as:
  - Forward kinematics and inverse kinematics.

- The forward kinematics problem is quite simple and there is no complexity in deriving the equations.

- Hence, there is always a forward kinematics solution of a manipulator.

- Inverse kinematics is a much more difficult problem than forward kinematics.

- The solution of the inverse kinematics problem is computationally expensive and generally consume more time in the real time control of manipulators.

Robot kinematics (3 of 3)

- The relationship between forward and inverse kinematics:

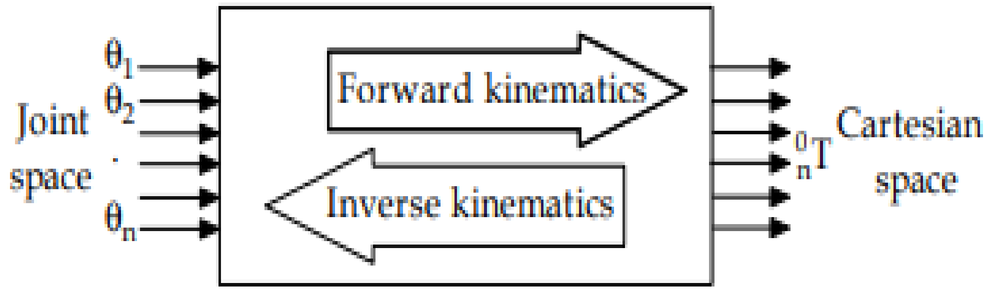


Figure: The schematic representation of forward and inverse kinematics.

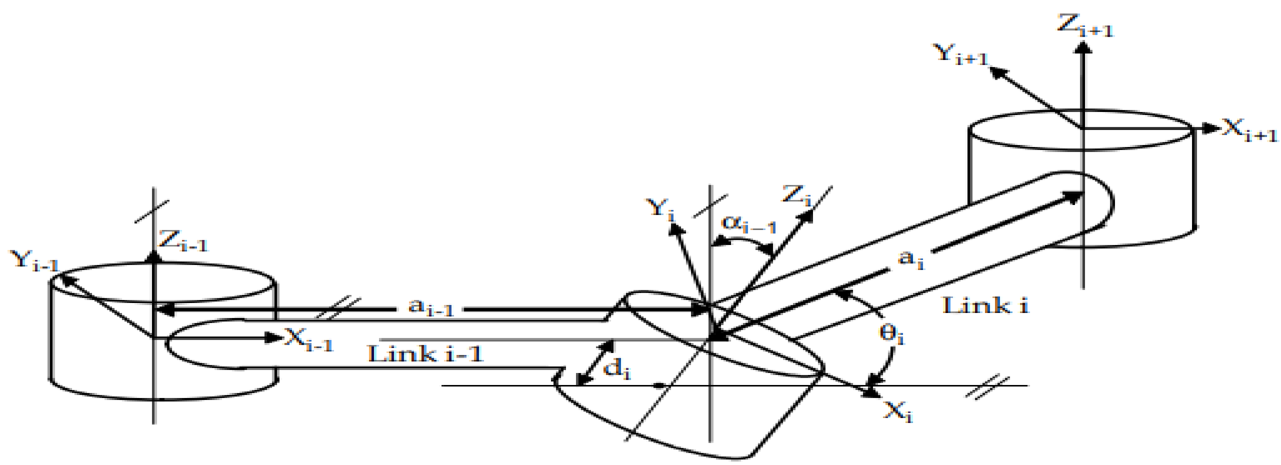
- The two major solutions for the inverse kinematics problem are:
  - The *analytical* and *numerical* methods
- In analytical: The joint variables are solved analytically according to the given configuration-data.
- In numerical: The joint variables are obtained founded on the numerical techniques.
- Two solutions in analytical method:
  - Geometric and algebraic solutions.
- Geometric approach is applied to the simple robot structures (2-DOF planar manipulator or less DOF manipulator with parallel joint axes)
- Algebraic approach is explored for the inverse kinematics solution:
  - In the cases, For the manipulators with more links and whose arms extend into 3 dimensions or more.

Homogenous transformation modelling convention (1 of 2)

IBM ICE (Innovation Centre for Education)

Forward Kinematics:

- The coordinate frame assignment for a general manipulator:



- The general transformation matrix for a single link can be obtained as follows:

$$
{}^{i-1}_{\ \ i}T = R_x(\alpha_{i-1})D_x(a_{i-1})R_z(\theta_i)Q_i(d_i)
$$

$$
= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_{i-1} & -s\alpha_{i-1} & 0 \\ 0 & s\alpha_{i-1} & c\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Homogenous transformation modelling convention (2 of 2)

- The forward kinematics of the end-effector with respect to the base frame is determined by multiplying all of the matrices. $^{base}_{end\_effector}T = {}^{0}_{1}T\,{}^{1}_{2}T \ldots {}^{n-1}_{n}T$

- An alternative representation of $^{base}_{end\ effector}T$ can be written as:

$$^{base}_{end-effector}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

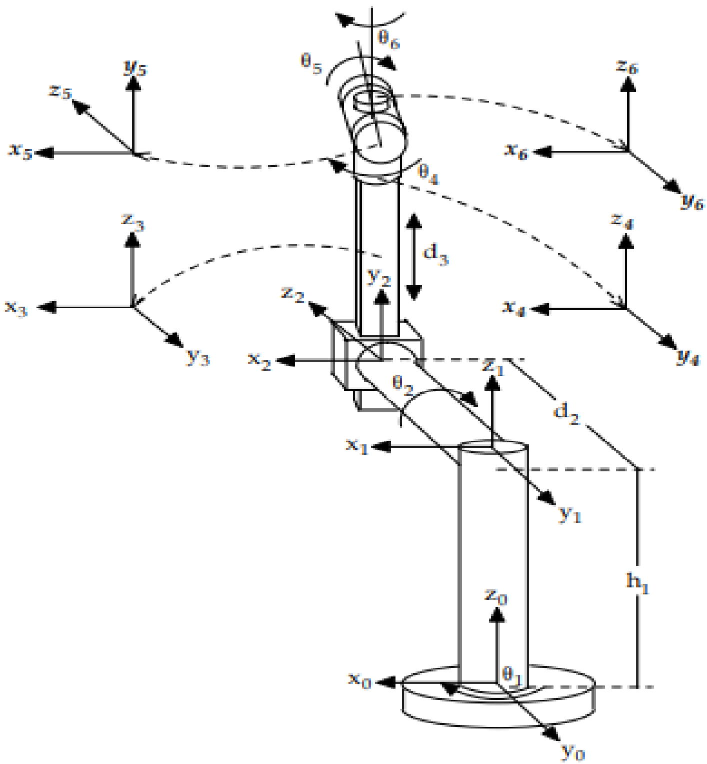
  – Where
  – r_{kj}'s - rotational elements of transformation matrix (k and j=1, 2 and 3).
  – p_x, p_y and p_z - elements of the position vector.

- For a six jointed manipulator, the position and orientation of the end-effector with respect to the base is given by:

$$^{0}_{6}T = {}^{0}_{1}T(q_1)\,{}^{1}_{2}T(q_2)\,{}^{2}_{3}T(q_3)\,{}^{3}_{4}T(q_4)\,{}^{4}_{5}T(q_5)\,{}^{5}_{6}T(q_6)$$

where q_i is the joint variable (revolute or prismatic joint) for joint i, (i=1, 2, ...6).

Example of forward kinematics (1 of 4)



i	q_i	ai-1	ai-1	d_i
1	q_1	0	0	h_1
2	q_2	90	0	d_2
3	0	-90	0	d_3
4	q_4	0	0	0
5	q_5	90	0	0
6	q_6	-90	0	0

Figure: Rigid body and coordinate frame assignment for the Stanford Manipulator.

Table: DH parameters for the Stanford Manipulator.

Example of forward kinematics (2 of 4)

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & h_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.3) $$

$$ {}^{3}_{4}T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.4) $$

$$ {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.5) $$

$$ {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.6) $$

$$ {}^{4}_{5}T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.7) $$

$$ {}^{5}_{6}T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.8) $$

Example of forward kinematics (3 of 4)

$$ {}^{0}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \ldots(4.9) $$

where

$$
\begin{aligned}
r_{11} &= -s\theta_6(c\theta_4 s\theta_1 + c\theta_1 c\theta_2 s\theta_4) - c\theta_6(c\theta_5(s\theta_1 s\theta_4 - c\theta_1 c\theta_2 c\theta_4) + c\theta_1 s\theta_2 s\theta_5) \\
r_{12} &= s\theta_6(c\theta_5(s\theta_1 s\theta_4 - c\theta_1 c\theta_2 c\theta_4) + c\theta_1 s\theta_2 s\theta_5) - c\theta_6(c\theta_4 s\theta_1 + c\theta_1 c\theta_2 s\theta_4) \\
r_{13} &= s\theta_5(s\theta_1 s\theta_4 - c\theta_1 c\theta_2 c\theta_4) - c\theta_1 c\theta_5 s\theta_2 \\
r_{21} &= s\theta_6(c\theta_1 c\theta_4 - c\theta_2 s\theta_1 s\theta_4) + c\theta_6(c\theta_5(c\theta_1 s\theta_4 + c\theta_2 c\theta_4 s\theta_1) - s\theta_1 s\theta_2 s\theta_5) \\
r_{22} &= c\theta_6(c\theta_1 c\theta_4 - c\theta_2 s\theta_1 s\theta_4) - s\theta_6(c\theta_5(c\theta_1 s\theta_4 + c\theta_2 c\theta_4 s\theta_1) - s\theta_1 s\theta_2 s\theta_5) \\
r_{23} &= -s\theta_5(c\theta_1 s\theta_4 + c\theta_2 c\theta_4 s\theta_1) - c\theta_5 s\theta_1 s\theta_2 \\
r_{31} &= c\theta_6(c\theta_2 s\theta_5 + c\theta_4 c\theta_5 s\theta_2) - s\theta_2 s\theta_4 s\theta_6 \\
r_{32} &= -s\theta_6(c\theta_2 s\theta_5 + c\theta_4 c\theta_5 s\theta_2) - c\theta_6 s\theta_2 s\theta_4 \\
r_{33} &= c\theta_2 c\theta_5 - c\theta_4 s\theta_2 s\theta_5 \\
p_x &= d_2 s\theta_1 - d_3 c\theta_1 s\theta_2 \\
p_y &= -d_2 c\theta_1 - d_3 s\theta_1 s\theta_2 \\
p_z &= h_1 + d_3 c\theta_2
\end{aligned}
$$

Example of forward kinematics (4 of 4)

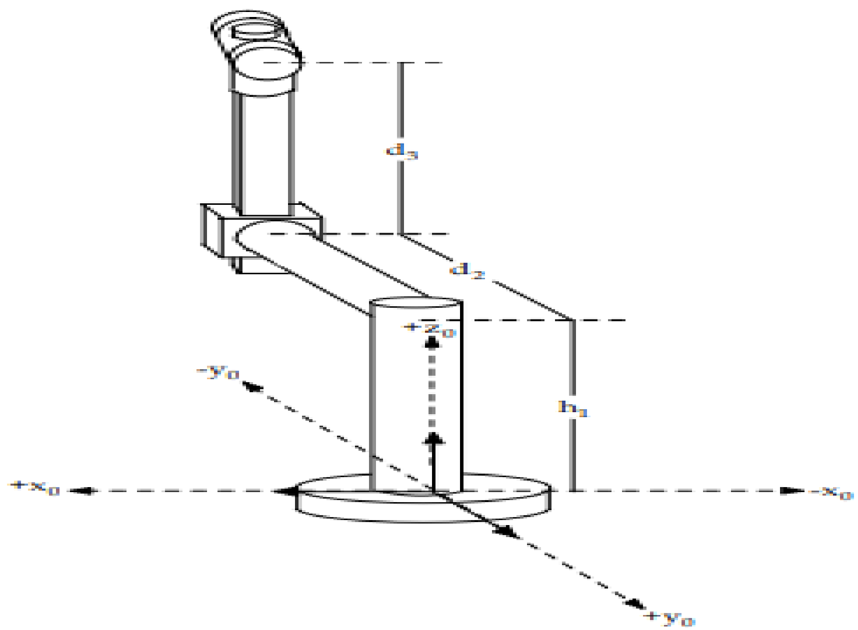


Figure: Zero position for the Stanford Manipulator.

- In order to obtain the zero position in terms of link parameters, set $q_1=q_2=0°$ in equation 4.10.

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} d_2 s\theta_1 - d_3 c\theta_1 s\theta_2 \\ -d_2 c\theta_1 - d_3 s\theta_1 s\theta_2 \\ h_1 + d_3 c\theta_2 \end{bmatrix} \quad \ldots(4.10)$$

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} d_2 s(0^\circ) - d_3 c(0^\circ) s(0^\circ) \\ -d_2 c(0^\circ) - d_3 s(0^\circ) s(0^\circ) \\ h_1 + d_3 c(0^\circ) \end{bmatrix} = \begin{bmatrix} 0 \\ -d_2 \\ h_1 + d_3 \end{bmatrix} \quad \ldots(4.11)$$

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} 0 \\ -d_2 \\ h_1 + d_3 \end{bmatrix} \quad \ldots(4.12)$$

Inverse kinematics (1 of 6)

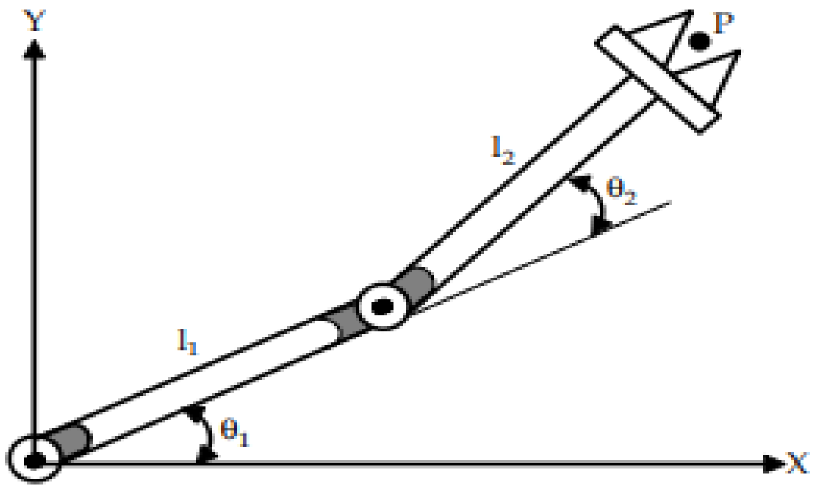


Figure (a): Planner manipulator

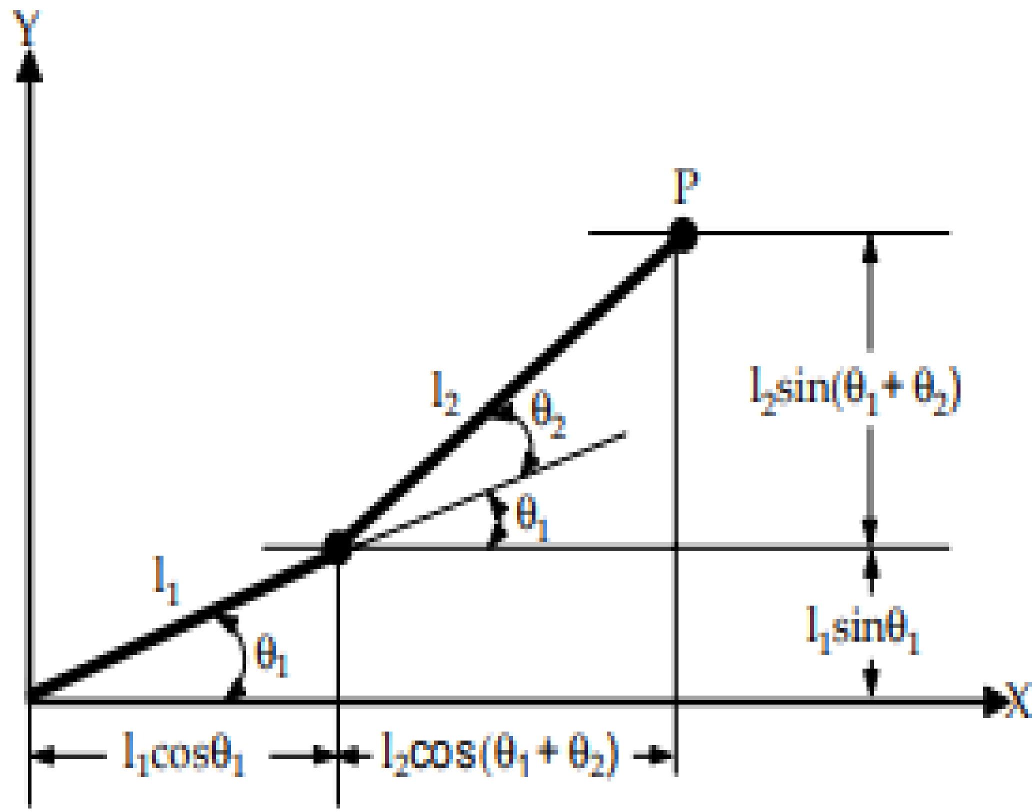


Figure (b): Solving the inverse kinematics based on trigonometry

Inverse kinematics (2 of 6)

- The components of the point P (p_x and p_y) are determined as follows:

$$p_x = l_1 c\theta_1 + l_2 c\theta_{12} \qquad \ldots(4.13)$$

$$p_y = l_1 s\theta_1 + l_2 s\theta_{12} \qquad \ldots(4.14)$$

where $c\theta_{12} = c\theta_1 c\theta_2 - s\theta_1 s\theta_2$ and $s\theta_{12} = s\theta_1 c\theta_2 + c\theta_1 s\theta_2$

$$\begin{aligned}
p_x^2 &= l_1^2 c^2\theta_1 + l_2^2 c^2\theta_{12} + 2l_1 l_2 c\theta_1 c\theta_{12} \\
p_y^2 &= l_1^2 s^2\theta_1 + l_2^2 s^2\theta_{12} + 2l_1 l_2 s\theta_1 s\theta_{12} \qquad \ldots(4.15) \\
p_x^2 + p_y^2 &= l_1^2(c^2\theta_1 + s^2\theta_1) + l_2^2(c^2\theta_{12} + s^2\theta_{12}) + 2l_1 l_2(c\theta_1 c\theta_{12} + s\theta_1 s\theta_{12})
\end{aligned}$$

$$\begin{aligned}
p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2(c\theta_1[c\theta_1 c\theta_2 - s\theta_1 s\theta_2] + s\theta_1[s\theta_1 c\theta_2 + c\theta_1 s\theta_2]) \\
p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2(c^2\theta_1 c\theta_2 - c\theta_1 s\theta_1 s\theta_2 + s^2\theta_1 c\theta_2 + c\theta_1 s\theta_1 s\theta_2) \qquad \ldots(4.16) \\
p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2(c\theta_2[c^2\theta_1 + s^2\theta_1]) \\
p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2 c\theta_2
\end{aligned}$$

Inverse kinematics (3 of 6)

- And hence,

$$c\theta_2 = \frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \qquad \ldots(4.17)$$

Since, $c^2\theta_i + s^2\theta_i = 1 \ (i = 1,2,3,\ldots\ldots)$, $s\theta_2$ is obtained as

$$s\theta_2 = \pm\sqrt{1 - \left(\frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2}\right)^2}$$

$$\theta_2 = A\tan 2\left(\pm\sqrt{1 - \left(\frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2}\right)^2}, \frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2}\right) \qquad \ldots(4.18)$$

$$\begin{aligned}
c\theta_1 p_x &= l_1 c^2\theta_1 + l_2 c^2\theta_1 c\theta_2 - l_2 c\theta_1 s\theta_1 s\theta_2 \\
s\theta_1 p_y &= l_1 s^2\theta_1 + l_2 s^2\theta_1 c\theta_2 + l_2 s\theta_1 c\theta_1 s\theta_2 \qquad \ldots(4.19) \\
c\theta_1 p_x + s\theta_1 p_y &= l_1(c^2\theta_1 + s^2\theta_1) + l_2 c\theta_2 (c^2\theta_1 + s^2\theta_1)
\end{aligned}$$

$$c\theta_1 p_x + s\theta_1 p_y = l_1 + l_2 c\theta_2 \qquad \ldots(4.20)$$

Inverse kinematics (4 of 6)

- In this step, multiply both sides of equation 4.13 by $-s\theta_1$ and equation 4.14 by $c\theta_1$ and then adding the resulting equations produce.

$$-s\theta_1 p_x = -l_1 s\theta_1 c\theta_1 - l_2 s\theta_1 c\theta_1 c\theta_2 + l_2 s^2\theta_1 s\theta_2$$

$$c\theta_1 p_y = l_1 s\theta_1 c\theta_1 + l_2 c\theta_1 s\theta_1 c\theta_2 + l_2 c^2\theta_1 s\theta_2$$

$$-s\theta_1 p_x + c\theta_1 p_y = l_2 s\theta_2 (c^2\theta_1 + s^2\theta_1)$$

$$-s\theta_1 p_x + c\theta_1 p_y = l_2 s\theta_2 \qquad \dots(4.21)$$

$$c\theta_1 p_x^2 + s\theta_1 p_x p_y = p_x(l_1 + l_2 c\theta_2)$$

$$-s\theta_1 p_x p_y + c\theta_1 p_y^2 = p_y l_2 s\theta_2$$

$$c\theta_1 (p_x^2 + p_y^2) = p_x(l_1 + l_2 c\theta_2) + p_y l_2 s\theta_2$$

$$c\theta_1 = \frac{p_x(l_1 + l_2 c\theta_2) + p_y l_2 s\theta_2}{p_x^2 + p_y^2} \qquad \dots(4.22)$$

$$s\theta_1 = \pm\sqrt{1 - \left(\frac{p_x(l_1 + l_2 c\theta_2) + p_y l_2 s\theta_2}{p_x^2 + p_y^2}\right)^2} \qquad \dots(4.23)$$

$$\theta_1 = A\tan 2\left(\pm\sqrt{1 - \left(\frac{p_x(l_1 + l_2 c\theta_2) + p_y l_2 s\theta_2}{p_x^2 + p_y^2}\right)^2}, \frac{p_x(l_1 + l_2 c\theta_2) + p_y l_2 s\theta_2}{p_x^2 + p_y^2} \right) \qquad \dots(4.24)$$

Inverse kinematics (5 of 6)

Algebraic solution approach:

- Consider the equation 4.9 to find the inverse kinematics solution for a six-axis manipulator.

$$ {}^{0}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{0}_{1}T(q_1)\ {}^{1}_{2}T(q_2)\ {}^{2}_{3}T(q_3)\ {}^{3}_{4}T(q_4)\ {}^{4}_{5}T(q_5)\ {}^{5}_{6}T(q_6) $$

- To find inverse kinematics solution for first joint (q_1) as a function of the known elements of ${}^{base}_{end\,effector}T$

- The link transformation inverses are pre-multiplied as follows:

$$ \left[{}^{0}_{1}T(q_1)\right]^{-1}\ {}^{0}_{6}T = \left[{}^{0}_{1}T(q_1)\right]^{-1}\ {}^{0}_{1}T(q_1)\,{}^{1}_{2}T(q_2)\ {}^{2}_{3}T(q_3)\ {}^{3}_{4}T(q_4)\ {}^{4}_{5}T(q_5)\ {}^{5}_{6}T(q_6) $$

$$ \left[{}^{0}_{1}T(q_1)\right]^{-1}\ {}^{0}_{1}T(q_1) = I $$ where I is the identity matrix.

- In this case the above equation is given by:

$$ \left[{}^{0}_{1}T(q_1)\right]^{-1}\ {}^{0}_{6}T = {}^{1}_{2}T(q_2)\ {}^{2}_{3}T(q_3)\ {}^{3}_{4}T(q_4)\ {}^{4}_{5}T(q_5)\ {}^{5}_{6}T(q_6) \qquad \ldots(4.25) $$

Inverse kinematics (6 of 6)

- To find the other variables,

$$\left[{}^0_1T(q_1)\,{}^1_2T(q_2)\right]^{-1}\,{}^0_6T = {}^2_3T(q_3)\,{}^3_4T(q_4)\,{}^4_5T(q_5)\,{}^5_6T(q_6) \quad \ldots(4.26)$$

$$\left[{}^0_1T(q_1)\,{}^1_2T(q_2)\,{}^2_3T(q_3)\right]^{-1}\,{}^0_6T = {}^3_4T(q_4)\,{}^4_5T(q_5)\,{}^5_6T(q_6) \quad \ldots(4.27)$$

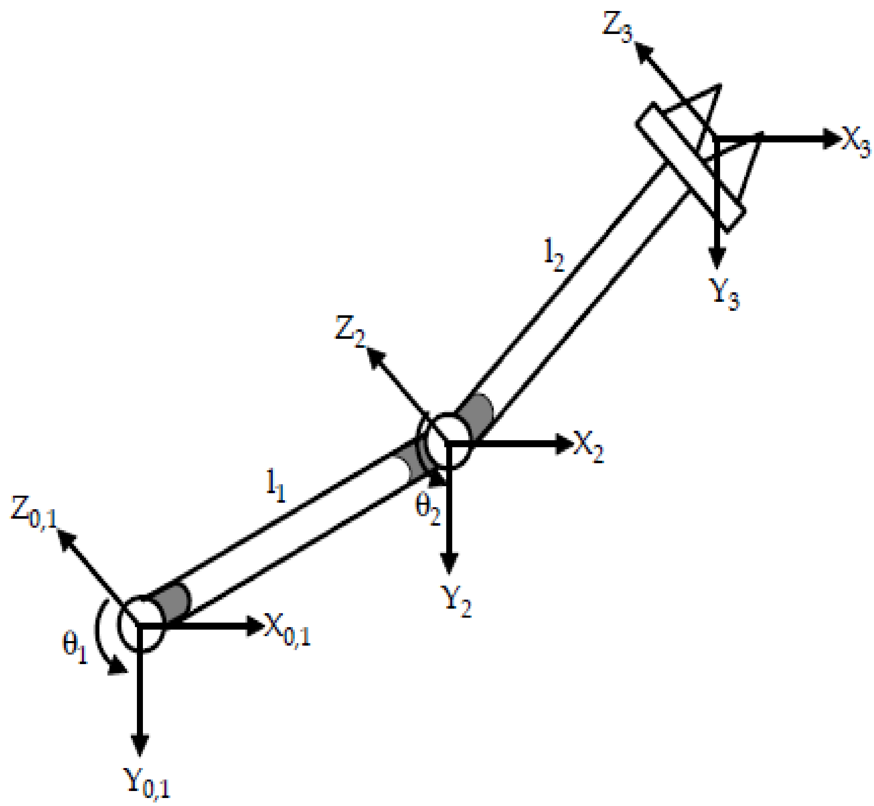
$$\left[{}^0_1T(q_1)\,{}^1_2T(q_2)\,{}^2_3T(q_3)\,{}^3_4T(q_4)\right]^{-1}\,{}^0_6T = {}^4_5T(q_5)\,{}^5_6T(q_6) \quad \ldots(4.28)$$

$$\left[{}^0_1T(q_1)\,{}^1_2T(q_2)\,{}^2_3T(q_3)\,{}^3_4T(q_4)\,{}^4_5T(q_5)\right]^{-1}\,{}^0_6T = {}^5_6T(q_6) \quad \ldots(4.29)$$

- Table for some trigonometric equations and solutions used in inverse kinematics.

	Equations	Solutions
1	$a\sin\theta + b\cos\theta = c$	$\theta = A\tan 2(a, b) \mp A\tan 2\left(\sqrt{a^2 + b^2 - c^2}, c\right)$
2	$a\sin\theta + b\cos\theta = 0$	$\theta = A\tan 2(-b, a)$ or $\theta = A\tan 2(b, -a)$
3	$\cos\theta = a$ and $\sin\theta = b$	$\theta = A\tan 2(b, a)$
4	$\cos\theta = a$	$\theta = A\tan 2\left(\mp\sqrt{1-a^2}, a\right)$
5	$\sin\theta = a$	$\theta = A\tan 2\left(a, \mp\sqrt{1-a^2}\right)$

Algebraic solution approach: Example (1 of 4)



i	θ_i	α_{i-1}	a_{i-1}	d_i
1	θ_1	0	0	0
2	θ_2	0	l_1	0
3	0	0	l_2	0

Figure: Coordinate frame assignment for the planar manipulator.

Table: DH parameters for the planar manipulator.

Algebraic solution approach: Example (2 of 4)

- The link transformation matrices are given by:

$$ {}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{0}_{3}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T $$

…(4.30) …(4.31) …(4.32) …(4.33)

- Multiply each side of equation 4.33 by ${}^{0}_{1}T^{-1}$.

$$ {}^{0}_{1}T^{-1}\,{}^{0}_{3}T = {}^{1}_{2}T\,{}^{2}_{3}T \qquad \text{…(4.34)} $$

$$ {}^{0}_{1}T^{-1} = \begin{bmatrix} {}^{0}_{1}R^{T} & -{}^{0}_{1}R^{T}\,{}^{0}P_1 \\ 0\,0\,0 & 1 \end{bmatrix} \qquad \text{…(4.33)} $$

- In equation 4.35, and denote the transpose of rotation and position vector of respectively.
- Since, ${}^{0}_{1}T^{-1}$ =I , equation 4.34 can be rewritten as follows:

$$ {}^{0}_{1}T^{-1}\,{}^{0}_{3}T = {}^{1}_{2}T\,{}^{2}_{3}T \qquad \text{…(4.36)} $$

Algebraic solution approach: Example (3 of 4)

- Substituting the link transformation matrices into equation yields:

$$\begin{bmatrix} c\theta_1 & s\theta_1 & 0 & 0 \\ -s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.37)$$

$$\begin{bmatrix} . & . & . & c\theta_1 p_x + s\theta_1 p_y \\ . & . & . & -s\theta_1 p_x + c\theta_1 p_y \\ . & . & . & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & l_2 c\theta_2 + l_1 \\ . & . & . & l_2 s\theta_2 \\ . & . & . & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Squaring the (1, 4) and (2, 4) matrix elements of each side in equation 4.37.

$$c^2\theta_1 p_x^2 + s^2\theta_1 p_y^2 + 2p_x p_y c\theta_1 s\theta_1 = l_2^2 c^2\theta_2 + 2l_1 l_2 c\theta_2 + l_1^2$$

$$s^2\theta_1 p_x^2 + c^2\theta_1 p_y^2 - 2p_x p_y c\theta_1 s\theta_1 = l_2^2 s^2\theta_2$$

- And then adding the resulting equations above gives:

$$p_x^2(c^2\theta_1 + s^2\theta_1) + p_y^2(s^2\theta_1 + c^2\theta_1) = l_2^2(c^2\theta_2 + s^2\theta_2) + 2l_1 l_2 c\theta_2 + l_1^2$$

$$p_x^2 + p_y^2 = l_2^2 + 2l_1 l_2 c\theta_2 + l_1^2$$

$$c\theta_2 = \frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2}$$

Algebraic solution approach: Example (4 of 4)

- Finally, two possible solutions for θ_2 are computed as follows using the 4th trigonometric equation given in Table:

$$\theta_2 = A\tan 2\left(\mp\sqrt{1-\left[\frac{p_x^2+p_y^2-l_1^2-l_2^2}{2l_1l_2}\right]^2}, \frac{p_x^2+p_y^2-l_1^2-l_2^2}{2l_1l_2} \right) \qquad \ldots(4.38)$$

$$c\theta_1 p_x + s\theta_1 p_y = l_2 c\theta_2 + l_1 \qquad \ldots(4.39)$$

- Using the 1st trigonometric equation in Table 4.2 produces two potential solutions:

$$\theta_1 = A\tan 2(p_y, p_x) \mp A\tan 2\left(\sqrt{p_y{}^2 + p_x - (l_2 c\theta_2 + l_1)^2}, l_2 c\theta_2 + l_1\right) \qquad \ldots(4.40)$$

Algebraic solution approach: Example (1 of 2)

- The inboard joint variables (first three joints) can be solved using the position vectors:

$$\left[{}^{0}_{1}T\,{}^{1}_{2}T\right]^{-1}\,{}^{0}_{6}T = {}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T \qquad \ldots(4.41)$$

$$\begin{bmatrix} . & . & . & c\theta_2(c\theta_1 p_x + s\theta_1 p_y) + s\theta_2(p_z - h_1) \\ . & . & . & -s\theta_2(c\theta_1 p_x + s\theta_1 p_y) + c\theta_2(p_z - h_1) \\ . & . & . & s\theta_1 p_x - c\theta_1 p_y - d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & 0 \\ . & . & . & d_3 \\ . & . & . & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Using the 1st& 2nd trigonometric equations in Table 4.2, respectively:

$$\theta_1 = A\tan 2(p_x, -p_y) \pm A\tan 2\left(\sqrt{p_x^{\,2} + p_y^{\,2} - d_2^{\,2}}, d_2\right) \qquad \ldots(4.42)$$

$$\theta_2 = \pm A\tan 2(c\theta_1 p_x + s\theta_1 p_y, -p_z + h_1) \qquad \ldots(4.43)$$

- The prismatic joint variable d_3 is extracted from the (2, 4) elements of each side in equation 4.41 as follows:

$$d_3 = -s\theta_2(c\theta_1 p_x + s\theta_1 p_y) + c\theta_2(p_z - h_1) \qquad \ldots(4.44)$$

Algebraic solution approach: Example (2 of 2)

- The rotation matrices are given by:

$$\begin{bmatrix} . & . & r_{33}s\theta_2 + r_{13}c\theta_1 c\theta_2 + r_{23}c\theta_2 s\theta_1 & . \\ d & e & r_{33}c\theta_2 - r_{13}c\theta_1 s\theta_2 - r_{23}s\theta_1 s\theta_2 & . \\ . & . & r_{13}s\theta_1 - r_{23}c\theta_1 & . \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} . & . & -c\theta_4 s\theta_5 & . \\ c\theta_6 s\theta_5 & -s\theta_5 s\theta_6 & c\theta_5 & . \\ . & . & s\theta_4 s\theta_5 & . \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots(4.45)$$

where $d = r_{31}c\theta_2 - r_{11}c\theta_1 s\theta_2 - r_{21}s\theta_1 s\theta_2$ and $e = r_{32}c\theta_2 - r_{12}c\theta_1 s\theta_2 - r_{22}s\theta_1 s\theta_2$.

- The revolute joint variables θ5 is determined by equating the (2,3) elements of both sides in equation 4.45 and using the 4th trigonometric equation in Table, as:

$$\theta_5 = A\tan 2\left(\pm\sqrt{1-(r_{33}c\theta_2 - r_{13}c\theta_1 s\theta_2 - r_{23}s\theta_1 s\theta_2)^2}, r_{33}c\theta_2 - r_{13}c\theta_1 s\theta_2 - r_{23}s\theta_1 s\theta_2\right)$$

$$\theta_4 = A\tan 2\left(\frac{r_{13}s\theta_1 - r_{23}c\theta_1}{s\theta_5}, -\frac{r_{33}s\theta_2 + r_{13}c\theta_1 c\theta_2 + r_{23}c\theta_2 s\theta_1}{s\theta_5}\right) \quad \ldots(4.46)$$

$$\theta_6 = A\tan 2\left(-\frac{r_{32}c\theta_2 - r_{12}c\theta_1 s\theta_2 - r_{22}s\theta_1 s\theta_2}{s\theta_5}, \frac{r_{31}c\theta_2 - r_{11}c\theta_1 s\theta_2 - r_{21}s\theta_1 s\theta_2}{s\theta_5}\right) \quad \ldots(4.47)$$

Advanced robotics (1 of 5)

- Due to low cost computing power and enhanced functionality of existing mechanisms, initiatives by European Community in advanced robotic activities.

- Extended the application domain into new and potentially gainful market sectors by exploring the innovations in robotics.

- An advanced robot was defined as a machine or system capable of accepting high level mission-oriented commands, navigation to a workplace and performing complex tasks in a semi-structured environment with a minimum of human intervention.

- The application areas include:
  - The nuclear industry, space, underwater, construction, health care and general service functions such as visual surveillance and cleaning.

- Advanced robotics is mainly dealing with the development of sensor-based control of mechanisms and automation and the development of appropriate system architectures to generate appropriate levels of functionality in order to execute the tasks.

Advanced robotics (2 of 5)

Implementation of a three-layer architecture:

- Three-layer systems architecture.

Strategic Identifies goals and strategies for achieving the goals which result in the achievement of an overall objective
Tactical Breeds the tactics by which a goal is to be achieved
Executive Executes the tactical plan.

- A basic functional representation of a representative hierarchical architecture:

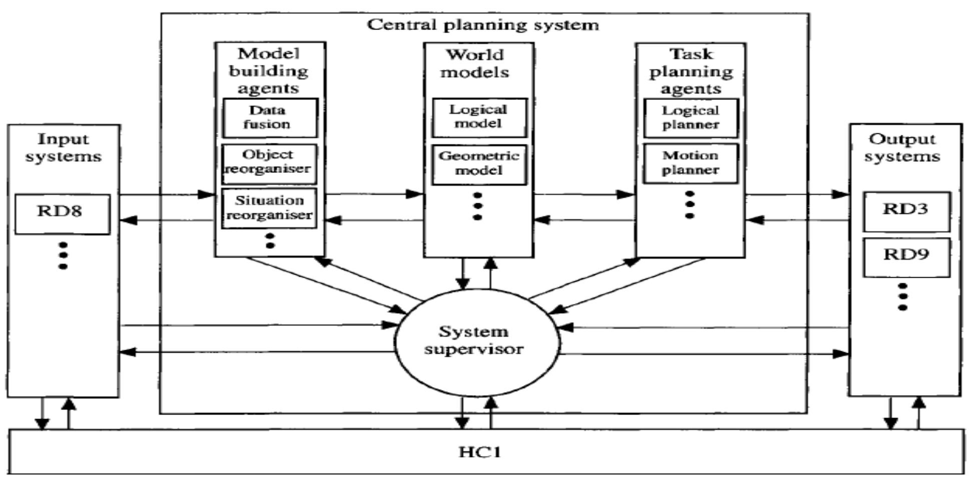


Figure: A hierarchy-based robot architecture

Advanced robotics (3 of 5)

- Pre-specified priority of operation :
  - A layer with a high level of priority can inhibit the operation of a layer with a low level of priority and
  - The overall architecture consists of an assembly of such layers, as shown in figure 4.20
  - The overall behavior of the robot being determined by how the various outputs are combined.
  - Each layer has its own relatively simple computing engine,
  - Parallel operation is possible and
  - The overall computing overheads are generally low.

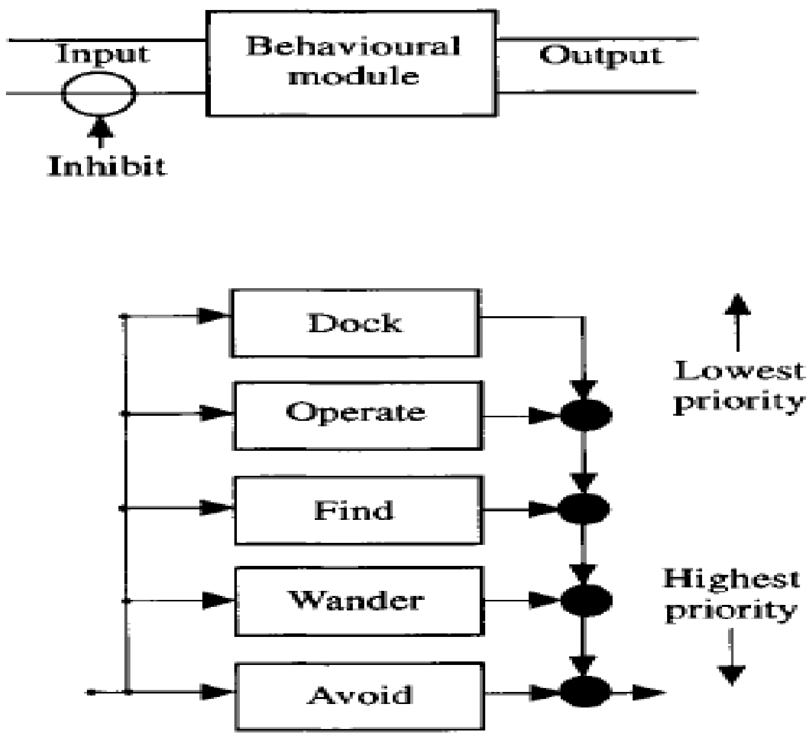


Figure: Behavior based robot architecture

Advanced robotics (4 of 5)

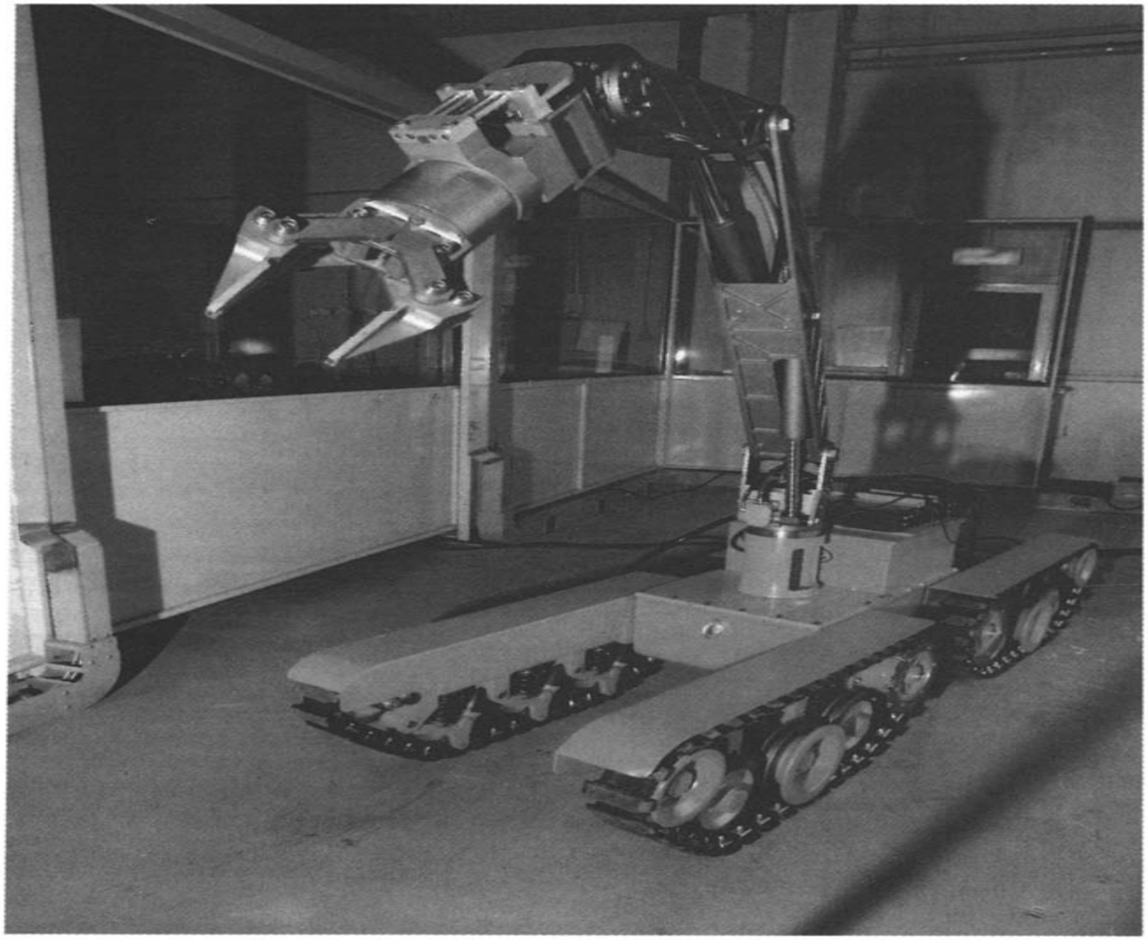


Figure: An advanced robot for the nuclear industry

Advanced robotics (5 of 5)

- Medical field applications:
  - Robot controlled prostate surgery.
  - The precision robotic machining of bones for hip surgery utilizing geometrical models derived from precisely computed-tomography image-information.
  - Use of autonomous/machine guided vehicles to aid the disabled, and aspects of microsurgery that employ micromanipulators.
  - Virtual reality and force feedback concepts.

- Mobile automata
  - The ESPRIT Project PANORAMA:
    - Development of practical perception and navigation systems suitable for the control of vehicles.
    - Implications for modern high-density motorway utilization.
    - By enhancing safety and the overall throughput of the existing roadway systems.

- Applications in aerospace:
  - Advanced tele-operated devices.
  - Mobility for satellite maintenance.
  - Constructional purposes and planetary exploration.

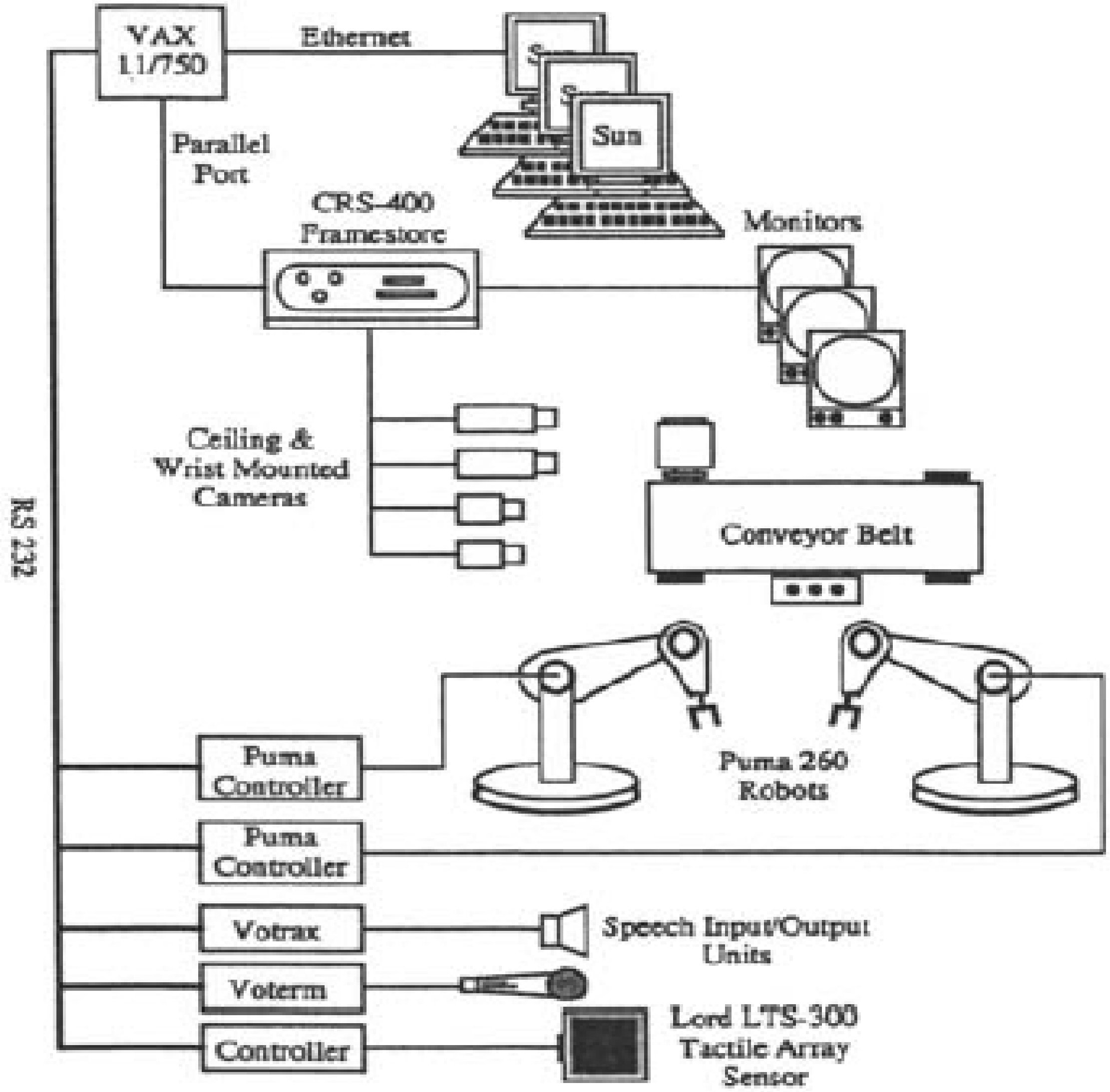


Figure: The Freddy 3 advanced robotic research test-bed.

Architectures for intelligent control (1 of 2)

- “The interconnection of physical systems, or the task undertaken by the system, does not make a machine intelligent.
- Intelligence comes from the manner in which the system is controlled or from the reasoning and decision making that the machine performs.
- Therefore intelligent-control is closely associated with machine intelligence” and is depicted in Figure.

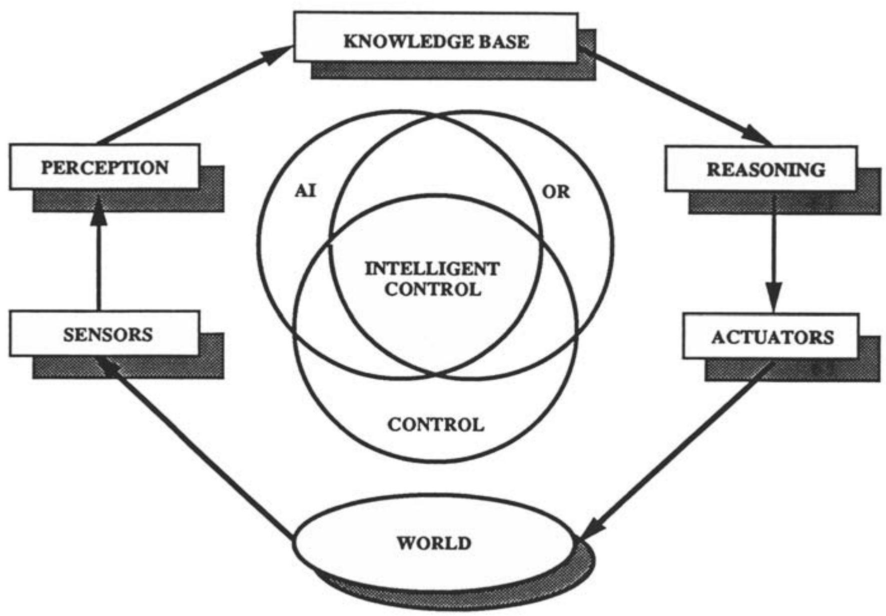


Figure: Architecture for intelligent control system.

Architectures for intelligent control (2 of 2)

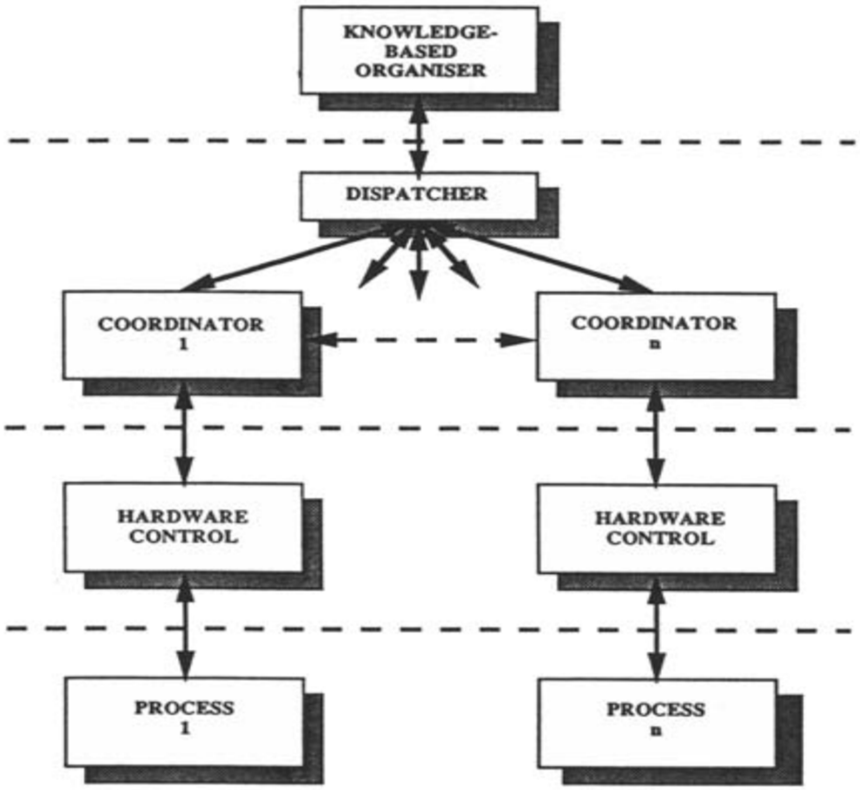


Figure: Architecture for intelligent control system

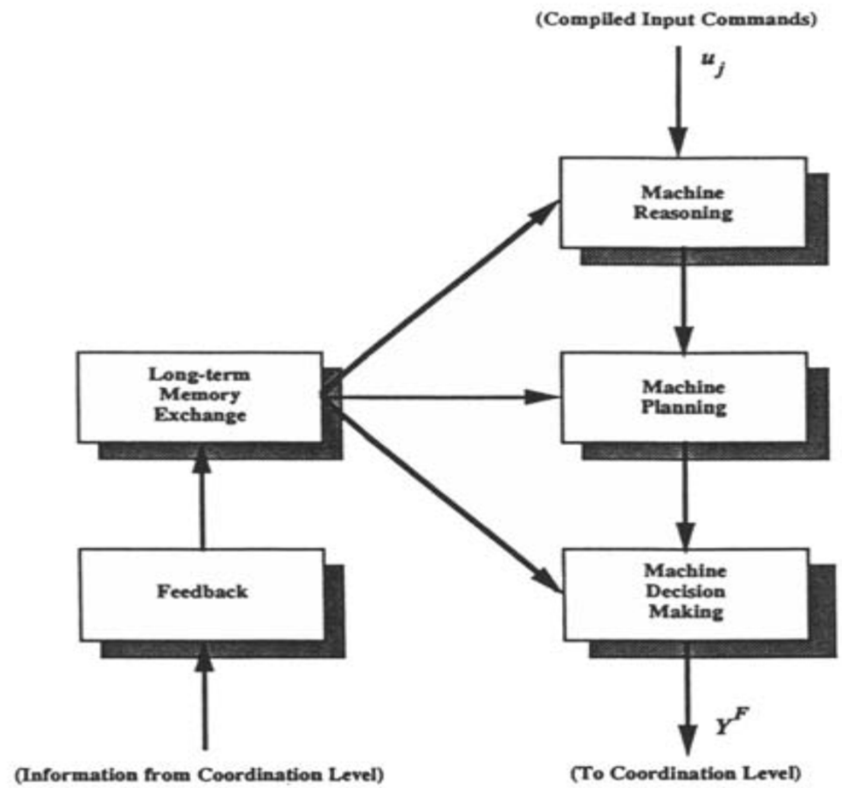


Figure: Saridis Processing Modules

Machine learning

- Expert systems are the most successful implementation of AI.

- Applications include: medical diagnosis, mineral analysis, control of complex processes, and fault diagnosis and monitoring; systems for fault diagnosis and monitoring account for half their applications.

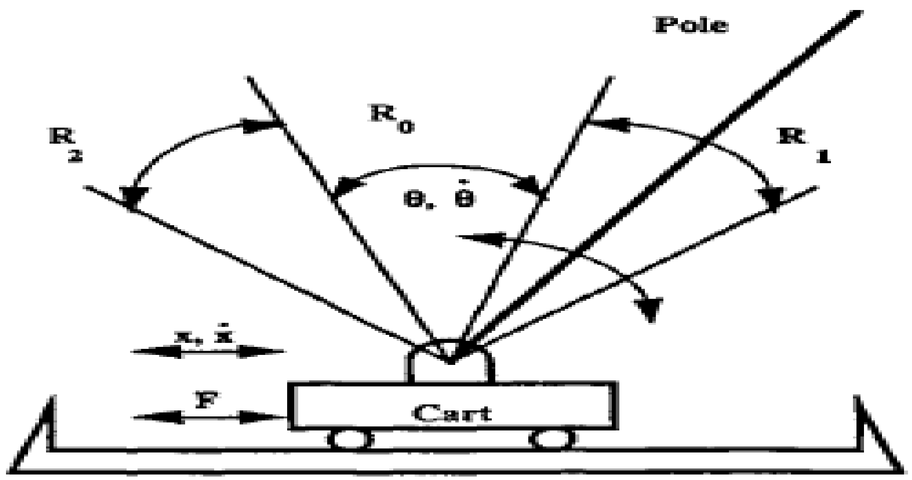
- Machine learning is described as going from the specific to the general.

- Here, we will look at:
  – How the production rules for 'rule-based' control can be produced manually, and automatically, and
  – Discuss two approaches for achieving machine learned control (MLC).

- Firstly, we will describe a controller based on the machine learning algorithm BOXES,
  – An algorithm that uses a reinforcement learning approach.

- Secondly, we will discuss the implementation of neural networks for control;
  – Both reinforcement learning and competitive learning are considered.

Machine learning: Rule-based control (1 of 3)



$$\ddot{\theta} = \frac{g \sin\theta - \cos\theta \left[\dfrac{F + m_p l \dot{\theta}^2 \sin\theta}{m_c + m_p} \right]}{l \left[\dfrac{4}{3} - \dfrac{m_p \cos\theta^2}{m_c + m_p} \right]}$$

$$\ddot{x} = \frac{F + m_p l [\dot{\theta}^2 \sin\theta - \ddot{\theta} \cos\theta]}{m_c + m_p}$$

Figure: Pole and cart.

Machine learning: Rule-based control (2 of 3)

- If theta_dot > THRESHOLD theta_dot then push RIGHT
- If theta_dot < -THRESHOLD theta_dot then push LEFT
- if theta > THRESHOLD theta then push RIGHT
- if theta < -THRESHOLD theta then push LEFT
- ifx_dot >THRESHOLD x_dot then push RIGHT
- ifx_dot < -THRESHOLD x_dot then push LEFT
- if x > THRESHOLD x then push RIGHT
- if x < -THRESHOLD x then push LEFT

Machine learning: Rule-based control (3 of 3)


```
if (theta(k) >THRESHOLD)
then
if((theta(k)<theta(k-1))
and (ltheta(k)-theta(k-1)l > ltheta(k-1)-theta(k-2)l)
then
apply a RIGHT force
else
apply a LEFT force
if (theta(k) <-THRESHOLD)
then
if((theta(k)>theta(k-1))
and (ltheta(k)-theta(k-1)l > ltheta(k-1)-theta(k-2)l)
then
apply a RIGHT force
else
apply a LEFT force
if (Itheta(k)l <=THRESHOLD)
then
if(x(k)>=0)
then
if ((x(k)<x(k-1) and (lx(k)-x(k-1)-x(k-2)l))|
then
apply a RIGHT force
else
apply a LEFT force
if(x(k)<0)
then
if ((x(k)>x(k-1)and (lx(k)-x(k-1)l>lx(k-1) -x(k-2)l))
then
apply a RIGHT force
else
apply a LEFT force
```

Machine learning: Machine learned control

- Machine learning is classified into two areas:
  - artificial-intelligence type learning on symbolic computation, and
  - neural networks.
- Expert systems based on artificial-intelligence learning:
  - have exceeded the performances of human experts and,
  - could communicate what they have learned to human experts for the purposes of verification.
- Artificial-intelligence type learning originated from an investigation into the possibility of using decision-trees/production-rules for concept representation.
- The field use decision-trees/production-rules in order to handle the most conventional data types, including those with noisy data, and as a knowledge acquisition tool.

Machine learning: Reinforcement learning

- Reinforcement learning is similar to supervised learning.
- It uses feedback for adaptation.
- The feedback reinforcement learning gets is only an indication of the value of the systems action.
- Reinforcement is feedback on the correctness of an action.
- Reinforcement learning is useful in cases where:
  - Supervisory information is not available.
- Reinforcement learning falls into two categories:
  - Non-associative type.
  - Associative reinforcement learning.

Advanced control systems for robotic arms

- Robotic arm is a mechanical arm to perform the desired task.
- The links of the manipulator can be considered to form a kinematic chain.
- The business end of the kinematic chain of the schemer is called the end effector and it is analogous to the human hand.
- The end effector can be a gripper or designed to perform any desired task such as welding, painting, assembly, etc.
- Robot must:
  - Control the position of the tool with respect to the work as a function of time.
  - Control the operation of the tool. Actuators – move the joints by a particular form of drive system.
- Common drive systems used in robotics are:
  - Electric drive.
  - Hydraulic drive, and
  - Pneumatic drive.

Kinematic and dynamic control

- Most industrial robot arms are designed so that the principal of wrist partitioning can be used.
- In machine design, a mechanism is synthesized that functions with the minimum complexity of the structure.
- Generally claimed with six degrees-of-freedom non-redundant robot arm is a general purpose device.
- Welding robots, which usually do not need rotation about the welding torch, have five actuators.
- Kinematic redundancy: When a robot arm possesses more degrees of freedom than the minimum number required to execute a given task.
- A six degree-of-freedom non-redundant robot arm can 'freely' position and orient an object in the Cartesian workspace.

Intelligent gripping systems

- Robotic grasping and restraint have evolved into an important field of robotics research, which is due partly to the evolution of industrial automation towards flexible automation, and partly to the progress in the study of fundamentals.
- Attention has been given to the mechanics of grasps and to the grasping of a broader class of objects, both in precise assembly and in hazardous environments.
- The need is to determine the grasp-ability of objects, so as to plan a stable and optimal grasp, and to impart fine motion and force control.
- A versatile grasping system integrates sensors and grippers, together with appropriate software.
- At present, the gripping systems and mechanical hands are weakly related to the human hand, which raises the issue of the study of anthropomorphic hands.
- Anthropomorphic hands make use of human knowledge, relying upon a large database for gripper models and sensors for feedback.

Overview of the Salford theories (1 of 2)

- Salford is used to analyze and synthesize the grasp, with the application of:
  - screw algebra, convex algebra and algebraic geometry, and a set of new approaches.

- A known grasp can be analyzed. A grasp of a known and unknown objects can be synthesized and planned.

Theories on restraint and grasping:

- The minimum number of frictionless point contacts to complete restraint is needed to perform either translational restraint or complete restraint.

- Table Minimum number of contacts in restraint.

$$\sum_{i=1}^{n+1} f_i \$_i = \boldsymbol{W}$$

	Translational	Complete
Planar	3	4
Spatial	4	6

Overview of the Salford theories (2 of 2)

- In consideration of frictional restraint, equation (4.48) is in the form:

$$\sum_{i=1}^{vn} f_i \$_i = W$$

- Where *n* is the number of contact points, and *v* is 2/3 in planar or spatial cases respectively.

- A further equation of geometric compatibility of contact points has to be introduced in the form:

$$u = [J]^T [\Delta] D$$

  – where [J] is a Jacobian matrix formed by contact screws $\$_i i=l,...,vn$,

    u is the displacement along vn contact screws.

    [Δ] is the matrix operator to exchange both parts of D, and

    D is the general twisting displacement of an object.

Need and provision of fingertip sensor system

- The implications of the above results suggest that for a grasp to be analyzed it is necessary to have a knowledge of the positions of the contact points, and the directions of the normal to the surface of the object at those contact points.

- If friction is involved, then it is also necessary to know the elastic properties of the contacts or fingers, and also the coefficients of limiting friction at the contacts.

- The new fingertip sensor system is :
  - capable of allowing the detection of position of contact point.
  - relative to the known position and orientation of the supporting frame or fingertip.
  - to a satisfactory level of accuracy, typically to 0.5 mm in any co-ordinate direction.

Computer software package implementation (1 of 2)

- A software has been developed:
  - To implement the new theories and approaches.
  - To effectively use the information from the fingertip system.
  - To facilitate the analysis and synthesis of grasps both in the analytical and experimental work.

- The software is written in C++:
  - Consists of a user-oriented package.
  - Two libraries of linear algebra and screw algebra, and
  - A set of independent programmes.

- It is capable of:
  - Interrogating contact information, normal, position vectors etc. From the fingertip sensor system.
  - Giving detailed force distribution from the detection.
  - Assessing a grasp and giving the diagnosis of the grasp.
  - Planning a grasp on a set of contact information and further optimizing the grasp.
  - Analyzing a grasp with different properties such as stiffness.
  - Predicting preload and force distribution corresponding to any given external force.

Computer software package implementation (2 of 2)

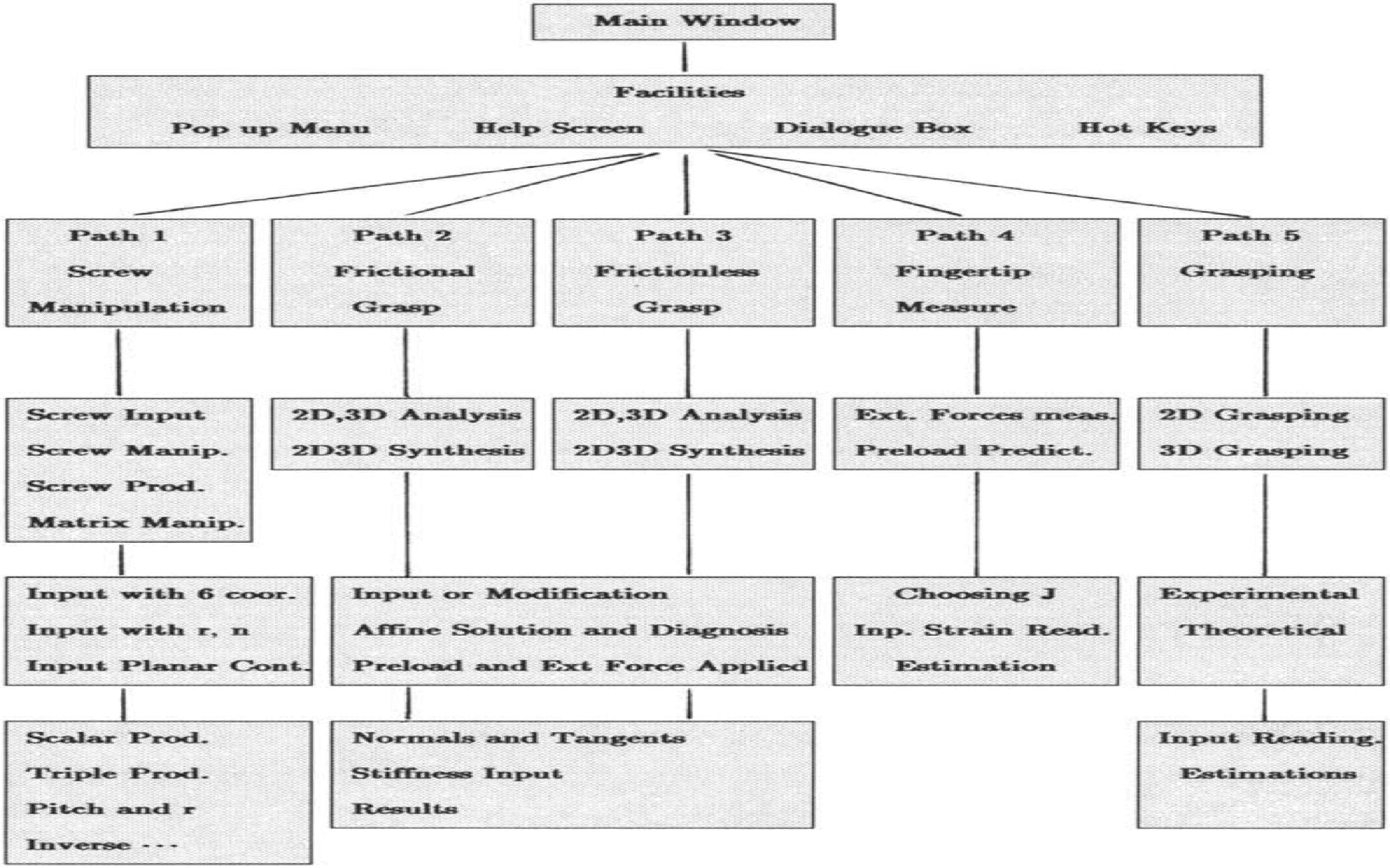


Figure: Flowchart of RASP package

Force feedback control in robots and its application to decommissioning

- Force feedback control in robots and its application.
- Force control is a central requirement if robot arms are to use tools or interact with work-pieces in an unstructured environment.
- The paradigm considered is the use of tele-robots' - arms which are operated under human supervision while carrying out complex tasks.
- Robot force control - a technique to modify an arm's servo behavior when contacting its environment.
- Decommissioning is a task which requires the use of tools, such as drills and saws, in a radioactive contaminated environment. A number of systems have been, or are being, developed to accomplish this task.
- The experimental system at oxford is based on a very high-performance parallel robot used as an input device coupled to a remote slave arm.

Force feedback strategies

- One of the earliest force feedback strategies was active stiffness control.

- In this original formulation the joint torque to the motors is given by:

$$\tau = J^T K_p (q_d - q) + K_v (\dot{q}_d - \dot{q}) \qquad \ldots(4.50)$$

Where *J* is the arm Jacobian,

K is the position gain chosen to achieve the requisite stiffness.

- This is a simple form of impedance control as developed by Hogan.

- The philosophy of impedance control is to specify the apparent robot end point dynamic impedance as seen by the environment.

- The term impedance is used within the context of the classical analogy of Bond graph theory, which maps force into effort (electrical voltage) and velocity into flow(electrical current).

Introduction to mobile robots

- Mobile robots or autonomous guided vehicles (AGVs) follow fixed paths either by using an inductive sensor to locate a buried wire or a simple optical sensor to follow a white line.

- Various sensors can be used in robotic systems to achieve autonomous navigation:
  - Encoders.
  - Cameras.
  - Detectors.
  - Diversified range finders.
  - Radar.
  - Sonar.
  - Optoelectronic devices.

Environment capturing with common sensors

- Sensors measure a physical quantity and convert it into a useable signal for your robot.
- We have a huge range of sensors that can be applied to a variety of projects which includes:
  - Environmental monitoring.
  - Distance.
  - Force.
  - Speed.
  - Pressure.
  - Temperature.
  - Magnetic flux.
  - Vibration.
  - Humidity.
  - Rotation.
  - Touch.
  - Imaging.
  - Light.
  - Biometrics.
  - Gas.
  - Acceleration.
  - Current.

CCD cameras (1 of 2)

- A CCD camera was first used in the late 1960s.
- It captures and stores images in digital memory.
- CCDs are found in photocopiers, security surveillance cameras, fax machines.
- The beauty of CCD cameras is that it provides a low-noise, high quality image at a highly pixilated resolution.

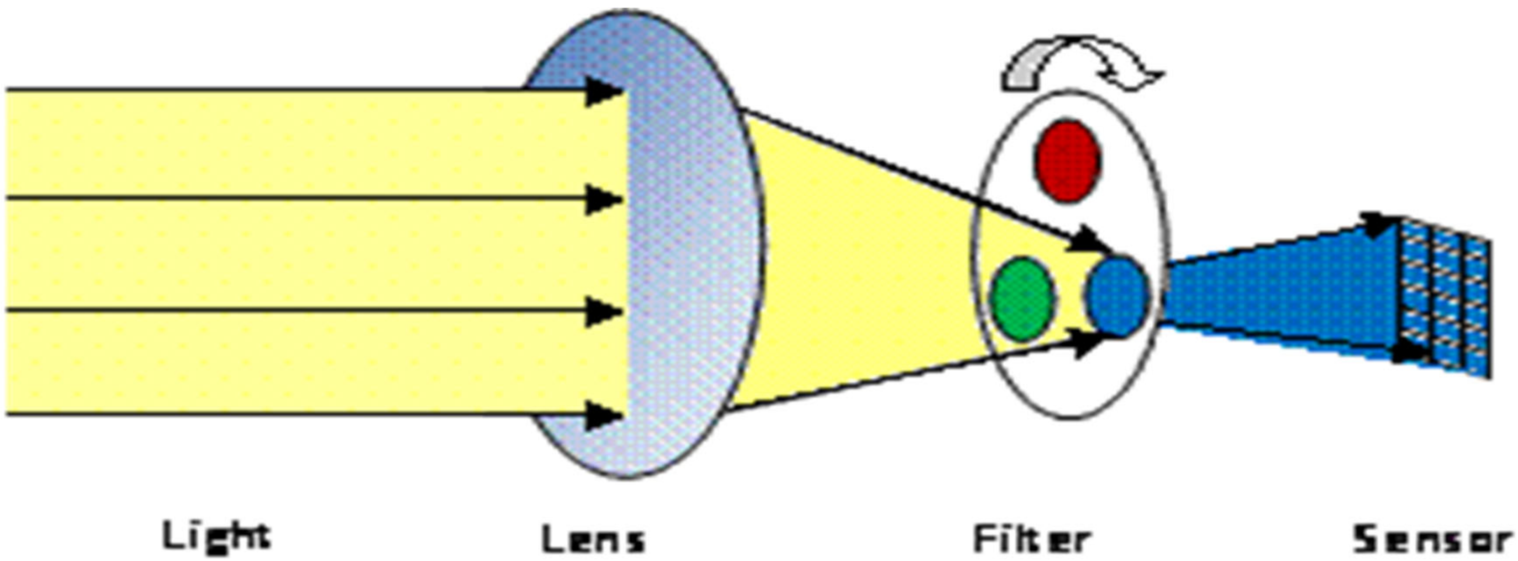
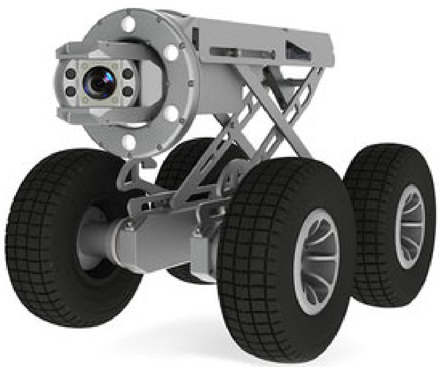


Figure: Image sensor setup to capture blue component.

CCD cameras (2 of 2)



Figure(a): CCD sensor-based sewer pipe inspection robot camera

Figure(b): CCD Sensor based crawling pipe inspection camera

Figure(c): Outdoor wireless camera

Figure(d): Pipeline Inspection CAM robot camera

CCD Vs. CMOS

- CCD sensors create high-quality, low-noise images. CMOS sensors are generally more susceptible to noise.

- Because each pixel on a CMOS sensor has several transistors located next to it, the light sensitivity of a CMOS chip is lower. Many of the photons hit the transistors instead of the photodiode.

- CMOS sensors traditionally consume little power. CCDs, on the other hand, use a process that consumes lots of power. CCDs consume as much as 100 times more power than an equivalent CMOS sensor.

- CCD sensors have been mass produced for a longer period of time, so they are more mature. They tend to have higher quality pixels, and more of them.

Sonar sensors (1 of 2)

- The sonar sensors in various robotic applications:
  - Wall Following.
  - Obstacle Avoidance.
  - Positioning.
  - Room Mapping.
  - People Detection.
  - Collision Avoidance.

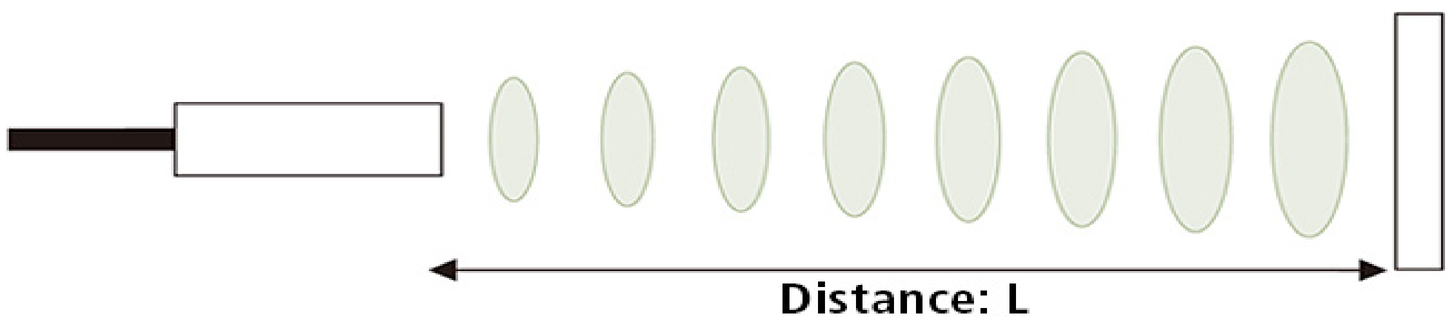


Figure: Sonar sensor technology

Sonar sensors (2 of 2)

- The following list shows typical characteristics enabled by the detection system:
  - Transparent object detectable: Since ultrasonic waves can reflect off a glass or liquid surface and return to the sensor head, even transparent targets can be detected.
  - Resistant to mist and dirt: Detection is not affected by accumulation of dust or dirt.
  - Complex shaped objects detectable: Presence detection is stable even for targets such as mesh trays or springs.

Figure: High performance ultrasonic sensor

Figure: Ultrasonic range finder

Optoelectronic sensors

- An optoelectronic sensor is a device that produces an electrical signal proportional to the amount of light incident on its active area.

- Integrated optoelectronic sensors are designed to respond to light so that they can recognize things such as patterns, images, motion, intensity, and color.

- Different types of Optoelectronic Sensors:

Optoelectronic Sensor Type	Description
Light-to-voltage converters	Produce a linear output voltage proportional to light intensity
Light-to-frequency converters	Convert light intensity to digital format for direct connection to a microcontroller or DSP
Ambient light sensors	Measure what the human eye sees
Linear sensor arrays	Measure spatial relationships and light intensity
Color sensors	RGB (red/green/blue) filtered sensors for color discrimination, determination, and measurement
Reflective light sensors	Convert reflective light intensity to a voltage output

Sensor integration

- The use of multiple, physically different sensors can help to overcome the shortcomings of each individual device.
- Integration of redundant data from multiple sensors can minimize the uncertainty.
- Typical methods for integrating data from multiple sensors can be categorized as follows:
  - Qualitative approach.
  - Quantitative approach.

Qualitative approaches (1 of 3)

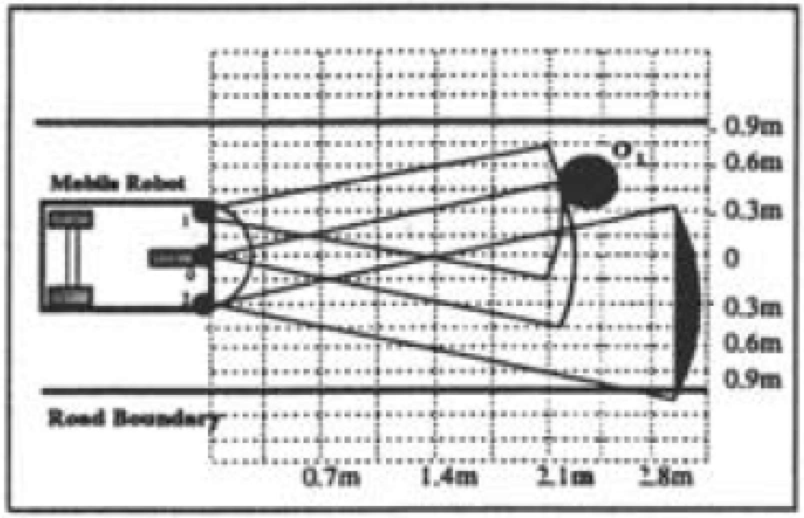
- Qualitative approaches use powerful representation mechanisms to describe complex sensor information and heuristics to guide data interpretation.

- Rule bases are convenient when the statements are in the form 'if., then'.

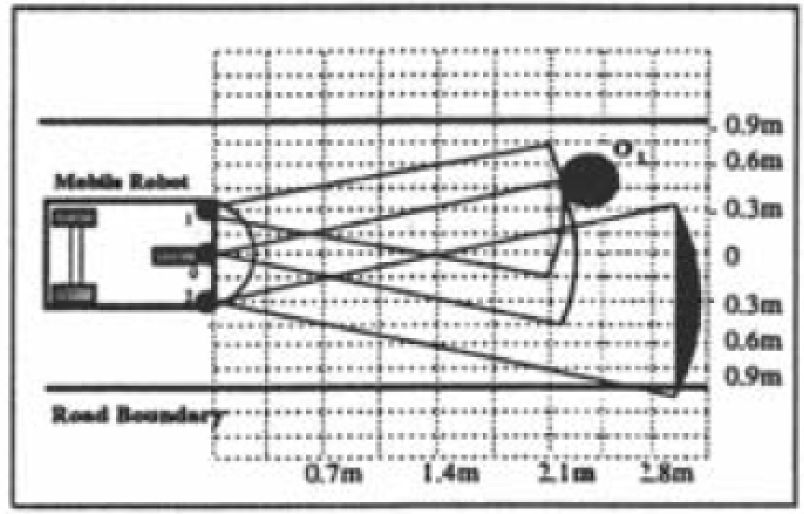
- The rule base provided information to guide the robot as it followed a predetermined path in a factory with a number of allowable pathways.

- It provided an estimate of the following states of the robot's current path:
  - The path was clear.
  - It was blocked by a small obstacle (such that a side-step maneuver past it can be performed within the pathway).
  - It was blocked by a large obstacle (which entirely blocks the pathway, requiring the robot to backtrack and find a new pathway).

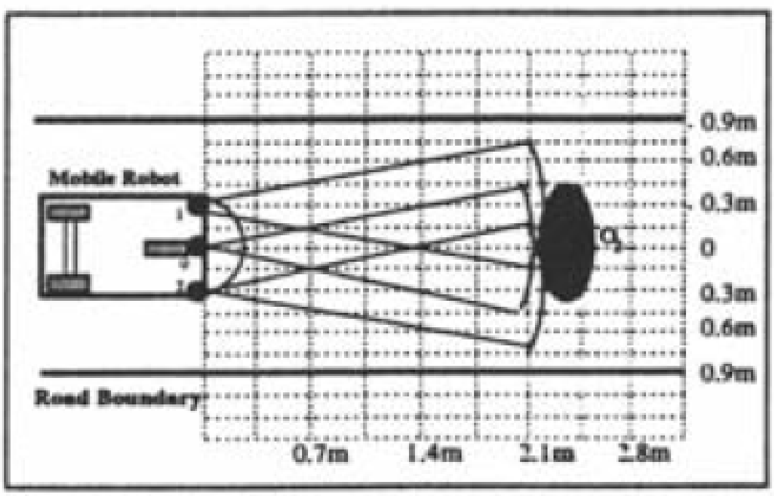
Qualitative approaches (2 of 3)



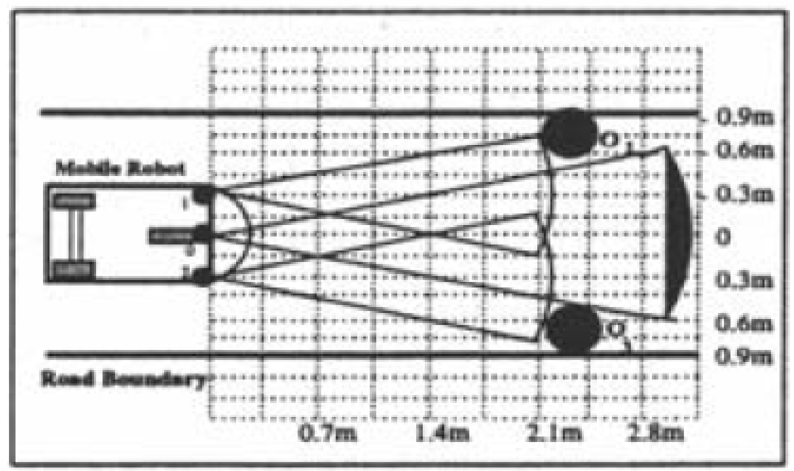
Figure(a): An obstacle appears at the left side



Figure(b): An obstacle appears at the right side



Figure(c): An obstacle appears in the middle



Figure(d): An obstacle appears on both sides

Quantitative approaches (3 of 3)

- The quantitative approach to sensor integration requires numbers.
- The information obtained from sensors is quantified so that statistical and decision theoretic methods can be adopted.
- Through quantitative analysis and mathematical modelling, considerable success has been demonstrated with such approaches.
- We describe two approaches in this section:
  - Bayes statistics.
  - The Kalman filter.

Bayes statistics

- Bayes statistics, which use well established mathematical reasoning, can be used to integrate information from various sources (for example from multiple sensors or over time).

- Unlike the simple sonar rule base we introduced earlier, the Bayes formulation distinguishes between the measurement of a quantity and the true state of the quantity.

- To account for uncertainty, it reasons using probability density functions.

- The Bayes update rule, which updates an existing hypothesis on a quantity with new information:

$$P(O|Z) = \frac{P(Z|O)P(O)}{\sum P(Z|O)P(O)}$$

  - Where O is the true state of the system and
  - Z is the possibly incorrect observation of the state.

Kalman filter

- The filter uses two stages:
  - The previous state and a model of how it changes between iterations.
  - An observation of the new state.

- Therefore, like the Bayes update rule, it combines both prior expectation with measurements.

- Kalman filters are used too for spatial integration of data.

- The advantage of this scheme over Flynn's is that the map includes information on the variance as well as the expectation of each state variable, so the risk of plans based on this map can be quantified.

Machine vision system

- Because humans are good at vision, the task of implementing vision artificially is often underestimated.
- Even without any financial constraints, it is still not possible to artificially implement vision comparable to that of humans.
- Fortunately, this is not required or even desirable for industrial sensors, since the sensing problem can invariably be greatly reduced in complexity by applying a priori knowledge.
- An area of research, commonly referred to as industrial machine vision, is concerned with the development of visual systems for use in constrained industrial environments.

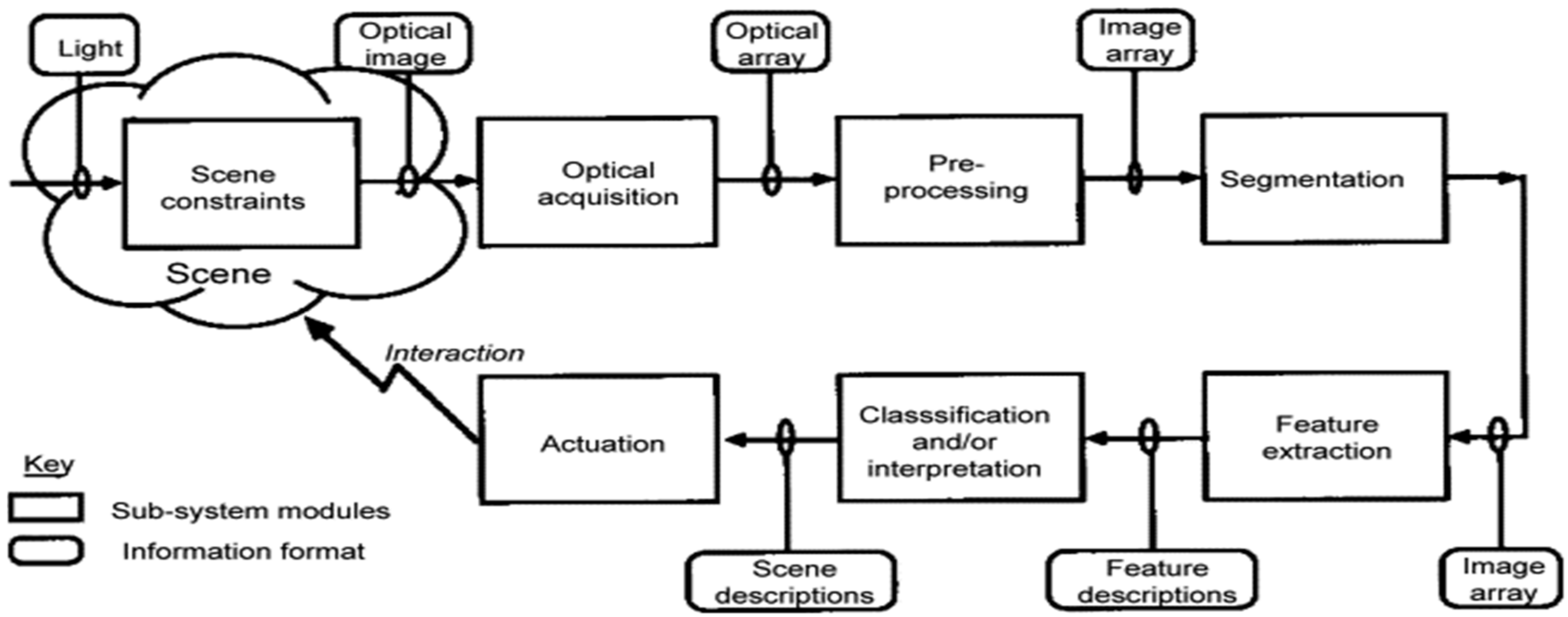


Figure: A generic model of a machine vision system

Phases of a machine vision system (1 of 2)

- The task of the machine vision system is to classify an object into a correct class based on the measurements about the object.

- The possible classes are usually well-defined already before the design of the machine vision system.

- Many machine vision systems can be thought to consist of five stages:
  - Sensing.
  - Pre-processing and segmentation.
  - Segmentation.
  - Classification.
  - Post-processing.

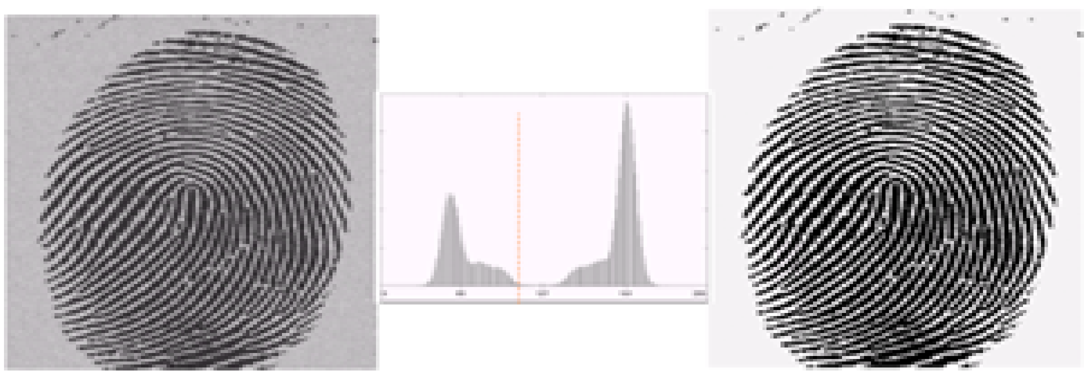


Figure: An image with bi-modal histogram

Phases of a machine vision system (2 of 2)

Feature Extraction:

- Some of the well-known boundary descriptors where we describe region borders expressed in some mathematical form are as follows:
  - Chain codes.
  - Shape numbers.
  - Geometric representations.
  - Boundary length.
  - Curvature.
  - Bending energy.
  - Signature.
  - Chord distribution.
  - Fourier descriptors.
  - Segment sequence-based representations.
  - Polygonal representation.
  - B-spline representation.
  - Other representation.
  - Hough transforms.
  - Mathematical morphology.
  - Neural networks.

Post-processing:

- The final task of the machine vision system is to decide upon an action based on the classification result.

Tool condition monitoring systems

- The following four essential components namely:
  - Sensing technique systems.
  - Feature extraction systems.
  - Decision making systems.
  - Knowledge learning systems.

- Any automated tool condition monitoring system has to emulate the human monitoring action.

- The major goals for tool condition monitoring are to develop self-adjusting and integrated monitoring systems able to function under various working conditions with minimum operator supervision.

- Basically, a monitoring process has two parts:
  - Sensing.
  - Monitoring.

Neural networks for tool condition monitoring systems

- Neural network learning methods provide a robust approach to approximating real-valued, discrete-valued, and vector-valued target functions.

- Neural network learning methods provide a robust approach to approximating real-valued, discrete-valued, and vector-valued target functions.

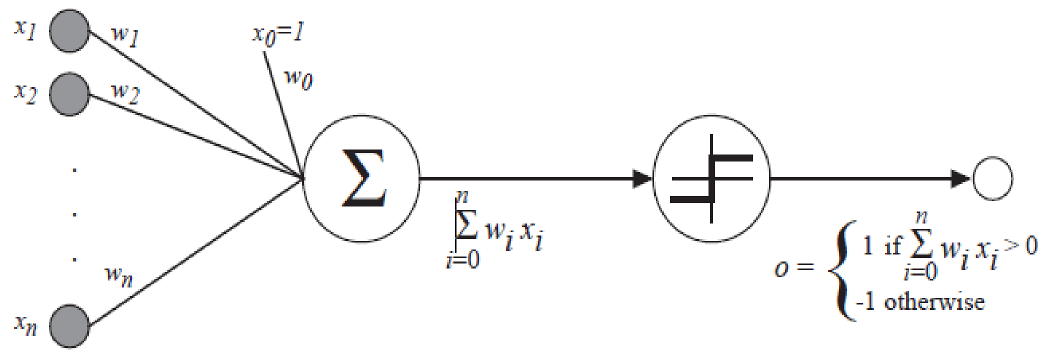
- Appropriate problems for neural network learning.

- The training examples may contain errors:
  - Long training times are acceptable.
  - Fast evaluation of the learned target function may be required.
  - The ability of humans to understand the learned target function is not important.

Basic understanding of neural networks

- One type of ANN system is based on a unit called a perceptron, illustrated in figure given below. Figure: A perceptron

>



- Given inputs xl through x,, the output o(x1, . . . , x,) computed by the perceptron is:

$$o(x_1, \dots, x_n) = \begin{cases} 1 \; if \; w_0 + w_1 x_1 + w_2 x_2 + \; \dots \; + w_n x_n > 0 \\ -1 \; otherwise \end{cases}$$

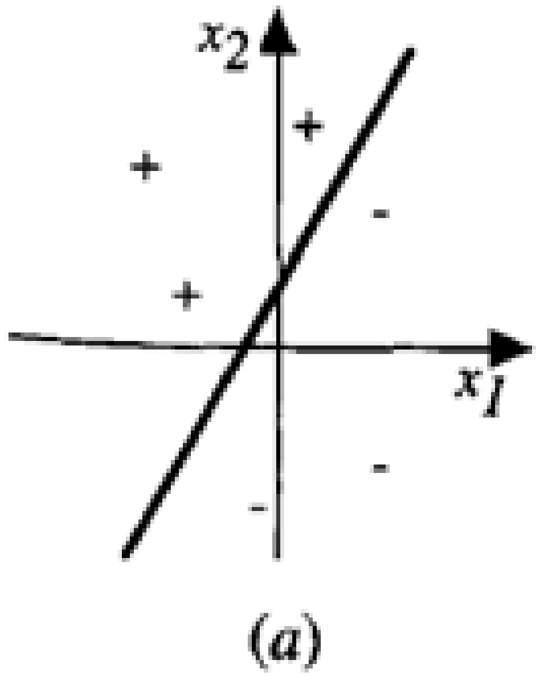
- To simplify notation, we imagine an additional constant input xo=1, allowing us to write the above inequality as:

$$\sum_{i=0}^{n} w_i x_i > 0 \qquad \vec{w}.\vec{x} > 0$$

- For brevity, we will sometimes write the perceptron function as:

$$o(\vec{x}) = sgn(\vec{w}.\vec{x}) \qquad sgn(y) = \begin{cases} 1 & if \; y > 0 \\ -1 & otherwise \end{cases}$$

Representational power of perceptrons



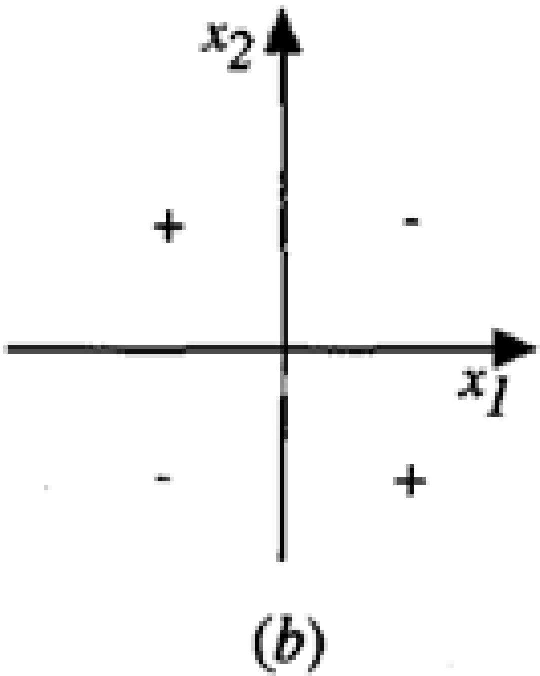


Figure: Perceptron representation.

- The decision surface represented by a two-input perceptron:
  - A set of training examples and the decision surface of a perceptron that classifies them correctly.
  - A set of training examples that is not linearly separable.

Architecture of neural networks

- An artificial neural network can be divided into three parts, named layers, which are known as:
  - Input layer.
  - Hidden, intermediate, or invisible layers.
  - Output layer.

- The layers of architecture are as follows:
  - Single-layer feed forward network.
  - Multilayer feed forward networks.
  - Recurrent networks.
  - Mesh networks.

Single-layer feed-forward architecture

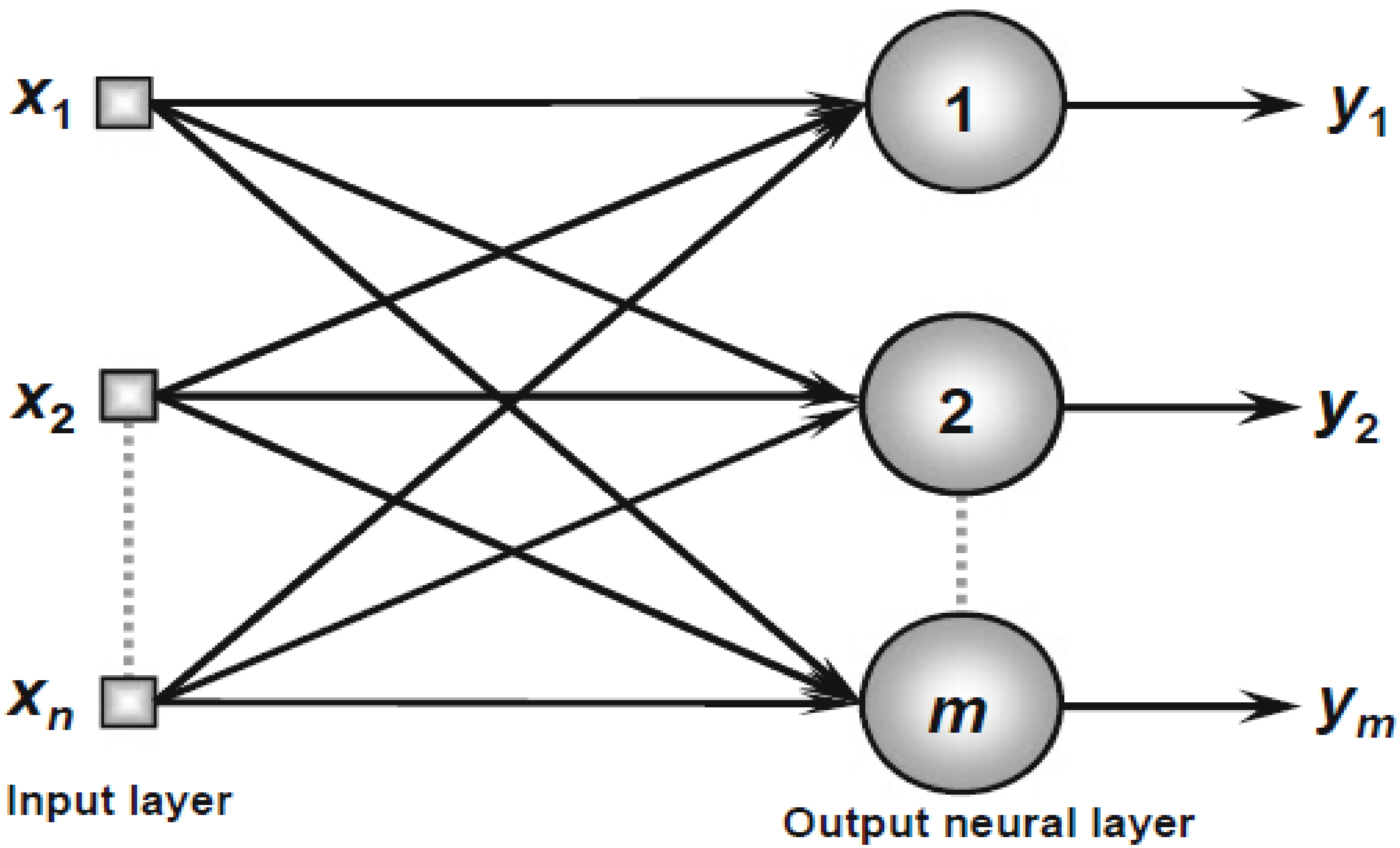


Figure: Example of single layer feed forward network

Multiple-layer feed-forward architecture

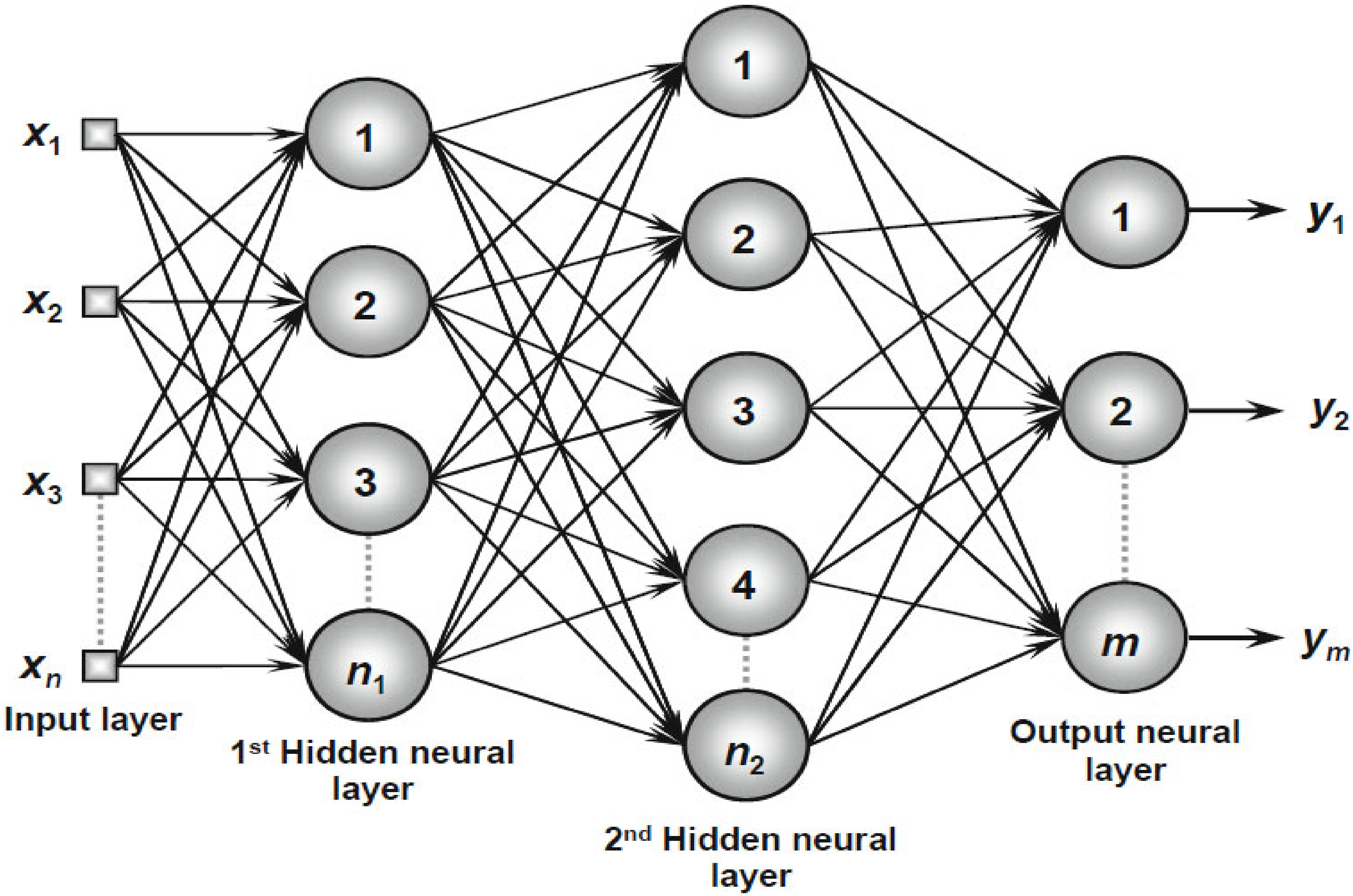


Figure: Example of a feedforward network with multiple layers

Recurrent or feedback architecture

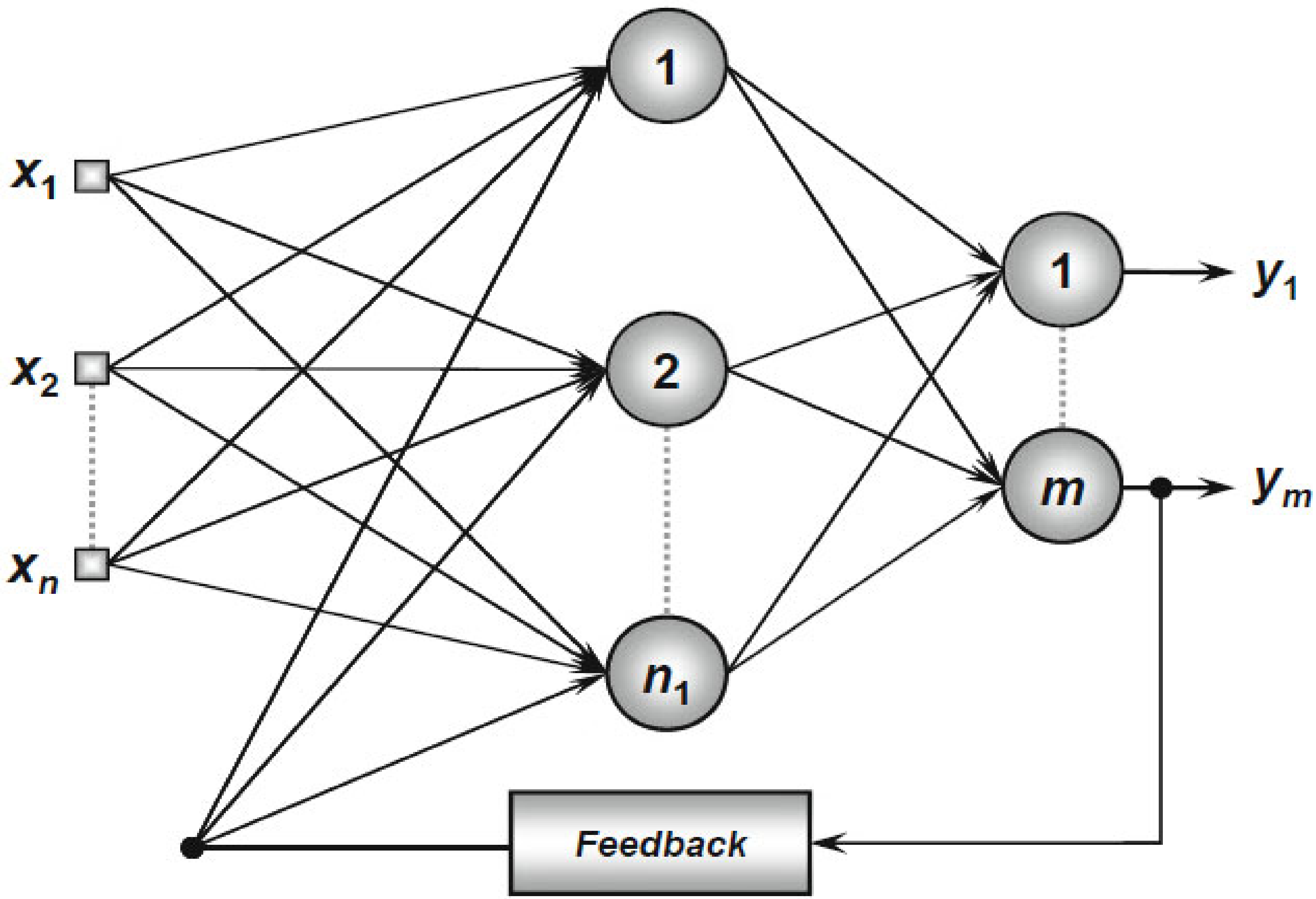


Figure: Example of a recurrent network

Mesh architecture

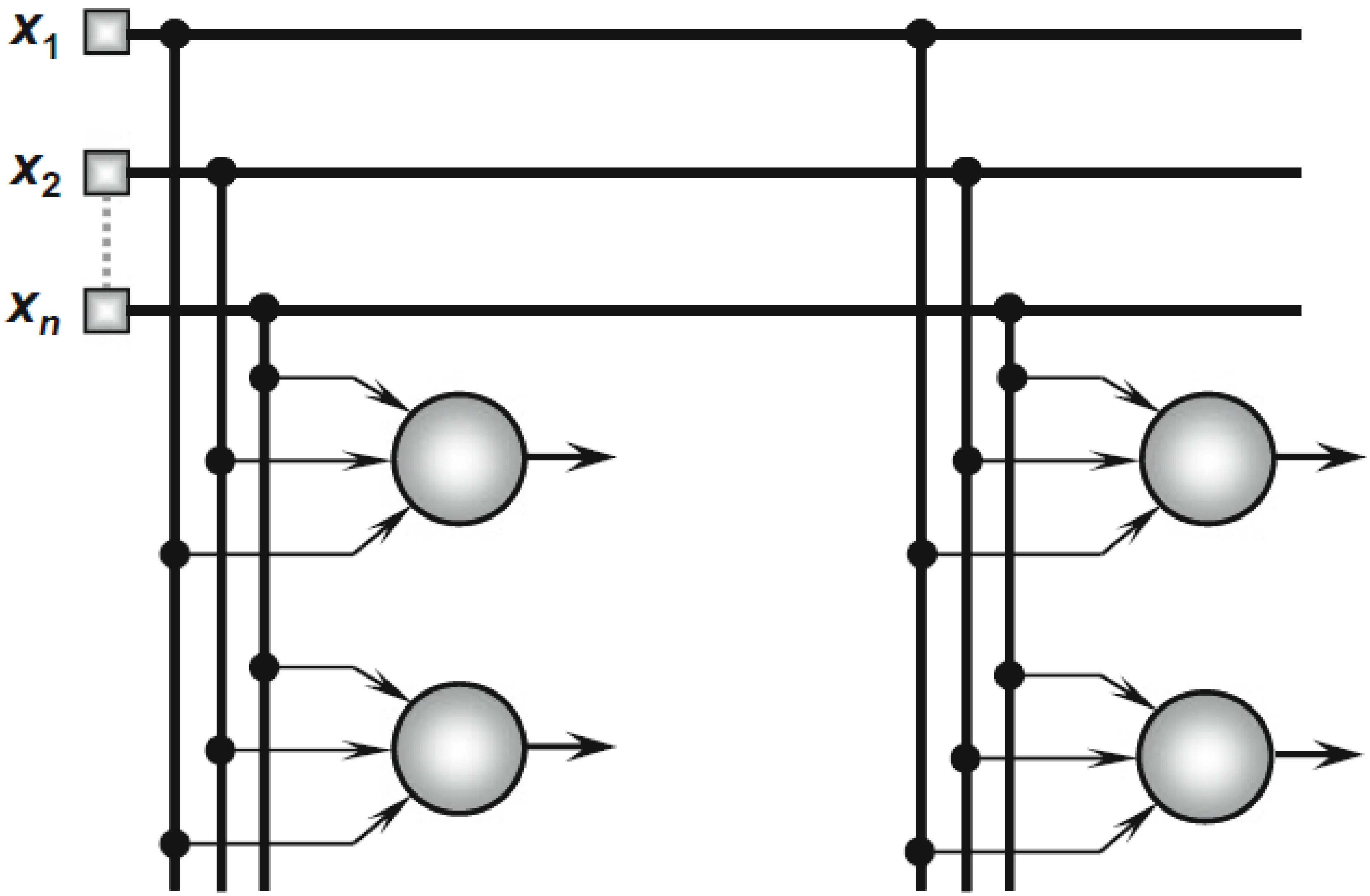


Figure: Structure of a mesh network

The perceptron training rule

- Precise learning problem is to determine a weight vector that causes the perceptron to produce the correct ±1 output for each of the given training examples.

- Several algorithms are known to solve this learning problem. Here we consider two:
  - The perceptron rule.
  - The delta rule.

- One way to learn an acceptable weight vector is to begin with random weights, then iteratively apply the perceptron to each training.

Gradient descent and the delta rule

- The delta training rule is best understood by considering the task of training an un-thresholded perceptron; that is, a linear unit for which the output o is given by:

$$o(\vec{x}) = \vec{w} \cdot \vec{x}$$

- Although there are many ways to define this error, one common measure that will turn out to be especially convenient is training examples:

$$E(\vec{w}) \equiv \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2$$

  - Where D is the set of training examples, td is the target output for training exampled
  - Od is the output of the linear unit for training example d.
  - By this definition, E() is simply half the squared difference between the target output td and the hidden unit output od, summed over all

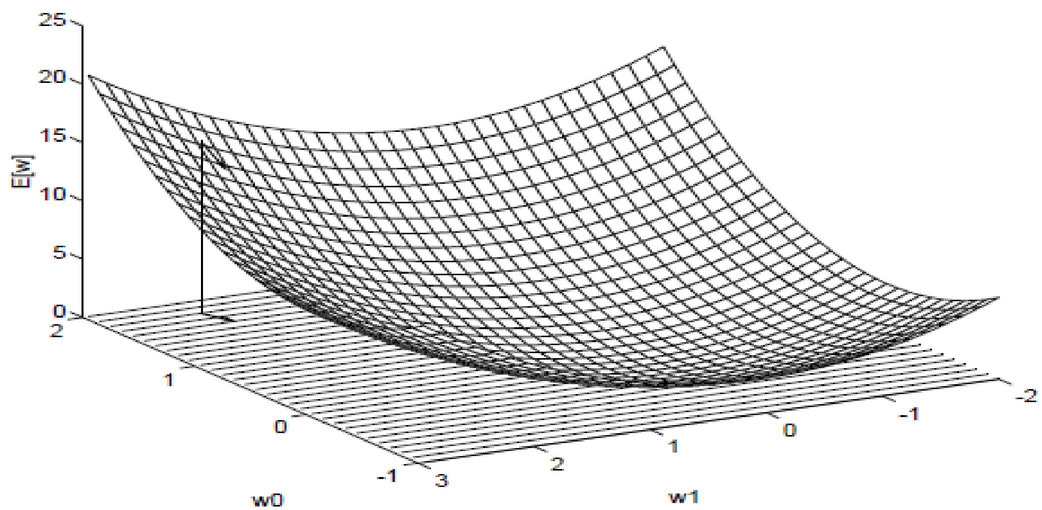


Figure: Error of different hypotheses

Gradient descent algorithm

- Algorithm: Gradient Descent Algorithm for training a linear unit.

- GRADIENT-DESCENT(training examples, η).

- Each training example is a pair of the form <x, t > , where x is the vector of input values, and t is the target output value. η is the learning rate.

Stochastic approximation to gradient descent (1 of 2)

- The key practical difficulties in applying gradient descent are:
  - Converging to a local minimum can sometimes be quite slow (i.e., It can require many thousands of gradient descent steps).
  - If there are multiple local minima in the error surface, then there is no guarantee that the procedure will find the global minimum.
- The modified training rule is like the training rule except that as we iterate through each training example, we update the weight according to:

$$\Delta w_i = \eta(t - o)\, x_i$$

  - where t, o, and xi are the target value, unit output, and ith input for the training example in question.
- One way to view this stochastic gradient descent is to consider a distinct error function $E_d(\vec{w})$ defined for each individual training example d as follows:

$$E_d(\vec{w}) = \frac{1}{2}(t_d - o_d)^2$$

  - Where td, and od are the target value and the unit output value for training example d.
  - Stochastic gradient descent iterates over the training examples d in D, at each iteration altering the weights according to the gradient with respect to $E_d(\vec{w})$.
  - The sequence of these weight updates, when iterated over all training examples, provides a reasonable approximation to descending the gradient with respect to our original error function E($\vec{w}$).
  - By making the value of η (the gradient descent step size) sufficiently small, stochastic gradient descent can be made to approximate true gradient descent arbitrarily closely.

Stochastic approximation to gradient descent (2 of 2)

- The key differences are listed below:

Standard Gradient Descent	Stochastic Gradient Descent
1.Error is summed over all examples before updating weights	1.Weights are updated upon examining each training example
2. Requires more computation per weight update step	2. Require less computation
3. Converges to local minima	3. Sometimes avoid falling into these local minima

- We have considered two similar algorithms for iteratively learning perceptron weights.
- The key difference between these algorithms are listed below:

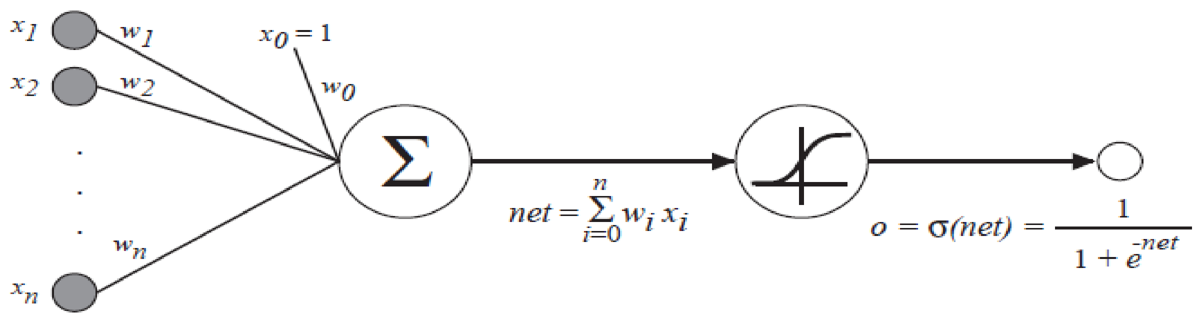
Perceptron training rule	Delta rule
Updates weights based on the error in the thresholded perceptron output	Updates weights based on the error in the un-thresholded linear combination of inputs
Converges after a finite number of iterations to a hypothesis that perfectly classifies the training data, provided the training examples are linearly separable.	Converges only asymptotically toward the minimum error hypothesis, possibly requiring unbounded time, but converges regardless of whether the training data are linearly separable.

Multilayer networks and back-propagation algorithm

- The kind of multilayer networks learned by the backpropacation algorithm are capable of expressing a rich variety of nonlinear decision surfaces.

- It is possible for the multilayer network to represent highly nonlinear decision surfaces that are much more expressive than the linear decision surfaces of single units.

- Figure: A sigmoid threshold unit



- The sigmoid unit computes its output o as $$\sigma(y) = \frac{1}{1+e^{-y}}$$
  - where $o = \sigma(\vec{w} \cdot \vec{x})$ σ is often called the sigmoid function.
  - Note its output ranges between 0 and 1, increasing monotonically with its input.
  - Because it maps a very large input domain to a small range of outputs, it is often referred to as the squashing function of the unit.
  - The sigmoid function has the useful property that its derivative is easily expressed in terms of its output. $\sigma'(y) = \sigma(y)(1- \sigma(y))$

The back-propagation algorithm

- Because we are considering networks with multiple output units rather than single units as before.

- We begin by redefining E to sum the errors over all of the network output units:

$$E(\vec{w}) \equiv \frac{1}{2} \sum_{d \in D} \sum_{k \in outputs} (t_{kd} - o_{kd})^2$$

  – Where outputs are the set of output units in the network, and t_{kd} and o_{kd} are the target and output values associated with the kth output unit and training example d.
  – The learning problem faced by Backpropagation search a large hypothesis space defined by all possible weight values for all the units in the network.
  – The situation can be visualized in terms of an error surface similar to that shown for linear units in Figure 5.15.
  – The error in that diagram is replaced by our new definition of E, and the other dimensions of the space correspond now to all of the weights associated with all of the units in the network.
  – As in the case of training a single unit, gradient descent can be used to attempt to find a hypothesis to minimize E.

Multiple principal component fuzzy neural networks

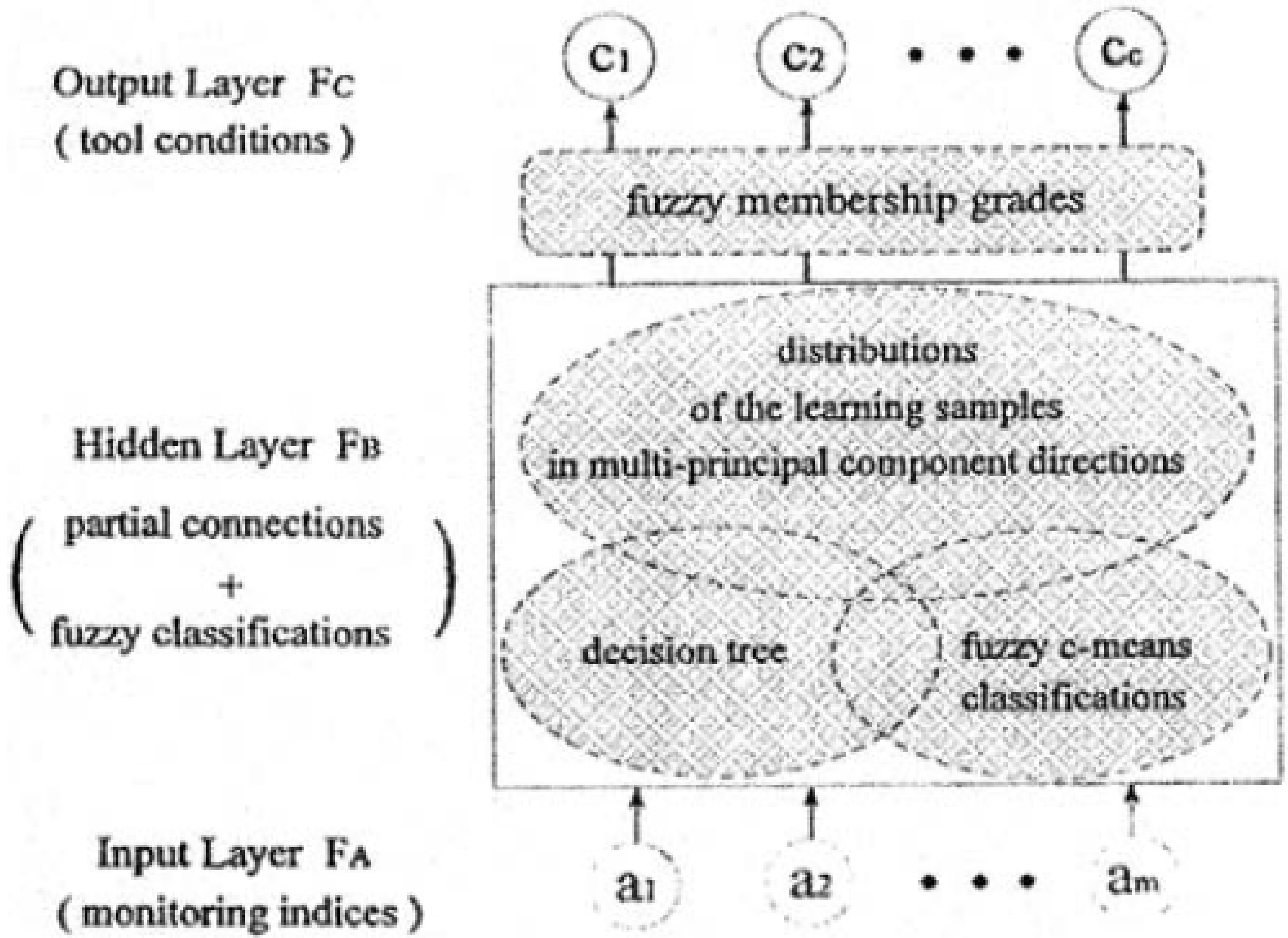


Figure: The Multiple Principal Component (MPC) fuzzy Neural Network

Fuzzy classification and uncertainties in tool condition monitoring

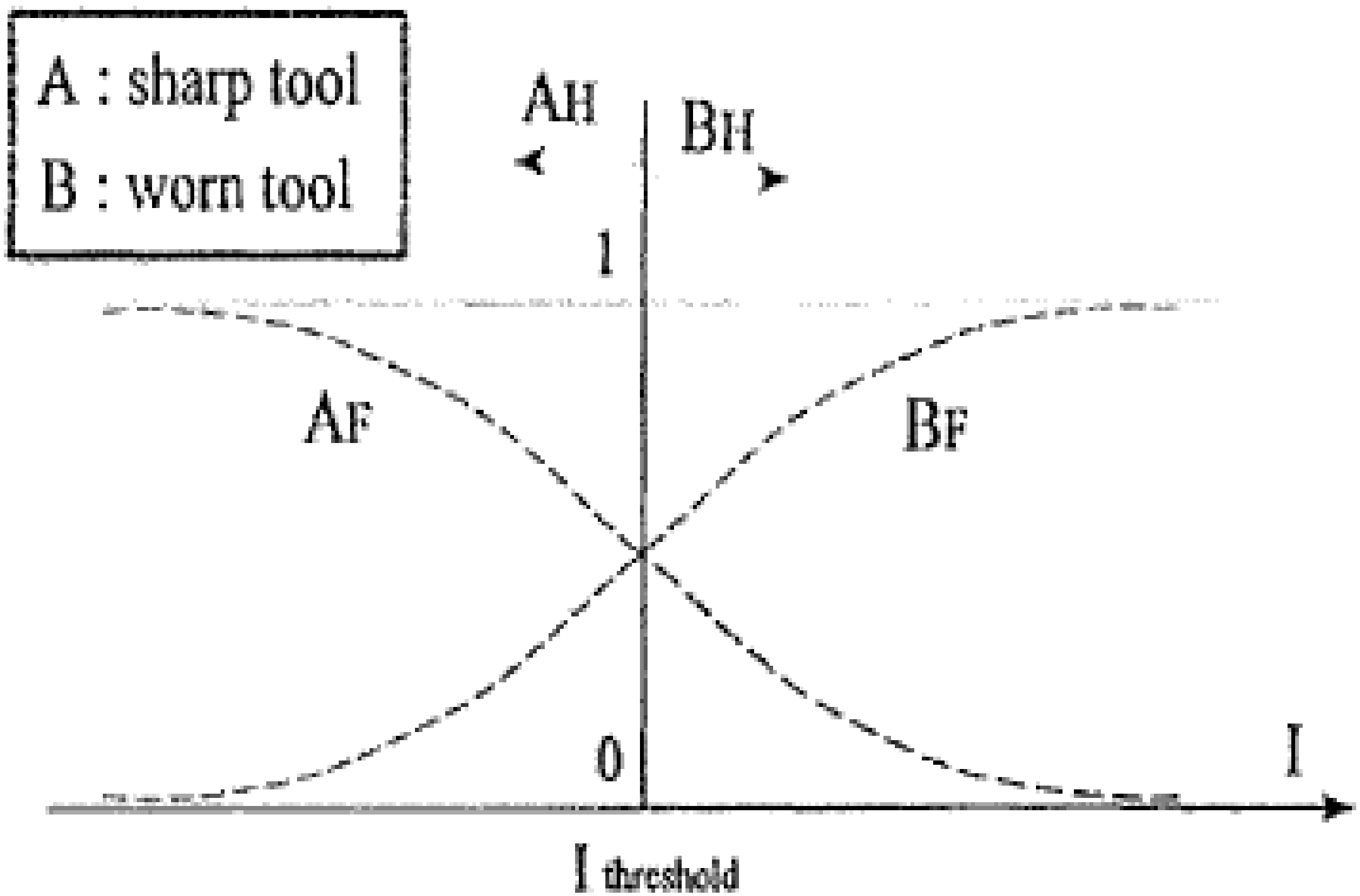


Figure: Soft boundaries in fuzzy classification

Checkpoint (1 of 2)

Multiple choice questions:

1. Robotics is an inter disciplinary branch of engineering and science that deals with ______.
   a) Construction
   b) Operation
   c) Use of robots
   d) All the above

2. Wrist has three degrees of freedom _______.
   a) Roll
   b) Pitch
   c) Both a) and b)
   d) None of these

3. Machine learning is classified into two areas ______.
   a) AI type learning on symbolic computation
   b) Neural network
   c) Both a) and b)
   d) None of these

Checkpoint solutions (1 of 2)

Multiple choice questions:

1. Robotics is an inter disciplinary branch of engineering and science that deals with ______.
   a) Construction
   b) Operation
   c) Use of robots
   d) All the above

2. Wrist has three degrees of freedom _______.
   a) Roll
   b) Pitch
   c) Both a) and b)
   d) None of these

3. Machine learning is classified into two areas ______.
   a) AI type learning on symbolic computation
   b) Neural network
   c) Both a) and b)
   d) None of these

Checkpoint (2 of 2)

Fill in the blanks:

1. The traces of the first robot can be found deep into the ____________century.
2. ___________is a much more difficult problem than forward kinematics.
3. A software has been developed to effectively use the information from the _____________.
4. The dominant lag in the force feedback path generates ___________around the force reflection path and so eases the operator's problems in stabilizing the system.

True or False:

1. Force control is a central requirement if robot arms are to use tools or interact with work-pieces in an unstructured environment. True / False
2. The forward kinematics problem is quite complicated and there is complexity in deriving the equations. True /False
3. An objective of high-performance tele-operation is to give a human operator a sense of feel which can aid in the implementation of a remote task. True / False

Checkpoint solutions (2 of 2)

Fill in the blanks:

1. The traces of the first robot can be found deep into the **18th century**.
2. **Inverse kinematics** is a much more difficult problem than forward kinematics.
3. A software has been developed to effectively use the information from the **fingertip system**.
4. The dominant lag in the force feedback path generates **a phase lead** around the force reflection path and so eases the operator's problems in stabilizing the system.

True or False:

1. Force control is a central requirement if robot arms are to use tools or interact with work-pieces in an unstructured environment. **True**
2. The forward kinematics problem is quite complicated and there is complexity in deriving the equations. **False**
3. An objective of high performance tele-operation is to give a human operator a sense of feel which can aid in the implementation of a remote task. **True**

Question bank

Two mark questions:

1. Define the robotics?
2. Define inverse kinematics?
3. Discuss the factors that affect the design of force controller.
4. What is Salford theories?

Four mark questions:

1. Explain robot kinematics with examples?
2. Define inverse kinematics. Why it is complex.
3. Discuss different types of Saridis architecture?
4. Define machine learning. Explain rule-based machine learning system?

Eight mark questions:

1. Explain the three layer architecture implementation in advanced robotics.
2. What is Force feedback strategy? Explain.

Unit summary

Having completed this unit, you should be able to:

- Gain knowledge on the basic concepts of robotics and its components
- Gain an insight into the role of machine learning in modern day robotics industry
- Learn about the kinematic and dynamic control concept with a focus on intelligent gripping systems
- Gain an insight into the design and development of robotic components
- Learn about environment capturing sensors like CCD cameras
- Gain knowledge on the integration of sensors with real time robotic system
- Learn about the fuzzy classification and uncertainties in tool condition monitoring system